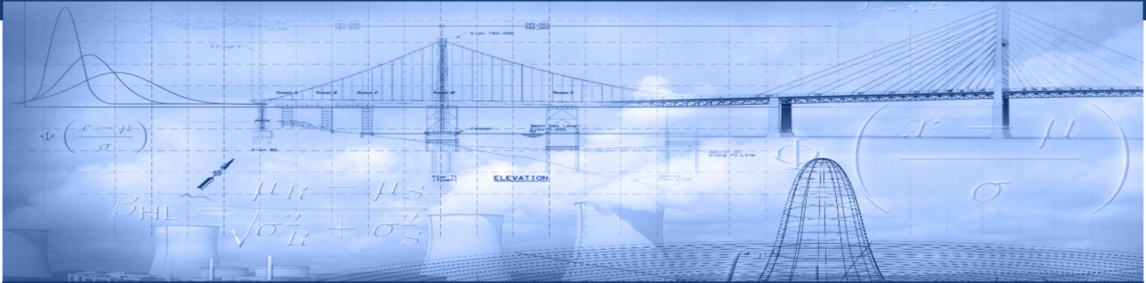
Benchmark of active learning methods for structural reliability analysis

Author(s):
Moustapha, Maliki; Marelli, Stefano; Sudret, Bruno

Publication date:
2022-07-20

Permanent link:
https://doi.org/10.3929/ethz-b-000559274



Benchmark of active learning methods for structural reliability analysis

15th International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing (MCQMC), Linz, July 17-22, 2022

Maliki Moustapha, Stefano Marelli and Bruno Sudret

Chair of Risk, Safety and Uncertainty Quantification | ETH Zürich

How to cite?

This presentation is a talk given at the 15th International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing (MCQMC) on July 20, 2022.

How to cite

Moustapha, M., Marelli, S. and Sudret, B., *Benchmark of active learning methods for structural reliability analysis*, 15th International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing (MCQMC), Linz (Austria), July 17-22, 2022.

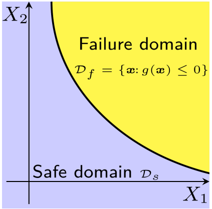
Structural reliability analysis

- Estimate the probability of occurrence of an adverse event

$$P_f = \int_{\mathcal{D}_f} f_{\boldsymbol{X}}(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x}$$

$f_{\boldsymbol{X}}(\boldsymbol{x})$:	Joint distribution of the random vector $\boldsymbol{X}$
$\mathcal{D}_f = \{\boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}} : g(\boldsymbol{x}, \mathcal{M}(\boldsymbol{x}) \leq 0)\}$:	Failure domain

- Failure is assessed by a limit-state function $g : \boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}} \mapsto \mathbb{R}$, based on a computational model $\mathcal{M}$
- Multi-dimensional integral ($d = 10 - 100^{+}$), implicit domain of integration
- 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

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

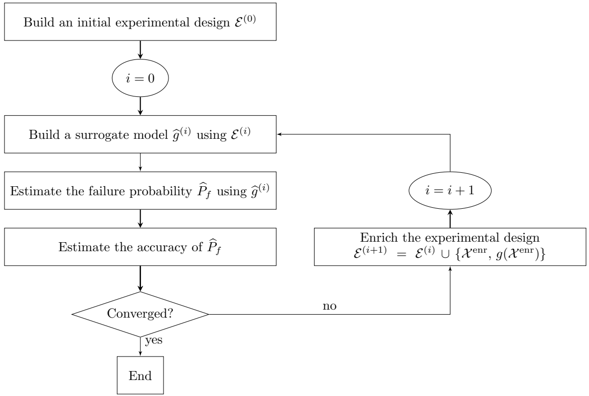
Name	Shape	Parameters
Polynomial chaos expansions	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} a_{\boldsymbol{\alpha}}\, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{x})$	$\boldsymbol{a}_{\boldsymbol{\alpha}}$
Low-rank tensor approximations	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \sum_{l=1}^{R} b_l \left(\prod_{i=1}^{M} v_l^{(i)}(x_i) \right)$	$b_l,\ z_{k,l}^{(i)}$
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	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \sum_{i=1}^{m} a_i\, K(\boldsymbol{x}_i, \boldsymbol{x}) + b$	$\boldsymbol{a}\ ,\ b$

Outline

Active learning reliability framework

Bichon *et al.* (2009), Echard *et al.* (2011)

Principle: A surrogate model, built by adaptively enriching an experimental design $\mathcal{E} = \{\mathcal{X}, g(\mathcal{X})\}$ so as to be accurate in the vicinity of the limit-state surface, is used within a reliability analysis



Active learning reliability illustration

Active Kriging - Monte Carlo simulation (AK-MCS)

Echard *et al.* (2011)

- Gaussian process model to emulate the limit-state
- ED locally enriched using the deviation number U

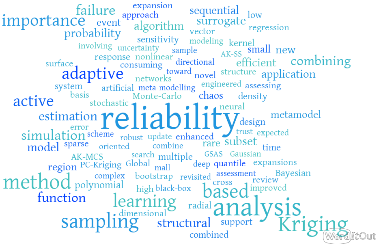
- Probability of failure estimated using Monte Carlo simulation
- Convergence assumed when U is sufficiently large

Active learning reliability methods

Teixeira *et al.* (2021), Moustapha *et al.* (2022)

Numerous papers on active learning called AK-XXX-YYY 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.* (2022)

	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 & Shafieezadeh (2019) Wang & Shafieezadeh (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	Chojazyck et al. (2015) Gomes et al. (2019) Li & Wang (2020) [Deep NN]	Sundar & Shields (2016)	Chojazyck 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 β
SVR	Importance sampling	FBR	Stability of P_f
PC-Kriging	Line sampling	CMM	Bounds on β
Neural networks	Directional sampling	SUR	Bounds on P_f
...	...	...	...

Outline

Extensive benchmark: set-up

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

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

Moustapha, M., Marelli, S. & Sudret, B. Active learning reliability: survey, general framework and benchmark (2022), Struct. Saf., 96

Gaussian process modelling or Kriging

Rasmussen & Williams (2006)

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

$$\mathcal{M}(\boldsymbol{x}) = \boldsymbol{\beta}^T \boldsymbol{f}(\boldsymbol{x}) + \sigma^2 Z(\boldsymbol{x})$$

$\boldsymbol{\beta}^T \boldsymbol{f}(\boldsymbol{x})$: trend - $Z(\boldsymbol{x})$: zero-mean, unit variance Gaussian process - σ^2 process variance

- The experimental design response $\mathcal{Y}$ and the response at new point $\widehat{Y}(\boldsymbol{x})$ are jointly Gaussian

$$\begin{Bmatrix} \widehat{Y}(\boldsymbol{x}) \\ \mathcal{Y} \end{Bmatrix} \sim \mathcal{N}_{N+1}\left(\begin{Bmatrix} \mathbf{f}(\boldsymbol{x})^T \beta \\ \mathbf{F}\beta \end{Bmatrix}, \quad \sigma^2 \begin{Bmatrix} 1 & \mathbf{r}^T(\boldsymbol{x}) \\ \mathbf{r}(\boldsymbol{x}) & \mathbf{R} \end{Bmatrix} \right)$$

- The prediction is given by the conditional mean (and variance)

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

$$\sigma^2_{\widehat{Y}(\boldsymbol{x})} = \widehat{\sigma}^2\left(1 - \mathbf{r}^T(\boldsymbol{x})\mathbf{R}^{-1}\mathbf{r}(\boldsymbol{x}) + \mathbf{u}^T(\boldsymbol{x})(\mathbf{F}^T\mathbf{R}^{-1}\mathbf{F})^{-1}\mathbf{u}(\boldsymbol{x})\right)$$

$R_{ij} = R\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}; \widehat{\gamma}\right)$ - $\boldsymbol{r}(\boldsymbol{x}) = R\left(\boldsymbol{x}, \boldsymbol{x}^{(i)}; \widehat{\gamma}\right)$ - $\boldsymbol{F} = F_{ij} = f_j\left(\boldsymbol{x}^{(i)}\right)$

- $\left\{\widehat{\beta}, \widehat{\sigma}^2, \widehat{\boldsymbol{\theta}}\right\}$ are estimated by maximum likelihood

Polynomial chaos expansions

- The random variable $Y = \mathcal{M}(\boldsymbol{X})$ can be cast as a polynomial expansion in the form

Xiu & Karniadakis (2002)

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

$\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$: 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
- Approximation obtained by truncating the infinite series

$$Y = \mathcal{M}(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} y_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) + \varepsilon_P$$

- Coefficients can computed by ordinary least square
- Sparsity enforced here using advanced truncation scheme and least angle regression

Blatman & Sudret (2011)

- Analytical approximation of the leave-one-out error speeds up calibration

Polynomial-Chaos Kriging

- Universal Kriging with a sparse PCE model as trend

Schöebi *et al.* (2015,2016)

$$\mathcal{M}(\boldsymbol{x}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} y_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) + \sigma^2 Z(\boldsymbol{x}),$$

- Combines both advantages of PCE and Kriging:
  - PCE approximates the global behaviour of the model
  - Kriging captures local variations and provides an in-built error estimate

- Both the coefficients of the expansion and the auto-correlation parameters are calibrated
  - Sequential PC-Kriging: LAR to detect basis then universal Kriging model calibration
  - Optimal PC-Kriging: Universal Kriging model calibration at each iteration of LAR

Reliability estimation algorithms

Melchers & A.T. Beck (2018), Au & Beck (2001)

Crude Monte Carlo simulation

$$P_{f,\text{MC}} = \frac{1}{N}\sum_{k=1}^{N} \mathbf{1}_{\mathcal{D}_f}\left(\boldsymbol{x}^{(k)}\right)$$

- Universal and easy to implement
- Unbiased but slow convergence
- Difficulty to sample in the failure domain for very small P_f

Importance sampling

$$P_{f,\text{IS}} = \frac{1}{N}\sum_{k=1}^{N} \mathbf{1}_{\mathcal{D}_f}\left(\boldsymbol{x}^{(k)}\right)\frac{f_{\boldsymbol{X}}\left(\boldsymbol{x}^{(k)}\right)}{\Psi\left(\boldsymbol{x}^{(k)}\right)}$$

- Sample from an instrumental density with higher weight in the failure domain
- e.g., a Gaussian centered on the most probable failure point
- Other advanced techniques not considered here.

Subset simulation

$$P_{f,\text{SuS}} = \mathbb{P}\left(\mathcal{D}_1\right)\prod_{i=1}^{m-1}\mathbb{P}\left(\mathcal{D}_{i+1}|\mathcal{D}_i\right)$$

- Solve a series of problems with larger target probabilities
- Split the domain:

$$\mathcal{D}_1 \supset \mathcal{D}_2 \supset \cdots \supset \mathcal{D}_m = \mathcal{D}_f$$

- Conditional samples are obtained using Markov Chain Monte Carlo (MCMC)
- The initial and conditional probabilities are estimated by Monte Carlo simulation

Reliability estimation algorithms

Melchers & A.T. Beck (2018), Au & Beck (2001)

Crude Monte Carlo simulation

$$P_{f,\text{MC}} = \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_{\mathcal{D}_f}\left(\boldsymbol{x}^{(k)}\right)$$

- Universal and easy to implement
- Unbiased but slow convergence
- Difficulty to sample in the failure domain for very small P_f

Importance sampling

$$P_{f,\text{IS}} = \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_{\mathcal{D}_f}\left(\boldsymbol{x}^{(k)}\right) \frac{f_{\boldsymbol{X}}\left(\boldsymbol{x}^{(k)}\right)}{\Psi\left(\boldsymbol{x}^{(k)}\right)}$$

- Sample from an instrumental density with higher weight in the failure domain
- e.g., a Gaussian centered on the most probable failure point
- Other advanced techniques not considered here.

Subset simulation

$$P_{f,\text{SuS}} = \mathbb{P}\left(\mathcal{D}_1\right) \prod_{i=1}^{m-1} \mathbb{P}\left(\mathcal{D}_{i+1} | \mathcal{D}_i\right)$$

- Solve a series of problems with larger target probabilities
- Split the domain:

$$\mathcal{D}_1 \supset \mathcal{D}_2 \supset \cdots \supset \mathcal{D}_m = \mathcal{D}_f$$

- Conditional samples are obtained using Markov Chain Monte Carlo (MCMC)
- The initial and conditional probabilities are estimated by Monte Carlo simulation

Reliability estimation algorithms

Melchers & A.T. Beck (2018), Au & Beck (2001)

Crude Monte Carlo simulation

$$P_{f,\text{MC}} = \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_{\mathcal{D}_f}(\boldsymbol{x}^{(k)})$$

- Universal and easy to implement
- Unbiased but slow convergence
- Difficulty to sample in the failure domain for very small P_f

Importance sampling

$$P_{f,\text{IS}} = \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_{\mathcal{D}_f}(\boldsymbol{x}^{(k)}) \frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(k)})}{\Psi(\boldsymbol{x}^{(k)})}$$

- Sample from an instrumental density with higher weight in the failure domain
- e.g., a Gaussian centered on the most probable failure point
- Other advanced techniques not considered here.

Subset simulation

$$P_{f,\text{SuS}} = \mathbb{P}(\mathcal{D}_1) \prod_{i=1}^{m-1} \mathbb{P}\left(\mathcal{D}_{i+1} | \mathcal{D}_i\right)$$

- Solve a series of problems with larger target probabilities
- Split the domain:

$$\mathcal{D}_1 \supset \mathcal{D}_2 \supset \cdots \supset \mathcal{D}_m = \mathcal{D}_f$$

- Conditional samples are obtained using Markov Chain Monte Carlo (MCMC)
- The initial and conditional probabilities are estimated by Monte Carlo simulation

Extensive benchmark: options for the various methods

Kriging	PCE	PC-Kriging
• Trend: Constant • Kernel: Gaussian • Calibration: MLE	• Degree: $1-20$ • q-norm : 0.8 • Calibration: LAR	• Same as Kriging • same as PCE but... • Degree $1-3$

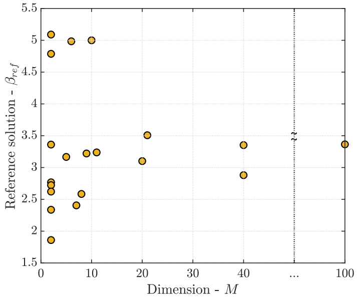
Monte Carlo simulation	Importance sampling	Subset simulation
• Max. sample size: 10^7 • Target C.o.V: 2.5% • Batch size: 10^5	• Max. sample size: 10^4 • Target C.o.V: 2.5% • Instrumental density: Standard Gaussian centered on the MPFP	• Max. sample size: 10^7 • Target C.o.V: 2.5% • Batch size: 10^5 • Conditional probability: $p_0 = 0.25$

Overkill setting in reliability estimation algorithms

- Reduce the stochastic error due to the reliability estimation algorithm
- Increase the likelihood of finding enrichment points in the remote failure domains

Extensive benchmark: selected problems

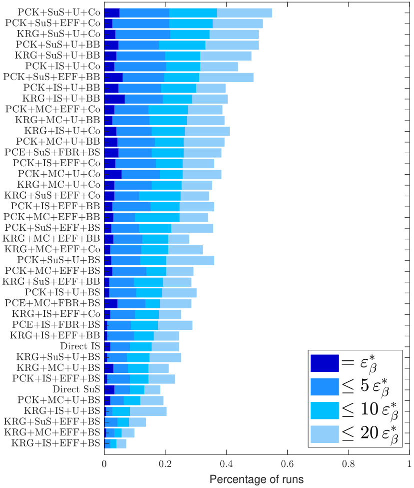
- 20 problems selected from the literature
- 11 come from the TNO benchmark (https://rprepo.readthedocs.io/en/latest/)
- Wide spectrum of problems in terms of
  - Dimensionality
  - Reliability index $\beta = -\Phi^{-1}(P_f)$



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	$\mathbf{1.31 \cdot 10^{-7}}$	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	$\mathbf{3.14 \cdot 10^{-2}}$	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)

Outline

Ranking of the strategies: accuracy of β

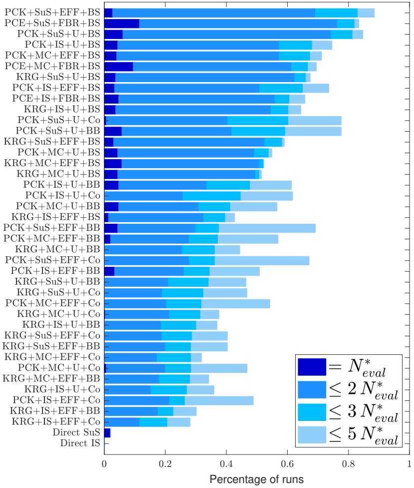


How many times a method ranks best in terms of smallest error on beta (resp. within 5, 10 or 20 times this relative error)?

$$\varepsilon = \left|\beta - \beta_{\text{ref}}\right| / \beta_{\text{ref}}$$

- Best approach: PC-Kriging + SuS + U + Combined stopping criterion
- Worst approaches: Kriging + IS + EFF + BS

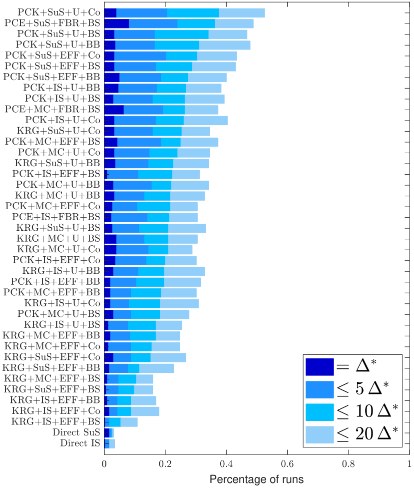
Ranking of the strategies: number of model evaluations



How many times a method ranks best (resp. within 2, 3, 5 times the lowest cost denoted N^*_{eval}) ?

- Best approach: PC-Kriging + SuS + EFF + BS
- Worst approaches: Direct SuS and Direct IS

Ranking of the strategies: efficiency



How many times a method ranks best according to efficiency Δ (resp. within 5, 10, 20 times the best)?

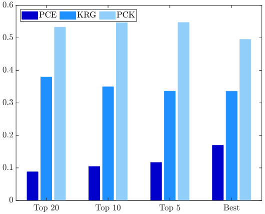
$$\Delta = \varepsilon_\beta \frac{N_{\text{eval}}}{\overline{N}_{\text{eval}}}$$

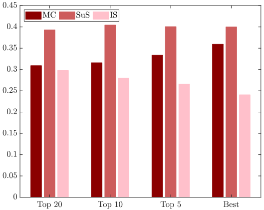
where $\overline{N}_{\text{eval}}$ is the median number of model evaluations for a particular problem (over all methods and replications)

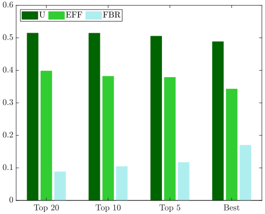
- Best approach: PC-Kriging + SuS + U + Combined stopping criterion
- Worst approaches: Direct SuS and Direct IS

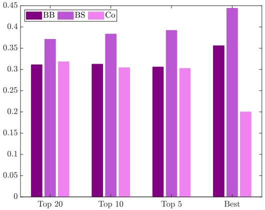
Results aggregated by method

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





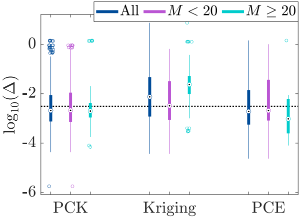


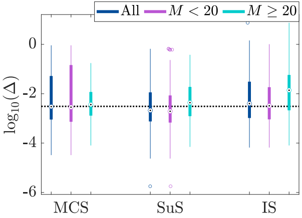


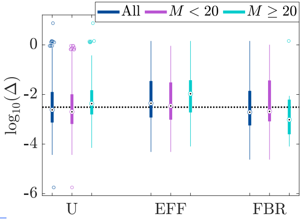
- Surrogates: PC-Kriging dominates by far
- Reliability: Slight advantage to subset simulation
- Learning function: U dominates both EFF and FBR
- Stopping criterion: Slight advantage to the stability criterion

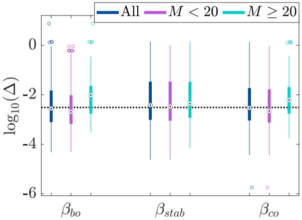
Performance w.r.t. problem feature: dimension

Results split in dimension: $M < 20$ vs. $M \geq 20$



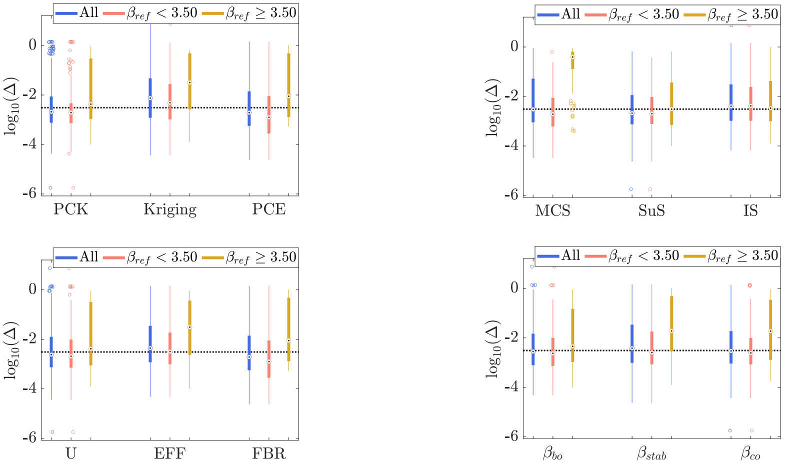






Performance w.r.t. problem feature: P_f range

Results split in reliability index: $\beta < 3.5$ vs. $\beta \geq 3.5$



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	β_{bo},β_{co}	β_{bo} / β_{co}	β_{bo},β_{co}	β_{bo}

Main take-away

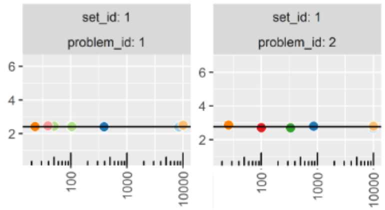
- The active learning method inherits the pros and cons of the reliability method
- Surrogate allows reducing the stochastic error due to the reliability estimation algorithm

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

TNO Benchmark: performance of UQLab "ALR" module

Rozsas & 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)



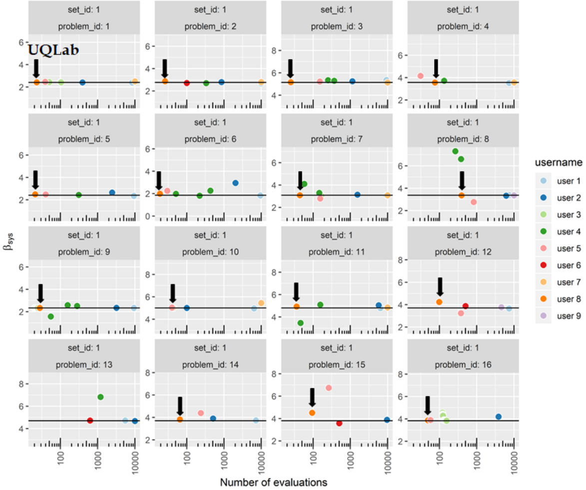
Summary plot (TNO)

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

best approach: "on the line / to the left"

TNO Benchmark: performance of UQLab “ALR” module

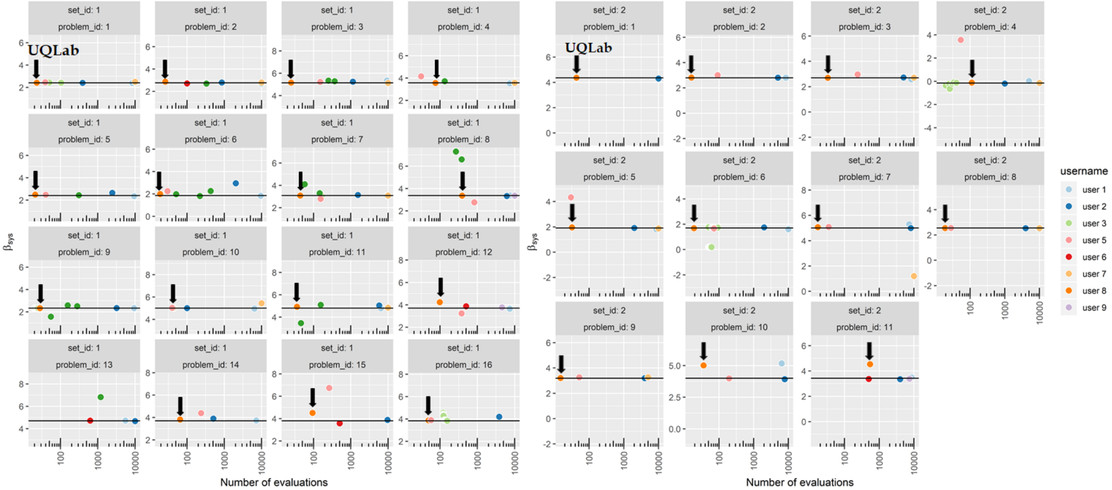
Rozsas & Slobbe (2019)



Component reliability

TNO Benchmark: performance of UQLab “ALR” module

Rozsas & Slobbe (2019)



Component reliability | **System reliability**

Conclusions

- Extensive survey and identification of an underlying recurring scheme
- Global framework for active learning reliability considering four components or modules
- Extensive benchmark running approximately $12,000$ reliability analyses
- Best performance from our benchmark: combination of PC-Kriging and subset simulation
- The flexibility of the proposed framework allows building strategies on-the-fly considering the features of the problem
- Surrogates should be used to harness the benefits of the most sophisticated reliability estimation algorithms

Questions ?

Chair of Risk, Safety & Uncertainty Quantification

www.rsuq.ethz.ch

The Uncertainty Quantification Software

www.uqlab.com

UQLab

www.uqpylab.uq-cloud.io

UQ[py]Lab

The Uncertainty Quantification Community

www.uqworld.org

UQWorld

Thank you very much for your attention !