

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/351992197>

# Surrogate models and active learning for reliability analysis

Presentation · May 2021

DOI: 10.13140/RG.2.2.30330.62405

---

CITATIONS  
0

READS  
449

---

3 authors:



Bruno Sudret  
ETH Zurich  
433 PUBLICATIONS 17,320 CITATIONS

[SEE PROFILE](#)



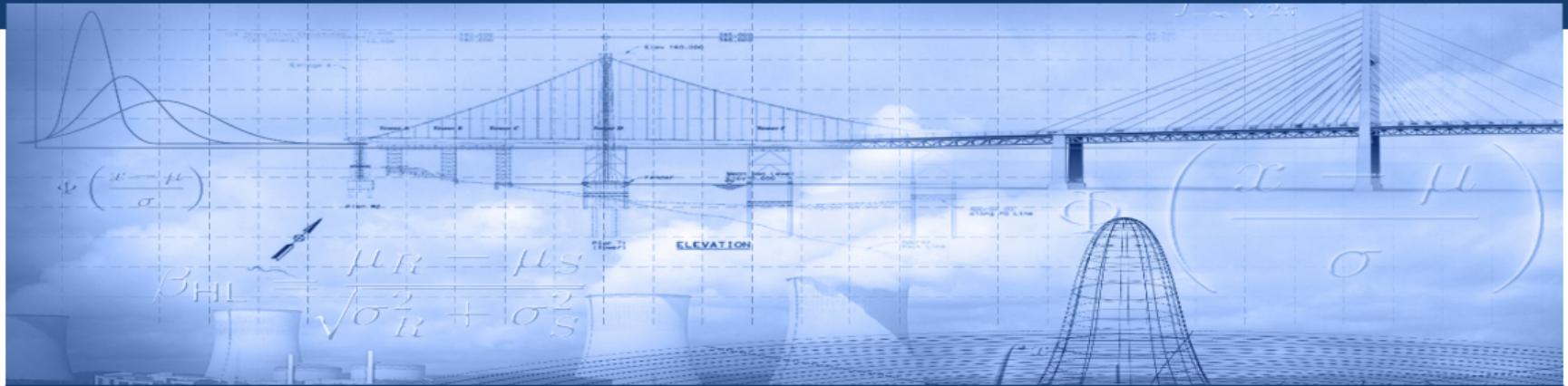
Maliki Moustapha  
ETH Zurich  
37 PUBLICATIONS 666 CITATIONS

[SEE PROFILE](#)



Stefano Marelli  
ETH Zurich  
126 PUBLICATIONS 4,154 CITATIONS

[SEE PROFILE](#)



## Surrogate models and active learning for reliability analysis

B. Sudret, M. Moustapha, S. Marelli

Chair of Risk, Safety and Uncertainty Quantification

## How to cite?

This presentation is an invited lecture given at 13th International Workshop on Rare-Event Simulation on May 19, 2021.

### How to cite

Sudret, B., Moustapha, M. and Marelli, S., *Surrogate models and active learning for reliability analysis*,  
13th International Workshop on Rare-Event Simulation, Paris (France), May 18-21, 2021.

# 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

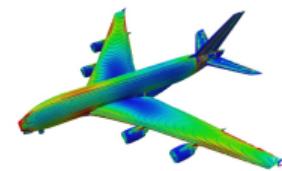
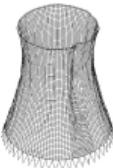


<http://www.rsuq.ethz.ch>

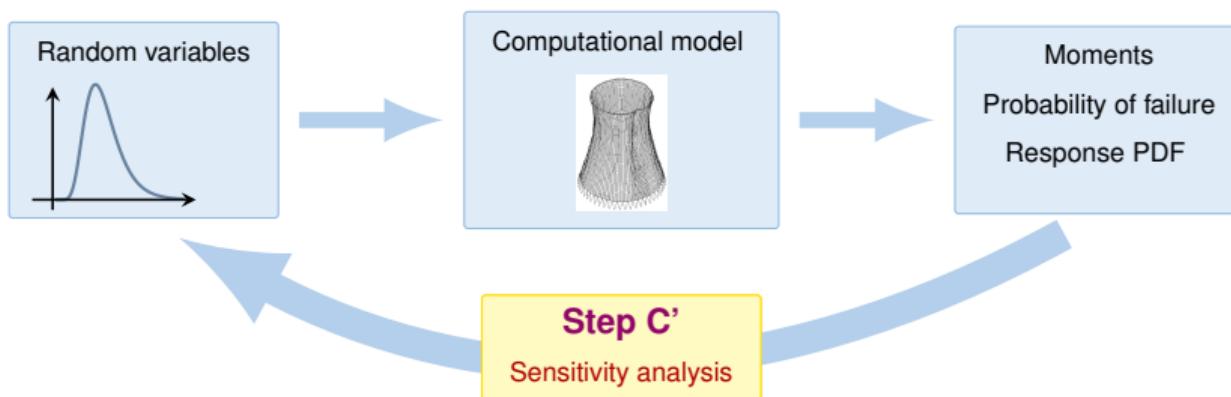
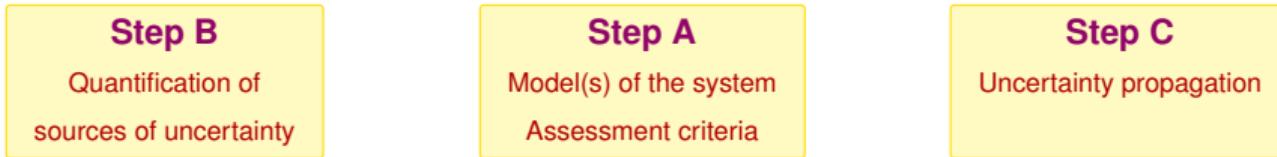
# Computational models in engineering

Complex engineering systems are designed and assessed using **computational models**, a.k.a **simulators** that allows us:

- 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



# Global framework for uncertainty quantification



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

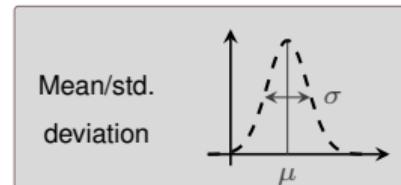
## Step C: uncertainty propagation

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

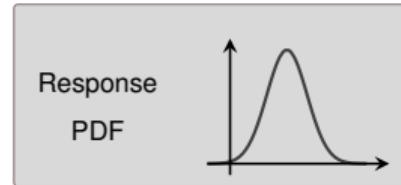
- Output statistics, *i.e.* mean, standard deviation, etc.

$$\mu_Y = \mathbb{E}_{\mathbf{X}} [\mathcal{M}(\mathbf{X})]$$

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

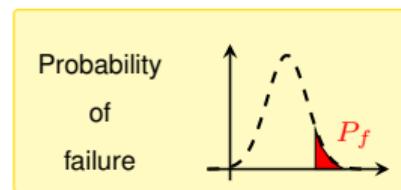


- Distribution of the QoI



- Probability of exceeding an admissible threshold  $y_{adm}$

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



## Limit state function

- For the assessment of the system's performance, failure criteria are defined, e.g. :

$$\text{Failure} \Leftrightarrow QoI = \mathcal{M}(x) \geq q_{adm}$$

Examples:

- + admissible stress / displacements in civil engineering
- + max. temperature in heat transfer problems
- + crack propagation criterion in fracture mechanics

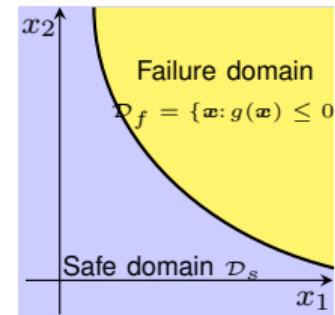
- The failure criterion is cast as a limit state function (performance function)  $g : x \in \mathcal{D}_X \mapsto \mathbb{R}$  such that:

$$g(x, \mathcal{M}(x)) \leq 0 \quad \text{Failure domain } \mathcal{D}_f$$

$$g(x, \mathcal{M}(x)) > 0 \quad \text{Safety domain } \mathcal{D}_s$$

$$g(x, \mathcal{M}(x)) = 0 \quad \text{Limit state surface}$$

e.g.  $g(x) = q_{adm} - \mathcal{M}(x)$

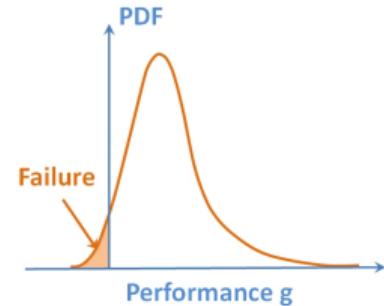


# Probability of failure

## Definition

$$P_f = \mathbb{P} (\{ \mathbf{X} \in D_f \}) = \mathbb{P} (g(\mathbf{X}, \mathcal{M}(\mathbf{X})) \leq 0)$$

$$P_f = \int_{\mathcal{D}_f = \{ \mathbf{x} \in \mathcal{D}_{\mathbf{X}} : g(\mathbf{x}, \mathcal{M}(\mathbf{x})) \leq 0 \}} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}$$



## Features

- Multidimensional integral, whose dimension is equal to the number of basic input variables  $M = \dim \mathbf{X}$
- Implicit domain of integration defined by a condition related to the sign of the limit state function:

$$\mathcal{D}_f = \{ \mathbf{x} \in \mathcal{D}_{\mathbf{X}} : g(\mathbf{x}, \mathcal{M}(\mathbf{x})) \leq 0 \}$$

- Failures are (usually) rare events: sought probability in the range  $10^{-2}$  to  $10^{-8}$

# Classical methods

## Approximation methods

Hasofer & Lind (1974), Rackwitz & Fiessler (1978)

- First-/Second- order reliability method (FORM/SORM)
  - Relatively **inexpensive** semi-analytical methods
  - Convergence is not guaranteed (*e.g.* in presence of multiple failure regions)

## Simulation methods

Melchers (1989), Au & Beck (2001), Koutsourelakis *et al.* (2001)

- Monte Carlo simulation
  - **Unbiased** but **slow** convergence rate
- Variance-reduction methods
  - *e.g.* Importance sampling, subset simulation, line sampling, etc.
  - Their computational costs remain high (*i.e.*  $\mathcal{O}(10^{3-4})$  model runs)

Surrogate models can be used to leverage the computational cost of simulation methods

# Outline

Introduction

Surrogate modelling

- General principles

- Gaussian processes

- Kriging predictor

Active learning for structural reliability

- Principle

- Different enrichment criteria

- General framework and benchmark

## 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:

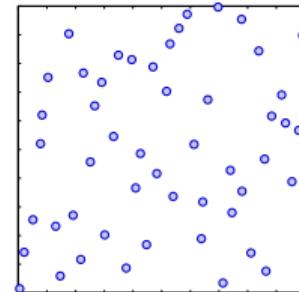
- It is built from a **limited** set of runs of the original model  $\mathcal{M}$  called the **experimental design**  
 $\mathcal{X} = \{\boldsymbol{x}^{(i)}, i = 1, \dots, n\}$
- It assumes some regularity of the model  $\mathcal{M}$  and some general functional shape

Name	Shape	Parameters
Polynomial chaos expansions	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \sum_{\alpha \in \mathcal{A}} a_{\alpha} \Psi_{\alpha}(\boldsymbol{x})$	$a_{\alpha}$
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)	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \boldsymbol{\beta}^T \cdot \boldsymbol{f}(\boldsymbol{x}) + Z(\boldsymbol{x}, \omega)$	$\boldsymbol{\beta}, \sigma_Z^2, \theta$
Support vector machines	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \sum_{i=1}^n a_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b$	$\boldsymbol{a}, b$
(Deep) Neural networks	$\tilde{\mathcal{M}}(\boldsymbol{x}) = f_n(\cdots f_2(b_2 + f_1(b_1 + \boldsymbol{w}_1 \cdot \boldsymbol{x}) \cdot \boldsymbol{w}_2))$	$\boldsymbol{w}, \boldsymbol{b}$

- It is **fast to evaluate**

## 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
<b>Kriging</b>	<b>maximum likelihood, Bayesian inference</b>
Support vector machines	quadratic programming

## Advantages of surrogate models

### Usage

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

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

Introduction

Surrogate modelling

General principles

Gaussian processes

Kriging predictor

Active learning for structural reliability

## Gaussian process modelling

Gaussian process modelling (a.k.a. Kriging) assumes that the map  $y = \mathcal{M}(\boldsymbol{x})$  is a realization of a Gaussian process:

$$Y(\boldsymbol{x}, \omega) = \sum_{j=1}^p \beta_j f_j(\boldsymbol{x}) + \sigma Z(\boldsymbol{x}, \omega)$$

where:

- $\boldsymbol{f} = \{f_j, j = 1, \dots, p\}^\top$  are predefined (e.g. polynomial) functions which form the trend or regression part
- $\boldsymbol{\beta} = \{\beta_1, \dots, \beta_p\}^\top$  are the regression coefficients
- $\sigma^2$  is the variance of  $Y(\boldsymbol{x}, \omega)$
- $Z(\boldsymbol{x}, \omega)$  is a stationary, zero-mean, unit-variance Gaussian process

$$\mathbb{E}[Z(\boldsymbol{x}, \omega)] = 0 \quad \text{Var}[Z(\boldsymbol{x}, \omega)] = 1 \quad \forall \boldsymbol{x} \in \mathbb{X}$$



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

## Assumptions on the trend and the zero-mean process

Prior assumptions are made based on the existing knowledge on the model to surrogate (linearity, smoothness, etc.)

### Trend

- Simple Kriging: known constant  $\beta$
- Ordinary Kriging:  $p = 1$ , unknown constant  $\beta$
- Universal Kriging:  $f_j$ 's is a set of e.g. polynomial functions,  
e.g.  $\{f_j(x) = x^{j-1}, j = 1, \dots, p\}$  in 1D

### Type of auto-correlation function of $Z(x)$

A family of auto-correlation function  $R(\cdot; \theta)$  is selected:

$$\text{Cov} [Z(x), Z(x')] = \sigma^2 R(x, x'; \theta)$$

e.g. square exponential, generalized exponential, Matérn, etc.

$$R_{\text{Matérn}}(h; \nu = 5/2) = (1 + \sqrt{5} h + \frac{5}{3} h^2) \exp(-\sqrt{5} h)$$

# Outline

Introduction

## Surrogate modelling

General principles

Gaussian processes

Kriging predictor

Active learning for structural reliability

# Kriging equations

## Data

- Given is an experimental design  $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$  and the output of the computational model  $\mathbf{y} = \{y_1 = \mathcal{M}(\mathbf{x}_1), \dots, y_N = \mathcal{M}(\mathbf{x}_N)\}$
- We assume that  $\mathcal{M}(\mathbf{x})$  is a realization of a Gaussian process  $Y(\mathbf{x})$  such that the values  $y_i = \mathcal{M}(\mathbf{x}_i)$  are known at the various points  $\{\mathbf{x}_1, \dots, \mathbf{x}_N\}$
- Of interest is the prediction at a new point  $\mathbf{x}_0 \in \mathbb{X}$ , denoted by  $\hat{Y}_0 \equiv \hat{Y}(\mathbf{x}_0, \omega)$ , which will be used as a surrogate  $\tilde{\mathcal{M}}(\mathbf{x}_0)$

$\hat{Y}_0$  is obtained as a conditional Gaussian variable:

$$\hat{Y}_0 = Y(\mathbf{x}_0 | Y(\mathbf{x}_1) = y_1, \dots, Y(\mathbf{x}_N) = y_N)$$

## Joint distribution of the predictor / observations

- For each point  $\mathbf{x}_i \in \mathcal{X}$ ,  $Y_i \equiv Y(\mathbf{x}_i)$  is a Gaussian variable:

$$Y_i = \sum_{j=1}^p \beta_j f_j(\mathbf{x}_i) + \sigma Z_i = \mathbf{f}_i^\top \cdot \boldsymbol{\beta} + \sigma Z_i \quad Z_i \sim \mathcal{N}(0, 1)$$

- The joint distribution of  $\{Y_0, Y_1, \dots, Y_N\}^\top$  is Gaussian:

$$\begin{Bmatrix} Y_0 \\ \vdots \\ Y_N \end{Bmatrix} \sim \mathcal{N}_{1+N} \left( \begin{Bmatrix} \mathbf{f}_0^\top \boldsymbol{\beta} \\ \mathbf{F} \boldsymbol{\beta} \end{Bmatrix}, \sigma^2 \begin{bmatrix} 1 & \mathbf{r}_0^\top \\ \mathbf{r}_0 & \mathbf{R} \end{bmatrix} \right)$$

- Regression matrix  $\mathbf{F}$  of size  $(N \times p)$

$$\mathbf{F}_{ij} = f_j(\mathbf{x}_i)$$

$$i = 1, \dots, N, \quad j = 1, \dots, p$$

- Correlation matrix  $\mathbf{R}$  of size  $(N \times N)$

$$\mathbf{R}_{ij} = R(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta})$$

- Cross-correlation vector  $\mathbf{r}_0$  of size  $N$

$$\mathbf{r}_{0i} = R(\mathbf{x}_i, \mathbf{x}_0; \boldsymbol{\theta})$$

- Vector of regressors  $\mathbf{f}_0$  of size  $p$

$$\mathbf{f}_0 = \{f_1(\mathbf{x}_0), \dots, f_p(\mathbf{x}_0)\}$$

## Kriging mean predictor and variance

Santner, William & Notz (2003)

The conditional distribution of  $\widehat{Y}_0$  given the observations  $\{Y(\mathbf{x}_i) = y_i\}_{i=1}^n$  is a Gaussian variable:

$$\widehat{Y}_0 \sim \mathcal{N}(\mu_{\widehat{Y}_0}, \sigma_{\widehat{Y}_0}^2)$$

Mean predictor : used as surrogate model

$$\mu_{\widehat{Y}_0} = \mathbf{f}_0^\top \widehat{\boldsymbol{\beta}} + \mathbf{r}_0^\top \mathbf{R}^{-1} (\mathbf{y} - \mathbf{F} \widehat{\boldsymbol{\beta}})$$

where the regression coefficients  $\widehat{\boldsymbol{\beta}}$  are obtained from the generalized least-square solution:

$$\widehat{\boldsymbol{\beta}} = (\mathbf{F}^\top \mathbf{R}^{-1} \mathbf{F})^{-1} \mathbf{F}^\top \mathbf{R}^{-1} \mathbf{y}$$

Kriging variance : local prediction uncertainty

$$\sigma_{\widehat{Y}_0}^2 = \mathbb{E} [(\widehat{Y}_0 - Y_0)^2] = \sigma^2 \left( 1 - \mathbf{r}_0^\top \mathbf{R}^{-1} \mathbf{r}_0 + \mathbf{u}_0^\top (\mathbf{F}^\top \mathbf{R}^{-1} \mathbf{F})^{-1} \mathbf{u}_0 \right) \quad \mathbf{u}_0 = \mathbf{F}^\top \mathbf{R}^{-1} \mathbf{r}_0 - \mathbf{f}_0$$

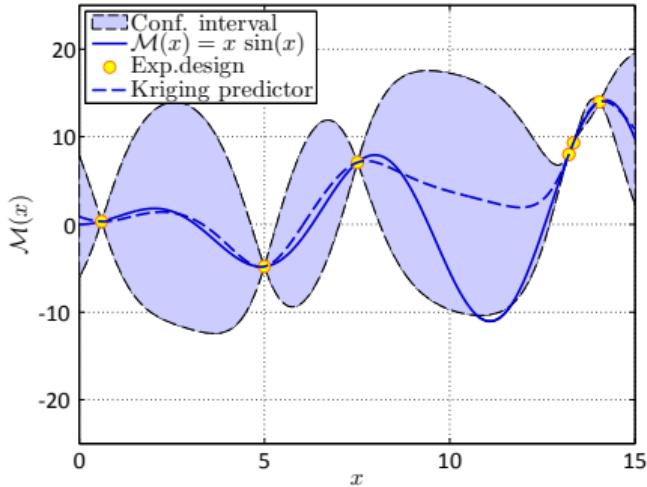
## One-dimensional example

Computational model

$$x \mapsto x \sin x \quad \text{for } x \in [0, 15]$$

Experimental design

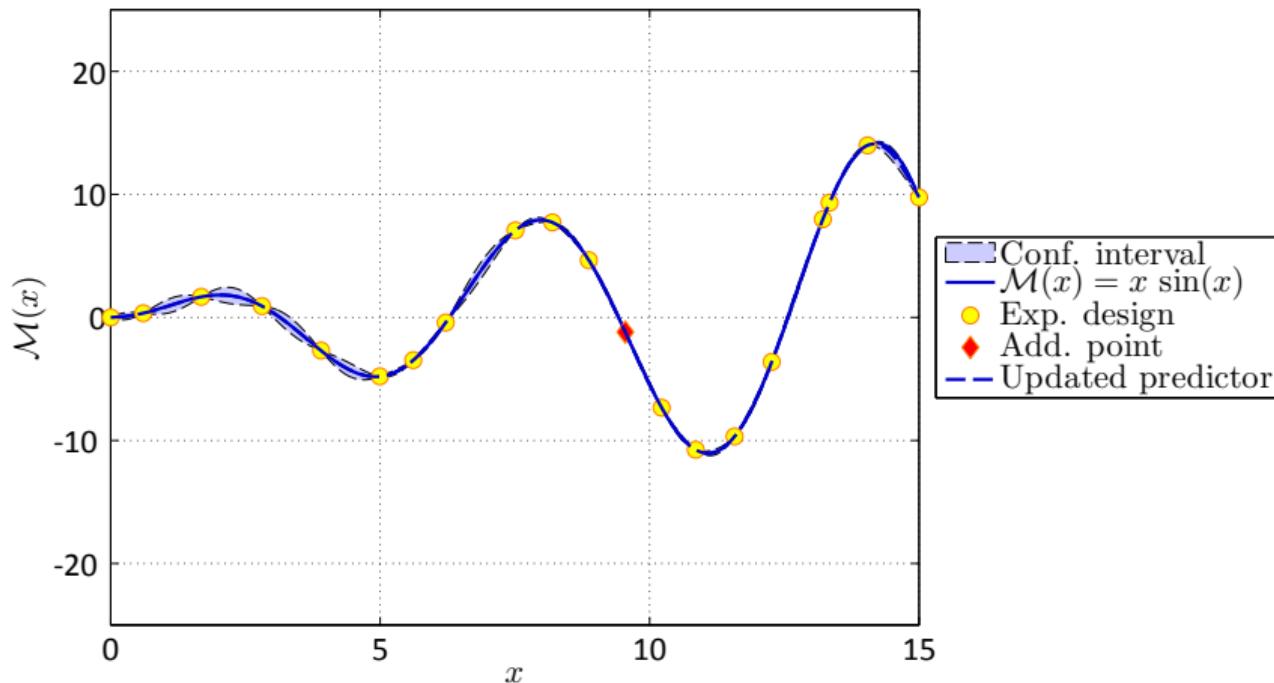
Six points selected in the range  $[0, 15]$  using Monte Carlo simulation



Confidence intervals With confidence level  $(1 - \alpha)$ , e.g. 95%, one gets:

$$\mu_{Y_0}^{\wedge} - 1.96 \sigma_{Y_0}^{\wedge} \leq \mathcal{M}(x_0) \leq \mu_{Y_0}^{\wedge} + 1.96 \sigma_{Y_0}^{\wedge}$$

## Sequential updating



## Outline

Introduction

Surrogate modelling

Active learning for structural reliability

Principle

Different enrichment criteria

General framework and benchmark

## Kriging for reliability analysis: basic approach

- From a given experimental design  $\mathcal{X} = \{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(n)}\}$ , Kriging yields a **mean predictor**  $\mu_{\hat{g}}(\boldsymbol{x})$  and the **Kriging variance**  $\sigma_{\hat{g}}(\boldsymbol{x})$  of the limit state function  $g$
- The mean predictor is **substituted for** the “true” limit state function, defining the **surrogate failure domain**

$$\mathcal{D}_f^0 = \{\boldsymbol{x} \in \mathcal{D}_{\mathbf{X}} : \mu_{\hat{g}}(\boldsymbol{x}) \leq 0\}$$

- The probability of failure is approximated by:

Kaymaz, Struc. Safety (2005)

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

- **Monte Carlo simulation** (resp. subset simulation, etc.) 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:

$$\mathcal{D}_f^- = \left\{ \mathbf{x} \in \mathcal{D}_{\mathbf{X}} : \mu_{\hat{g}}(\mathbf{x}) + k_{1-\alpha} \sigma_{\hat{g}}(\mathbf{x}) \leq 0 \right\}$$

$$\mathcal{D}_f^+ = \left\{ \mathbf{x} \in \mathcal{D}_{\mathbf{X}} : \mu_{\hat{g}}(\mathbf{x}) - k_{1-\alpha} \sigma_{\hat{g}}(\mathbf{x}) \leq 0 \right\}$$

- Interpretation ( $1 - \alpha = 95\%$ ):

- If  $\mathbf{x} \in \mathcal{D}_f^0$  it belongs to the true failure domain with a 50% chance
- If  $\mathbf{x} \in \mathcal{D}_f^+$  it belongs to the true failure domain with 95% chance: **conservative estimation**

## Bounds on the probability of failure

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

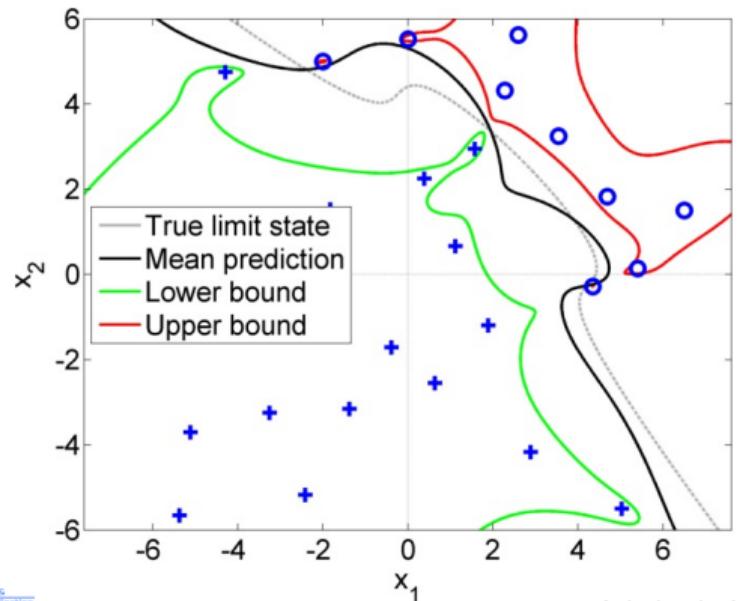
See also Picheny *et al.* (2010, 2013), Chevalier & Ginsbourger (2014), work on excursion sets by Azzimonti *et al.* (2016)

## Example: hat function

### Problem statement

$$g(\mathbf{x}) = 20 - (x_1 - x_2)^2 - 8(x_1 + x_2 - 4)^3$$

where  $X_1, X_2 \sim \mathcal{N}(0, 1)$



- Ref. solution:

$$P_f = 1.07 \cdot 10^{-4}$$

- Kriging surrogate:

$$P_f^- = 7.70 \cdot 10^{-6}$$

$$P_f^0 = 4.43 \cdot 10^{-4}$$

$$P_f^+ = 5.52 \cdot 10^{-2}$$

# How to improve the results?

## Heuristics

- The Monte Carlo estimate of  $P_f$  reads:

$$\widehat{P}_f = \frac{1}{N} \sum_{k=1}^N \mathbf{1}_{\mathcal{D}_f}(x_k) \approx \frac{1}{N} \sum_{k=1}^N \mathbf{1}_{\mathcal{D}_f^0}(x_k)$$

- The Kriging-based prediction is accurate when:

$$\mathbf{1}_{\mathcal{D}_f^0}(x_k) = \mathbf{1}_{\mathcal{D}_f}(x_k) \quad \text{for almost all } x_k$$

i.e. if  $\mu_{\hat{g}}(x)$  is **of the same sign** as  $g(x)$  for almost all sample points

Ensure that the mean predictor  $\mu_{\hat{g}}(x)$  **classifies** properly the MCS samples according to the sign of  $g(x)$

# Active learning reliability using a Kriging surrogate

## Procedure

- Start from an initial experimental design  $\mathcal{X}$  and build the initial Kriging surrogate
- At each iteration:
  - Compute an estimation of  $P_f$  and bounds from the current surrogate
  - Check a convergence criterion
  - Select the next point(s) to be added to  $\mathcal{X}$ : enrichment (a.k.a. in-fill) criterion
  - Update the Kriging surrogate

## Early approaches

- Efficient global reliability analysis (EGRA) Bichon *et al.* (2008)
- Active Kriging - Monte Carlo simulation (AK-MCS) Echard *et al.* (2011)
- Both use the same reliability estimation algorithm and surrogate model
- They introduce their own **learning functions** for enrichment

## Outline

Introduction

Surrogate modelling

**Active learning for structural reliability**

Principle

Different enrichment criteria

General framework and benchmark

## Different enrichment criteria

### Requirements

- It shall be based on the available information:  $(\mu_{\hat{g}}(x), \sigma_{\hat{g}}(x))$
- It shall favor new points in the vicinity of the limit state surface
- If possible, it shall yield the best  $K$  points when distributed computing is available

### Different enrichment criteria

- Margin indicator function Ph.D Deheeger (2008); Bourinet *et al.*, Struc. Safety (2011)
- Margin classification function Ph.D Dubourg (2011); Dubourg *et al.*, PEM (2013)
- Learning function  $U$  Ph.D Échard (2012); Échard & Gayton, RESS (2011)
- Expected feasibility function Bichon *et al.*, AIAA (2008); RESS (2011)
- Stepwise uncertainty reduction (SUR) Bect *et al.*, Stat. Comput. (2012)
- ... more on slide 35!

# Learning function $U(x)$

## Definition

- The learning function  $U$  is defined by:

Échard *et al.* (2011)

$$U(x) = \frac{|\mu_{\hat{g}}(x)|}{\sigma_{\hat{g}}(x)}$$

## Interpretation

- It describes the distance of the mean predictor  $\mu_{\hat{g}}$  to zero in terms of a number of Kriging standard deviations  $\sigma_{\hat{g}}$
- A small value of  $U(x)$  means that:
  - $\mu_{\hat{g}}(x) \approx 0$ :  $x$  is close to the limit state surface
  - and / or  $\sigma_{\hat{g}}(x) >> 0$ : the uncertainty in the prediction at point  $x$  is large
- The probability of misclassification of a point  $x$  is equal to  $\Phi(-U(x))$

Bect *et al.*, Stat. Comput. (2012)

# Expected feasibility function

## Heuristics

Bichon et al., AIAA (2008); Bichon et al., RESS (2011)

The **feasibility function**  $FF(\mathbf{x})$  describes the distance from the current point  $\mathbf{x}$  to the limit state surface

- In each point  $\mathbf{x}$ , one checks if the predictor is sufficiently close to zero: if  $|\hat{Y}(\mathbf{x})| > \varepsilon$ ,  $FF(\mathbf{x}) = 0$
- Otherwise,  $FF(\mathbf{x})$  is the distance to the boundary of the tube  $[-\varepsilon, \varepsilon]$ :

$$FF(\mathbf{x}) = \begin{cases} 0 & \text{if } \hat{Y}(\mathbf{x}) \notin [-\varepsilon, \varepsilon] \\ \varepsilon - |\hat{Y}(\mathbf{x})| & \text{otherwise} \end{cases} = \max(\varepsilon - |\hat{Y}(\mathbf{x})|, 0)$$

- As  $\hat{Y}(\mathbf{x})$  is a Gaussian random variable, the **expected feasibility function** is obtained by:

$$EFF(\mathbf{x}) = \mathbb{E}_{GP} \left[ \max(\varepsilon - |\hat{Y}(\mathbf{x})|, 0) \right] = \int_{\mathbb{R}} \max(\varepsilon - |z|, 0) f_{\hat{Y}(\mathbf{x})}(z) dz$$

where  $\mathbb{E}_{GP} [\cdot]$  is the expectation with respect to the **Gaussian measure associated with Kriging**

## Expected feasibility function

$$EFF(\mathbf{x}) = \int_{-\varepsilon}^{\varepsilon} (\varepsilon - |z|) f_{\hat{Y}(\mathbf{x})}(z) dz \quad f_{\hat{Y}(\mathbf{x})}(z) = \frac{1}{\sigma_{\hat{g}}(\mathbf{x})} \varphi \left( \frac{z - \mu_{\hat{g}}(\mathbf{x})}{\sigma_{\hat{g}}(\mathbf{x})} \right)$$

The **expected feasibility function** finally reads:

$$\begin{aligned} EFF(\mathbf{x}) &= \mu_{\hat{g}}(\mathbf{x}) \left[ 2\Phi \left( -\frac{\mu_{\hat{g}}(\mathbf{x})}{\sigma_{\hat{g}}(\mathbf{x})} \right) - \Phi \left( \frac{-\varepsilon - \mu_{\hat{g}}(\mathbf{x})}{\sigma_{\hat{g}}(\mathbf{x})} \right) - \Phi \left( \frac{\varepsilon - \mu_{\hat{g}}(\mathbf{x})}{\sigma_{\hat{g}}(\mathbf{x})} \right) \right] \\ &\quad - \sigma_{\hat{g}}(\mathbf{x}) \left[ 2\varphi \left( -\frac{\mu_{\hat{g}}(\mathbf{x})}{\sigma_{\hat{g}}(\mathbf{x})} \right) - \varphi \left( \frac{-\varepsilon - \mu_{\hat{g}}(\mathbf{x})}{\sigma_{\hat{g}}(\mathbf{x})} \right) - \varphi \left( \frac{\varepsilon - \mu_{\hat{g}}(\mathbf{x})}{\sigma_{\hat{g}}(\mathbf{x})} \right) \right] \\ &\quad + \varepsilon \left[ \Phi \left( \frac{\varepsilon - \mu_{\hat{g}}(\mathbf{x})}{\sigma_{\hat{g}}(\mathbf{x})} \right) - \Phi \left( \frac{-\varepsilon - \mu_{\hat{g}}(\mathbf{x})}{\sigma_{\hat{g}}(\mathbf{x})} \right) \right] \end{aligned}$$

NB: Usually a local value of  $\varepsilon$  is used in each point  $x$ , e.g.  $\varepsilon(x) = 2\sigma_{\hat{g}}(x)$

## Selection of the new ED points

Optimization of the enrichment criterion

$$\mathbf{x}_{EFF}^* = \arg \max_{\mathbf{x} \in \mathcal{D}_X} EFF(\mathbf{x})$$

$$\mathbf{x}_U^* = \arg \min_{\mathbf{x} \in \mathcal{D}_X} U(\mathbf{x})$$

Requires to solve a complex optimization problem in each iteration

Discrete optimization over a **large** Monte Carlo sample  $\mathfrak{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$

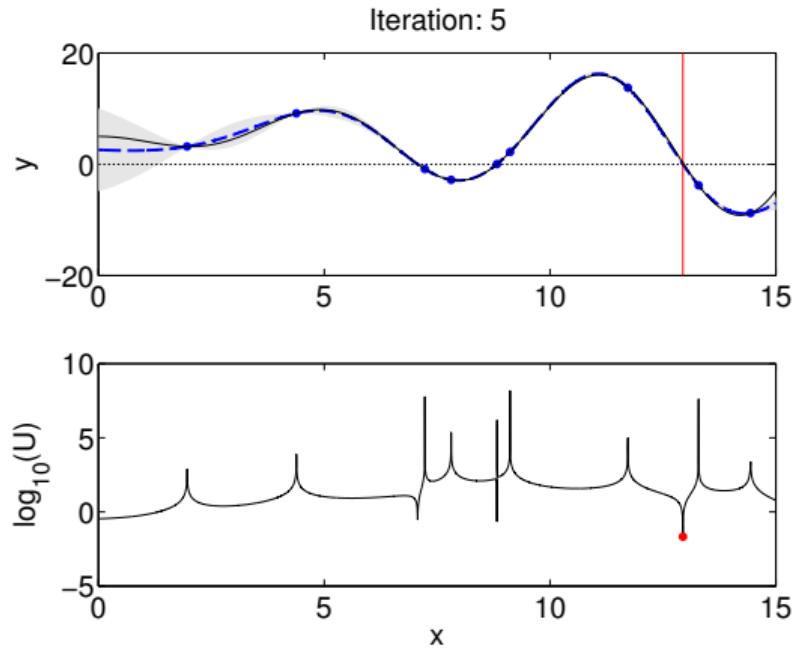
$$\mathbf{x}_{EFF}^* = \arg \max_{i=1, \dots, n} \{EFF(\mathbf{x}_1), \dots, EFF(\mathbf{x}_n)\}$$

$$\mathbf{x}_U^* = \arg \min_{i=1, \dots, n} \{U(\mathbf{x}_1), \dots, U(\mathbf{x}_n)\}$$

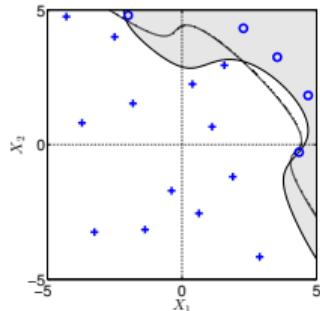
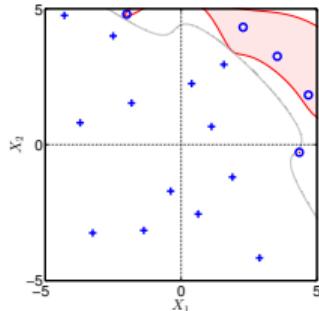
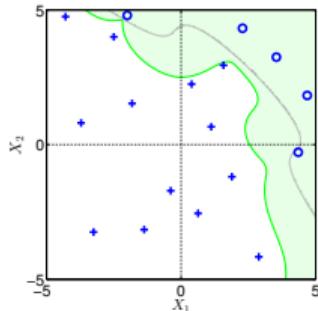
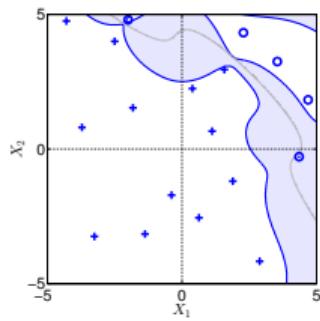
"AKMCS", Échard *et al.* (2011)

## 1D Application example - $U$ function

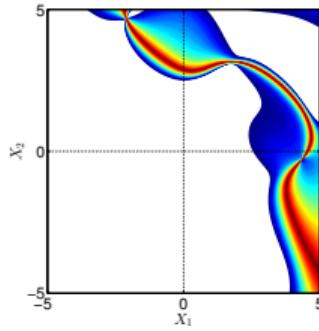
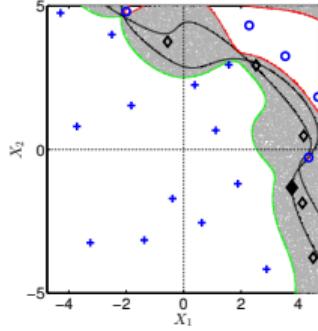
Limit state function:  $g(x) = 5 - x \sin x$



## Example: hat function

Mean  $D_f^0$ Lower bound  $D_f^-$ Upper bound  $D_f^+$ 

Limit-state margin

Probability of misclassification  
Active learning for reliability

Additional samples

RESIM2021 – May 19, 2021

B. Sudret

30 / 50

# PC-Kriging

Schöbi & Sudret, IJUQ (2015); Kersaudy *et al.*, J. Comp. Phys (2015), Schöbi & Sudret, ASCE/ASME JRUEng (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

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

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

# Series system

Schöbi et al., ASCE J. Risk Unc. (2016)

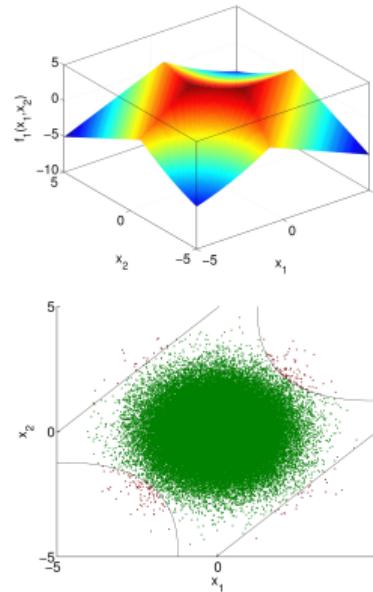
Consider the system reliability analysis defined by:

$$g(\mathbf{x}) = \min \left( \begin{array}{l} 3 + 0.1(x_1 - x_2)^2 - \frac{x_1 + x_2}{\sqrt{2}} \\ 3 + 0.1(x_1 - x_2)^2 + \frac{x_1 + x_2}{\sqrt{2}} \\ (x_1 - x_2) + \frac{6}{\sqrt{2}} \\ (x_2 - x_1) + \frac{6}{\sqrt{2}} \end{array} \right)$$

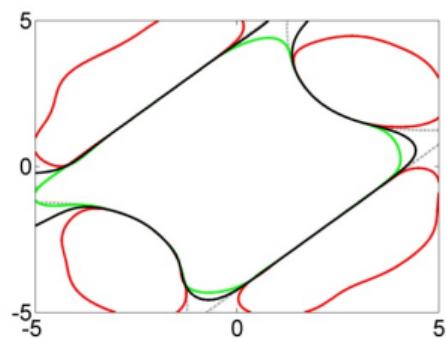
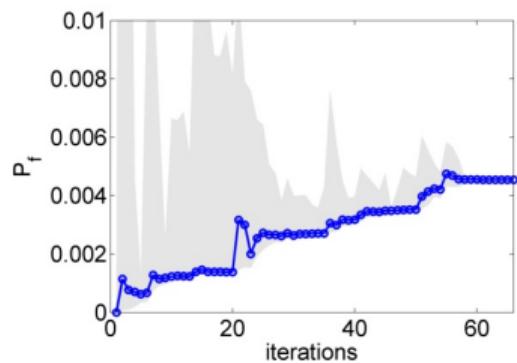
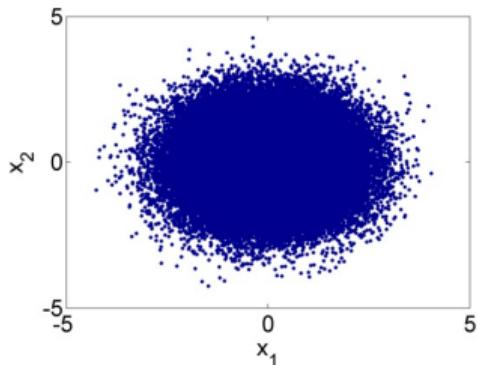
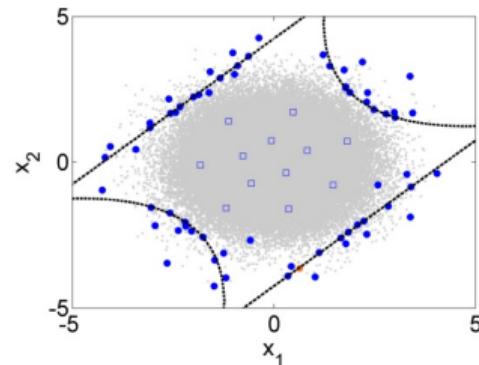
where  $X_1, X_2 \sim \mathcal{N}(0, 1)$

- Initial design: LHS of size 12 (transformed into the standard normal space)
- In each iteration, one point is added (maximize the probability of missclassification)
- The mean predictor  $\hat{\mu}_{\mathcal{M}}(\mathbf{x})$  is used, as well as the bounds  $\hat{\mu}_{\mathcal{M}}(\mathbf{x}) \pm 2\sigma_{\mathcal{M}}(\mathbf{x})$  so as to get bounds on  $P_f$ :  

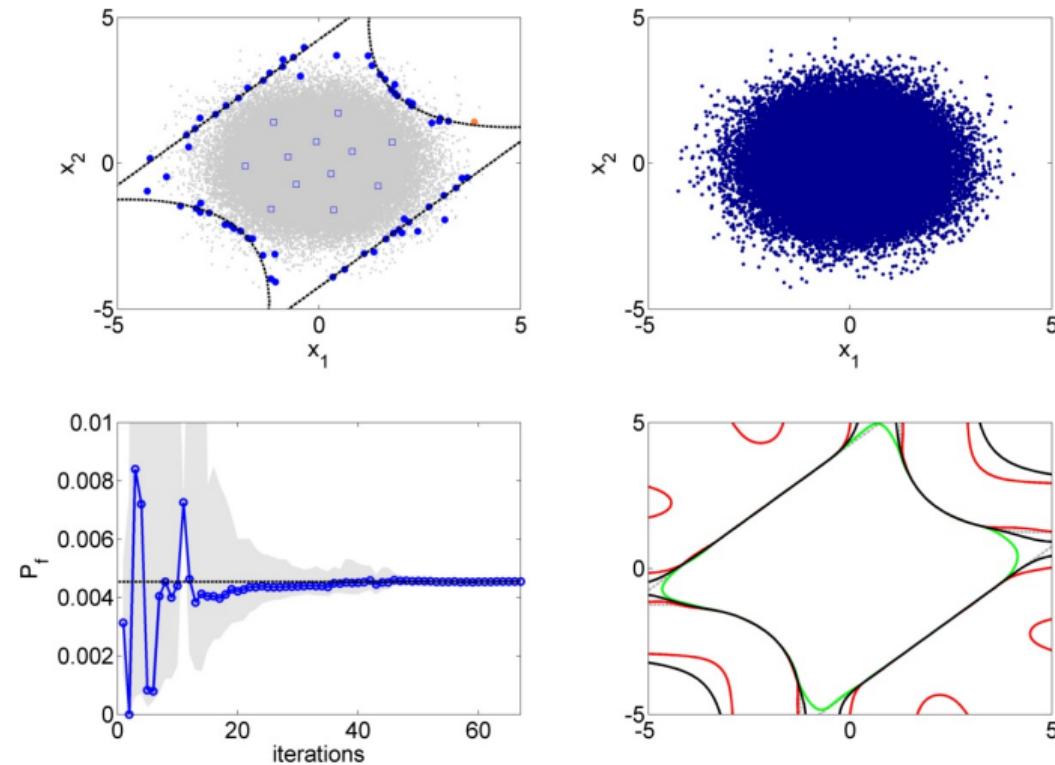
$$\hat{P}_f^- \leq \hat{P}_f^0 \leq \hat{P}_f^+$$



## Results with classical Kriging



## Results with PC Kriging



# Outline

Introduction

Surrogate modelling

Active learning for structural reliability

Principle

Different enrichment criteria

General framework and benchmark

# Active learning reliability methods

Teixeira et al. (2021), Moustapha et al. (2021) (submitted)

Numerous papers on active learning in the last few years!

- AK-MCS is a cornerstone for the development of active learning reliability strategies
- Most methods in the literature are built by modifying:
  - the surrogate model
  - the learning function
  - the algorithm for reliability estimation
  - the stopping criterion



# A module-oriented survey

Moustapha et al. (2021) (submitted)

	Monte Carlo simulation	Subset simulation	Importance sampling	Other
Kriging	Bichon et. al (2008) Echard et. al (2011) Hu & Mahadevan (2016) Wen et al. (2016) ) Fauriat & Gayton (2017) Jian et. al (2017) Peijuan et al. (2017) Sun et al. (2017) Lelievre et al. (2018) Xiao et al. (2018) Jiang et al. (2019) Tong et al. (2019) Wang & Shafeezadeh (2019) Wang & Shafeezadeh (SAMO, 2019) Zhang, Wang et al. (2019)	Huang et al. (2016) Tong et al. (2015) Ling et al. (2019) Zhang et al. (2019)	Dubourg et al. (2012) Balesdent et al. (2013) Echard et al. (2013) Cadini et al. (2014) Liu et al. (2015) Zhao et al. (2015) Gaspar et al. (2017) Razaaly et al. (2018) Yang et al. (2018) Zhang & Taflanidis (2018) Pan et al. (2020) Zhang et al. (2020)	Lv et al. (2015) Bo & HuiFeng (2018) Guo et al. (2020)
PCE	Chang & Lu (2020) Marelli & Sudret (2018) Pan et al. (2020)			
SVM	Basudhar & Missoum (2013) Lacaze & Missoum (2014) Pan et al. (2017)	Bourinet et al. (2011) Bourinet (2017)		
RSM/RBF	Li et al. (2018) Shi et al. (2019)			Rajakeshir (1993) Rous-souly et al. (2013)
Neural networks	Chojazyk et al. (2015) Gomes et al. (2019) Li & Wang (2020) [Deep NN]	Sundar & Shields (2016)	Chojazyk et al. (2015)	
Other	Schoebi & Sudret (2016) Sadoughi et al. (2017) Wagner et al. (2021)			

— U — EFF — Other variance-based — Distance-based — Bootstrap-based — Sensitivity-based — Cross-validation/Ensemble-based — ad-hoc/other

## General framework

Modular framework which consists of independent blocks that can be assembled in a black-box fashion

Surrogate model	Reliability estimation	Learning function	Stopping criterion
Kriging	Monte Carlo	U	LF-based
PCE	Subset simulation	EFF	Stability of $\beta$
SVR	Importance sampling	FBR	Stability of $P_f$
PC-Kriging	Line sampling	CMM	Bounds on $\beta$
Neural networks	Directional sampling	SUR	Bounds on $P_f$
...	...	...	...

# Active learning for reliability analysis

---

```
1: Initialization
2:   Initial experimental design  $\mathcal{ED} = \{\boldsymbol{\chi}^{(1)}, \dots, \boldsymbol{\chi}^{(n)}\}$ 
3:   Converged = FALSE
4:   while not(Converged) do
5:     Train a surrogate model  $\tilde{g}$  on the current experimental design
6:     Compute the failure probability  $\hat{P}_f^0$ , and its bounds  $[\hat{P}_f^-, \hat{P}_f^+]$  using  $\tilde{g}$ 
7:     if Stopping criterion fulfilled then
8:       Converged = TRUE
9:     else
10:      Evaluate the learning function  $LF$  on  $\mathcal{X}$ 
11:      Enrich the ED:  $\boldsymbol{\chi}^* = \arg \min_{\boldsymbol{x} \in \mathcal{X}} LF(\boldsymbol{x})$ 
12:      Update the experimental design:  $\mathcal{ED} \leftarrow \mathcal{ED} \cup \{\boldsymbol{\chi}^*\}$ 
13:    end
14:  end
15:  Return Probability of failure  $\hat{P}_f^0$  and confidence interval  $[\hat{P}_f^-, \hat{P}_f^+]$ 
```

---

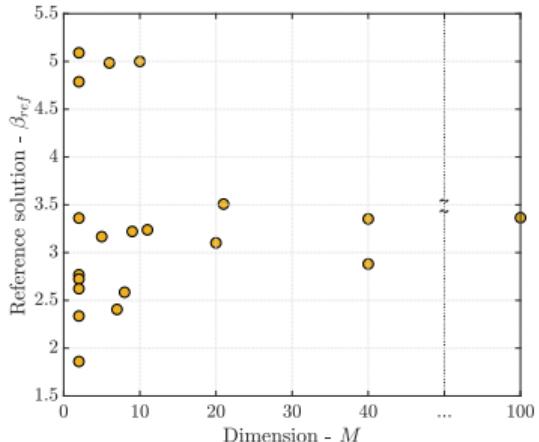
## Extensive benchmark: Set-up

Reliability method	Surrogate model	Learning function	Stopping criterion	
Monte Carlo simulation	Kriging	U	Beta bounds	
Subset simulation	PC-Kriging	EFF	Beta stability	$3 \cdot 2 \cdot 2 \cdot 3 = 36 \text{ strategies}$
Importance sampling			Combined	
Monte Carlo simulation				
Subset simulation	PCE	FBR	Beta stability	<b>3 strategies</b>
Importance sampling				
Subset simulation, Importance sampling w/o metamodel				<b>2 strategies</b>

In total  $39 + 2 = 41$  strategies are tested

## Selected problems

- 20 problems selected from the literature
- 11 come from the TNO benchmark  
(<https://rprepo.readthedocs.io/en/latest/>)
- Wide spectrum of problems in terms of
  - Dimensionality
  - Reliability index  $\beta = -\Phi^{-1}(P_f)$



Problem	$M$	$P_{f, \text{ref}}$	Reference
01 (TNO RP14)	5	$7.69 \cdot 10^{-4}$	Rozsas & Slobbe 2019
02 (TNO RP24)	2	$2.90 \cdot 10^{-3}$	Rozsas & Slobbe 2019
03 (TNO RP28)	2	$1.31 \cdot 10^{-7}$	Rozsas & Slobbe 2019
04 (TNO RP31)	2	$3.20 \cdot 10^{-3}$	Rozsas & Slobbe 2019
05 (TNO RP38)	7	$8.20 \cdot 10^{-3}$	Rozsas & Slobbe 2019
06 (TNO RP53)	2	$3.14 \cdot 10^{-2}$	Rozsas & Slobbe 2019
07 (TNO RP54)	20	$9.79 \cdot 10^{-4}$	Rozsas & Slobbe 2019
08 (TNO RP63)	100	$3.77 \cdot 10^{-4}$	Rozsas & Slobbe 2019
09 (TNO RP7)	2	$9.80 \cdot 10^{-3}$	Rozsas & Slobbe 2019
10 (TNO RP107)	10	$2.85 \cdot 10^{-7}$	Rozsas & Slobbe 2019
11 (TNO RP111)	2	$7.83 \cdot 10^{-7}$	Rozsas & Slobbe 2019
12 (4-branch series)	2	$3.85 \cdot 10^{-4}$	Echard et al. (2011)
13 (Hat function)	2	$4.40 \cdot 10^{-3}$	Schoebi et al. (2016)
14 (Damped oscillator)	8	$4.80 \cdot 10^{-3}$	Der Kiureghian (1990)
15 (Non-linear oscillator)	6	$3.47 \cdot 10^{-7}$	Echard et al. (2011,2013)
16 (Frame)	21	$2.25 \cdot 10^{-4}$	Echard et al. (2013)
17 (HD function)	40	$2.00 \cdot 10^{-3}$	Sadoughi et al. (2017)
18 (VNL function)	40	$1.40 \cdot 10^{-3}$	Bichon et al. (2008)
19 (Transmission tower 1)	11	$5.76 \cdot 10^{-4}$	FEM (172 bars, 51 nodes)
20 (Transmission tower 2)	9	$6.27 \cdot 10^{-4}$	FEM (172 bars, 51 nodes)

## Comparison of the various strategies

Approximately 12,000 reliability analyses were run:  
41 strategies - 20 problems - 15 replications

Three evaluation criteria:

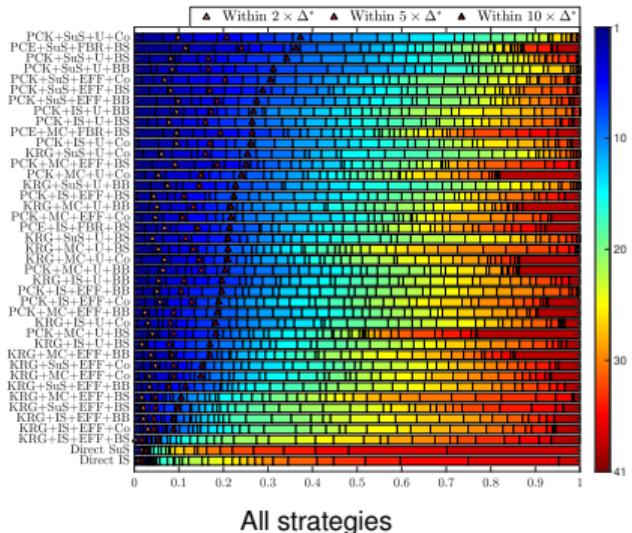
- Number of model evaluations:  $N_{\text{eval}}$
- Accuracy:  $\varepsilon = |\beta - \beta_{\text{ref}}| / \beta_{\text{ref}}$
- Efficiency:  $\Delta = \varepsilon N_{\text{eval}} / N_{\text{med}}$   
where  $N_{\text{med}}$  is the median number of model evaluations for each problem

For each criterion:

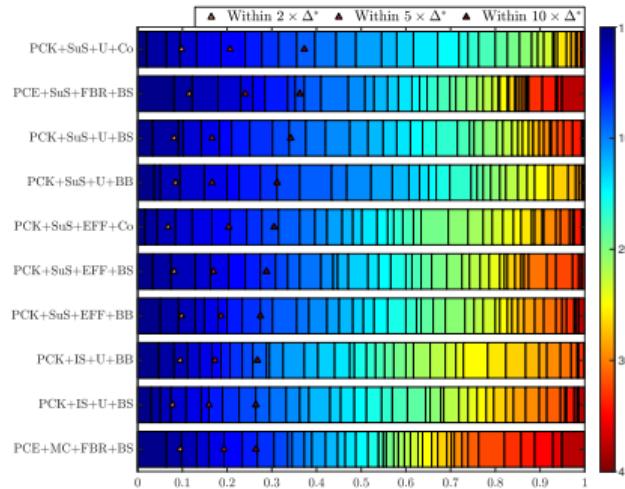
- Ranking of the strategies as a whole
- Ranking of the methods within each block
- Performance of the methods w.r.t. problem feature (dimensionality, range of  $P_f$ )

# Ranking of the strategies

Percentage of times a strategy is ranked 1st, 2nd, ..., 41st w.r.t.  $\Delta$



All strategies

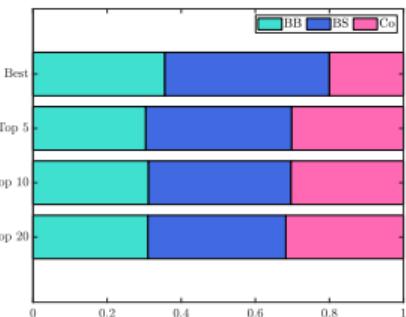
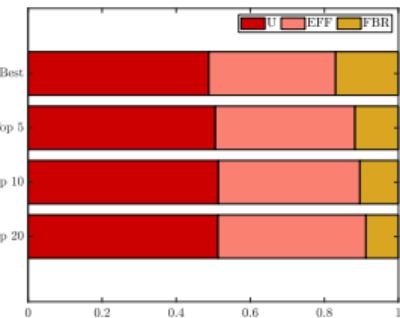
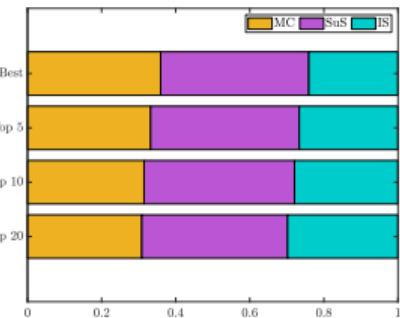
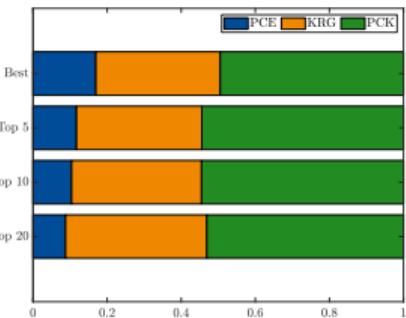


Best 10 strategies

- Best approach: PC-Kriging + SuS + U + Combined stopping criterion
- Worst approaches: Direct SuS and Direct IS

## Results aggregated by method

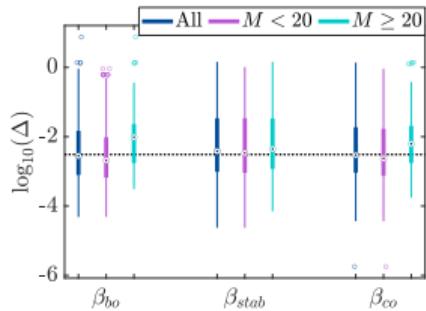
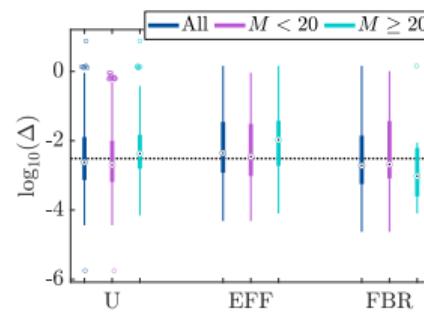
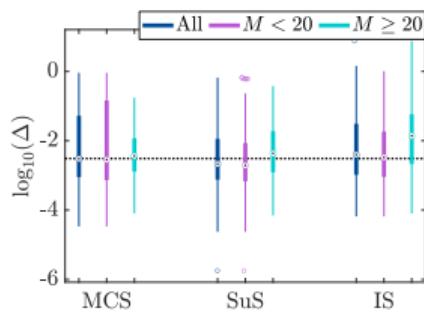
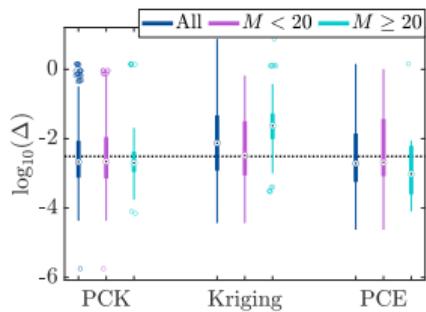
Percentage of times a method is first or in the Top 5, 10, 20 w.r.t.  $\Delta$  (regardless of the strategy)



- Surrogates: PC-Kriging dominates by far
- Reliability: Slight advantage to subset simulation
- Learning function:  $U$  dominates both  $EFF$  and  $FBR$
- Stopping criterion: Slight advantage to the stability criterion

## Influence of the dimension

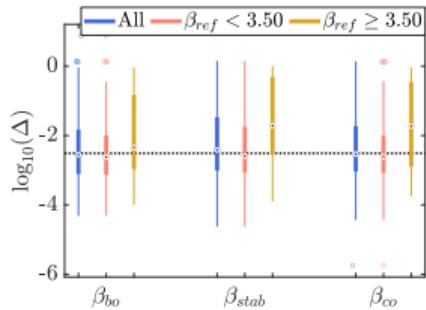
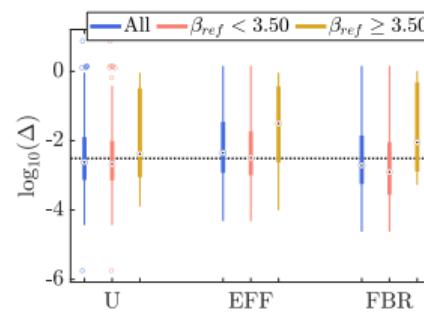
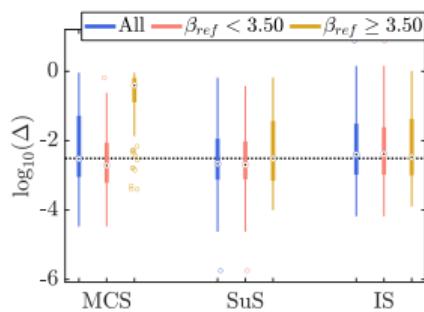
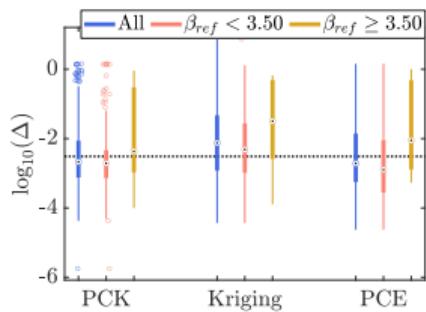
Results aggregated w.r.t. to dimension ( $M < 20$  v.s.  $20 \leq M \leq 100$ )



- Kriging performs worse in large dimension but not PCK and PCE
- As expected MCS is insensitive to dimension, SuS and especially IS have worse performance for  $M \geq 20$
- The learning functions reproduce the performance of the surrogate models
- The  $\beta$  bound criterion which is based on the Kriging variance performs poorly in high dimension

# Influence of the target reliability index

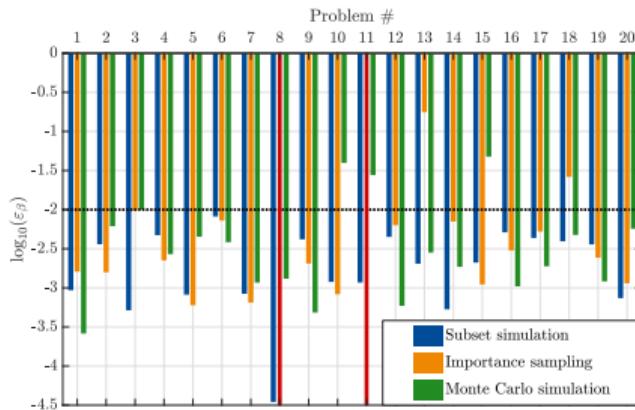
Results aggregated w.r.t. to dimension ( $\beta < 3.5$  vs.  $\beta \geq 3.5$ )



- PC-Kriging is not so much sensitive to  $\beta_{ref}$  while Kriging and PCE perform worse for larger values of  $\beta_{ref}$
- SuS and IS are not so much affected by  $\beta_{ref}$  compared to MCS
- EFF performs worse than U for larger values of  $\beta_{ref}$
- $\beta$ -stability and combined criteria have a noticeably poorer performance for larger values of  $\beta_{ref}$

## Results without surrogates

Relative error for overkill reliability methods without surrogates



- Red lines correspond to non-convergence
- The problems that could not be solved with ALR were not solved by a direct approach (*i.e.*, without surrogate) either

In most cases, the surrogate model was not the cause of failure of the ALR strategy, but rather the reliability estimation algorithm

## Summary of the results

Recommendations w.r.t. the problem feature

Module	Dimensionality		Magnitude of the reliability index	
	$M < 20$	$20 \leq M \leq 100$	$\beta < 3.5$	$\beta \geq 3.5$
Surrogate model	PCK	PCE	PCE/PCK	PCK
Reliability method	SuS	SuS	SuS	SuS
Learning function	U	FBR	U/FBR	U
Stopping criterion	$\beta_{bo}, \beta_{co}$	$\beta_{bo} / \beta_{co}$	$\beta_{bo}, \beta_{co}$	$\beta_{bo}$

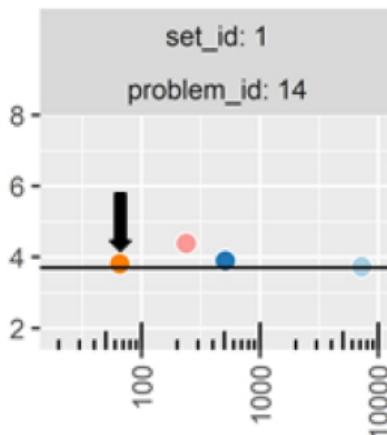
Main take-away

There is no drawback in using surrogates compared to a direct solution

# TNO Benchmark: performance of UQLab “ALR” module

Rozsas &amp; Slobbe (2019)

- Truly black-box benchmark with 27 problems
- Limit state functions not known to the participants and only accessible through an anonymous server
- Our solution: the “best approach” previously highlighted (PCK + SuS + U + Co)



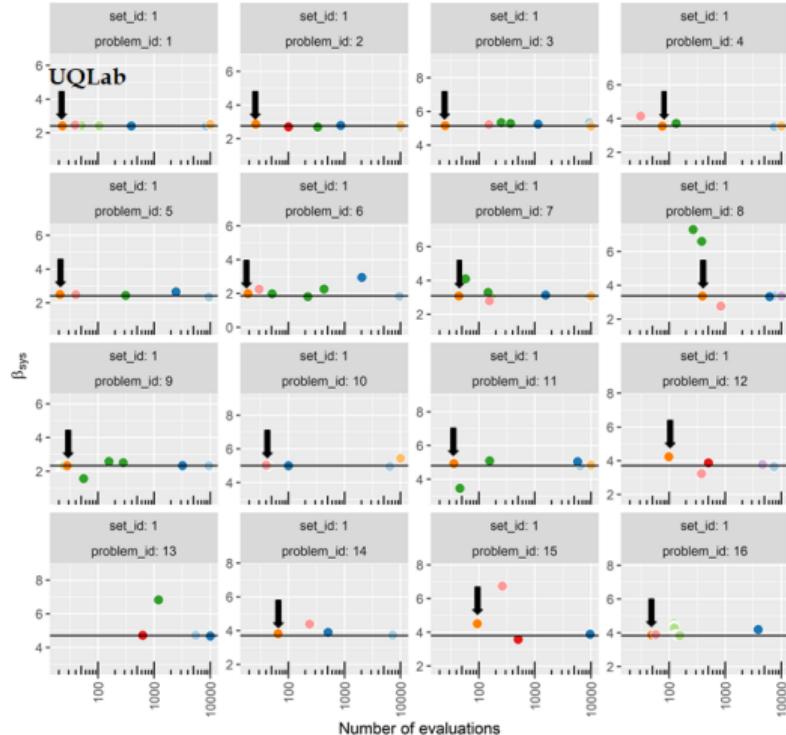
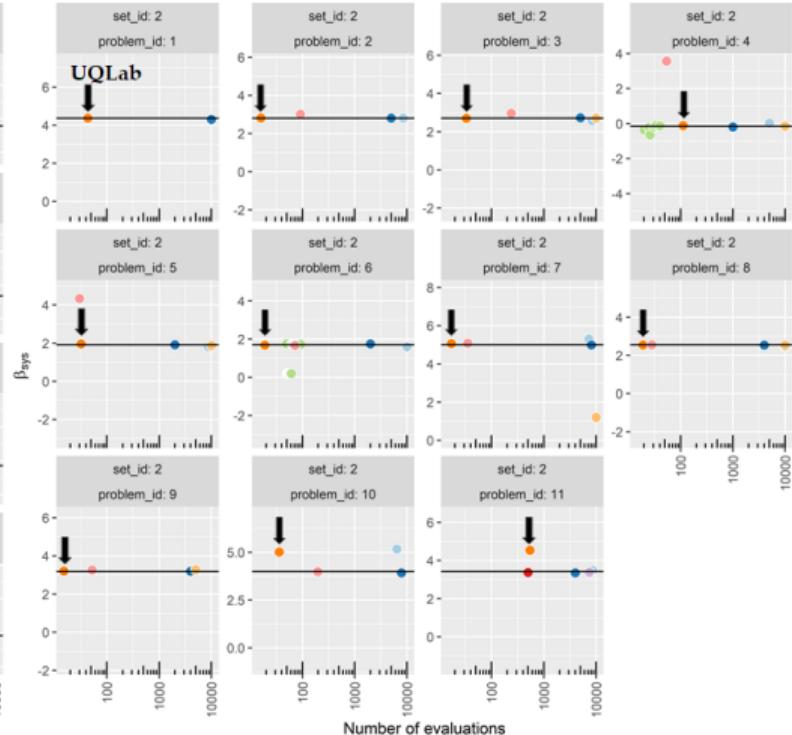
Summary plot (TNO)

- Reference solution: black line
- Zero, one or more points per participant
- X: number of runs (log scale)
- Y: obtained  $\beta$  index

best approach: “on the line / to the left”

# TNO Benchmark: performance of UQLab “ALR” module

Rozsas &amp; Slobbe (2019)

**Component reliability****System reliability**

## Conclusions

- Estimating low probabilities of failure in high-dimensional problems requires more refined algorithms than plain MCS
- Recent research on surrogate models (e.g. Kriging and polynomial chaos expansions) and **active learning** has brought new extremely efficient algorithms
- Accurate estimations of  $P_f$ 's (not of  $\beta$  !) are obtained with  $\mathcal{O}(100)$  runs of the computer code independently of their magnitude
- All the presented algorithms are available in the general-purpose **uncertainty quantification software UQLab** (V.1.4, "Active learning reliability" module)

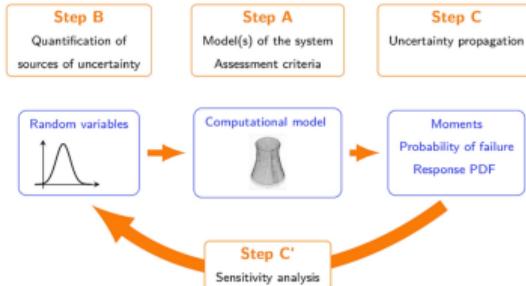
# UQLab

The Framework for Uncertainty Quantification



OVERVIEW   FEATURES   DOCUMENTATION   DOWNLOAD/INSTALL   ABOUT   COMMUNITY

"Make uncertainty quantification available for anybody,  
in any field of applied science and engineering"



[www.uqlab.com](http://www.uqlab.com)

- MATLAB®-based Uncertainty Quantification framework
- State-of-the art, highly optimized open source algorithms
- Fast learning curve for beginners
- Modular structure, easy to extend
- Exhaustive documentation

## UQLab: The Uncertainty Quantification Software



- free access to academia
- More than 3,600 registered users
- 1,400+ active users from 92 countries

<http://www.uqlab.com>

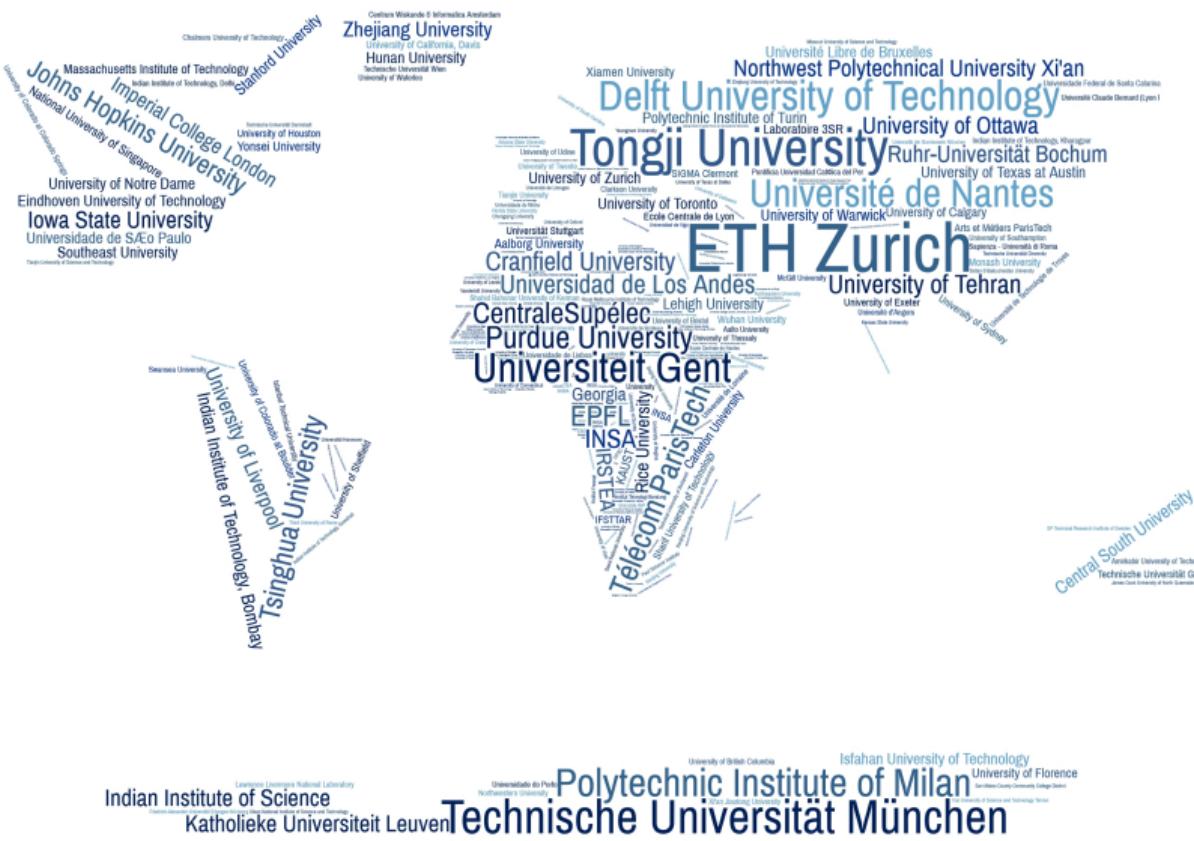


- The **cloud version** of UQLab, accessible via an API (SaaS)
- Available with **python bindings** for beta testing

<https://uqpylab.uq-cloud.io/>

Country	# Users
United States	582
China	500
France	339
Switzerland	285
Germany	270
United Kingdom	157
Italy	145
Brazil	126
India	120
Canada	87

As of May 15, 2021



# UQWorld: the community of UQ

<https://uqworld.org/>

The screenshot shows the homepage of UQWorld. At the top, there is a navigation bar with links for "All About UQ", "UQ Resources", "UQ with UQLab", "Sign Up", "Log In", a search icon, and a menu icon. The main header is "Welcome to UQWorld!". Below the header, there is a brief introduction: "Connect with fellow uncertainty quantification (UQ) practitioners across scientific disciplines to discuss the practice of UQ in science and engineering, use cases, and best practices. You can share and discuss your problem, experience, and expertise in all topics related to UQ and UQLab." The page features three main sections: "All About UQ", "UQ Resources", and "UQ with UQLab". Each section has a brief description and a small icon. Below these sections, there are buttons for "all categories", "all tags", and "Categories" (which is highlighted in red). There are also "Latest" and "Top" links. The "All About UQ" section includes a "Char's Blog" and "UQ Discussion Forum". The "UQ Resources" section includes a "UQWorld" logo. The "UQ with UQLab" section includes a "UQLab" logo. At the bottom, there is a footer with the text "Active learning for reliability", the date "RESIM2021 – May 19, 2021", the name "B. Sudret", and the page number "50 / 50".

UQWorld

All About UQ UQ Resources UQ with UQLab Sign Up Log In

Welcome to UQWorld!

Connect with fellow uncertainty quantification (UQ) practitioners across scientific disciplines to discuss the practice of UQ in science and engineering, use cases, and best practices. You can share and discuss your problem, experience, and expertise in all topics related to UQ and UQLab.

**All About UQ**  
Discuss and learn more about UQ important concepts, best practices, and current topics with the community.

**UQ Resources**  
News, updates, and other resources from the UQ community.

**UQ with UQLab**  
Community-powered resources you need to use UQLab for UQ.

all categories all tags Categories Latest Top

Category Topics

**All About UQ** 24  
Connect with members of the community across scientific disciplines to discuss current topics, best practices, important concepts in uncertainty quantification (UQ). Learn more about UQ good practices from the RSUQ Chair.  
Char's Blog UQ Discussion Forum

**UQ Resources** 1 / month  
Here you can find news, updates, case studies, and other resources from our own community and the uncertainty quantification (UQ) community at large.

Active learning for reliability

RESIM2021 – May 19, 2021 B. Sudret 50 / 50

## Questions ?



**Chair of Risk, Safety & Uncertainty Quantification**

[www.rsuq.ethz.ch](http://www.rsuq.ethz.ch)

## The Uncertainty Quantification Software

[www.uqlab.com](http://www.uqlab.com)



## The Uncertainty Quantification Community

[www.uqworld.org](http://www.uqworld.org)

