ETH zürich

Risk, Safety & Uncertainty Quantification

Recent developments in surrogate modelling for uncertainty quantification

Bruno Sudret

Contributions from: K. Konakli, C.V. Mai, S. Marelli, J. Nagel, R. Schöbi

Chair of Risk, Safety and Uncertainty Quantification
ETH Zurich

ICVRAM/ISUMA/Uncertainties Conference - Florianopolis, Brazil
April 9, 2018

How to cite?

This presentation is an extended version of the keynote lecture given at the 3rd International Conference on Vulnerability, Risk Analysis and Management (ICVRAM), jointly organized with the 7th International Symposium on Uncertainty Modelling and Analysis (ISUMA) and the 4th International Symposium on Uncertainty Quantification and Stochastic Modelling in Florianopolis (Brazil), April 8-11, 2018 (`http://icvramisuma2018.org/`).

How to cite

Sudret, B. *Recent developments in surrogate modelling for uncertainty quantification* (2018), Proc. 3rd Int. Conf. on Vulnerability, Risk Analysis and Management (ICVRAM2018), Florianopolis (Brazil), April 8-11.

Computational models in engineering

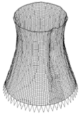
Complex engineering systems are designed and assessed using computational models, a.k.a simulators

A computational model combines:

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

$$\begin{aligned} \nabla \cdot \mathbf{D} &= \rho \\ \nabla \cdot \mathbf{B} &= 0 \\ \nabla \times \mathbf{E} &= -\frac{\partial \mathbf{B}}{\partial t} \\ \nabla \times \mathbf{H} &= \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t} \end{aligned}$$

- Discretization techniques which transform continuous equations into linear algebra problems

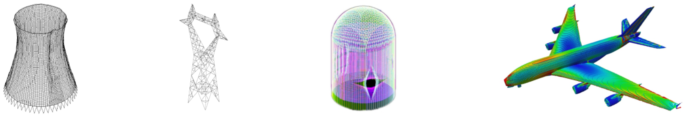


- Algorithms to solve the discretized equations

Computational models in engineering

Computational models are used:

- Together with experimental data for calibration purposes
- 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



Computational models: the abstract viewpoint

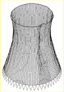
A computational model may be seen as a black box program that computes quantities of interest (QoI) (a.k.a. model responses) as a function of input parameters

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

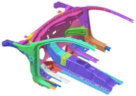


- Analytical formula
- Finite element model
- Comput. workflow

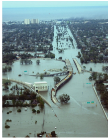
- Displacements
- Strains, stresses
- Temperature, etc.

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)

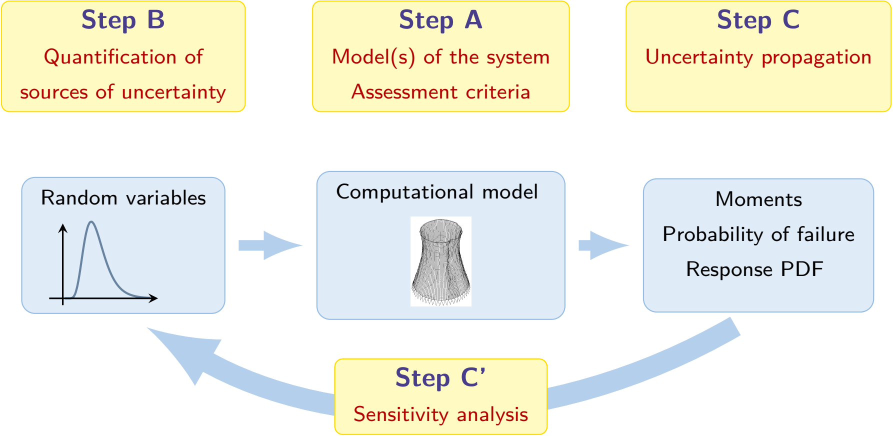


- Unforecast exposures: exceptional service loads, natural hazards (earthquakes, floods, landslides), climate loads (hurricanes, snow storms, etc.), accidental human actions (explosions, fire, etc.)



Outline

Global framework for uncertainty quantification



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

Step B: Quantification of the sources of uncertainty

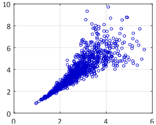
Goal: represent the uncertain parameters based on the *available data and information*

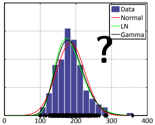
Probabilistic model $f_{\boldsymbol{X}}$

Experimental data is available

- What is the distribution of each parameter ?
- What is the dependence structure ?

Copula theory

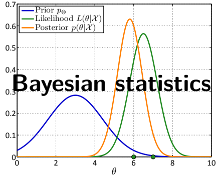




No data is available: expert judgment

- Engineering knowledge (*e.g.* reasonable bounds and uniform distributions)
- Statistical arguments and literature (*e.g.* extreme value distributions for climatic events)

Scarce data + expert information



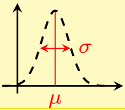
Step C: uncertainty propagation

Goal: estimate the uncertainty / variability of the quantities of interest (QoI) $Y = \mathcal{M}(\boldsymbol{X})$ due to the input uncertainty $f_{\boldsymbol{X}}$

- Output statistics, *i.e.* mean, standard deviation, etc.

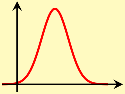
$$\mu_Y = \mathbb{E}_{\boldsymbol{X}}\left[\mathcal{M}(\boldsymbol{X})\right]$$
$$\sigma_Y^2 = \mathbb{E}_{\boldsymbol{X}}\left[\left(\mathcal{M}(\boldsymbol{X}) - \mu_Y\right)^2\right]$$

Mean/std. deviation



- Distribution of the QoI

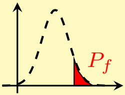
Response PDF



- Probability of exceeding an admissible threshold y_{adm}

$$P_f = \mathbb{P}\left(Y \geq y_{adm}\right)$$

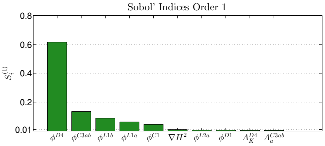
Probability of failure



Step C': sensitivity analysis

Goal: determine what are the input parameters (or combinations thereof) whose uncertainty explains the variability of the quantities of interest

- Screening: detect input parameters whose uncertainty has no impact on the output variability
- Feature setting: detect input parameters which allow one to best decrease the output variability when set to a deterministic value
- Exploration: detect interactions between parameters, *i.e.* joint effects not detected when varying parameters one-at-a-time

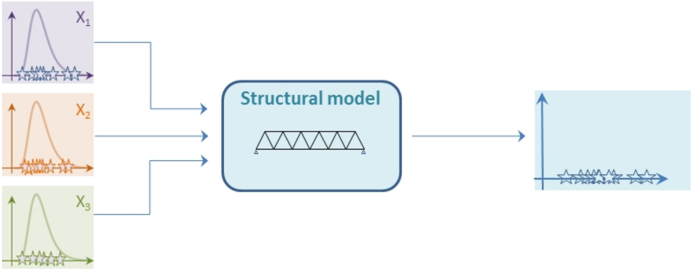


Variance decomposition (Sobol' indices)

Uncertainty propagation using Monte Carlo simulation

Principle: Generate virtual prototypes of the system using random numbers

- A sample set $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ is drawn according to the input distribution $f_{\boldsymbol{X}}$
- For each sample the quantity of interest (resp. performance criterion) is evaluated, say $\mathcal{Y} = \{\mathcal{M}(\boldsymbol{x}_1), \ldots, \mathcal{M}(\boldsymbol{x}_n)\}$



- The set of quantities of interest is used for moments-, distribution- or reliability analysis

Advantages/Drawbacks of Monte Carlo simulation

Advantages

- Universal method: only rely upon sampling random numbers and running repeatedly the computational model
- Sound statistical foundations: convergence when $N_{MCS} \to \infty$
- Suited to High Performance Computing: "embarrassingly parallel"

Drawbacks

- Statistical uncertainty: results are not exactly reproducible when a new analysis is carried out (handled by computing confidence intervals)
- Low efficiency: convergence rate $\propto n^{-1/2}$

Monte Carlo for reliability analysis

To compute $P_f = 10^{-k}$ with an accuracy of $\pm 10\%$ (coef. of variation of 5%), $4 \cdot 10^{k+2}$ runs are required

Surrogate models for uncertainty quantification

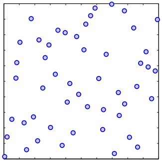
A surrogate model $\tilde{\mathcal{M}}$ is an approximation of the original computational model $\mathcal{M}$ with the following features:

- 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); Sudret & Der Kiureghian (2000); Xiu & Karniadakis (2002); Soize & Ghanem (2004)

- Consider the input random vector $\boldsymbol{X}$ (dim $\boldsymbol{X} = M$) with given probability density function (PDF) $f_{\boldsymbol{X}}(\boldsymbol{x}) = \prod_{i=1}^{M} f_{X_i}(x_i)$

- Assuming that the random output $Y = \mathcal{M}(\boldsymbol{X})$ has finite variance, it can be cast as the following polynomial chaos expansion:

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

  where :

  - $\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$: basis functions
  - $y_{\boldsymbol{\alpha}}$: coefficients to be computed (coordinates)

- The PCE basis $\left\{\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}),\ \boldsymbol{\alpha} \in \mathbb{N}^M\right\}$ is made of multivariate orthonormal polynomials

Multivariate polynomial basis

Univariate polynomials

- For each input variable X_i, univariate orthogonal polynomials $\{P_k^{(i)},\ k \in \mathbb{N}\}$ are built:

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

e.g. , Legendre polynomials if $X_i \sim \mathcal{U}(-1,1)$, Hermite polynomials if $X_i \sim \mathcal{N}(0,1)$

- Normalization: $\Psi_j^{(i)} = P_j^{(i)} / \sqrt{\gamma_j^{(i)}} \quad i = 1, \dots, M, \quad j \in \mathbb{N}$

Tensor product construction

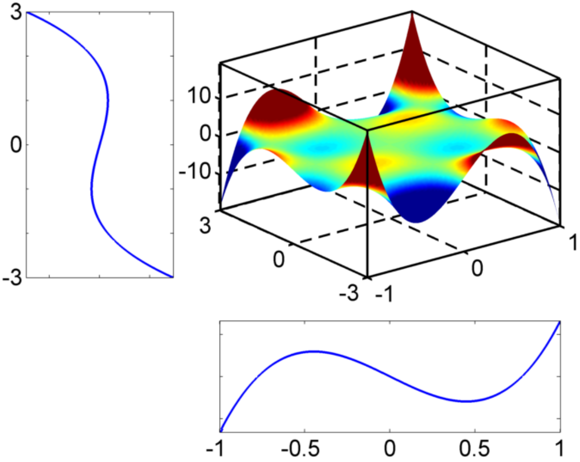
$$\Psi_{\boldsymbol{\alpha}}(\boldsymbol{x}) \stackrel{\text{def}}{=} \prod_{i=1}^{M} \Psi_{\alpha_i}^{(i)}(x_i) \qquad \mathbb{E}\left[\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\Psi_{\boldsymbol{\beta}}(\boldsymbol{X})\right] = \delta_{\boldsymbol{\alpha\beta}}$$

where $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_M)$ are multi-indices (partial degree in each dimension)

Example: $M = 2$

Xiu & Karniadakis (2002)

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



- $X_1 \sim \mathcal{U}(-1,1)$: Legendre polynomials
- $X_2 \sim \mathcal{N}(0,1)$: Hermite polynomials

Isoprobabilistic transform

- 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

Need for isoprobabilistic transforms

Isoprobabilistic transform

Independent variables

- Given the marginal CDFs $X_i \sim F_{X_i} \qquad i = 1, \dots, M$
- A one-to-one mapping to reduced variables is used:

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

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

- The best choice is dictated by the least non linear transform

General case: addressing dependence

Sklar's theorem (1959)

- The joint CDF is defined through its marginals and copula

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

- Rosenblatt or Nataf isoprobabilistic transform is used

Outline

Computing the coefficients by least-square minimization

Isukapalli (1999); Berveiller, Sudret & Lemaire (2006)

Principle

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 : $\mathbf{Y} = \{y_{\boldsymbol{\alpha}},\, \boldsymbol{\alpha} \in \mathcal{A}\} \equiv \{y_0, \ldots, y_{P-1}\}$ (P unknown coef.)

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

Least-square minimization

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

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

Discrete (ordinary) least-square minimization

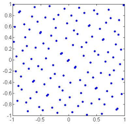
An estimate of the mean square error (sample average) is minimized:

$$\hat{\mathbf{Y}} = \arg\min_{\mathbf{Y}\in\mathbb{R}^P} \frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{Y}^{\mathsf{T}}\boldsymbol{\Psi}(\boldsymbol{x}^{(i)}) - \mathcal{M}(\boldsymbol{x}^{(i)})\right)^2$$

Procedure

- Select a truncation scheme, *e.g.* $\mathcal{A}^{M,p} = \left\{\boldsymbol{\alpha}\in\mathbb{N}^M \;:\; |\boldsymbol{\alpha}|_1 \leq p\right\}$
- Select an experimental design and evaluate the model response

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



- Compute the experimental matrix

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

- Solve the resulting linear system

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

Simple is beautiful !

Error estimators

- In least-squares analysis, the generalization error is defined as:

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

- The empirical error based on the experimental design $\mathcal{X}$ is a poor estimator in case of overfitting

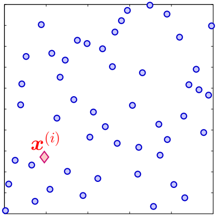
$$E_{emp} = \frac{1}{n}\sum_{i=1}^{n}\left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}^{\mathsf{PC}}(\boldsymbol{x}^{(i)})\right)^2$$

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

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

R^2 is a poor estimator of the accuracy of the PCE when there is overfitting

Leave-one-out cross validation



- An experimental design $\mathcal{X} = \{\boldsymbol{x}^{(j)},\ j = 1, \ldots, n\}$ is selected
- Polynomial chaos expansions are built using all points but one, *i.e.* based on $\mathcal{X} \backslash \boldsymbol{x}^{(i)} = \{\boldsymbol{x}^{(j)},\ j = 1, \ldots, n,\ j \neq i\}$

- Leave-one-out error (PRESS)

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

- Analytical derivation from a single PC analysis

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

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

Least-squares analysis: Wrap-up

Algorithm 1: Ordinary least-squares

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

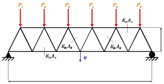
Outline

Curse of dimensionality

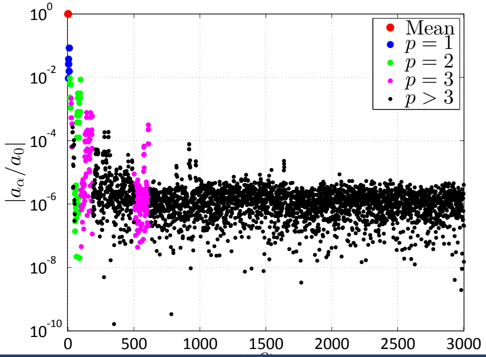
- The cardinality of the truncation scheme $\mathcal{A}^{M,p}$ is $P = \dfrac{(M+p)!}{M!\,p!}$
- Typical computational requirements: $n = OSR \cdot P$ where the oversampling rate is $OSR = 2 - 3$

However ... most coefficients are close to zero !

Example



- Elastic truss structure with $M = 10$ independent input variables
- PCE of degree $p = 5$ ($P = 3,003$ coeff.)



Hyperbolic truncation sets

Sparsity-of-effects principle

Blatman & Sudret, Prob. Eng. Mech (2010); J. Comp. Phys (2011)

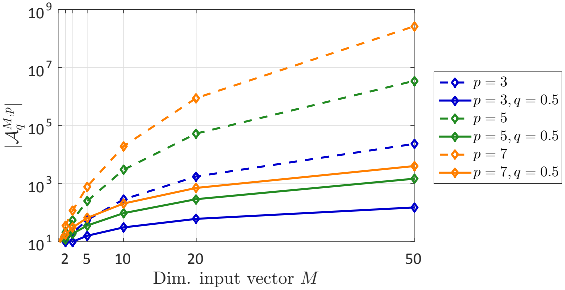
In most engineering problems, only low-order interactions between the input variables are relevant

- $q-$norm of a multi-index $\boldsymbol{\alpha}$:

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

- Hyperbolic truncation sets:

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



Compressive sensing approaches

Blatman & Sudret (2011); Doostan & Owhadi (2011); Ian, Guo, Xiu (2012); Sargsyan *et al.* (2014); Jakeman *et al.* (2015)

- Sparsity in the solution can be induced by ℓ_1-regularization:

$$\boldsymbol{y}_{\boldsymbol{\alpha}} = \arg\min \frac{1}{n} \sum_{i=1}^{n} \left(\mathsf{Y}^{\mathsf{T}} \boldsymbol{\Psi}(\boldsymbol{x}^{(i)}) - \mathcal{M}(\boldsymbol{x}^{(i)}) \right)^2 + \lambda \parallel \boldsymbol{y}_{\boldsymbol{\alpha}} \parallel_1$$

- Different algorithms: LASSO, orthogonal matching pursuit, Bayesian compressive sensing

Least Angle Regression

Efron *et al.* (2004)
Blatman & Sudret (2011)

- Least Angle Regression (LAR) solves the LASSO problem for different values of the penalty constant in a single run without matrix inversion
- Leave-one-out cross validation error allows one to select the best model

Sparse PCE: wrap up

Algorithm 2: LAR-based Polynomial chaos expansion

```
Input: Computational budget n
Initialization
    Sample experimental design 𝒳 = {x^(1), ..., x^(n)}
    Evaluate model response 𝒴 = {ℳ(x^(1)), ..., ℳ(x^(n))}
PCE construction
    for p = p_min : p_max do
        for q ∈ 𝒬 do
            Select candidate basis 𝒜_q^{M,p}
            Run LAR for extracting the optimal sparse basis 𝒜*(p, q)
            Compute coefficients {y_α, α ∈ 𝒜*(p, q)} by OLS
            Compute e_LOO(p, q)
        end
    end
    (p*, q*) = arg min e_LOO(p, q)
Return Optimal sparse basis 𝒜*(p, q), PCE coefficients, e_LOO(p*, q*)
```

Outline

Post-processing sparse PC expansions

Statistical moments

- Due to the orthogonality of the basis functions $(\mathbb{E}\left[\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\Psi_{\boldsymbol{\beta}}(\boldsymbol{X})\right] = \delta_{\boldsymbol{\alpha\beta}})$ and using $\mathbb{E}\left[\Psi_{\boldsymbol{\alpha}\neq\boldsymbol{0}}\right] = 0$ the statistical moments read:

$$\text{Mean:} \qquad \hat{\mu}_Y = y_0$$

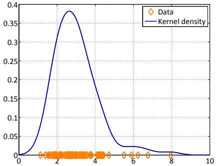
$$\text{Variance:} \qquad \hat{\sigma}_Y^2 = \sum_{\boldsymbol{\alpha}\in\mathcal{A}\setminus\boldsymbol{0}} y_{\boldsymbol{\alpha}}^2$$

Distribution of the QoI

- The PCE can be used as a response surface for sampling:

$$\mathfrak{y}_j = \sum_{\boldsymbol{\alpha}\in\mathcal{A}} y_{\boldsymbol{\alpha}}\,\Psi_{\boldsymbol{\alpha}}(\boldsymbol{x}_j) \quad j = 1, \ldots, n_{big}$$

- The PDF of the response is estimated by histograms or kernel smoothing



Sensitivity analysis

Goal

Sobol' (1993); Saltelli *et al.* (2000)

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

Hoeffding-Sobol' decomposition

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

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

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

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

- Sobol' indices:

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

- 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:

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

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

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

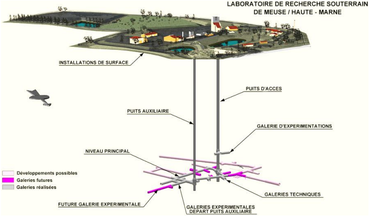
$$\mathcal{M}^{\text{PC}}(\boldsymbol{x}) = \mathcal{M}_0 + \sum_{\mathbf{u} \subset \{1, \dots, M\}} \mathcal{M}_{\mathbf{u}}(\boldsymbol{x}_{\mathbf{u}}) \qquad \text{where} \qquad \mathcal{M}_{\mathbf{u}}(\boldsymbol{x}_{\mathbf{u}}) \stackrel{\text{def}}{=} \sum_{\boldsymbol{\alpha} \in \mathcal{A}_{\mathbf{u}}}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

PC-based Sobol' indices

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

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

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

Joint work with University of Neuchâtel

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

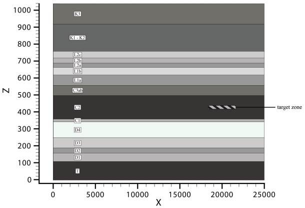
- Two-dimensional idealized model of the Paris Basin (25 km long / 1,040 m depth) with 5×5 m mesh (10^6 elements)
- Steady-state flow simulation with Dirichlet boundary conditions:

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

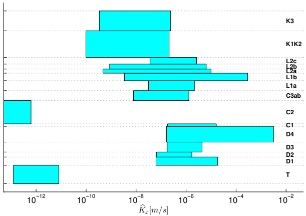
- 15 homogeneous layers with uncertainties in:
  - Porosity (resp. hydraulic conductivity)
  - Anisotropy of the layer properties (inc. dispersivity)
  - Boundary conditions (hydraulic gradients)

78 input parameters

Sensitivity analysis



Geometry of the layers



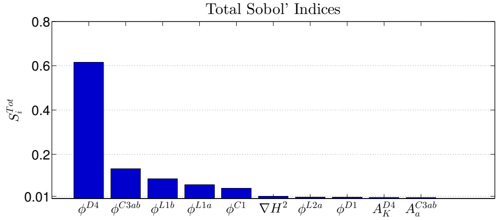
Conductivity of the layers

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

Technique: Sobol'indices computed from polynomial chaos expansions



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

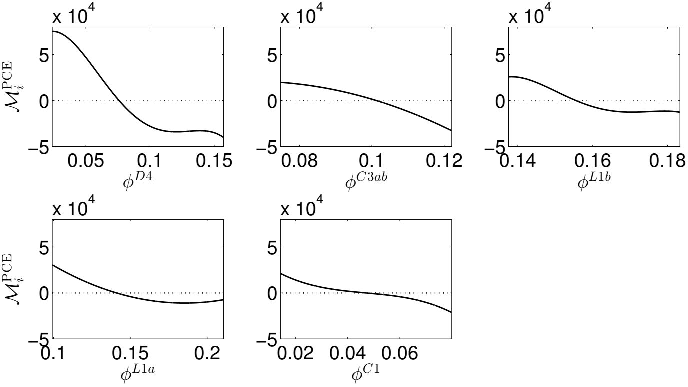
Conclusions

- 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

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

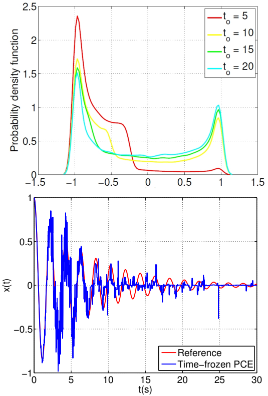


Polynomial chaos expansions in structural dynamics

Spiridonakos et al. (2015); Mai, Spiridonakos, Chatzi & Sudret, IJUQ (2016); Mai & Sudret, SIAM JUQ (2017)

Premise

- For dynamical systems, the complexity of the map $\boldsymbol{\xi} \mapsto \mathcal{M}(\boldsymbol{\xi}, t)$ increases with time.
- Time-frozen PCE does not work beyond first time instants



PC-NARX

- Use of non linear autoregressive with exogenous input models (NARX) to capture the dynamics:

$$y(t) = \mathcal{F}\left(x(t), \ldots, x(t-n_x), y(t-1), \ldots, y(t-n_y)\right) + \epsilon_t \equiv \mathcal{F}\left(\boldsymbol{z}(t)\right) + \epsilon_t$$

Earthquake engineering – Bouc-Wen oscillator

Governing equations

Kafali & Grigoriu (2007), Spiridonakos & Chatzi (2015)

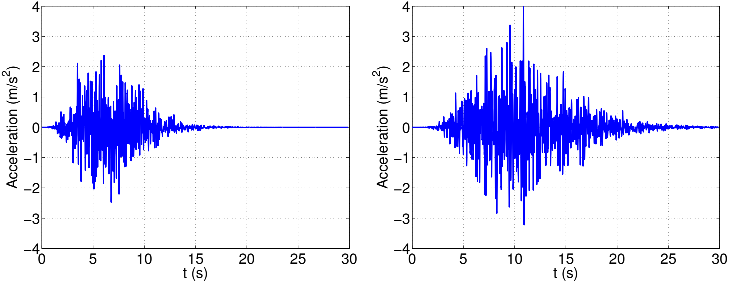
$$\ddot{y}(t) + 2\,\zeta\,\omega\,\dot{y}(t) + \omega^2(\rho\,y(t) + (1-\rho)\,z(t)) = -x(t),$$
$$\dot{z}(t) = \gamma\dot{y}(t) - \alpha\,|\dot{y}(t)|\,|z(t)|^{n-1}\,z(t) - \beta\,\dot{y}(t)\,|z(t)|^n\,,$$

Excitation

$x(t)$ is generated by a probabilistic ground motion model

Rezaeian & Der Kiureghian (2010)

$$x(t) = q(t, \boldsymbol{\alpha}) \sum_{i=1}^{n} s_i\left(t, \boldsymbol{\lambda}(t_i)\right)\,U_i$$

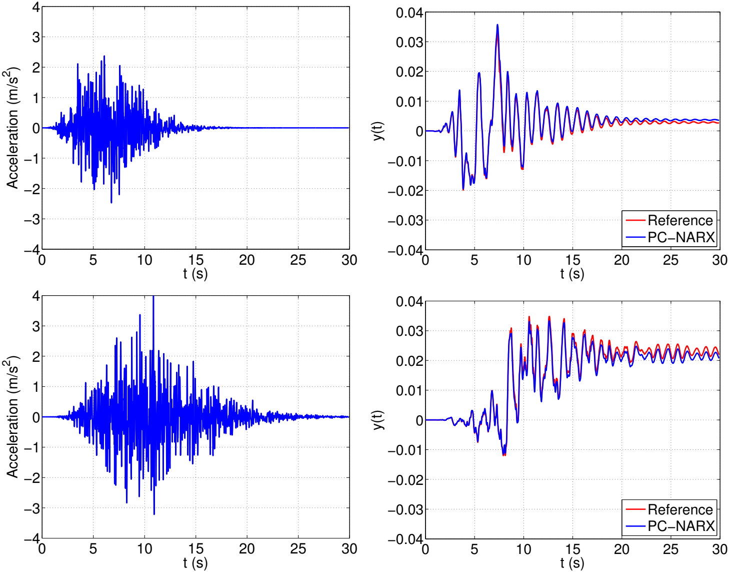


Bouc-Wen model

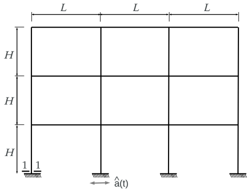
Marginal distributions of the model parameters

Parameters	Distribution	Support	Mean	Std
ω (rad/s)	Uniform	$[5.373, 6.567]$	5.97	0.3447
α (1/m)	Uniform	$[45, 55]$	50	2.887
I_a (s.g)	Lognormal	(0, $+\infty$)	0.0468	0.164
D_{5-95} (s)	Beta	[5, 45]	17.3	9.31
t_{mid} (s)	Beta	[0.5, 40]	12.4	7.44
$\omega_{mid}/2\pi$ (Hz)	Gamma	(0, $+\infty$)	5.87	3.11
$\omega'/2\pi$ (Hz)	Two-sided exponential	[-2, 0.5]	-0.089	0.185
ζ_f (.)	Beta	[0.02, 1]	0.213	0.143

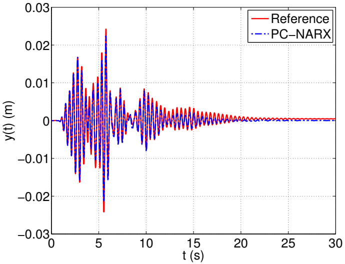
Bouc-Wen model: prediction

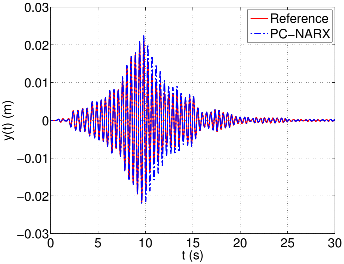


Earthquake engineering – frame structure



- 2D steel frame with uncertain properties submitted to synthetic ground motions
- Experimental design of size 300



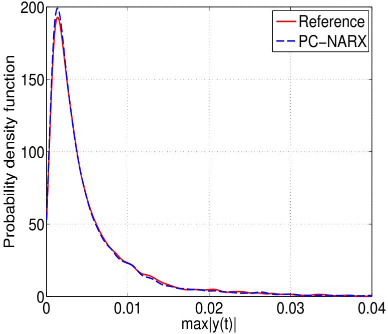


Surrogate model of single trajectories

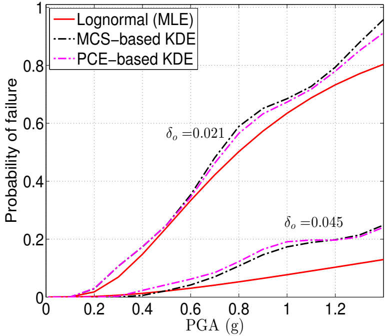
Frame structure – fragility curves

First-storey drift

- PC-NARX calibrated based on 300 simulations
- Reference results obtained from 10,000 Monte Carlo simulations



PDF of max. drift



Fragility curves for two drift thresholds

Other usage of polynomial chaos expansions

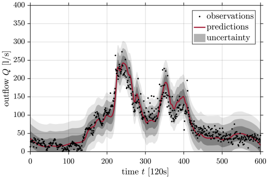
Bayesian inversion

- PCE of the forward model used in conjunction with Markov Chain Monte Carlo (MCMC) simulation

Nagel & Sudret, PEM (2016)

- Spectral likelihood expansions

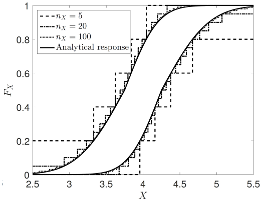
Nagel & Sudret, J. Comp. Phys. (2016)



Propagation of mixed epistemic/aleatory uncertainties

- Input uncertainty modelled by free (resp.) parametric p-boxes
- Uncertainty propagation using augmented spaces and optimization

Schöbi & Sudret, PEM (2017) ; J. Comp. Phys (2017)



Outline

Introduction

- Polynomial chaos expansions (PCE) represent the model output on a fixed, predetermined basis:

$$Y = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^M} y_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \qquad\qquad \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) = \prod_{i=1}^{M} P_{\alpha_i}^{(i)}(X_i)$$

- Sparse PCEs are built from a pre-selected set of candidate basis functions $\mathcal{A}$
- High-dimensional problems (*e.g.* $M > 50$) may still be challenging for sparse PCE in case of small experimental designs ($n < 100$)

Low-rank tensor representations

Rank-1 function

A rank-1 function of $\boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}}$ is a product of univariate functions of each component:

$$w(\boldsymbol{x}) = \prod_{i=1}^{M} v^{(i)}(x_i)$$

Canonical low-rank approximation (LRA)

A canonical decomposition of $\mathcal{M}(\boldsymbol{x})$ is of the form

Nouy, Arch. Comput. Meth. Eng. (2010)

$$\mathcal{M}^{\mathrm{LRA}}(\boldsymbol{x}) = \sum_{l=1}^{R} b_l \left(\prod_{i=1}^{M} v_l^{(i)}(x_i) \right)$$

where:

- R is the rank (# terms in the sum)
- $v_l^{(i)}(x_i)$ are univariate function of x_i
- b_l are normalizing coefficients

Low-rank tensor representations

Polynomial expansions

Doostan et al., 2013

By expanding $v_l^{(i)}(X_i)$ onto polynomial basis orthonormal w.r.t. f_{X_i} one gets:

$$\widehat{Y} = \sum_{l=1}^{R} b_l \left(\prod_{i=1}^{M} \left(\sum_{k=0}^{p_i} z_{k,l}^{(i)} \, P_k^{(i)}(X_i) \right) \right)$$

where:

- $P_k^{(i)}(X_i)$ is k-th degree univariate polynomial of X_i
- p_i is the maximum degree of $P_k^{(i)}$
- $z_{k,l}^{(i)}$ are coefficients of $P_k^{(i)}$ in the l-th rank-1 term

Complexity

Assuming an isotropic representation $(p_i = p)$, the number of unknown coefficients is $R(p \cdot M + 1)$

Linear increase with dimensionality M

Greedy construction of the LRA

Chevreuil et al. (2015); Konakli & Sudret (2016)

- An greedy construction is carried out by iteratively adding rank-1 terms. The r-th approximation reads $\widehat{Y}_r = \mathcal{M}_r(\boldsymbol{X}) = \sum_{l=1}^{r} b_l w_l(\boldsymbol{X})$
- In each iteration, alternate least-squares are used (correction and updating steps)

Correction step: sequential updating of $\boldsymbol{z}_r^{(j)}$, $j = 1, \ldots, M$, to build w_r:

$$\boldsymbol{z}_r^{(j)} = \arg\min_{\boldsymbol{\zeta} \in \mathbb{R}^{p_j}} \left\| \mathcal{M} - \widehat{\mathcal{M}}_{r-1} - \left(\prod_{i \neq j} \sum_{k=0}^{p_i} z_{k,r}^{(i)} \, P_k^{(i)} \right) \left(\sum_{k=0}^{p_j} \zeta_k \, P_k^{(j)} \right) \right\|_{\mathcal{E}}^2$$

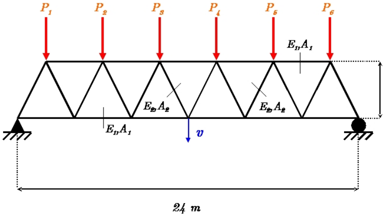
Updating step: evaluation of normalizing coefficients $\{b_1, \ldots, b_r\}$:

$$\boldsymbol{b} = \arg\min_{\boldsymbol{\beta} \in \mathbb{R}^r} \left\| \mathcal{M} - \sum_{l=1}^{r} \beta_l w_l \right\|_{\mathcal{E}}^2$$

Elastic truss

Structural model

Blatman & Sudret (2011)



- Response quantity: maximum deflection U
- Reliability analysis:

$$P_f = \mathbb{P}\left(U \geq u_{\text{lim}}\right)$$

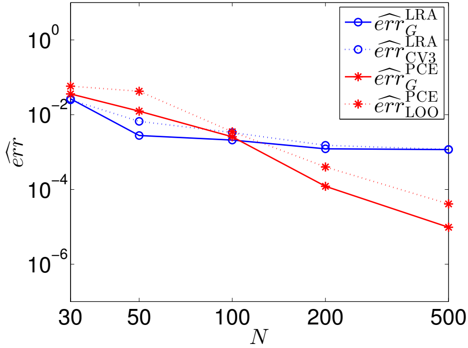
Probabilistic model

Variable	Distribution	mean	CoV
Hor. bars cross section A_1 [m]	Lognormal	0.002	0.10
Oblique bars cross section A_2 [m]	Lognormal	0.001	0.10
Young's moduli E_1, E_2 [MPa]	Lognormal	210,000	0.10
Loads $P_1, \ldots, P_6$ [KN]	Gumbel	50	0.15

Elastic truss

Konakli & Sudret, Prob. Eng. Mech (2016)

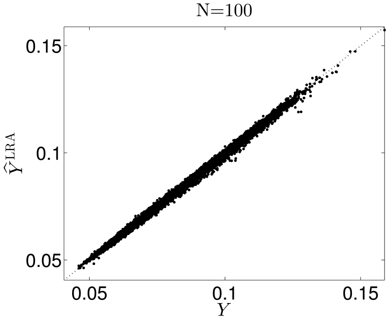
Surrogate modelling error



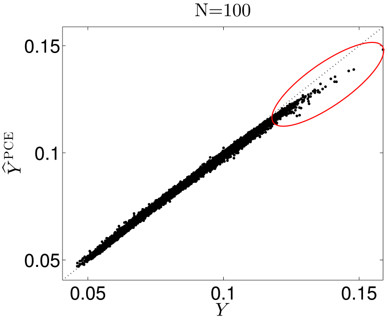
- Smaller validation error for LRA when ED is small ($N < 100$)
- Faster error decrease for PCE
- However ...

Elastic truss: validation plots

Konakli & Sudret, Prob. Eng. Mech (2016)



Low-rank approximation



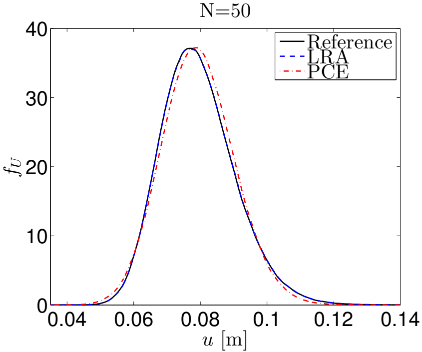
Polynomial chaos expansion

Polynomial chaos approximation is biased in the high values

PDF of the truss deflection

Size of the experimental design: 50 (resp. 100) samples from Sobol' sequence

Kernel density estimates of the PDF **in the linear scale**

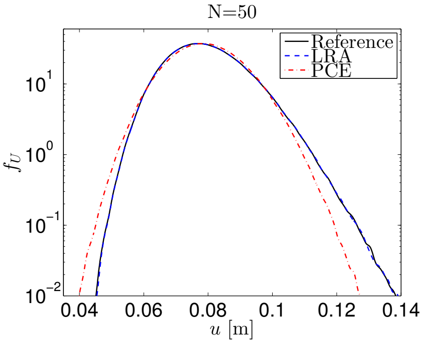


50 samples **100 samples**

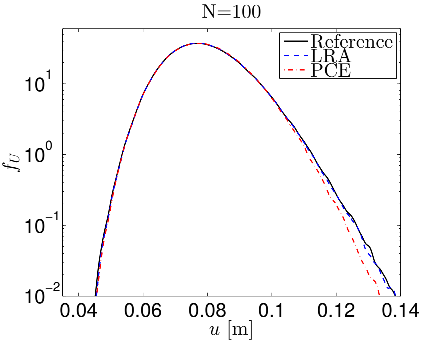
PDF of the truss deflection

Size of the experimental design: 50 (resp. 100) samples from Sobol' sequence

Kernel density estimates of the PDF **in the log scale**



50 samples

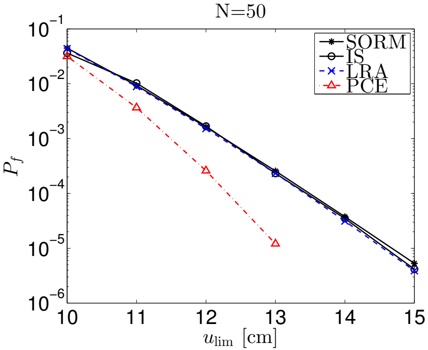


100 samples

Truss deflection - reliability analysis

Probability of failure

- LRA/PCE built from 50 samples
- Post-processing by crude Monte Carlo simulation: $P_f = \mathbb{P}\left(U \geq \boldsymbol{u}_{\text{lim}}\right)$



Number of model evaluations

u_{lim}(m)	SORM	IS
0.10	387	375
0.11	365	553
0.12	372	660
0.13	367	755
0.14	379	1,067
0.15	391	1,179

Full curve at the cost of 50 finite element analyses

Outline

Gaussian process modelling (a.k.a Kriging)

Santner, Williams & Notz (2003)

Kriging assumes that $\mathcal{M}(\boldsymbol{x})$ is a trajectory of an underlying Gaussian process

$$\mathcal{M}(\boldsymbol{x}) \approx \mathcal{M}^{(\mathsf{K})}(\boldsymbol{x}) = \boldsymbol{\beta}^{\mathsf{T}} \boldsymbol{f}(\boldsymbol{x}) + \sigma^2 \, Z(\boldsymbol{x}, \omega)$$

where:

- $\boldsymbol{\beta}^{\mathsf{T}} \boldsymbol{f}(\boldsymbol{x})$: trend
- $Z(\boldsymbol{x}, \omega)$: zero mean, unit variance Gaussian process with autocorrelation function, *e.g.* :

$$R\left(\boldsymbol{x},\, \boldsymbol{x}'\right) = \exp\left(\sum_{k=1}^{M} - \left(\frac{x_k - x_k'}{\theta_k}\right)^2\right)$$

- σ^2: variance

The Gaussian measure artificially introduced is different from the aleatory uncertainty on the model parameters $\boldsymbol{X}$

Kriging prediction

Unknown parameters

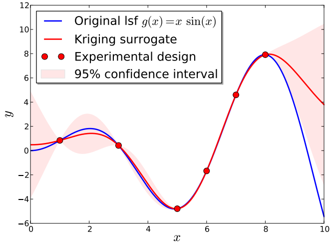
- Parameters $\{\boldsymbol{\theta},\, \boldsymbol{\beta},\, \sigma^2\}$ are estimated from the experimental design $\mathcal{Y} = \{y_i = \mathcal{M}(\boldsymbol{\chi}_i)\,,\, i = 1,\, \dots\,, n\}$ by maximum likelihood estimation, cross validation or Bayesian calibration

Mean predictor

$$\mu_{\widehat{Y}}(\boldsymbol{x}) = \boldsymbol{f}\,(\boldsymbol{x})^{\mathsf{T}}\;\hat{\boldsymbol{\beta}} + \boldsymbol{r}\,(\boldsymbol{x})^{\mathsf{T}}\,\boldsymbol{R}^{-1}\,\left(\mathcal{Y} - \boldsymbol{F}\,\hat{\boldsymbol{\beta}}\right)$$

where:

$$\begin{aligned} r_i(\boldsymbol{x}) &= R\left(\boldsymbol{x} - \boldsymbol{x}^{(i)},\, \boldsymbol{\theta}\right) \\ \boldsymbol{R}_{ij} &= R\left(\boldsymbol{x}^{(i)} - \boldsymbol{x}^{(j)},\, \boldsymbol{\theta}\right) \\ \boldsymbol{F}_{ij} &= f_j\left(\boldsymbol{x}^{(i)}\right) \end{aligned}$$



Kriging variance

$$\sigma^2_{\widehat{Y}}\,(\boldsymbol{x}) = \sigma^2_Y\,\left(1 - \left\langle\; \boldsymbol{f}\,(\boldsymbol{x})^{\mathsf{T}} \quad \boldsymbol{r}\,(\boldsymbol{x})^{\mathsf{T}} \;\right\rangle \begin{bmatrix} \boldsymbol{0} & \boldsymbol{F}^{\mathsf{T}} \\ \boldsymbol{F} & \boldsymbol{R} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{f}\,(\boldsymbol{x}) \\ \boldsymbol{r}\,(\boldsymbol{x}) \end{bmatrix}\right)$$

Use of Kriging for structural reliability analysis

- From a given experimental design $\mathcal{X} = \left\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}\right\}$, Kriging yields a mean predictor $\mu_{\widehat{Y}}(\boldsymbol{x})$ and the Kriging variance $\sigma_{\widehat{Y}}(\boldsymbol{x})$
- The mean predictor is substituted for the "true" limit state function, defining the surrogate failure domain

$$\mathcal{D}_f{}^0 = \left\{\boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}} \;:\; \mu_{\widehat{Y}}(\boldsymbol{x}) \leq 0\right\}$$

- The probability of failure is approximated by:

Kaymaz, Struc. Safety (2005)

$$P_f^0 = \mathbb{P}\left[\mu_{\widehat{Y}}(\boldsymbol{X}) \leq 0\right] = \int_{\mathcal{D}_f^0} f_{\boldsymbol{X}}(\boldsymbol{x})\, d\boldsymbol{x} = \mathbb{E}\left[\mathbf{1}_{\mathcal{D}_f^0}(\boldsymbol{X})\right]$$

- Monte Carlo simulation can be used on the surrogate model:

$$\widehat{P_f^0} = \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_{\mathcal{D}_f^0}(\boldsymbol{x}_k)$$

Confidence bounds on the probability of failure

Shifted failure domains

Dubourg *et al.* , Struct. Mult. Opt. (2011)

- Let us define a confidence level $(1-\alpha)$ and $k_{1-\alpha} = \Phi^{-1}(1-\alpha/2)$, *i.e.* 1.96 if $1-\alpha = 95\%$, and:

$$\mathcal{D}_f^- = \left\{ \boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}} \ : \ \mu_{\widehat{Y}}(\boldsymbol{x}) + k_{1-\alpha}\, \sigma_{\widehat{Y}}(\boldsymbol{x}) \leq 0 \right\}$$

$$\mathcal{D}_f^+ = \left\{ \boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}} \ : \ \mu_{\widehat{Y}}(\boldsymbol{x}) - k_{1-\alpha}\, \sigma_{\widehat{Y}}(\boldsymbol{x}) \leq 0 \right\}$$

- Interpretation ($1-\alpha = 95\%$):
  - If $\boldsymbol{x} \in \mathcal{D}_f^0$ it belongs to the true failure domain with a 50% chance
  - If $\boldsymbol{x} \in \mathcal{D}_f^+$ it belongs to the true failure domain with 95% chance: conservative estimation

Bounds on the probability of failure

$$\mathcal{D}_f^- \subset \mathcal{D}_f^0 \subset \mathcal{D}_f^+ \qquad \Leftrightarrow \qquad P_f^- \leq P_f^0 \leq P_f^+$$

Adaptive designs for reliability analysis

Premise

- When using high-fidelity computational models for assessing structural reliability, the goal is to minimize the number of runs
- Adaptive experimental designs allow one to start from a small ED and enrich it with new points in suitable regions (*i.e.* close to to the limit state surface)

Enrichment (infill) criterion

Bichon *et al.* , (2008, 2011); Echard *et al.* (2011); Bect *et al.* (2012)

The following learning function is used:

$$LF(\boldsymbol{x}) = \frac{|\mu_{\hat{\mathcal{M}}}(\boldsymbol{x})|}{\sigma_{\hat{\mathcal{M}}}(\boldsymbol{x})}$$

- Small if $\mu_{\hat{\mathcal{M}}}(\boldsymbol{x}) \approx 0$ ($\boldsymbol{x}$ close to the limit state surface) and/or $\sigma_{\hat{\mathcal{M}}}(\boldsymbol{x}) >> 0$ (poor local accuracy)
- The probability of misclassification is $\Phi(-LF(\boldsymbol{x}))$
- At each iteration, the new point is: $\boldsymbol{\chi}^* = \arg\min LF(\boldsymbol{x})$

PC-Kriging

Schöbi & Sudret, IJUQ (2015); Kersaudy *et al.* , J. Comp. Phys (2015); chöbi & Sudret, ASME J. Risk (2016);

Heuristics: Combine polynomial chaos expansions (PCE) and Kriging

- PCE approximates the global behaviour of the computational model
- Kriging allows for local interpolation and provides a local error estimate

Universal Kriging model with a sparse PC expansion as a trend

$$\mathcal{M}(\boldsymbol{x}) \approx \mathcal{M}^{(\mathsf{PCK})}(\boldsymbol{x}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} a_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{x}) + \sigma^2 Z(\boldsymbol{x}, \omega)$$

PC-Kriging calibration

- Sequential PC-Kriging: least-angle regression (LAR) detects a sparse basis, then PCE coefficients are calibrated together with the auto-correlation parameters
- Optimized PC-Kriging: universal Kriging models are calibrated at each step of LAR

Conclusions

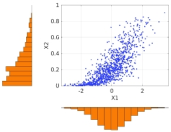
- Surrogate models are unavoidable for solving uncertainty quantification problems involving costly computational models (*e.g.* finite element models)
- Depending on the analysis, specific surrogates are most suitable: polynomial chaos expansions for distribution- and sensitivity analysis, low-rank tensor approximations and Kriging for reliability analysis
- Kriging and PC-Kriging are suitable for adaptive algorithms (enrichment of the experimental design)
- All these techniques are non-intrusive: they rely on experimental designs, the size of which is a user's choice
- They are versatile, general-purpose and field-independent
- All the presented algorithms are available in the general-purpose uncertainty quantification software UQLab

UQLab

www.uqlab.com

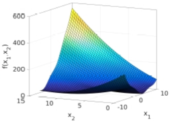
Probabilistic Input Modelling

- Common marginals
- Support for user-defined marginals
- Support for bounds on all distributions (including user-defined)
- Gaussian copula



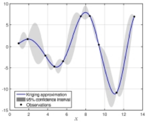
Modelling facilities

- Simple text strings
- MATLAB m-files
- MATLAB handles
- Simple API to produce wrappers to commercial/external solvers



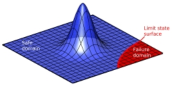
Advanced metamodelling

- Sparse degree-adaptive Polynomial Chaos Expansions
- Gaussian process modelling (Kriging)
- Polynomial-Chaos Kriging
- Low-rank tensor approximations



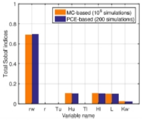
Reliability Analysis (Rare Event Estimation)

- FORM/SORM approximation
- Monte Carlo Simulation (MCS)
- Importance Sampling
- Subset Simulation
- Adaptive Kriging (AK-MCS)



Sensitivity Analysis

- Correlation-based indices
- Standard Regression Coefficients
- Cotter measure
- Morris indices
- Sampling-based Sobol' indices
- PCE-based Sobol' indices



Upcoming features

- UQLINK: easily connect UQLAB to external modelling software
- Bayesian model calibration/inversion toolbox
- Random fields discretization and sampling toolbox
- Support vector machines for regression and classification
- Reliability-based design optimization (RBDO)
- Advanced dependence modelling and inference with vine copulas

UQLab: The Uncertainty Quantification Laboratory

http://www.uqlab.com

- Release of V0.9 on July 1st, 2015; V0.92 on March 1st, 2016
- Release of V1.0 on April 28th, 2017 UQLabCore + Modules
- 1250 downloads, 700+ active users from 59 countries

Country	# Users
United States	237
France	150
Switzerland	126
China	101
Germany	77
United Kingdom	71
Italy	47
India	36
Canada	32
Belgium	30

As of April 1st, 2018

UQLab users


Universidad Nacional de Colombia
University of Sheffield
Stanford University
Pontificia Universidad Católica del Perú
Institut Fresnel
University of Zurich
University of Colorado at Boulder
Beijing Institute of Applied Physics and Computational Mathematics
University of Maryland, College Park
Université d'Angers
Sapienza - Università di Roma
Rheinisch Westfälische Technische Hochschule Aachen
Hunan University
Cranfield University
University of Liverpool
Télécom ParisTech
University of Florida
Université de Strasbourg
University of Minnesota
Yeungnam University
University of Crete
Université de Lorraine
University of Surrey
Iowa State University
Ohio State University, Columbus
Chalmers University of Technology
Johns Hopkins University
Tohoku University
Technische Universität Darmstadt
Pontifícia Universidade Católica do Rio de Janeiro
Katholieke Universiteit Leuven
National University of Singapore
Beijing Jiaotong University
Sistan & Balouchestan University
Shanghai Jiao Tong University
University of Science and Technology of China
Arts et Métiers ParisTech
Ecole Supérieure d'Electricité
Tongji University
University of Leeds
Indian Institute of Technology, Delhi
San Mateo County Community College District
Bocconi University
CentraleSupélec
University of Notre Dame
Ruhr-Universität Bochum
European Commission - Joint Research Centre
Max Planck Institute - Magdeburg
Cornell University
Central South University
University of Exeter
Technische Universität Wien
Polytechnic Institute of Turin
Harbin Institute of Technology
Indian Institute of Technology, Bombay
Norwegian University of Science and Technology
Technical University of Denmark
Swansea University
Laboratoire 3SR
Monash University
French Geological Survey
Tsinghua University
University of Twente
Universidad Politécnica de Madrid
Kunsan National University
Polytechnic Institute of Milan
Technical University of Budapest
University of New South Wales
Northwest Polytechnical University Xi'an
University of Ottawa
North Carolina State University
Institut National des Sciences Appliquées de Rouen
Texas A&M University - College Station
University of Michigan - Ann Arbor
Huazhong University of Science and Technology
San Francisco State University
University of Utah
INRIA
King Abdullah University of Science and Technology
Indian Institute of Technology, Madras
Zhejiang University
Beihang University
ETH Zurich
Technische Universität München
Aalborg University
National University of Defense Technology
Virginia Polytechnic Institute and State University
ONERA - Office National d'Etudes et de Recherches Aérospatiales
IRSTEA
Universiteit Gent
Bauhaus Universität Weimar
University of Florence
Lawrence Livermore National Laboratory
Aalto University
Université de Technologie de Compiègne
University of Illinois at Urbana-Champaign
Universidade Federal de Santa Catarina
Southeast University
State University of New York at Buffalo
Universität Stuttgart
Indian Institute of Science
Nanyang Technological University
Universidade Federal do Rio de Janeiro
Université de Nantes
University of California, Davis
Singapore University of Technology and Design
Technische Universität Carolo-Wilhelmina Braunschweig
Delft University of Technology
Royal Melbourne Institute of Technology
Vanderbilt University
Université Libre de Bruxelles
EPFL - EPF Lausanne
University of Bristol
University of Tehran
University of Texas at Austin
Carleton University
SP Technical Research Institute of Sweden
University of Toronto
Third University of Rome
Sharif University of Technology
National Technical University of Athens
Universidad de Los Andes
Ecole Normale Supérieure de Cachan
Technische Universität Braunschweig
Université Claude Bernard (Lyon I)
University of Oxford
Université Paris-Est-Marne-la-Vallée
University of Warwick
Wuhan University
Universidade do Porto
Nanjing University of Aeronautics and Astronautics
Purdue University
Technische Universität Graz
Centrum Wiskunde & Informatica Amsterdam
Universidad Autónoma del Estado de México
University of California, Berkeley
SIGMA Clermont
Technische Universität Chemnitz
Universität der Bundeswehr München
University of Delaware
Eindhoven University of Technology
McMaster University
University of Connecticut
Universidad de Salamanca
Imperial College London
Indian Institute of Technology, Roorkee
University of Calgary
Rice University
Massachusetts Institute of Technology
Universidade de Lisboa

Questions ?

Chair of Risk, Safety & Uncertainty Quantification

www.rsuq.ethz.ch

The Uncertainty Quantification Laboratory

www.uqlab.com

Thank you very much for your attention !