ETHzürich

Active learning for structural reliability: survey, general framework and benchmark

M. Moustapha, S. Marelli and B. Sudret

Chair of Risk, Safety and Uncertainty Quantification
Stefano-Franscini-Platz 5
CH-8093 Zürich

Risk, Safety &
Uncertainty Quantification

Data Sheet

Active learning for structural reliability: survey, general framework and benchmark

M. Moustapha[1], S. Marelli[1], and B. Sudret[1]

[1] *Chair of Risk, Safety and Uncertainty Quantification,*
ETH Zurich, Stefano-Franscini-Platz 5, 8093 Zurich, Switzerland

Abstract

Active learning methods have recently surged in the literature due to their ability to solve complex structural reliability problems within an affordable computational cost. These methods are designed by adaptively building an inexpensive surrogate of the original limit-state function. Examples of such surrogates include Gaussian process models which have been adopted in many contributions, the most popular ones being the efficient global reliability analysis (EGRA) and the active Kriging Monte Carlo simulation (AK-MCS), two milestone contributions in the field. In this paper, we first conduct a survey of the recent literature, showing that most of the proposed methods actually span from modifying one or more aspects of the two aforementioned methods. We then propose a generalized modular framework to build on-the-fly efficient active learning strategies by combining the following four ingredients or modules: surrogate model, reliability estimation algorithm, learning function and stopping criterion. Using this framework, we devise 39 strategies for the solution of 20 reliability benchmark problems. The results of this extensive benchmark (more than $12,000$ reliability problems solved) are analyzed under various criteria leading to a synthesized set of recommendations for practitioners. These may be refined with a priori knowledge about the feature of the problem to solve, *i.e.* dimensionality and magnitude of the failure probability. This benchmark has eventually highlighted the importance of using surrogates in conjunction with sophisticated reliability estimation algorithms as a way to enhance the efficiency of the latter.

Keywords: Structural reliability – Active learning – Surrogate models – Benchmark – Gaussian process (Kriging) – Polynomial chaos expansions

1 Introduction

Structural reliability analysis is a central tool for the design and assessment of complex engineering systems. Such systems are affected by uncertainties, which may arise from natural variability in their physical properties (*e.g.*, material strength, manufacturing tolerances), operating conditions (*e.g.*, variable loads, environmental conditions) or simply because of an incomplete or lack of knowledge (*e.g.*, in the non-destructive assessment of existing structures). Structural reliability analysis aims at assessing the effects of such uncertainties, by estimating the associated failure probability with respect to some relevant limit states. In this paper, we consider a probabilistic setting, in which the uncertainties are represented through a set of random parameters $\boldsymbol{X} \in \mathcal{D}_{\boldsymbol{X}} \subset \mathbb{R}^M$ completely defined by their joint probability distribution function (PDF) $f_{\boldsymbol{X}}$. These parameters represent the state of the system, which can be evaluated through a so-called performance function (a.k.a. limit-state function), herein denoted by $g(\boldsymbol{X})$. By convention, the system is assumed to be in a failure (resp. safe) state when $g(\boldsymbol{x}) \leq 0$ (resp. $g(\boldsymbol{x}) > 0$). The probability of failure of the system can then be defined as

$$P_f = \mathbb{P}\left(g(\boldsymbol{X}) \leq 0\right) = \int_{\mathcal{D}_f} f_{\boldsymbol{X}}\left(\boldsymbol{x}\right) \mathrm{d}\boldsymbol{x}. \tag{1}$$

This integration over an implicitly defined domain $\mathcal{D}_f = \{\boldsymbol{x} : g(\boldsymbol{x}) \leq 0\}$ is not straightforward to solve and has motivated the development of a rich variety of techniques (Ditlevsen and Madsen, 1996; Lemaire, 2009; Melchers, 2018). These techniques can be broadly grouped in several classes. These include *approximation methods*, where the limit-state function is linearized (or otherwise approximated) around a so-called design point, *e.g.*, the most probable failure point (MPFP) in a suitably transformed probabilistic input space. This step allows one to then derive (semi-)analytically an approximation of the failure probability. This class includes the well-known first-order and second-order reliability methods (FORM and SORM) (Hasofer and Lind, 1974; Rackwitz and Fiessler, 1978). This family, however, is known to suffer severe limitations when the limit-state function is strongly non linear, or in the presence of multiple failure modes. A second class of methods, namely that of *simulation techniques*, is widely used for the solution of Eq. 1. Monte Carlo simulation is certainly among the most widely-used methods in this category. It is known to be robust and unbiased, yet its convergence rate is extremely slow, especially when the target failure probability is small. This is problematic when the computational model used in the evaluation of the limit-state function is costly, a common occurrence when *e.g.*, finite element analysis is involved. More advanced methods, based on variance-reduction techniques, are constantly being developed. A non-exhaustive list of the latter include importance sampling (Melchers, 1989), subset simulation (Au and Beck, 2001), directional simulation (Ditlevsen et al., 1990), line sampling (Koutsourelakis et al., 2004) and asymptotic sampling (Bucher, 2009). Numerous variants of these methods have been introduced in the recent literature in an attempt to further accelerate their convergence rates, *e.g.*, Papaioannou et al. (2016);

Wang et al. (2019); Geyer et al. (2019). However, the computational cost remains unaffordably high (*i.e.* $\mathcal{O}(10^{3-4})$ model runs) when considering time-consuming computational models.

In the past decade, a different avenue that offers substantial savings in the computational budget, while retaining the favourable properties of simulation methods, has been explored in the reliability analysis literature: *surrogate-model aided methods*. Surrogate models are inexpensive approximations of the original computational model, which have consistently shown superior performance when combined with the traditional simulation methods introduced earlier. Originally, simple polynomial response surface models (RSM) were built using a set of carefully designed computer experiments (Faravelli, 1989; Lemaire, 1998) These RSM were then used *in lieu* of the original computational model to approximately solve Eq. 1. Borrowing from the machine learning community, this process has evolved into a more sophisticated methodology known as *active learning* (Bichon et al., 2008; Echard et al., 2011). In active learning, the surrogate model is not used as a mere proxy of the original computational model, but as a tool to help explore the random input variable space efficiently. The idea is to start with an initial small set of model evaluations, known as the *experimental design*, which is then sequentially enriched following a so-called *learning function*. The latter aims at finding which model evaluation would bring the most useful information for the purpose of accurately assessing the failure probability of the system under consideration. This starts from the premise that in Eq. 1, only the sign of the limit-state is required to characterize the failure domain $\mathcal{D}_f$ in simulation-based reliability algorithms. The goal is then to approximate the limit-state surface as parsimoniously as possible (*i.e.*, using the least number of model evaluations) to achieve the best possible accuracy for the estimated failure probability. In the past few years, an increasingly large number of contributions have been proposed in the field of active learning for reliability analysis. The most popular approaches are based on Kriging, a.k.a. Gaussian process modelling, owing to its built-in error measure. The most prominent examples are the efficient global reliability analysis (EGRA) proposed by Bichon et al. (2008) and the active Kriging Monte Carlo simulation (AK-MCS) developed by Echard et al. (2011). The latter is a cornerstone of various methods derived incrementally by modifying one or another aspect of the AK-MCS algorithm (Lelièvre et al., 2018) and commonly referred to as *AK methods*. For instance, replacing the Monte Carlo simulation part of the algorithm with importance sampling or subset simulation leads respectively to AK-IS (Echard et al., 2013) or AK-SS (Huang et al., 2016). Similarly, other contributions have targeted the surrogate model type, introducing for instance support vector machines (Hurtado, 2004; Deheeger and Lemaire, 2007; Bourinet et al., 2011; Bourinet, 2018) or polynomial chaos expansions (Marelli and Sudret, 2018). A comprehensive overview of recent developments in active-learning based reliability analysis can be found in Teixeira et al. (2021).

This paper aims at achieving two goals. The first is to provide an in-depth characteriza-

tion of the current trends in active-learning-based reliability analysis through a comprehensive survey, following the footsteps of Teixeira et al. (2021). In doing so, however we focus on highlighting common aspects in the numerous literature contributions. More specifically, we classify the methods with respect to the specific *novelty* put forward in each contribution.

We then propose a generalized framework that summarizes the survey and puts the entire reviewed literature under a consistent formal umbrella. This framework is built by combining non-intrusively four identified common ingredients of active learning-based methods: i. a surrogate model, ii. a reliability estimation algorithm, iii. a learning function and iv. a stopping criterion. In the second part of the paper, we then conduct the first-ever extensive benchmark of active learning methods considering, on the one hand, a collection of 20 problems of diverse characteristics and on the other hand, a total of 39 active learning schemes built by combining selected methods in each of the four aforementioned components.

This benchmark is mainly aimed at illustrating how easily the proposed framework can be configured to reproduce a wide class of recently published studies, and hence only a limited number of methods per component was considered. The methods were selected for their maturity and ease of deployment, *i.e.*, they do not require extensive tuning by the user and are relatively fast to run on standard workstations. For instance, only Kriging, polynomial chaos expansions and PC-Kriging surrogate models are considered in this study due to their prevalence in the reliability analysis literature and their off-the-shelf availability. Similarly the scope of the problems solved within the benchmark is limited to the ones typically considered in the reviewed active learning papers. More specifically, we do not consider time-variant (such as in Kroetz et al. (2020)), dynamic or extremely high-dimensional problems (*i.e.*, in the order of hundreds) as they would require special treatment. For the former, dimensionality reduction techniques are often combined with surrogate modelling as in manifold learning (Zheng and Xue, Zheng and Xue), active subspace method (Constantine et al., 2014; Zhou and Peng, 2020) or in a more general setting as in Lataniotis et al. (2020). Even though some of these methods may be used for mildly high dimensional problems, they are not considered in the survey or benchmark carried out in this paper.

The large batch of analyses resulting from the selected methods is repeated 15 times, to obtain statistically significant estimates on the stability of each method. This results in a set of over $12,000$ reliability analyses which allows us to validate, repeat and assess most of the methods introduced in the recent literature, and at the same time to explore a large portion of new methods and combinations that have not been published yet. The results of this benchmark are used to give recommendations as to which type of methods performs the best generally or at least to be preferred given features of the reliability problem at hand.

The remainder of the paper is organized as follows. Section 2 presents a literature review of the current state-of-the art in surrogate-modeling based reliability analysis. Section 3 introduces a generalized active learning reliability framework inferred from the literature review. In Section 4, an extensive comparative benchmark study is carried out on a wide

class of methods and benchmark problems. Finally, recommendations and conclusions are given in Section 5 and 6.

2 A short overview of recent literature

2.1 Common rationale

At the core of active-learning reliability lies the idea of reducing the cost of simulation algorithms by introducing a surrogate model as an inexpensive approximation of the expensive-to-evaluate limit-state function. Surrogate models were first introduced in a static scheme to globally replace computer codes mainly for the purpose of visualization or optimization. Active learning pushes this concept further by aiming to an efficient allocation of resources, *i.e.*, computer simulations are performed sparingly and only when most needed. Active learning reliability (ALR) algorithms are practically devised using the general framework illustrated in Figure 1. In the initialization step, a so-called *experimental design* $\mathcal{E}^{(0)} = \left\{ \left(\boldsymbol{\mathcal{X}}^{(i)}, \mathcal{Y}^{(i)} \right) : \mathcal{Y}^{(i)} = g\left(\boldsymbol{\mathcal{X}}^{(i)} \right) \in \mathbb{R}, \boldsymbol{\mathcal{X}}^{(i)} \in \mathbb{X} \subset \mathbb{R}^M, i = 1, \ldots, m_0 \right\}$ is initially generated. The input sample set $\left\{ \boldsymbol{\mathcal{X}}^{(i)}, i = 1, \ldots, m_0 \right\}$ is often drawn using space-filling methods such as Latin hypercube sampling (LHS, McKay et al. (1979)) or randomized low-discrepancy sequences (Sobol', 1967). Typically m_0 is chosen small, *i.e.*, in the order of tens of samples. Following initialization, the algorithm enters in a four-step loop where:

1. A **surrogate model** is built using the current experimental design;

2. The failure probability is estimated using the current surrogate model and an appropriate **reliability estimation** algorithm;

3. The **convergence** of the algorithm is assessed;

4. An **enrichment of the experimental design** is carried out by appropriately selecting at least one pair of sample points $\left\{ \boldsymbol{\mathcal{X}}^{(\mathrm{enr})}, g\left(\boldsymbol{\mathcal{X}}^{(\mathrm{enr})} \right) \right\}$, when convergence is not achieved. This is often achieved by evaluating a so-called *learning function* which gives information as to which points are most likely to increase the accuracy of the surrogate (and subsequently of the estimated failure probability) when added to the experimental design.

One of the first implementations of this flowchart was proposed by Bichon et al. (2008) in their efficient global reliability analysis method (EGRA). In this work, they used Gaussian process regression as a surrogate model, Monte Carlo simulation as reliability estimation algorithm and the so-called expected feasibility function (EFF) as a means to find points to enrich the experimental design. The latter is actually an adaptation to contour estimation of the well-known expected improvement (EI) function (Ranjan et al., 2008) widely used in Bayesian optimization as first introduced in Jones et al. (1998). A noticeable improvement of EGRA was introduced by Echard et al. (2011) in the widely known active Kriging - Monte Carlo simulation (AK-MCS) method. Contrary to EGRA, where reliability estimation is
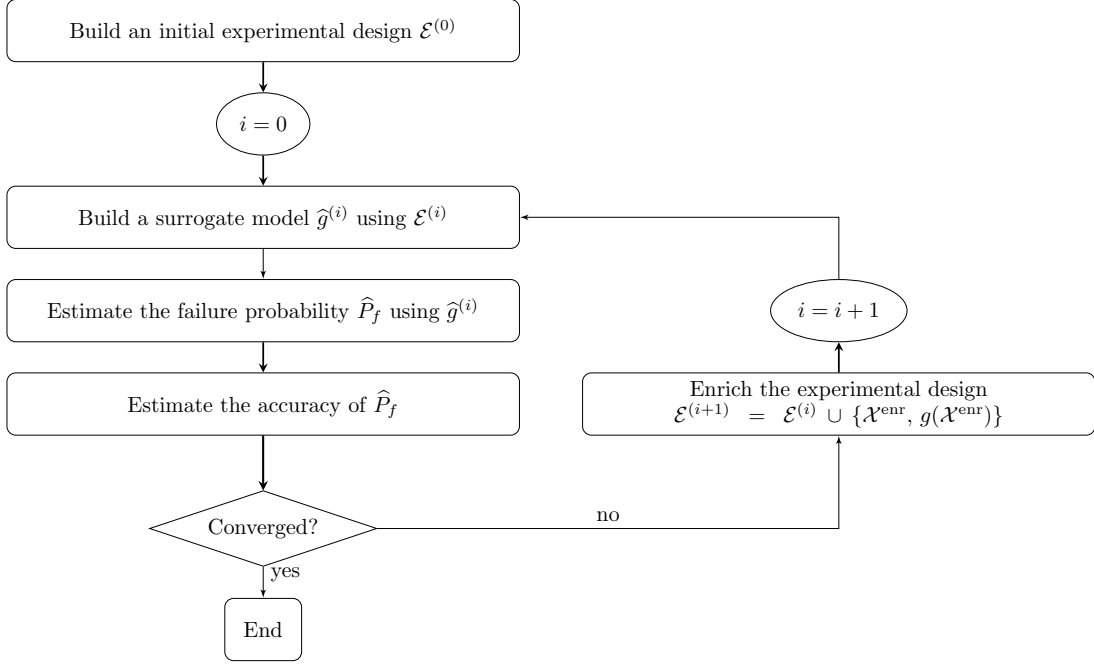
Figure 1: General flowchart of active learning reliability.

carried out only after the enrichment stage is completed, AK-MCS couples enrichment and reliability estimation. Furthermore, it introduces a new learning function, the so-called deviation number U, which is optimized with respect to a pre-defined sample set. This highly reduces the computational cost and the complexity of the active learning procedure.

While this algorithm is at the time of writing already ten years old, it has aged surprisingly well, with a number of recent methods proposing only minor variations to one or more of the steps just reported. A comprehensive survey on recent developments on this topic was recently proposed by Teixeira et al. (2021). In the following sections we identify and describe in more detail four key ingredients that are common to all of the aforementioned active-learning based methods.

2.2 Surrogate models in structural reliability

Various surrogate models were already used in adaptive schemes for the solution of reliability problems even before the emergence of the AK methods. Polynomial response surface models were arguably the first type of surrogates used in the context of structural reliability analysis. Faravelli (1989) uses a second-order polynomial while Bucher and Bourgund (1990) introduced a two-stage approach where a first quadratic response surface is used to locate the MPFP. A second response surface is then built close to that MPFP to refine the knowledge of the limit-state surface. A direct improvement of this approach which introduces an iterative procedure was proposed by Rajashekhar and Ellingwood (1993). Leonel et al. (2011) considered various schemes using response surfaces for reliability analysis in crack propagation and concluded that a direct coupling (*i.e.*, building the surrogate only once the

MPFP has been located) is more efficient. More recently, Roussouly et al. (2013) proposed an iterative scheme that combines trust regions, sparse response surface and bootstrap for the identification of regions where enrichment is necessary. Radial basis functions (RBF), which have been popular in static surrogate-assisted reliability analysis, were introduced in a sequential approach as well. Li et al. (2018) presented an MCS-based approach where an RBF is sequentially updated through a constrained min-max optimization problem which aims at finding points close to the limit-state surface while keeping a minimum distance to the existing ED points. Shi et al. (2019) proposed two other learning schemes based on RBF considering either an ensemble of surrogates or cross-validation. In the former case, the interquartile range of the predictions using an ensemble of surrogates is considered as a measure of uncertainty to derive a learning function similar to the U-function of Echard et al. (2011). Another learning function similar to U was developed by Marelli and Sudret (2018) using bootstrap and polynomial chaos expansions (PCE). More recently, sparse Bayesian PCE was used by Cheng and Lu (2020) where a new learning function relying on the Gaussian process variance was proposed. Pan et al. (2020) also used Bayesian regression PCE combined with the deviation number U.

Popular methods from the machine learning community, such as support vector machines or neural networks, have also been steadily introduced in structural reliability. Support vector machines for classification was first introduced by Hurtado (2004). Basudhar and Missoum (2008) proposed an adaptive scheme combining SVM classification and Monte Carlo simulation. The enrichment scheme is based on finding the point belonging to the limit-state surface approximation that is the furthest from the existing training points. This is a maximin problem solved using a general-purpose optimization algorithm. Lacaze and Missoum (2014) proposed an improvement of this maximin scheme by including a weight which accounts for the random variables joint PDF. Another improvement aiming at avoiding the optimization problem and relying on a candidate pool for enrichment has been proposed by Pan and Dias (2017). Combining SVM and subset simulation, Bourinet et al. (2011) proposed a learning scheme where a classifier is built in each iteration of the SuS algorithm. SVM has also been widely used in its regression form (SVR) for reliability analysis (Bourinet, 2018). Bourinet (2017) proposed an SVR scheme with three novelties: i. the sample set size is kept constant, meaning some samples are withdrawn from the training set as the algorithm is proceeding, ii. intermediate thresholds are used to approximate the limit-state functions and iii. the surrogate models built in each stage are combined in a weighted ensemble to keep information of all training points without increasing the computation time. Another popular machine learning method widely used in structural reliability analysis is neural networks. Even though most of the contributions are in a static scheme, the most recent ones consider adaptivity (Chojazyk et al., 2015; Kroetz et al., 2017). Sundar and Shields (2016) proposed a two-stage algorithm where an artificial neural network (ANN) is first used together with parallel Markov Chains to identify (possibly disjoint) failure regions. The

ANN is then enriched to accurately represent the limit-state surface. Finally, Gomes (2019) introduced an active scheme combining artificial neural networks and Monte Carlo simulation using the bootstrap-based learning function introduced in Marelli and Sudret (2018).

Various other surrogate model types have been used to propose new active learning reliability algorithms, following similar schemes as introduced earlier, *e.g.*, polynomial chaos-Kriging (Schöbi et al., 2016), high-dimensional model reduction (HDMR) (Sadoughi et al., 2017), deep neural networks (Li and Wang, 2020) or stochastic spectral embedding (Wagner et al., 2021), among others.

2.3 Reliability estimation algorithm

An immediate alternative strategy to AK-MCS can be devised by focusing on the reliability estimation algorithm. The benefits of replacing Monte Carlo simulation are two-fold. First, more sophisticated algorithms have been developed to reduce the variance of the failure probability estimate, and introducing them in active learning allows overcoming the pitfalls of MCS, *i.e.*, its slow convergence rate. Second, choosing another reliability estimation algorithm also allows one to modify the way sample candidates to enrichment are generated. In fact, for problems with low failure probability, the initial candidate set for enrichment may not contain any sample point at all in the actual failure domain. This can seriously reduce the chances of convergence of the active learning scheme. In contrast, more advanced reliability estimation algorithms may allow one to reach more easily areas associated with small probability densities, as well as disconnected failure regions.

Basically, almost all well-established simulation-based reliability estimation methods have been used together with active learning in the literature. A direct adaptation of AK-MCS, simply coined AK-IS, has been proposed by Echard et al. (2013) using importance sampling (Melchers, 2018) in lieu of Monte Carlo simulation. In this contribution, they first find the design point using FORM and the original model. They then build an importance density sample set around this point which is used both for computing the failure probability and as candidate pool for enrichment. Gaspar et al. (2017) proposed to use a surrogate model even for the location of the design point, hence further reducing the computational cost. Zhao et al. (2015) did not rely on the design point but rather uses Monte Carlo Markov Chain (MCMC) to generate points in the failure domains. Importance sampling is then performed around those points together with enrichment. This allows overcoming a major shortcoming of importance sampling related to the presence of multiple design points. Another line of research involving Kriging combined with IS includes the meta-IS algorithm where Dubourg et al. (2012) proposed to use the Kriging model to approximate the optimal importance density in an iterative scheme. Cadini et al. (2014) combined the work of Dubourg et al. (2012) and Echard et al. (2013) in a two-stage algorithm called metaAK-IS^2. Other sequential importance sampling methods have been adapted in an active Kriging strategy. For instance, Balesdent et al. (2013) sequentially built and enriched Kriging models

in intermediate steps of a cross-entropy and non-parametric adaptive importance sampling algorithm. Other contributions using adaptive importance sampling include Gaspar et al. (2017); Razaaly and Congedo (2018); Yang et al. (2018); Zhang and Taflanidis (2018); Liu et al. (2019); Pan et al. (2020); Zhang et al. (2020).

Another popular reliability estimation algorithm that has been used in an active learning scheme is subset simulation (Au and Beck, 2003). Huang et al. (2016) introduced AK-SS which, as its name suggests, is a declination of AK-MCS with the use of subset simulation for the computation of the failure probability. All other aspects are those of the original AK-MCS algorithm, including the candidate pool for enrichment which is obtained by an initial large Monte Carlo sample set. The obvious limitation here is that it may be difficult to find points in the failure region for problems where failure is an extremely rare event. Zhang et al. (2019) then proposed an improvement where the first and last levels of subset simulation are used as candidate pool for enrichment. The first level being a global Monte Carlo and the last one leading to points closest to the limit-state surface, this method allows both exploration and exploitation of the random input space. Ling et al. (2019) proposed an intermediate approach where a local Kriging model is built at each stage of subset simulation. Other similar methods include Bayesian subset simulation (Li et al., 2012; Bect et al., 2017) which combines subset simulation, sequential Monte Carlo and Kriging and AK-SSIS (Tong et al., 2015) which combines subset simulation and importance sampling in an active Kriging strategy.

Finally, we shall note that even though importance sampling and subset simulation have been widely exploited in active learning methods, the use of other variance-reduction simulation methods has been explored. Examples include algorithms such as directional importance sampling (Guo et al., 2020), radial basis importance sampling (Bo and HuiFeng, 2018) or line sampling (Lv et al., 2015).

2.4 Enrichment of the experimental design

A core feature of active-learning-based reliability methods is that the accuracy of the failure probability estimate is gradually increased by enriching the experimental design. A key component in this respect is the learning function (LF), which plays the central role of providing a measure of the information value of any experimental design enrichment candidates. Many authors have come up with new learning functions that can increase the efficiency of otherwise comparable methods. A direct improvement of the deviation number U was given for instance by Peijuan et al. (2017), where a line search step is added to get even closer to the limit-state surface once the best next point with respect to U is found. Arguing that errors due to regions with small density would be negligible in the final estimate of the failure probability, Wen et al. (2016) also used the random variables joint PDF to constrain the EFF learning function. Similarly, Sun et al. (2017) proposed the least improvement function (LIF) which weights the probability of misclassification $\Phi\left(-U\left(\boldsymbol{x}\right)\right)$ with the joint probability

density of the samples $f_{\boldsymbol{X}}(\boldsymbol{x})$, an idea already used in Dubourg et al. (2012). Tong et al. (2019) followed this idea and, adding more terms related to global/local uncertainty, they created a new learning function.

From another perspective, Lv et al. (2015) introduced a new learning function based on the information theory with an analytical expression similar to EFF. Hu and Mahadevan (2016) introduced a method which relies on computing the sensitivities of the failure probability to add new points to the experimental design. A new learning function using K-fold cross validation generalizable to any type of surrogate model was introduced by Xiao et al. (2018). More recently, Jiang et al. (2019) proposed an approach which is based on splitting the space using Voronoi cells and finding out those points with the largest sensitivities to the estimated failure probability. The use of Voronoi cells allows the authors to both reduce the computational cost and spread the sample points as much as possible. The latter goal is also achieved through a pre-processing step by Zhang et al. (2019) who then introduced a new LF inspired from the expected improvement.

2.5 Stopping criterion

The stopping criterion is an often overlooked yet crucial part of any active learning reliability algorithm. Three types of criteria have been proposed in the literature to halt the iterative enrichment scheme. The first one is directly based on the learning function. For instance, Bichon et al. (2008) proposed to stop the enrichment scheme when the value of the expected feasibility function is lower than 10^{-3}. Similarly, Echard et al. (2011) stops AK-MCS iterations when $U > 2$ for all candidate points. This actually means that the probability of misclassifying any point from the sample set used to evaluate the failure probability is below 2.28%. This criterion has shown to be extremely conservative, leading to unnecessarily added points. Some authors have tried softening it either by considering the whole candidate set through an average, for instance Jian et al. (2017); Sun et al. (2017); Lelièvre et al. (2018), or by considering convergence when only a small proportion of the candidate set does not comply with $U > 2$ (Moustapha et al., 2016; Fauriat and Gayton, 2017).

The second family of convergence criteria are those based on the accuracy of the failure probability. Using the Kriging variance, Dubourg et al. (2013) proposed a bound on the estimate $\widehat{P}_f$ which accounts for the Kriging epistemic uncertainty. Similarly, Sun et al. (2017); Jian et al. (2017) proposed an upper bound on $\left|\widehat{P}_f - P_f\right|$ using the probability of misclassification $\Phi(-U)$. For surrogate models which do not possess a built-in error measure, similar bounds have been derived considering either cross-validation (Shi et al., 2019) or bootstrap replicates (Marelli and Sudret, 2018).

Finally, the third family of stopping criteria has been built using the stabilization of either the limit-state surface or the failure probability estimates within enrichment iterations. Basudhar and Missoum (2008) tracked the fraction of some predefined convergence points that changed sign within two updates of an SVM model and assumed convergence when this

fraction was relatively small. This criteria, often with slight adjustments, has been used in numbers of SVM-based active learning schemes (Bourinet, 2018). As for the failure probability, the obvious approach is to track its variation within iterations. Stabilization criteria may often lead to premature convergence when the initial surrogate model is extremely inaccurate. A workaround consists in tracking the convergence over several iterations, on average 2 to 3 and in some contributions and up to 10 iterations (Bourinet, 2017). An alternative is to smooth out the convergence criterion as in Basudhar and Missoum (2008) by using an exponential curve fitted to the convergence criterion.

3   A generalized active learning reliability framework

3.1   Motivation

As anticipated in the previous section, the state-of-the-art in active learning reliability can essentially be summarized into four basic components or modules. These modules are namely the surrogate model, the reliability estimation algorithm, the learning function and the stopping criterion. Most of the contributions in the recent literature can be reconstructed by combining various methods within each module. In a few cases, elaborate ad-hoc techniques are devised by taking advantage of some highly specific combination of such methods. The modules are in these cases not independent anymore but intrusively linked to each other. However, the advantages brought by such configurations are most often only marginal and not systematically justified by benchmarks.

In this section, we present a modular framework for active learning whose first aim is to unify and present the plethora of active learning reliability methods from a single and consistent viewpoint. The interest of such an approach can be seen through the prism of the no-free lunch principle. Despite their claimed advantages, methods proposed in the literature perform best under certain conditions and do not generalize so well as to provide consistently superior performance in the wide spectrum of structural reliability problems. By framing active learning reliability under a modular framework, we can then take advantage of each method to solve a wide class of reliability problems, possibly identifying guidelines based on limited prior information, such as the problem dimensionality.

A second advantage of the framework we propose is that it decouples the four modules. This independence means that there is no need to alter or adapt a given method/module to fit in the overall workflow. Methods can be used solely through their input/output structures, as "black boxes" even allowing for a seamless interconnection to third-party software.

The core idea of the framework is depicted in Figure 2. The four modules are shown in columns with examples of popular methods. Most of the contributions surveyed in the previous section can be retrieved by appropriately combining the methods. For instance, combining all methods on the first row, *i.e.*, Kriging, Monte Carlo simulation, deviation number U and the LF-based stopping criterion, leads to the well-known AK-MCS. In princi-

ple, all methods within each module block can be combined with any from the other blocks. The only exceptions are from the methods that specifically rely on the surrogate built-in error, *e.g.*, the Kriging variance. It should be noted however that alternatives have been proposed in the literature to estimate comparable local error measures when not directly provided by the surrogate model itself.

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
...	...	...	...

Figure 2: Active learning reliability framework with example of methods.

3.1.1 Surrogate models

Surrogate models lie at the core of any active learning reliability algorithm. A relevant aspect that needs to be stressed here is that they are merely used as a tool to explore the random input space in conjunction with the original computational model. They are not meant to replace the latter *per se*. Many surrogates have been adopted in the literature for active learning and can be further classified on the basis of different properties. One way is to consider interpolation *vs.* regression approaches. The former are often preferred in active learning schemes as they allow to precisely approximate the limit-state function in the vicinity of any point that belongs to the experimental design. One may also consider a classification with regards to the availability of a built-in error measure. Built-in errors have been instrumental in the proliferation of active learning schemes, because most enrichment schemes rely on them. Kriging for instance, with its local variance estimator, is arguably the most adopted surrogate in recent active learning contributions. However, as shown in the literature review above, alternatives also exist and have even shown to be quite efficient. Such alternatives may include a similar error measure (either through statistical methods such as bootstrap and cross-validation or plug-in methods) or the use of alternative learning functions, as will be explained shortly.

3.1.2 Reliability estimation algorithm

Despite approximation methods have been sometimes used in early surrogate-assisted reliability methods (Bucher and Bourgund, 1990; Rajashekhar and Ellingwood, 1993), most of the recent contributions rely on simulation-based methods. Monte Carlo simulation, thanks

to its generality, is naturally one of the most commonly-used methods. The use of variance-reduction techniques such as subset simulation or importance sampling has also been widely explored. The latter can reduce the error due to the random nature of the sampling algorithm while keeping the number of model evaluations as low as possible, especially when the probability of failure is low.

In this contribution, we advocate for going even one step further by using an "overkill setting" of the reliability estimation algorithm, further capitalizing on the negligible computational costs associated with the use of surrogate models. By "overkill setting", we mean that the parameters of the algorithms are tuned to drastically reduce the coefficient of variation of the resulting failure probability estimate. This set of parameters depends on the reliability estimation algorithm. When using subset simulation for instance, the batch sample size and conditional failure probability are made larger than in the settings classically found in the literature: we choose here 10^5 samples per simulation step and an intermediate probability of 0.25 instead of the usual 0.10 value.

This approach has a two-fold benefit. First, the stochastic error due to the reliability estimation algorithm is reduced as much as possible, hence leaving only the surrogate-induced error to dominate the global estimation uncertainty on the failure probability. It should be noted that the overall computational time is of course increased, especially when Kriging is used. However, this overhead is expected to be marginal when compared to that of an actual computational model, *e.g.*, a finite element analysis. Second, by over-calibrating the reliability algorithm, we allow for the random space to be even more thoroughly explored, hence increasing the likelihood of finding sample points in the failure regions when the latter is considerably small. Examples of such settings will be shown in the benchmark section and compared to more traditional settings.

3.1.3 Learning function

The learning function is used as a driver to add new points in the experimental design. It is often intrinsically linked to both the surrogate model and the reliability estimation algorithm. Indeed, its very definition often draws from the characteristics of the surrogate model, *e.g.*, variance or built-in error measure. This does not need to be the case systematically, as the same features can be replaced by statistical methods that provide comparable error metrics, such as bootstrap and cross-validation or even mere distance measures to the existing experimental design points.

Generally, new candidate enrichment points are obtained by minimizing (or maximizing) the learning function over the input domain. The optimization problem is most often simplified into a discrete approximation where the enrichment samples are chosen from a finite candidate pool. In most of the literature (starting from the original AK-MCS algorithm (Echard et al., 2011) and in most subsequent variants), the candidate pool is defined prior to the analysis using Monte Carlo sampling. However this is not an optimal approach. For

instance, it is difficult to find points in the failure domain when the real failure probability is very small (*i.e.*, $P_f < 10^{-6}$) as this would require an extremely large candidate pool. An alternative and more efficient approach proposed here is to define the candidate pool as the set of samples generated by the chosen reliability estimation algorithm, which can in principle be updated throughout the ALR iterations. This allows us to fully exploit the benefits of more advanced reliability estimation algorithms that are more likely to locate the (multiple) failure domains and sample more points where needed most. To further avoid being intrusive, we consider as candidate pool for enrichment all samples that were used to estimate the failure probability in the previous iteration of the algorithm. To accelerate the procedure, it is possible to statistically reduce the size of the candidate pool by simple down-sampling or clustering. The latter approach can also serve as a way of simultaneously identifying multiple enrichment points so as to take advantage of any available parallelization capability.

3.1.4   Stopping criterion

The stopping criterion is an important part of the active learning scheme as the efficiency of the algorithm is ultimately and largely driven by its robustness. Too loose a stopping criterion can lead to premature convergence, while a too strict one would cause the unnecessarily addition of costly experimental design points. The criteria proposed in the literature can be classified into two groups. First are those based on the learning function value, *e.g.*, Bichon et al. (2008); Echard et al. (2011). These have shown to often be extremely conservative. The second family is derived by directly monitoring the accuracy of the estimated failure probability. Confidence bounds on the latter can be derived (Dubourg et al., 2012; Schöbi et al., 2016), and convergence is assumed when such bounds are small enough. Alternatively, one may monitor their evolution and assume convergence when a certain degree of stability is observed. Finally, increased robustness may be achieved by either combining different stopping criteria and/or considering convergence only when the criteria are satisfied consistently within a given number of consecutive iterations.

4   Comparative study

4.1   Benchmark set-up

The ingredients shown in the previous section can be assembled non-intrusively to build active learning schemes. In this section we perform an extensive comparison of several framework configurations on a set of benchmark reliability problems representative of a wide range of real case applications. We selected such configurations by considering some of the most widely-used methods in each module. Table 1 shows the different algorithms considered for the benchmark in this paper. The first part of the table deals with methods that use

the built-in surrogate model variance, while the second is based on a regression method and bootstrap error estimation. All possible combinations resulting from the tensor product of each compatible ingredient are considered. This amounts in a total of 39 strategies (36 for the first surrogate class and 3 for the second). UQLAB (Marelli and Sudret, 2014), a MATLAB framework for uncertainty quantification was used to run these analyses.

Table 1: Methods selected in each module to create the 39 solution strategies used in the benchmark. Further details about each method are given in Sections 4.2 and in the supplementary materials (Appendix B).

Reliability	Metamodel	Learning function	Stopping criterion
Monte Carlo simulation			Beta bounds
Subset simulation	Kriging	U	Beta stability
Importance sampling	PC-Kriging	EFF	Combined
Monte Carlo simulation			
Subset simulation	PCE	FBR	Beta stability
Importance sampling			

Using these 39 strategies, a collection of 20 reliability problems are solved. 11 of these problems were collected from the TNO reliability benchmark repository (Rozsas and Slobbe, 2019). The remaining were chosen from the literature with the aim of ensuring a large variety both in terms of limit-state function dimensionality and reference failure probability/reliability index. Most of the limit-state functions are analytical, except for two which are based on a truss finite element model. A list of the problems together with some references is given in the supplementary materials (Appendix B). Figure 3 summarizes these problems in terms of dimension against reference reliability index. They range from input dimension $M = 2$ to $M = 100$ and have a reliability index (resp. failure probability) that ranges from $\beta_{\mathrm{ref}} \approx 1.86$ (resp. $P_{f,\mathrm{ref}} \approx 3.14 \cdot 10^{-2}$) to $\beta_{\mathrm{ref}} \approx 5.15$ (resp. $P_{f,\mathrm{ref}} \approx 1.32 \cdot 10^{-7}$). The reference solutions are calculated using the original models and a large Monte Carlo set whose size is set adaptively until a coefficient of variation of 1% is reached. For the four problems that have a failure probability smaller than 10^{-7}, an overcalibrated subset simulation is used instead.

Throughout the benchmark, each analysis is repeated 15 times. The only exception being benchmark #8 of dimension 100, which is repeated only 10 times. It should be noted that within different strategies, the same initial random conditions/seeds are used for each of the repetitions. Hence, a total of $11,700$ reliability analyses (39 strategies x 20 problems x 15 repetitions) are carried out for this benchmark.
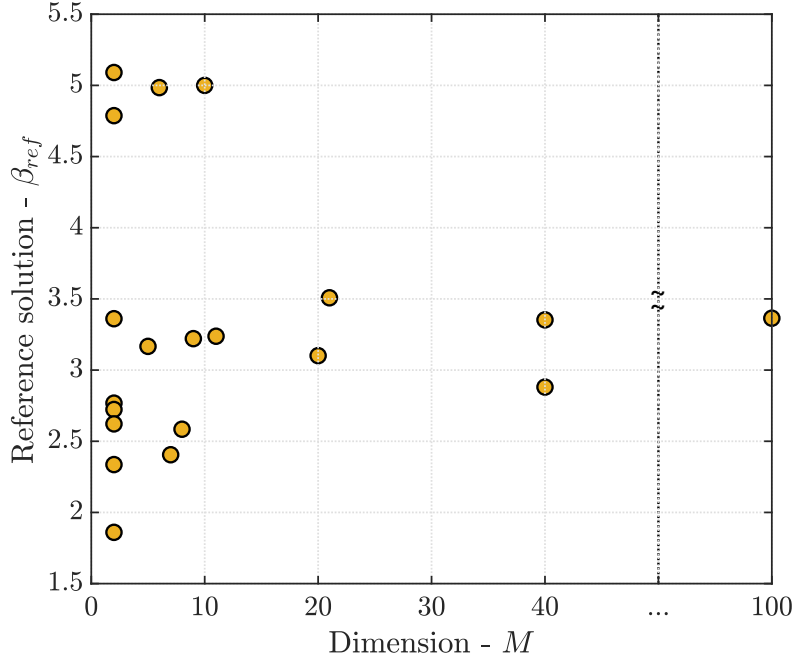
Figure 3: Collection of problems selected for the benchmark in terms of dimensions M and reference reliability indices β_{ref}.

4.2 Algorithmic settings

In this section, we will briefly review the algorithm settings used for each of the methods selected in Table 1. The methods selected for the first three components, *i.e.*, surrogate models, reliability estimation algorithm and learning functions are detailed in the supplementary materials (Appendix B). A summary of the most important settings for the surrogate models and reliability estimation algorithms is given in Figure 4. The learning functions however do not possess any special setting and are used exactly as described in the supplementary materials.

This section therefore focuses on the three stopping criteria mentioned in Table 1:

Beta bounds: This stopping criterion is based on the Kriging variance and reads:

$$\frac{\left|\widehat{\beta}^{+} - \widehat{\beta}^{-}\right|}{\widehat{\beta}} \leq \bar{\varepsilon}_{BB} \tag{2}$$

where $\widehat{\beta}^{+}$ and $\widehat{\beta}^{-}$ are the reliability indices respectively obtained using the limit-state functions $\mu_{\widehat{g}}(\boldsymbol{x}) - 2\sigma_{\widehat{g}}(\boldsymbol{x})$ and $\mu_{\widehat{g}}(\boldsymbol{x}) + 2\sigma_{\widehat{g}}(\boldsymbol{x})$, while $\widehat{\beta}$ is the reliability index obtained using the limit-state $\mu_{\widehat{g}}(\boldsymbol{x})$.

The threshold $\bar{\varepsilon}_{BB}$ is set to 0.01 which is arguably a relatively large value. However, convergence is assumed only when this criterion is respected three times in a row, hence ensuring some degree of robustness.

16

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: $p_0 = 0.25$

Figure 4: A summary of the most important settings for the surrogate models and reliability estimation algorithms considered in this paper. The meaning of each of these parameters can be found in the supplementary materials (Appendix B).

Beta stability   This convergence criterion ensures the stability of the failure probability estimate assuming that convergence is achieved when adding new points do not noticeably modify the estimate. Using the reliability index, it reads:

$$\frac{\left|\widehat{\beta}^{(i)} - \widehat{\beta}^{(i-1)}\right|}{\widehat{\beta}^{(i)}} \leq \bar{\varepsilon}_{BS}, \tag{3}$$

where $\widehat{\beta}^{(i)}$ represents the estimated reliability index at the i-th iteration.

The threshold is set to $\bar{\varepsilon}_{BS} = 0.005$ and convergence is considered only when this criterion is respected within three consecutive iterations.

Combined stopping criterion: This stopping criterion is simply a combination of the previous two. Convergence is assumed when the two criteria in Eq. 2 and 3 are met within two consecutive iterations.

4.2.1   Other common settings

Beside the method-specific settings introduced in the previous paragraph, others, which are common to all methods, need to be defined. These are mainly related to the initial experimental design which is drawn using the Latin hypercube sampling (LHS) method (McKay et al., 1979). The number of initial ED points is set to $\max(10, 2M)$, where M is the problem dimensionality. This allows one to ensure a minimum of 10 points for low-dimensional problem while at the same time making sure that there are enough sample points w.r.t. the dimension when the latter increases. Similarly at the other end of the

spectrum, the number of sample points is limited and only a maximum of $100 + 10M$ points can be added during the enrichment process. This number appears realistic for classical costly simulators used in engineering.

4.3 Criteria for the evaluation of the strategies

To properly compare different reliability analysis strategies, a performance measure needs to be defined. Due to the inherent complexity of the problem, we compare our benchmark results in terms of several different measures that take into account different performance metrics. Perhaps the most straightforward measures are i. how close the reliability estimate is to the reference and ii. how many points are needed to reach it. Focusing on the reliability index rather than the failure probability, a first criterion can be simply computed using the following relative reliability error estimator:

$$\varepsilon_{\beta_{i,j}}^{(k)} = \left| \frac{\widehat{\beta}_{i,j}^{(k)} - \beta_{\mathrm{ref},j}}{\beta_{\mathrm{ref},j}} \right|, \tag{4}$$

where $\widehat{\beta}_{i,j}^{(k)}$ denotes the reliability index resulting from the k-th replication of the i-th strategy applied to the j-th problem and $\beta_{\mathrm{ref},j}$ is the reference solution for the j-th problem. As a reminder, this benchmark comprises a set of 20 problems solved using 39 strategies, each repeated 15 times.

The criterion presented in Eq. 4 measures the accuracy of the resulting reliability index estimate. Additionally, we need to also assess the efficiency of the method, which simply relates to the number of model evaluations N_{eval} necessary to converge. The lower N_{eval}, the better the strategy. However N_{eval} alone is not a sufficient measure of the strategy efficiency, as premature convergence may occur. To avoid this, we will consider only solutions whose relative error, as computed in Eq. 4, is below a threshold arbitrarily set at 0.05 when ranking w.r.t. N_{eval}. Those with larger error will be automatically ranked in the last position, regardless of the number of model evaluations needed to converge.

Ideally, both criteria should be as low as possible, but they are by construction conflicting. Finding the best approach would therefore mean finding a good trade-off between relative error and computational cost. We therefore propose here a third criterion that combines these two criteria in one, making the ranking easier:

$$\Delta_{i,j}^{(k)} = \varepsilon_{\beta_{i,j}}^{(k)} \frac{N_{\mathrm{eval},i,j}^{(k)}}{N_{\mathrm{med},j}}, \tag{5}$$

where $N_{\mathrm{med},j}$ is the median number of model evaluations considering all strategies, repetitions included, to solve the j-th problem (in total, there are $15 \times 39 = 585$ runs for each problem).

All the 39 strategies are compared with each other and a ranking is established based on the three criteria defined above. The main goal of such a ranking is to find out if one strategy is consistently better than the others. If no such strategy were to be found, next is to find whether there are methods within each module that are consistently better than

18

the others. Finally, the ranking will be used to assess whether given methods are better when applied to a specific feature of the problem at hand, *i.e.*, dimensionality and failure probability magnitude.

4.4 Methods ranking over different problems

4.4.1 Ranking of the strategies

At first, we compare different strategies considering all the criteria previously defined, to which we add as reference two plain simulation methods, *i.e.*, importance sampling and subset simulation (the total number of reliability analysis runs becomes then $12,300$). They provide us with reference results without surrogates and serve as a benchmark baseline. For importance sampling, the MPFP is found using FORM and the MCS sample set is of size 10^3. For subset simulation, the subset sample set is of size 10^3 while p_0 is set to 0.1.

Additionally to the ranking, the robustness of each strategy is assessed. For each problem and replication, we observe whether a given strategy is within a certain distance from the best solution w.r.t. a chosen criterion. For the criterion "number of model evaluations", the distance is set to $\{2, 3, 5\} \times N_{\mathrm{eval}}^*$ where N_{eval}^* is the smallest number of model evaluations among strategies whose relative error is below the threshold of 0.05. For the "relative error" (resp. Δ-criterion), the distance is measured as $\{5, 10, 20\} \times \varepsilon_\beta^*$ (resp. $\{5, 10, 20\} \times \Delta^*$) where ε_β^* (resp. Δ^*) is the smallest relative error (resp. Δ value) among all strategies. This count is aggregated over all problems and replications (in total, $20 \times 15 = 300$ analyses) and given in terms of percentage as illustrated by the bars in Figures 5, 6 and 7. The mid-distance (*i.e.*, $3\,N_{\mathrm{eval}}^*$, $10\,\varepsilon_\beta^*$ and $10\,\Delta^*$) is used to rank the methods in these figures (the best solutions are in the upper positions).

In general, the larger the bars, the more robust and accurate the associated method is. For instance looking at the first line of Figure 5, the second bar shows that for the combination PCK + SuS + EFF + β-stability criterion and considering all the problems, the number of model evaluations required to converge in 69% of the repetitions is below 2 times the smallest number of model evaluations achieved for each given problem. The largest bar shows that this ratio increases to 88% when considering a threshold within 5 times the best achieved number of model evaluations. Finally, in each of these figures, the smallest and darkest bar represents the percentage of times a strategy was ranked first for a given experimental design.

Ranking with respect to the number of model evaluations

The first criterion we consider is the number of model evaluations, whose results are shown in Figure 5. As expected, when considering only the number of model evaluations, the direct solutions (*i.e.*, subset simulation and importance sampling without the use of surrogates) rank last. The most robust and efficient solution w.r.t. the number of model

evaluations is the combination of PCK with subset simulation, EFF and β-stability stopping criterion. Overall, considering the 10 best solutions, PCE or PC-Kriging as surrogates, subset simulation as reliability estimation algorithm and β-stability as stopping criterion seem to dominate. Regarding the learning function, there is no clear top performer as they all appear at least twice in the first ten positions.
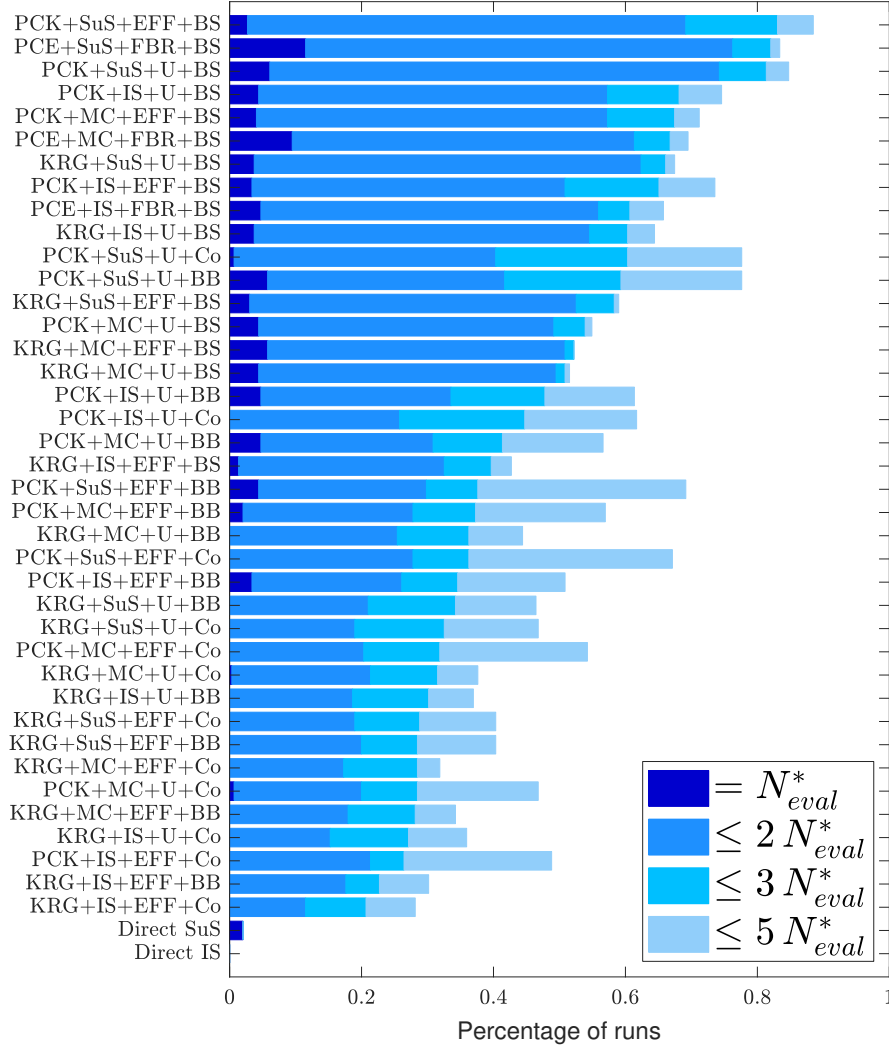

Figure 5: Ranking of the strategies w.r.t. N_{eval}. The overall ranking is based on the number of times the method performs within $3\,N_{\mathrm{eval}}^*$.

Ranking with respect to the relative error

The next criterion we consider is the relative error as illustrated in Figure 6. Here the direct solutions (without surrogates) are better ranked than with the previous criterion but they still rank worse than more than half of the methods considered. The better performance of surrogate-based methods is due to the "overkill" setup of the reliability solvers used in conjunction with the surrogates. As explained in Section 3.1.2, the computational efficiency of the surrogates allows one to use reliability solver configurations that maximize accuracy

and minimize the stochastic uncertainty in the reliability estimator, without the traditional trade-offs associated. This shows that the use of surrogate models as an instrument for the exploration of the random input space can lead to results at least as equally accurate as a direct solution, *i.e.*, without the use of surrogates.

Regarding the best strategy, we can observe a few differences with the previous ranking. The overall most robust and efficient strategy is the combination of PC-Kriging with subset simulation, deviation number U and the combined stopping criterion. PCE is not so well classified when focus is put solely on accuracy. PC-Kriging is dominating the top ranking with a few occurences of Kriging. As far as reliability estimation algorithm and learning function are concerned, subset simulation and the deviation number U are preferred. Finally, regarding the stopping criterion, β-stability (Eq. 3) seems not to favor accuracy, in sharp contrast to the combined criterion, which now appears in the top performing combinations.
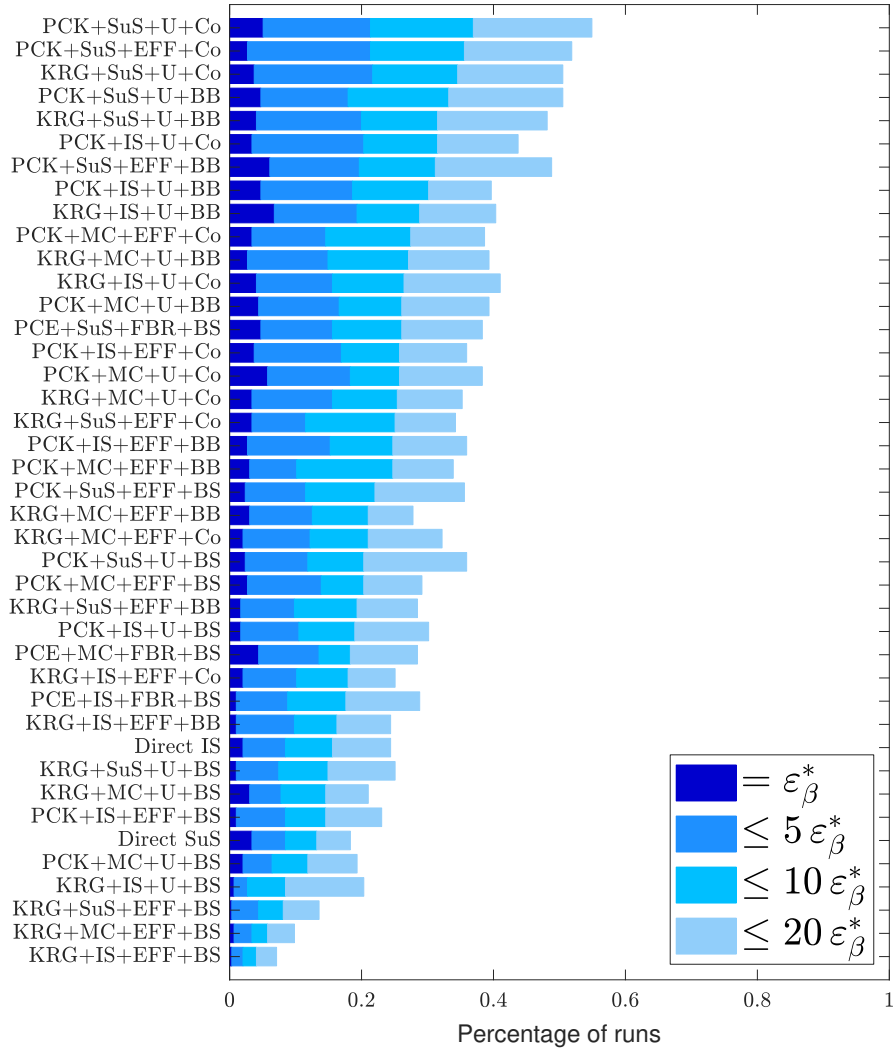

Figure 6: Ranking of the strategies w.r.t. ε_β. The overall ranking is based on the number of times the method performs within $10\,\varepsilon_\beta^*$ (one order of magnitude).

Ranking with respect to the Δ-criterion

The last criterion considered is Δ (Figure 7, Eq. 5), which as expected results in a combination of the two previous rankings. First the direct solutions are penalized by their relatively large number of model evaluations and rank again in the last two positions. Overall, this criterion favours solution accuracy, because the relative error in Eq. 4 can vary orders of magnitudes, while the range of variation of the number of model evaluations is not equally large (remember that the allowed number of model evaluations is limited to $100 + 10M$). Therefore the latter can only help make a difference within strategies that already lead to roughly the same accuracy.
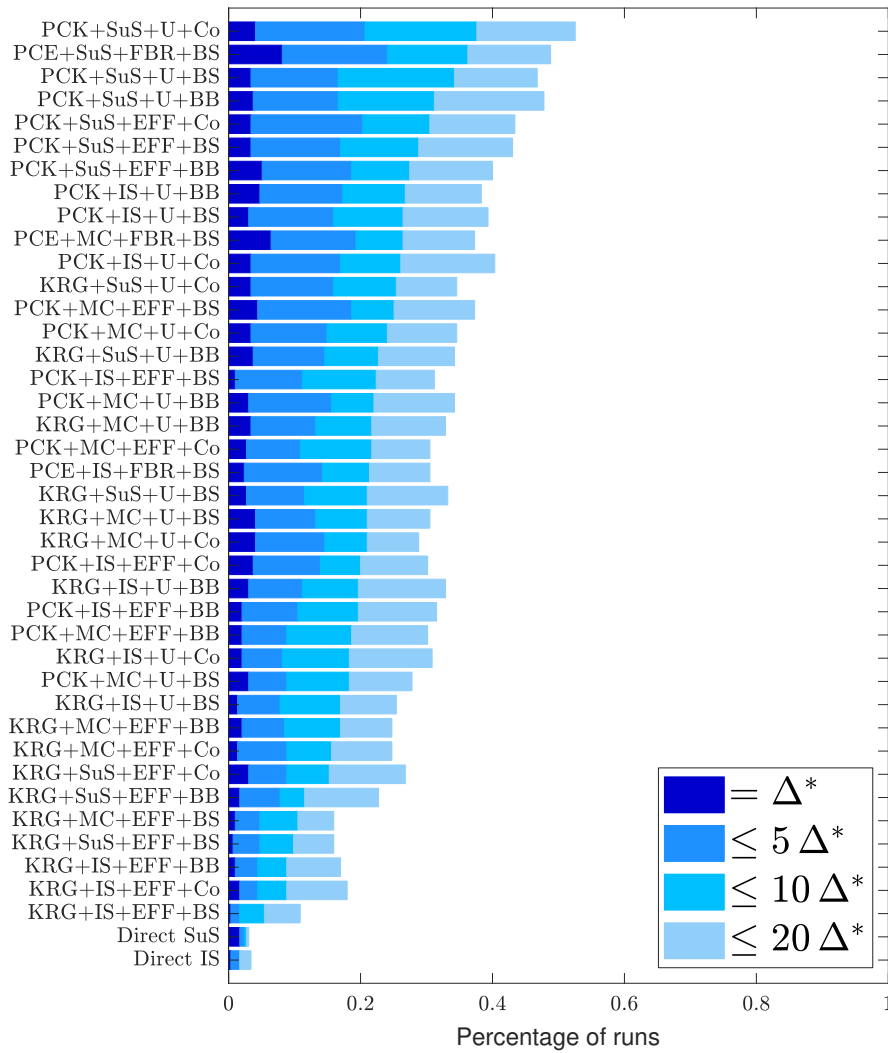

Figure 7: Ranking of the strategies w.r.t. Δ. The overall ranking is based on the number of times the method performs within $10\,\Delta^*$ (one order of magnitude).

As evidenced by Figures 5, 6 and 7, there is not a single strategy that outperforms all others in every benchmark. However, some trends are emerging from these results. The next section dives deeper into the specific methods selected for each module.

4.4.2   Ranking of the methods within each module

In the previous section, we analyzed the strategies as a block, now we split them into their four components and perform the same statistical analysis. The strategies are once again ranked for each problem and replication and we count the number of occurrences of each method in a given ranking. The results are summarized in Figures 8 and 9, where the percentage of times a given method is the best is shown in the last group of bars of each panel. To assess the variability in the ranking we also count the number of times a given method is within the first 5, 10 or 20 positions. Despite some minor variations in the share of each method in the top positions, the ranking remains unchanged if considering either the Δ-criterion or the relative error. In terms of surrogate models, PC-Kriging is the best performing choice, as it accounts for roughly half the occurrence in the best rankings. The reliability module is the most balanced, even though subset simulation shows a slight margin over the two others. In terms of learning function, the deviation number U outperforms both the expected feasibility function (EFF) and its PCE counterpart (FBR).

The convergence criterion is the only module, the results of which differ depending on the ranking criterion considered. The best method seems to be β-stability when it comes to the Δ-criterion. However this turns to β-bounds when considering the relative error. The explanation is simply that β-stability converges faster than β-bounds. Hence, when accuracy is the prior concern, the second criterion is more suitable. However, when the computational budget is limited, β-stability is a more appropriate convergence criterion.
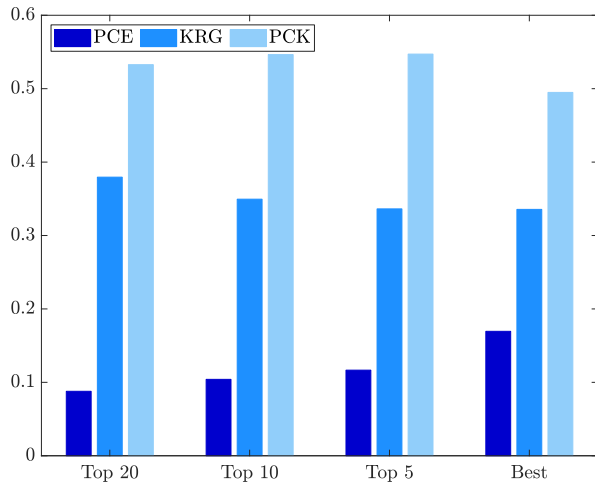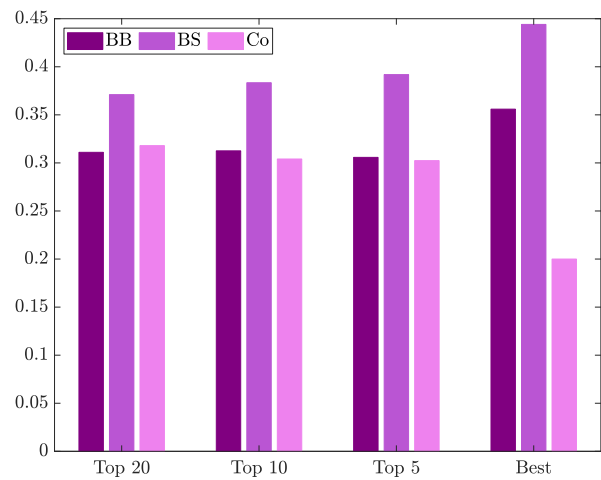
(a) Surrogate model
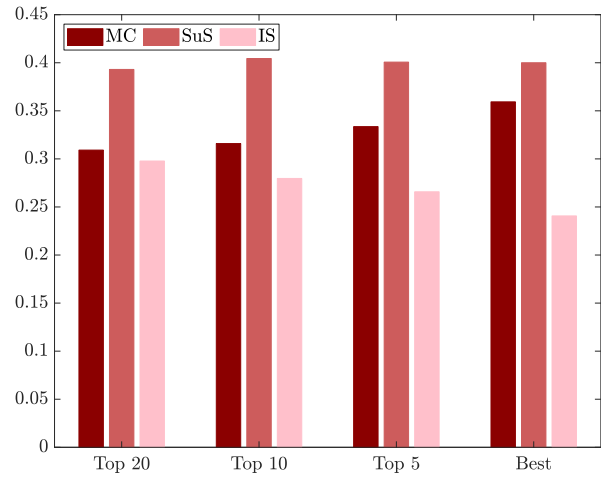
(b) Reliability estimation algorithm
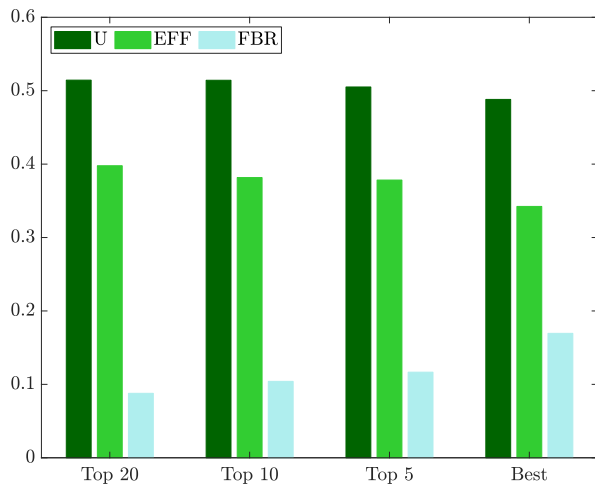
(c) Learning function

(d) Stopping criterion

Figure 8: Relative number of times a given method is among the top 20, top 10, top 5 or is the best. Ranking is made w.r.t. to the Δ-criterion.

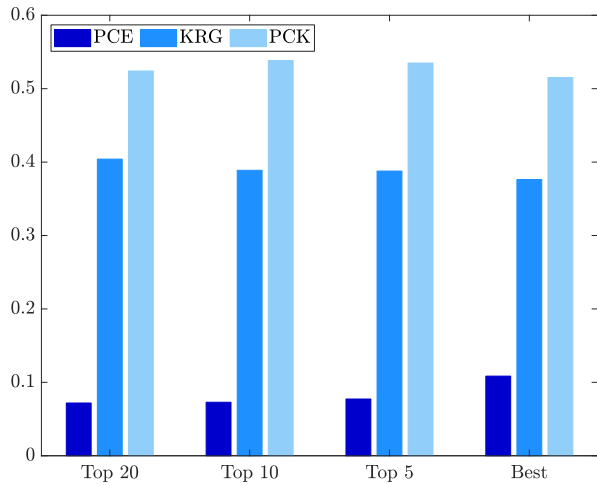(a) Surrogate model

(b) Reliability estimation algorithm

(c) Learning function
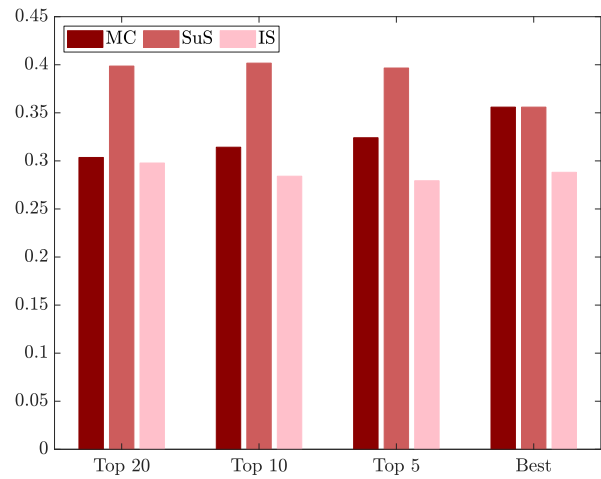
(d) Stopping criterion
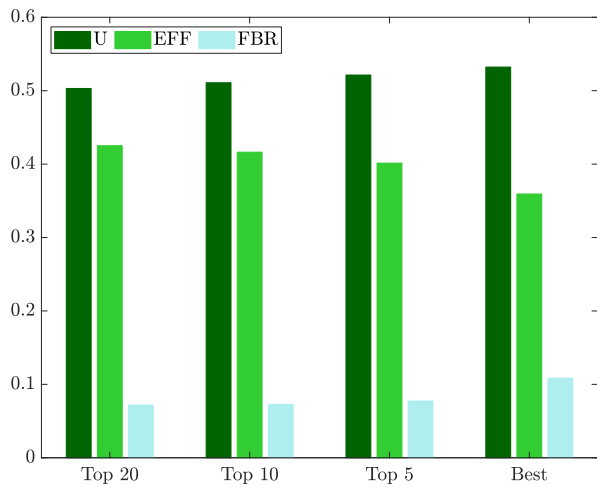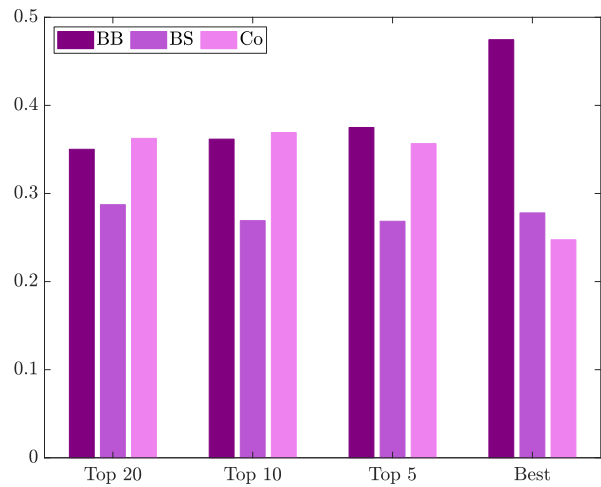
Figure 9: Relative number of times a given method is among the top 20, top 10, top 5 or is the best. Ranking is made w.r.t. the relative error ε_β

4.5 Aggregation of the results w.r.t. the problem features

Using the ranking from the previous two sections, it is clear now which methods lead on average to the best performance in terms of accuracy and efficiency. In this section, we go a step further and try to determine if these methods behave in the same way as a function of two selected features of the problem, namely input dimensionality and magnitude of the probability of failure, or if their performances are intrinsically linked to the type of problem at hand. We split the benchmark into low- ($M < 20$) and high-dimensional ($M \geq 20$) problems, as well as small ($\beta_{\text{ref}} < 3.5$) and high ($\beta_{\text{ref}} \geq 3.5$) reliability indices.

4.5.1 Performance with respect to dimensionality

Figures 10 and 11 respectively show the Δ-criterion values and relative errors aggregated over all problems and then split for each method. The horizontal dotted black line represents the median over all problems. In each panel, the boxplots represent the aggregated median results (over all 15 replications) considering all (blue), low- to medium- (magenta) and high-dimensional problems (cyan).

Starting with the surrogate models and looking at the Δ-criterion (Figure 10a), we observe that PC-Kriging does not seem to be strongly affected by the problem dimensionality, as in all cases the conditional median remains slightly below the overall median, while PCE seems to improve its relative performance in higher dimensional problems. Kriging on the other hand performs noticeably poorly. The same trend is observed with the accuracy criterion ε_β (Figure 11a).

Next are the reliability estimation algorithms (Figure 10b) and as expected Monte Carlo simulation is essentially insensitive to the dimension. Subset simulation performs slightly worse in high-dimension but not as much as importance sampling. The latter becomes slightly worse when considering the purely accuracy-oriented criterion (Figure 11b).

Regarding the learning functions (Figure 10c and 11c), the deviation number (U) also does not seem to be noticeably affected by the dimension, contrary to the expected feasibility function (EFF) which gets considerably worse as the dimension increases. The fraction of bootstrap replicates (FBR) mirrors the behavior of PCE as there is a one-to-one mapping between the two. Their performance gets somehow better for high-dimensional problems when considering either of the criteria.

Finally, as already observed earlier, contradicting conclusions are obtained when considering either the Δ-criterion value (Figure 10d) or the relative error (Figure 11d) for the stopping criterion. In the former case, β-stability seems to be the best option when dimension increases whereas in the latter β-bounds, or even the combined criterion, seem to be the best option. This again can be explained by the fact that β-bounds is a stricter convergence criterion, especially in high-dimensional cases where the Kriging variance hardly shrinks as the experimental design is enriched.
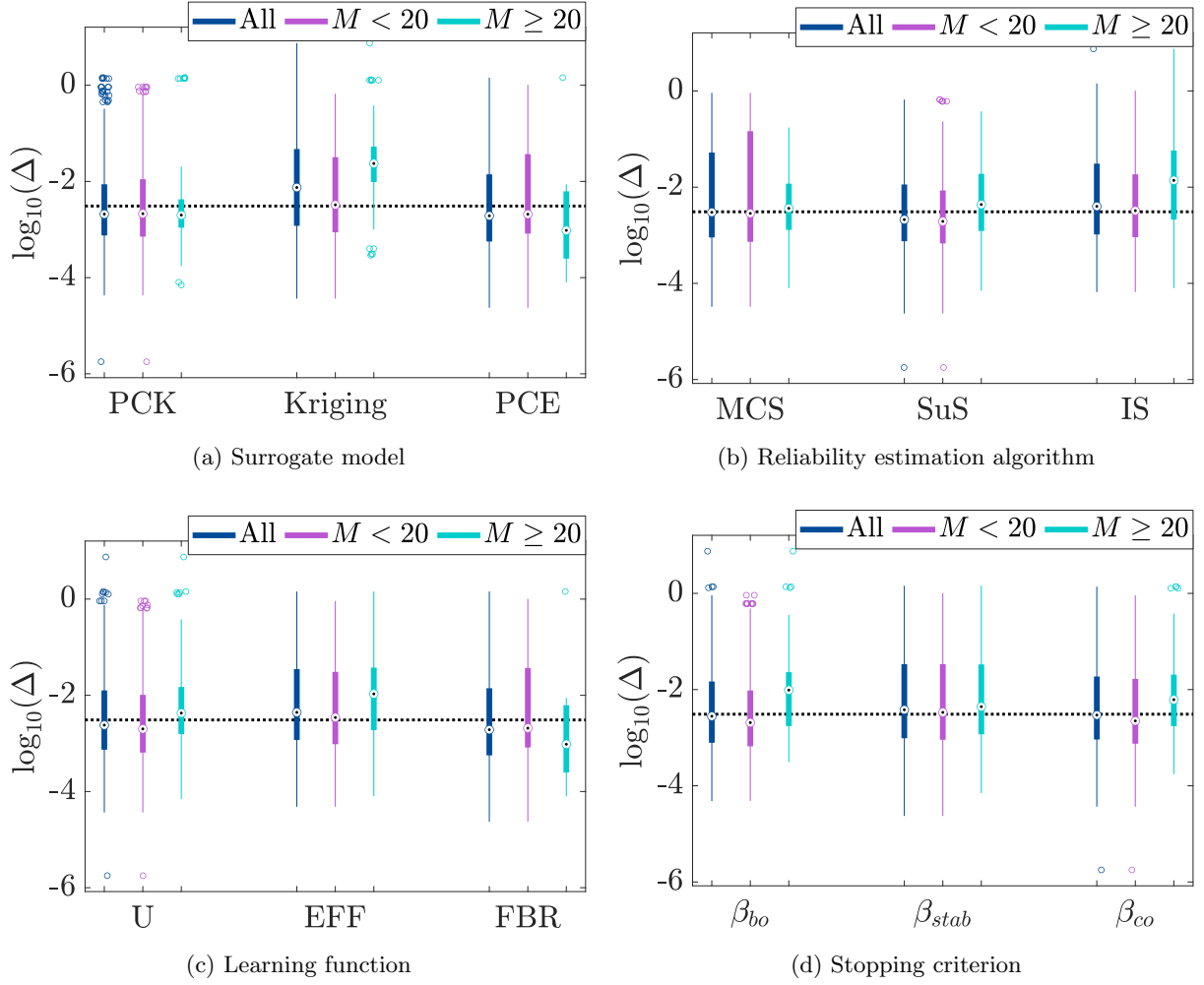
26

Figure 10: Different methods compared w.r.t. the combined criterion Δ with problems split in two: low- ($M < 20$) and high-dimensional ($M \geq 20$).

4.5.2 Performance with respect to the magnitude of the failure probability

In contrast to the dimensionality cases where some methods were not particularly affected by the increase in dimensions, the level of the reliability index seems to affect pretty much all the methods. More specifically, the extremely low failure probability cases result on average in systematically poorer performances both in terms of relative error and Δ (See Figures 12 and 13). As expected, Monte Carlo simulation performs considerably worse than its alternatives. The reason is that even in its "overkill" setting, the maximum number of model evaluations was set to 10^7, hence limiting the statistical estimator variance for extremely low failure probability problems.

(a) Surrogate model

(b) Reliability estimation algorithm

(c) Learning function

(d) Stopping criterion
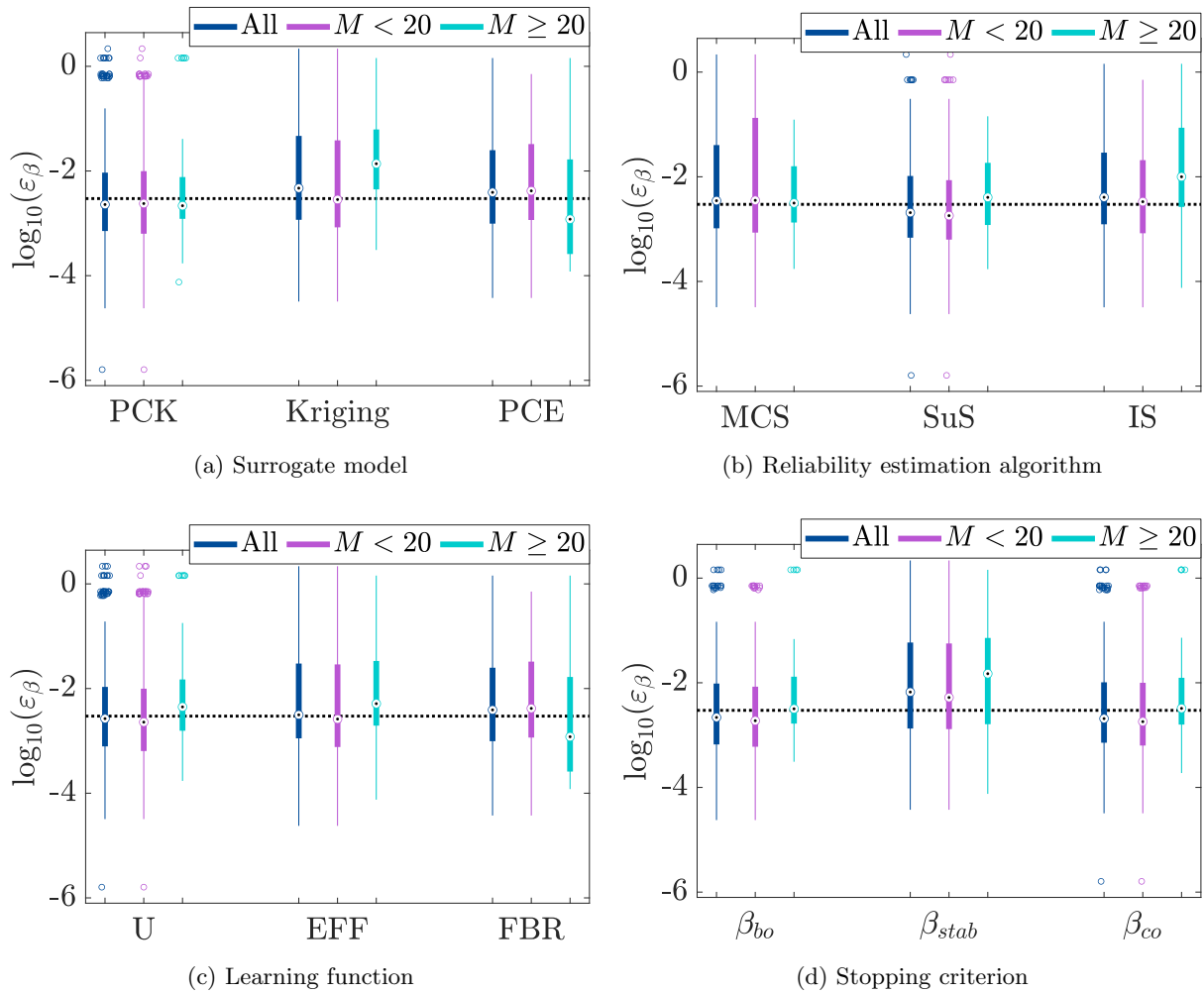
Figure 11: Different methods compared w.r.t. the relative error ε_β with problems split in two: low- ($M < 20$) and high-dimensional ($M \geq 20$).

(a) Surrogate model

(b) Reliability estimation algorithm

(c) Learning function

(d) Stopping criterion
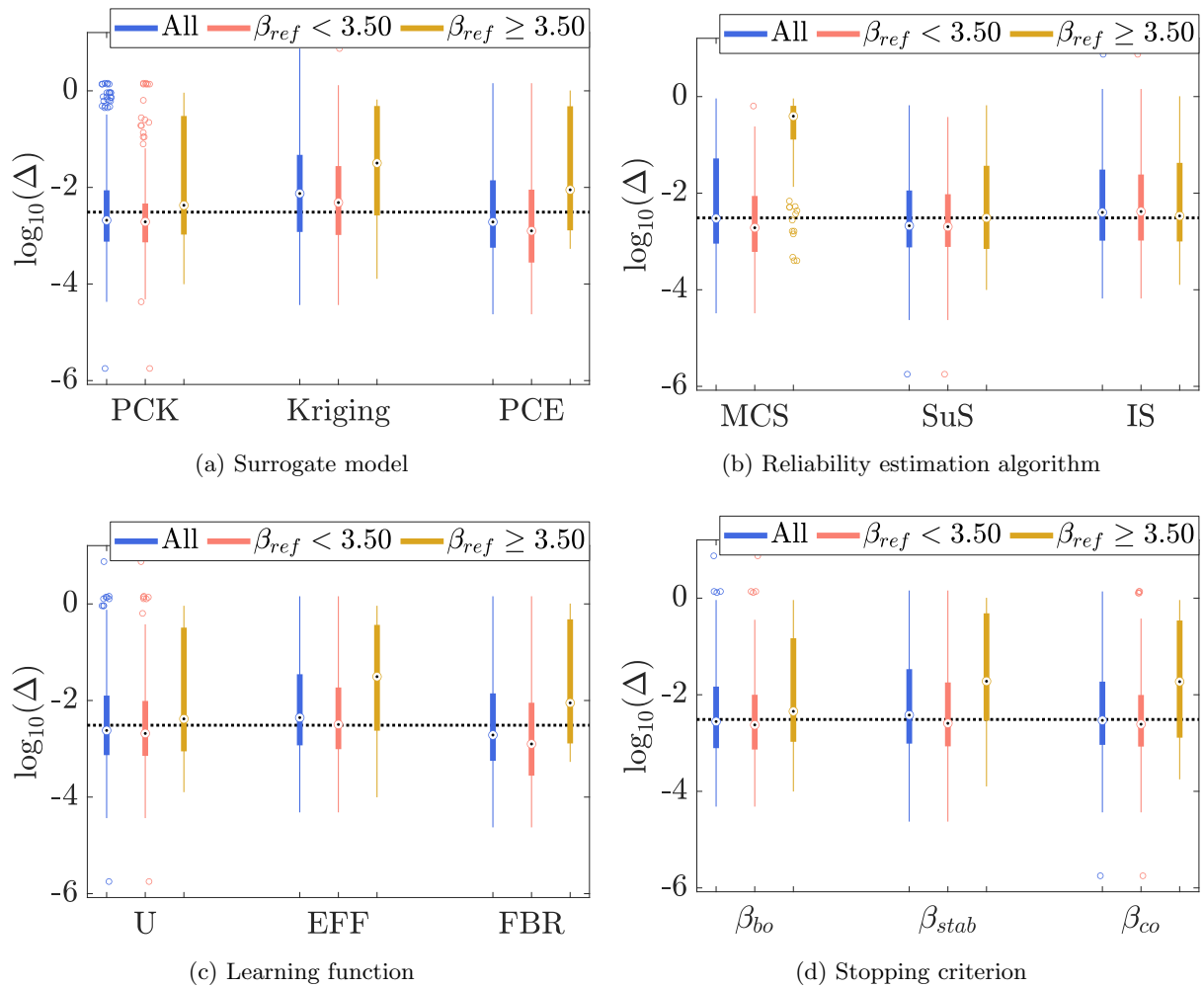
Figure 12: Different methods compared w.r.t. the combined criterion Δ with problems split in two: small ($\beta_{\text{ref}} < 3.5$) and large ($\beta_{\text{ref}} \geq 3.5$) reliability indices.

(a) Surrogate model

(b) Reliability estimation algorithm

(c) Learning function

(d) Stopping criterion
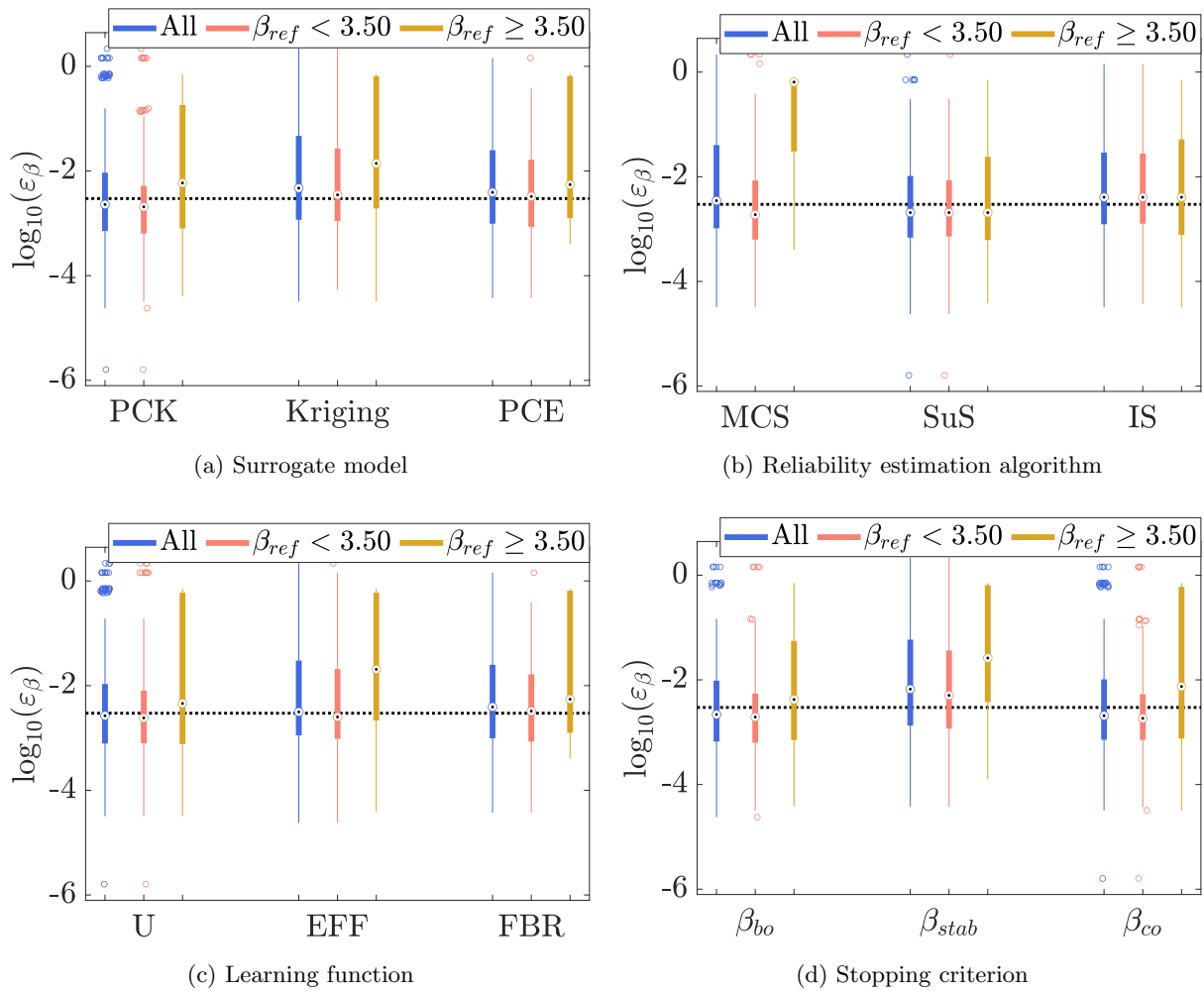
Figure 13: Different methods compared w.r.t. the relative error ε_β with problems split in two: small ($\beta_{\text{ref}} < 3.5$) and large ($\beta_{\text{ref}} \geq 3.5$) reliability indices.

4.6 Results with strategies aggregated over different problems

Now that we have a clear picture of the performances of each strategy as a whole and more particularly of different methods w.r.t. different problems and their features, let us have a look at the overall performance of all methods for each problem. The aggregated relative error and Δ values for all 20 problems (ordered in increasing dimensions) are shown in Figure 14 as violin plots, *i.e.*, boxplots with an additional indication of the probability density of the data. The three problems with the overall worst median performances are highlighted in red. The first observation from Figure 14 is that, regardless of the problem, there are still at least a few strategies that lead to good performances. As a matter of fact, the median values of both criteria are for most problems below the level arbitrarily set at 10^{-2}.

(a) Δ-criterion
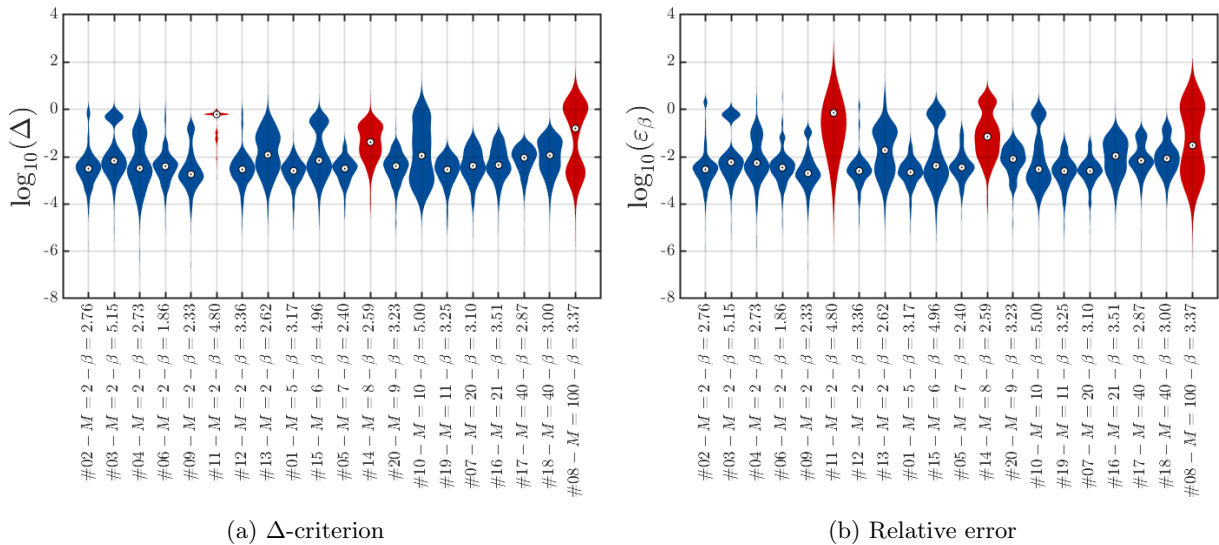
(b) Relative error

Figure 14: Results aggregated for all solutions strategies and shown for each problem considering the combined criterion and the relative error. The three problems with the worst median results are highlighted in red.

To end this benchmark, the methods which were ranked most often best are compared with the overall pool of strategies. This is shown in Figure 15. More specifically, Figures 15a and 15b show comparison on the Δ-criterion and relative error respectively, with respect to the combination of PC-Kriging with subset simulation, deviation number U and Combined criterion. This is the approach that was consistently best both in the strategies and methods ranking. In both cases, we can observe that this choice of methods improves the performance on most problems and also reduce the scatter in the results. This graph also confirms the no-free-lunch principle exhibited by this benchmark as the overall best strategy is not necessarily the best for each problem.
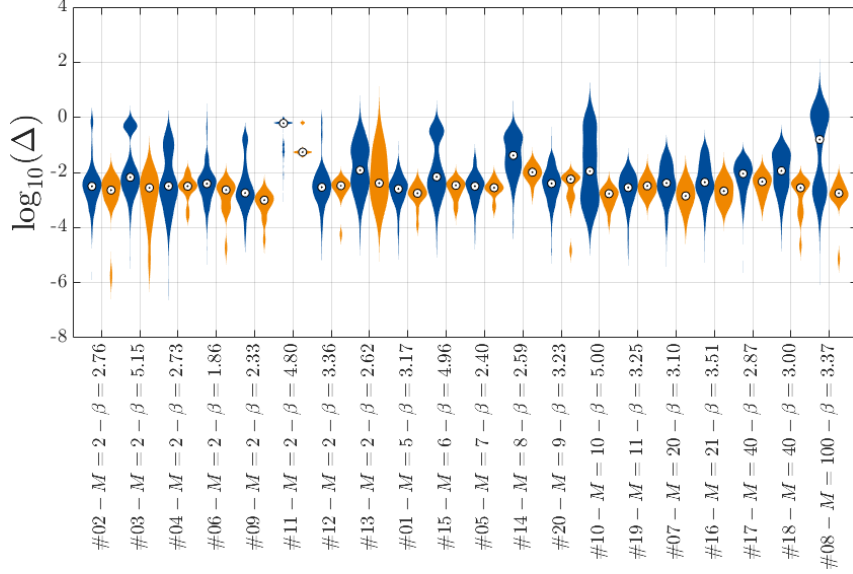
(a) Δ with PCK+SuS+U+Co
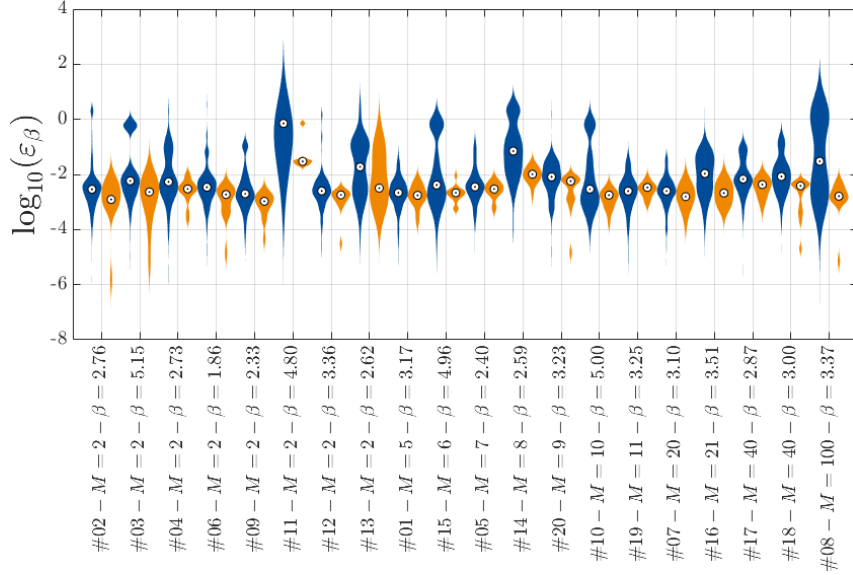
(b) ε_β with PCK+SuS+U+Co

Figure 15: Results aggregated for all solutions strategies (blue) compared to those corresponding to the overall best strategy (orange) w.r.t. to Δ and ε_β

4.7   Investigation of the most difficult problems

In this section, we closely look at the three problems that were most difficult to solve using the proposed ALR methods. By doing so, we note that some of these problems could not have been solved correctly even when considering the direct reliability estimation algorithms (without surrogates). Problem #11 contains four disjoint failure regions, which makes it impossible to solve using the standard importance sampling configuration considered in this benchmark. This also applies to the problem #8, which is spherical along the variables $\{X_2, X_3, \ldots, X_{100}\}$. Furthermore, problem #11 cannot be solved with direct Monte Carlo

simulation due to the low reference failure probability $p_f = 7.83 \cdot 10^{-7}$, which would require $N_{\text{MCS}} \approx 10^{9-10}$ samples to achieve sufficient accuracy.

These observations can be confirmed by Figure 16 which shows the results corresponding to the solution of the benchmark problems using the three reliability estimation algorithms, with their overkill settings and without the aid of surrogate models. The red lines correspond to problems whose relative error is larger than 1. Using the threshold for acceptable accuracy at 10^{-2} as in the previous sections, we can see that Monte Carlo simulation fails to solve problems #10, #11, and #15 while importance sampling fail with problems #8, #11, #13, #18. This illustrates an important result of the benchmark, namely that the surrogate models were never the main cause of failure of the ALR strategies.
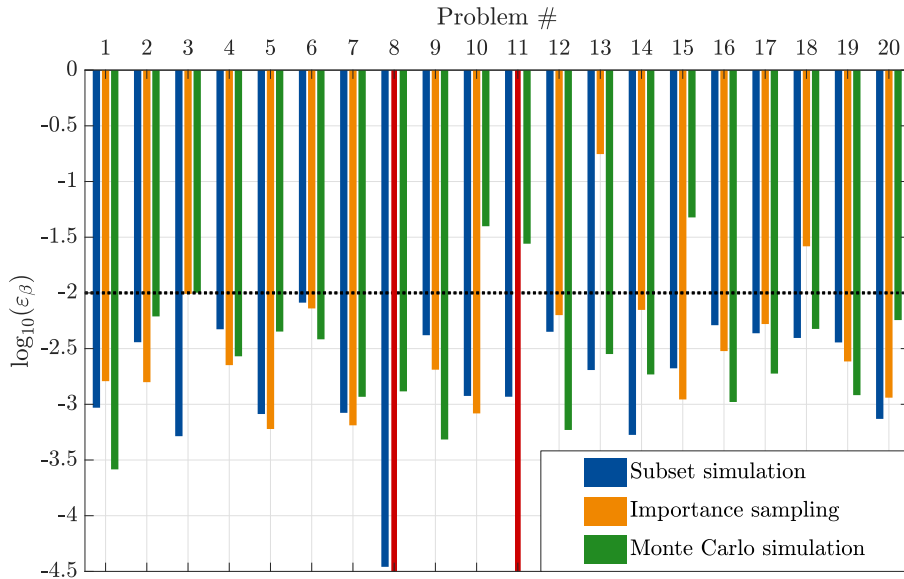
Figure 16: Solution of the 20 problems using the three "overkill" reliability settings without surrogate models.

5   Research questions and recommendations

Our extensive benchmarking exercise allows us to showcase the framework introduced in this paper. We compared strategies built using this generalized active learning reliability framework with respect to various metrics. In this section, we summarize the findings from this benchmark and set up some guidelines as to the choice of the methods within the modules of the framework.

The first question that we set out to answer in this benchmark was whether there was one strategy that would consistently outperform the others with respect to all metrics and throughout all problems. The answer is clearly that there is no such strategy (no-free-lunch principle), yet using a surrogate model is always beneficial. For each analysis, the best strategy would vary according to the metric of interest and the type of problems. A natural

follow-up question is whether any pattern could be uncovered by the benchmark. Without prior knowledge about the problem and using a non-intrusive combination of different methods, the conclusion is sharp: the best results are most likely obtained by combining PC-Kriging, subset simulation, deviation number U and β-bounds or combined stopping criterion.

This conclusion is obviously only limited to the methods selected for the benchmark. However, we may in some cases extrapolate to general guidelines by considering the characteristics of the different methods in relation to various features of the problems solved in the benchmark. We made the following observations regarding each module of the framework:

- **Surrogate model:** From this benchmark, it is clear that Kriging, which is the most used method in the literature, is not necessarily the best choice for active learning reliability. The main reason is that it performs poorly in fairly high-dimensional problems. PC-Kriging, which combines the global and local approximations capabilities of PCE and Kriging respectively, has shown consistently high performance throughout this benchmark. PCE has shown to perform better than Kriging for high-dimensional problems and this also benefits PC-Kriging to some extent. Finally, PC-Kriging possesses the same built-in error measure as Kriging, which makes it compatible with the various learning functions that take advantage of the Kriging variance.

- **Reliability estimation algorithm:** From this benchmark, it is clear that the active learning scheme inherits the pros and cons of the reliability estimation algorithm it uses, although this is somewhat mitigated by the ability to use "overkill" configurations. There is a direct correlation between the ability of a reliability estimation method to solve a given type of problems and the performance of the associated active learning scheme. Once again, Monte Carlo simulation, which is widely used in the literature, has proven not to be the best choice of algorithm according to the benchmark results. The introduction of surrogates does not eliminate the weakness of Monte Carlo simulation indeed when it comes to problems with extremely small failure probabilities. This is true also for the other algorithms, as importance sampling still showed limitations with problems where multiple failure regions exist. Note that other importance sampling densities may be used in practice when such a multiple-failure feature is known in advance (this was not considered in the benchmark). Regardless of the three algorithms used in the benchmark, it seems that the best course of action is to choose whatever algorithm the analyst thinks is best given his *a priori* knowledge on the problem. Finally, an important result highlighted by the benchmark is that over-calibrating the reliability estimation algorithm is beneficial and can lead to even better results than the non-surrogate equivalent using conventional settings. Therefore, surrogate models should be used to fully harness the benefits of the most sophisticated reliability estimation methods.

- **Learning function:** Considering this benchmark, the deviation number U seems to

outperform EFF. The fraction of bootstrap replicates (FBR) is better than U when it comes to high-dimensional problems but loses its advantage when considering problems with small failure probabilities. All the methods selected for the benchmark are however very similar. It would have been interesting in this specific case to explore other type of learning functions, *e.g.*, those that also include the PDF of the input random variables.

- **Stopping criterion:** The stopping criteria selected for the benchmark are based on the accuracy of the P_f (or β) estimate rather than on the learning function. This finding is consistent with those identified in previous contributions (Schöbi et al., 2016). This benchmark has however highlighted the difference between stopping criteria based on the local accuracy of the surrogate with those based on the stability of the limit-state surface. The former are shown to provide good results overall but to perform poorly in high-dimensional problems. This can be explained by the difficulty to sufficiently reduce the Kriging variance for high-dimensional problems. In contrast, stability criteria are somewhat more efficient for high-dimensional problems. However, they are prone to premature convergence. Combining these two criteria did not lead to a noticeable increase in performance.

To further summarize the observations made by analyzing the results of the benchmark, Table 2 gives a few recommendations on the basis of the benchmark.

Table 2: General recommendations on the basis of the benchmark carried out in this paper. β_{bo} stands for β-bounds convergence, β_{stab} stands for β-stability convergence and β_{co} is the combined criterion.

Module	Dimensionality		Failure probability magnitude	
	$M < 20$	$20 \leq M \leq 100$	$\beta \leq 3.5$	$\beta \geq 3.5$
Surrogate model	PCK	PCE	PCE/PCK	PCK
Reliability estimation algorithm	SuS	SuS	SuS	SuS
Learning function	U	FBR	FBR/U	U
Stopping criterion	$\beta_{\mathrm{bo}},\beta_{\mathrm{co}}$	$\beta_{\mathrm{bo}},\beta_{\mathrm{co}}$[1]/ β_{stab}[2]	$\beta_{\mathrm{bo}},\beta_{\mathrm{co}}$	β_{bo}

[1] When considering accuracy only ε_β.

[2] When factoring in efficiency Δ.

6   Conclusions

This paper investigated the use of active learning strategies for the solution of structural reliability problems. We first conducted a literature survey and identified an underlying and recurring scheme. This scheme was used to propose a global framework for active learning

reliability which is made of four components non-intrusively linked to each other. The four modules of this framework are surrogate model, reliability analysis, learning function and stopping criterion. We then showed that it is possible to combine various methods from each of the modules to build a wide array of viable solution strategies. On this basis, 39 solution strategies were built to solve a set of 20 selected problems. The results of this benchmark allowed us to identify patterns regarding the generalization capability of each method.

The first observation is that there is no strategy that consistently outperforms the others. We could however identify clear patterns to give recommendations on which types of methods should be preferred with regard to a given feature of the problem at hand. The flexibility of the presented framework is in this regard of great value as it allows the analyst to build tailored active learning reliability schemes.

The second observation is that there is essentially no drawbacks in using surrogate models. The latter allows one indeed to better exploit the reliability estimation algorithms through "overkill" settings.

Even though we ran an extensive benchmark a few aspects still need to be investigated. For instance, we did not explore the effect of the thresholds in the stopping criterion, nor did we explore more advanced learning functions. This includes techniques that weigh in the input PDF or allow for multiple points enrichment. By design, the scope of the analysis was limited to problems with single limit-state functions. More aspects need to be taken into account when considering multiple limit-state functions. Finally, extremely high-dimensional problems (M in the other of hundreds or even thousands) were not considered as they also would require special treatment. Finally the proposed optimal framework has been applied blindly to a structural reliability context organized by TNO (Rozsas and Slobbe, 2019), whose results are available at https://rprepo.readthedocs.io/en/latest/results.html (last accessed on October 4th, 2021). Out of 16 component- and 11 system-reliability problems, our approach was the most efficient for 24 problems. A short summary of the results is provided in A for the sake of completeness.

It is worth mentioning that the codes and the set of examples used are made publicly available through the UQLab release R1.4, such that it will be easy to extend the results presented here to new problems.

References

Au, S. K. and J. L. Beck (2001). Estimation of small failure probabilities in high dimensions by subset simulation. *Prob. Eng. Mech. 16*(4), 263–277.

Au, S. K. and J. L. Beck (2003). Subset simulation and its application to seismic risk based on dynamic analysis. *J. Eng. Mech. 129*(8), 901–917.

Balesdent, M., J. Morio, and J. Marzat (2013). Kriging-based adaptive Importance Sampling algorithms for rare event estimation. *Structural Safety 44*, 1–10.

Basudhar, A. and S. Missoum (2008). An improved adaptive sampling scheme for the construction of explicit boundaries. *Struct. Multidisc. Optim. 42*(4), 517–529.

Bect, J., L. Li, and E. Vazquez (2017). Bayesian subset simulation. *SIAM/ASA J. Unc. Quant. 5*, 762–786.

Bichon, B. J., M. S. Eldred, L. Swiler, S. Mahadevan, and J. McFarland (2008). Efficient global reliability analysis for nonlinear implicit performance functions. *AIAA Journal 46*(10), 2459–2468.

Blatman, G. and B. Sudret (2010). An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis. *Prob. Eng. Mech. 25*, 183–197.

Blatman, G. and B. Sudret (2011). Adaptive sparse polynomial chaos expansion based on Least Angle Regression. *J. Comput. Phys 230*, 2345–2367.

Bo, X. and T. HuiFeng (2018). A robust and efficient structural reliability method combining radial-based importance sampling and Kriging. *Sci. China Technol. Sci. 61*, 724–734.

Bourinet, J.-M. (2017). Anisotropic-kernel-based support vector regression for reliability assessment. In *Proc. 12th International Conference on Structural Safety and Reliability (ICOSSAR), August 6-10, 2017, Vienna, Austria.*

Bourinet, J.-M. (2018). *Reliability analysis and optimal design under uncertainty - Focus on adaptive surrogate-based approaches.* Université Blaise Pascal, Clermont-Ferrand, France. Habilitation à diriger des recherches, 243 pages.

Bourinet, J.-M., F. Deheeger, and M. Lemaire (2011). Assessing small failure probabilities by combined subset simulation and support vector machines. *Structural Safety 33*(6), 343–353.

Bucher, C. (2009). Asymptotic sampling for high-dimensional reliability analysis. *Prob. Eng. Mech. 24*, 504–510.

Bucher, C. and U. Bourgund (1990). A fast and efficient response surface approach for structural reliability problems. *Structural Safety 7*, 57–66.

Cadini, F., F. Santos, and E. Zio (2014). An improved adaptive Kriging-based importance technique for sampling multiple failure regions of low probability. *Reliab. Eng. Syst. Saf. 139*, 109–117.

Cheng, K. and Z. Lu (2020). Active learning polynomial chaos expansion for reliability analysis by maximizing expected indicator function predictor error. *Structural Safety 82*, 1–13.

Chojazyk, A. A., A. P. Teixeira, L. C. Neves, J. B. Cardoso, and C. Guedes Soares (2015). Review and application of artificial neural networks models in reliability analysis of steel structures. *Structural Safety 52*, 78–89.

Constantine, P. G., E. Dow, and Q. Wang (2014). Active subspace methods in theory and practice: Applications to K riging surfaces. *SIAM J. Sci. Comput. 36*(4), A1500–A1524.

Deheeger, F. and M. Lemaire (2007). Support vector machines for efficient subset simulations: [2]SMART method. In *Proc. 10th Int. Conf. on Applications of Stat. and Prob. in Civil Engineering (ICASP10), Tokyo, Japan.*

Der Kiureghian, A. and M. de Stefano (1990). An efficient algorithm for second-order reliability analysis. Technical Report UCB/SEMM-90/20, University of California, Berkeley. Dept of Civil and Environmental Engineering, University of California, Berkeley.

Ditlevsen, O. and H. Madsen (1996). *Structural reliability methods.* J. Wiley and Sons, Chichester.

Ditlevsen, O., R. E. Melchers, and H. Gluver (1990). General multi-dimensional probability integration by directional simulation. *Comput. Struct. 36*(2), 355–368.

Dubourg, V., B. Sudret, and J.-M. Bourinet (2012). Meta-model-based importance sampling for reliability sensitivity analysis. In *Proc. 11th ASCE Specialty Conference on Probabilistic Mechanics and Structural Reliability, Notre Dame, USA.*

Dubourg, V., B. Sudret, and F. Deheeger (2013). Metamodel-based importance sampling for structural reliability analysis. *Prob. Eng. Mech. 33*, 47–57.

Echard, B., N. Gayton, and M. Lemaire (2011). AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation. *Structural Safety 33*(2), 145–154.

Echard, B., N. Gayton, M. Lemaire, and N. Relun (2013). A combined importance sampling and Kriging reliability method for small failure probabilities with time-demanding numerical models. *Reliab. Eng. Syst. Safety 111*, 232–240.

Efron, B., T. Hastie, I. Johnstone, and R. Tibshirani (2004). Least angle regression. *Ann. Stat. 32*, 407–499.

Faravelli, L. (1989). Response surface approach for reliability analysis. *J. Eng. Mech. 115*(12), 2763–2781.

Fauriat, W. and N. Gayton (2017). AK-SYS: An application of the AK-MCS method for system reliability. *Reliab. Eng. Struct. Safety 123*, 137–144.

Gaspar, B., A. P. Teixeira, and C. Guedes Soares (2017). Adaptive surrogate model with active refinement combining Kriging and a trust region method. *Reliab. Eng. Syst. Saf. 165*, 277–291.

Geyer, S., I. Papaioannou, and D. Straub (2019). Cross entropy-based importance sampling using Gaussian densities revisited. *Structural Safety 76*, 15–27.

Gomes, W. J. S. (2019). Structural reliability analysis using artificial neural networks. *ASCE-ASME J. of Risk Uncertain. Eng. Syst., Part B: Mech.Eng. 5*, 1–8.

Guo, Q., Y. Liu, B. Chen, and Y. Zhao (2020). An active learning Kriging model combined with directional importance sampling method for efficient reliability analysis. *Prob. Eng. Mech. 60*, 1–9.

Hasofer, A.-M. and N.-C. Lind (1974). Exact and invariant second moment code format. *J. Eng. Mech. 100*(1), 111–121.

Hu, Z. and S. Mahadevan (2016). Global sensitivity analysis-enhanced surrogate (GSAS) modeling for reliability analysis. *Struct. Multidisc. Optim. 53*, 501–521.

Huang, X., J. Chen, and H. Zhu (2016). Assessing small failure probabilities by AK–SS: An active learning method combining Kriging and subset simulation. *Structural Safety 59*, 86–95.

Hurtado, J. E. (2004). An examination of methods for approximating implicit limit state functions from the viewpoint of statistical learning theory. *Structural Safety 26*, 271–293.

Jian, W., S. Zhili, Y. Qiang, and L. Rui (2017). Two accuracy measures of the Kriging model for structural reliability analysis. *Reliab. Eng. Sys. Safety 167*, 494–505.

Jiang, C., H. Qiu, Z. Yang, L. Chen, L. Gao, and P. Li (2019). A general failure-pursuing sampling framework for surrogate-based reliability analysis. *Reliab. Eng. Struct. Safety 183*, 47–59.

Jones, D. R., M. Schonlau, and W. J. Welch (1998). Efficient global optimization of expensive black-box functions. *J. Global Optim. 13*(4), 455–492.

Koutsourelakis, P. S., H. J. Pradlwarter, and G. I. Schuëller (2004). Reliability of structures in high dimensions, part I: algorithms and applications. *Prob. Eng. Mech. 19*, 409–417.

Kroetz, H. M., M. Moustapha, A. T. Beck, and B. Sudret (2020). A two-level Kriging-based approach with active learning for solving time-variant risk optimization problems. *Reliab. Eng. Syst. Saf. 203*, 107033.

Kroetz, H. M., R. K. Tessari, and A. T. Beck (2017). Performance of global metamodeling techniques in solution of structural reliability problems. *Adv. Eng. Softw. 114*, 394–404.

Lacaze, S. and S. Missoum (2014). A generalized "max-min" sample for surrogate update. *Struct. Multidisc. Optim. 49*, 683–687.

Lataniotis, C., S. Marelli, and B. Sudret (2017). UQLab user manual – Kriging. Technical report, Chair of Risk, Safety & Uncertainty Quantification, ETH Zurich. Report # UQLab-V1.0-105.

Lataniotis, C., S. Marelli, and B. Sudret (2020). Extending classical surrogate modeling to high dimensions through supervised dimensionality reduction: a data-driven approach. *Int. J. Uncertainty Quantification 10*(1), 55–82.

Lelièvre, N., P. Beaurepaire, C. Mattrand, and N. Gayton (2018). AK-MCSi: A Kriging-based method to deal with small failure probabilities and time-consuming models. *Structural Safety 73*, 1–11.

Lemaire, M. (1998). Finite element and reliability : combined methods by response surfaces. In G. Frantziskonis (Ed.), *Probamat-21st Century, Probabilities and Materials : Tests, Models and Applications for the 21st century*, pp. 317–331. Kluwer Academic Publishers.

Lemaire, M. (2009). *Structural reliability*. Wiley.

Leonel, E. D., A. T. Beck, and W. S. Venturini (2011). On the performance of response surface and direct coupling approachesin solution of random crack propagation problems. *Struct. Saf. 33*, 261–274.

Li, L., J. Bect, and E. Vazquez (2012). Bayesian subset simulation: a Kriging-based subset simulation algorithm for the estimation of small failure probabilities. In *11th International Probabilistic Assessment and Management Conference (PSAM11) and The Annual European Safety and Reliability Conference (ESREL 2012), Helsinki : Finland (2012)*. Curran.

Li, M. and Z. Wang (2020). Deep learning for high-dimensional reliability analysis. *Mechanical Systems and Signal Processing 139*, 1–18.

Li, X., C. Gong, L. Gu, W. Gao, Z. Jing, and H. Su (2018). A sequential surrogate method for reliability analysis based on radial basis function. *Structural Safety 73*, 42–53.

Ling, C., Z. Lu, K. Feng, and X. Zhang (2019). A coupled subset simulation and active learning Kriging reliability analysis method for rare failure events. *Struct. Multidisc. Optim. 60*, 2325–2341.

Liu, F., P. Wei, C. Zhou, and Z. Yue (2019). Reliability and reliability sensitivity analysis of structure by combining adaptive linked importance sampling and Kriging reliability method. *Chinese Journal of aeronautics 33*, 1218–1227.

Lv, Z., Z. Lu, and P. Wang (2015). A new learning function for Kriging and its applications to solve reliability problems in engineering. *Comput. Math. Appl. 70*, 1182–1197.

Marelli, S. and B. Sudret (2014). UQLab: A framework for uncertainty quantification in Matlab. In *Vulnerability, Uncertainty, and Risk (Proc. 2nd Int. Conf. on Vulnerability, Risk Analysis and Management (ICVRAM2014), Liverpool, United Kingdom)*, pp. 2554–2563.

Marelli, S. and B. Sudret (2016). Bootstrap-polynomial chaos expansions and adaptive designs for reliability analysis. In H. Huang, J. Li, J. Zhang, and J. Chen (Eds.), *Proc. 6th Asian-Pacific Symp. Struct. Reliab. (APSSRA'2016), Tongji University, Shanghai (China), May 27-31*.

Marelli, S. and B. Sudret (2017). UQLab user manual – Polynomial chaos expansions. Technical report, Chair of Risk, Safety & Uncertainty Quantification, ETH Zurich. Report # UQLab-V1.0-104.

Marelli, S. and B. Sudret (2018). An active-learning algorithm that combines sparse polynomial chaos expansions and bootstrap for structural reliability analysis. *Structural Safety 75*, 67–74.

McKay, M. D., R. J. Beckman, and W. J. Conover (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics 2*, 239–245.

Melchers, R. E. amd Beck, A. T. (2018). *Structural reliability analysis and prediction*. John Wiley & Sons.

Melchers, R. E. (1989). Importance sampling in structural systems. *Structural Safety 6*, 3–10.

Moustapha, M., B. Sudret, J.-M. Bourinet, and B. Guillaume (2016). Quantile-based optimization under uncertainties using adaptive Kriging surrogate models. *Struct. Multidisc. Optim. 54*(6), 1403–1421.

Pan, Q. and D. Dias (2017). An efficient reliability method combining adaptive support vector machines and Monte Carlo Simulation. *Structural Safety 67*, 85–95.

Pan, Q., X. Qu, L. Liu, and D. Dias (2020). A sequential sparse polynomial chaos expansion using Bayesian regression for geotechnical reliability estimations. *Int. J. Numer. Anal. Methods Geomech. 44*, 874–889.

Papaioannou, I., C. Papadimitriou, and D. Straub (2016). Sequential importance sampling for structural reliability analysis. *Structural Safety 62*, 66–75.

Peijuan, Z., W. Ming, Z. Zhouhong, and W. Liqi (2017). A new active learning method based on the learning function U of the AK-MCS reliability analysis method. *Eng. Struct. 148*, 185–194.

Rackwitz, R. and B. Fiessler (1978). Structural reliability under combined load sequences. *Comput. Struct. 9*, 489–494.

Rajashekhar, M.-R. and B.-R. Ellingwood (1993). A new look at the response surface approach for reliability analysis. *Structural Safety 12*, 205–220.

Ranjan, P., D. Bungham, and G. Michailidis (2008). Sequential experiment design for contour estimation from complex computer codes. *Technometrics 50*, 527–541.

Rasmussen, C. E. and C. K. I. Williams (2006). *Gaussian processes for machine learning* (Internet ed.). Adaptive computation and machine learning. Cambridge, Massachusetts: MIT Press.

Razaaly, N. and P. M. Congedo (2018). Novel algorithm using active metamodel learning and importance sampling: Application to multiple failure regions of low probability. *J. Comput. Phys. 368*, 92–114.

Roussouly, N., F. Petitjean, and M. Salaun (2013). A new adaptive response surface method for reliability analysis. *Prob. Eng. Mech. 32*, 103–115.

Rozsas, A. and A. Slobbe (2019). Repository and black-box reliability challenge 2019. https://gitlab.com/rozsasarpi/rprepo/. Accessed: 2021-05-04.

Sadoughi, M. K., M. Li, C. Hi, and C. A. Mackenzie (2017). High-dimensional reliability analysis of engineered systems involving computationally expensive black-box simulations. In *Proc. ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, August 6-9, 2017, Cleveland, Ohia, USA*.

Santner, T. J., B. J. Williams, and W. I. Notz (2003). *The Design and Analysis of Computer Experiments*. Springer, New York.

Schöbi, R., S. Marelli, and B. Sudret (2017). UQLab user manual – PC-Kriging. Technical report, Chair of Risk, Safety & Uncertainty Quantification, ETH Zurich. Report # UQLab-V1.0-109.

Schöbi, R., B. Sudret, and S. Marelli (2016). Rare event estimation using Polynomial-Chaos-Kriging. *ASCE-ASME J. Risk Uncertainty Eng. Syst., Part A: Civ. Eng. 3*(2). D4016002.

Shi, L., S. B., and D. S. Ibrahim (2019). An active learning reliability method with multiple kernel functions based on radial basis function. *Struct. Multidisc. Optim. 60*, 211–229.

Sobol', I. M. (1967). Distribution of points in a cube and approximate evaluation of integrals. *U.S.S.R Comput. Maths. Math. Phys. 7*, 86–112.

Sun, Z., J. Wang, R. Li, and C. Tong (2017). LIF: A new Kriging based learning function and its application to structural reliability analysis. *Reliab. Eng. Syst. Saf. 157*, 152–165.

Sundar, V. and M. D. Shields (2016). Surrogate-enhanced stochastic search algorithms to identify implicitly defined functions for reliability analysis. *Structural Safety 62*, 1–11.

Teixeira, R., M. Nogal, and A. O'Connor (2021). Adaptive approaches in metamodel-based reliability analysis: A review. *Structural Safety 89*, 102019.

Tong, C., Z. Sun, Q. Zhao, Q. Wang, and S. Wang (2015). A hybrid algorithm for reliability analysis combining Kriging and subset simulation importance sampling. *J. Mech sci. Tech. 29*, 3183–3193.

Tong, C., J. Wang, and L. J. (2019). A Kriging-based active learning algorithm for mechanical reliability analysis with time-consuming and nonlinear response. *Math. Probl. Eng. 2019*, 1–14.

Wagner, P.-R., S. Marelli, I. Papaioannou, D. Straub, and B. Sudret (2021). Rare event estimation using stochastic spectral embedding. *Structural Safety*.

Wang, Z., M. Broccardo, and J. Song (2019). Hamiltonian Monte Carlo methods for subset simulation in reliability analysis. *Structural Safety 76*, 51–67.

Wen, Z., H. Pei, H. Liu, and Z. Yue (2016). A sequential Kriging reliability analysis method with characteristics of adaptive sampling regions and parallelizability. *Reliab. Eng. Sys. Saf. 153*, 170–179.

Xiao, N.-C., M. J. Zuo, and W. Guo (2018). Efficient reliability analysis based on adaptive sequential sampling design and cross-validation. *Appl. Math. Model. 58*, 404–420.

Xiu, D. and G. E. Karniadakis (2002). The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput. 24*(2), 619–644.

Yang, X., Y. Liu, C. Mi, and X. Wang (2018). Active learning Kriging model combining with kernel-density estimation-based importance sampling method for the estimation of low failure probability. *J. Mech. Des. 140*, 1–9.

Zhang, J. and A. A. Taflanidis (2018). Adaptive Kriging stochastic sampling and density approximation and its application to rare-event estimation. *ASCE-ASME J. Risk Uncertainty Eng. Syst., Part A: Civ. Eng. 4*, 1–17.

Zhang, J., M. Xiao, and L. Gao (2019). An active learning reliability method combining Kriging constructed with exploration and exploitation of failure region and subset simulation. *Reliab. Eng. Syst. Saf. 188*, 90–102.

Zhang, X., L. Wang, and J. D. Sørensen (2019). REIF: A novel active-learning function toward adaptive Kriging surrogate models for structural reliability analysis. *Reliab. Eng. Syst. Saf. 185*, 440–454.

Zhang, X., L. Wang, and J. D. Sorensen (2020). AKOIS: An adaptive Kriging oriented importance sampling method for structural system reliability analysis. *Structural Safety 82*, 1–13.

Zhao, H., Z. yue, H. Liu, Z. Gao, and Y. Zhang (2015). An efficient reliability method combining adaptive importance sampling and Kriging metamodel. *Appl. Math. Model. 39*, 1853–1866.

Zheng, N. and J. Xue. *Manifold Learning*, Chapter 4, pp. 87–119. Springer, London.

Zhou, T. and Y. Peng (2020). Structural reliability analysis via dimension reduction, adaptive sampling, and Monte Carlo simulation. *Struct. Multidisc. Optim. 62*, 2629–2651.

A TNO Benchmark

The TNO benchmark is a truly black-box benchmark of structural reliability analysis methods organized by TNO (Netherlands) in 2019 (Rozsas and Slobbe, 2019). It is a two-part challenge which consists in a set of 16 component- and 11 system-reliability problems. It aims at assessing the efficiency and accuracy of various structural reliability methods. The limit-state functions were not known to the participants and were only accessible via an anonymous server API, *i.e.*, the participants could only submit a set of sample points and the server would return the corresponding model evaluations.

The methods highlighted by the benchmark in this paper, *i.e.*, a combination of PC-Kriging, subset simulation with overkill settings, deviation number U and combined stopping criterion, were used to participate in the challenge. The results were disclosed in terms of accuracy and efficiency.

Figure 17 shows the results submitted anonymously by the nine research groups that took part in the challenge. The black arrows point to the results submitted using our approach. Our approach turned out to be both the most accurate and efficient in 24 out of 27 problems. This confirms the potential of such a flexible framework for the solution of a wide variety of structural reliability problems.
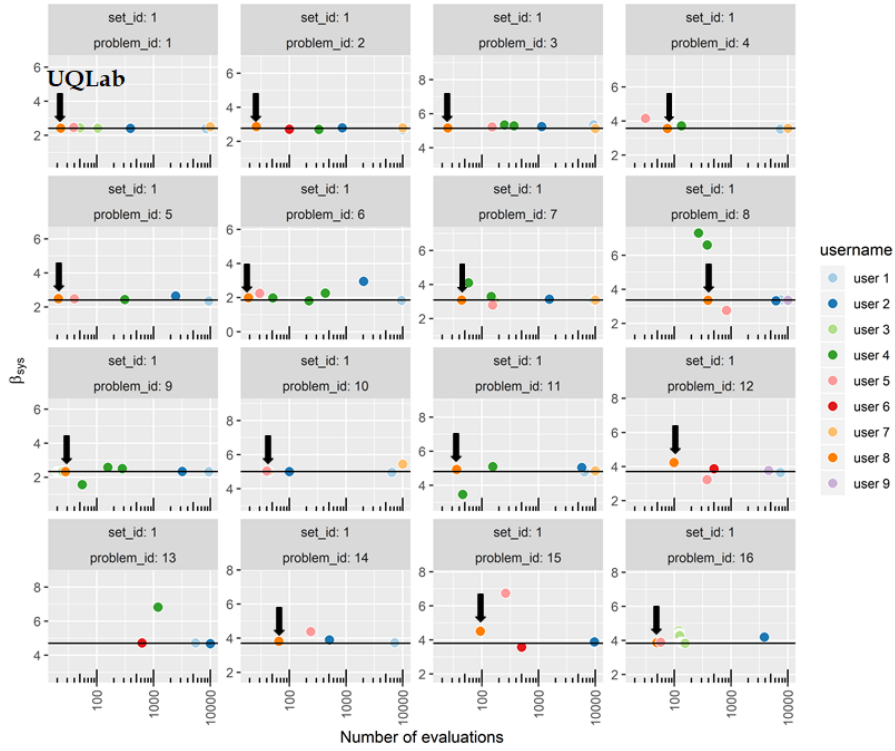
(a) Part 1: Component-reliability problems
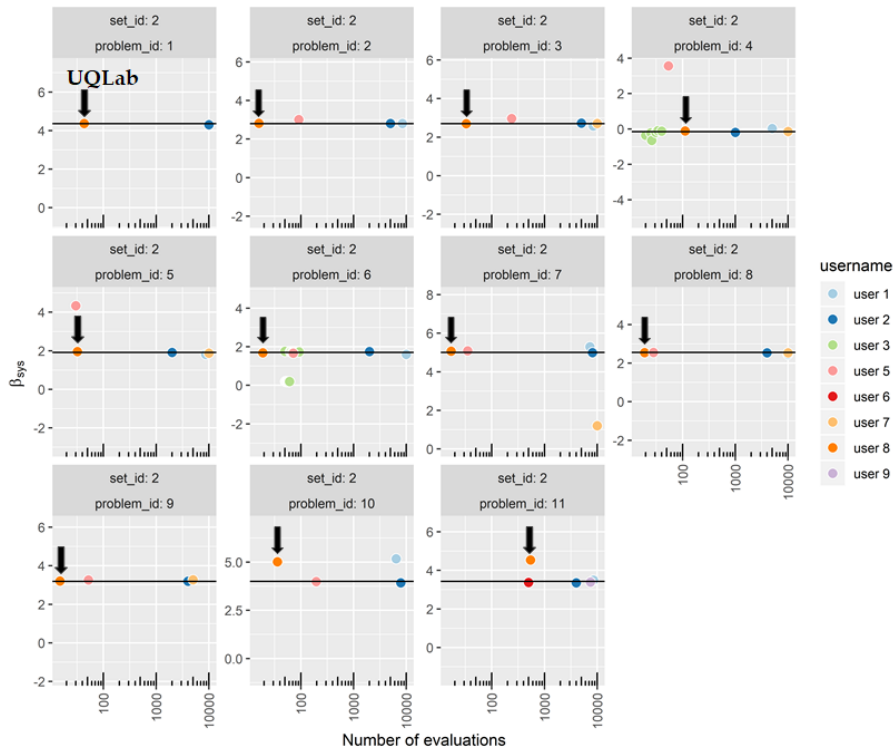
(b) Part 2: System-reliability problems

Figure 17: Results of the black-box reliability challenge as disclosed by Rozsas and Slobbe (2019). The results submitted using the approach highlighted in this paper are marked by the black arrow.

B Supplementary materials

B.1 Detailed list of problems

This section presents the different problems used in the benchmark. Only the transmission tower problems which were specifically developed for this benchmark are presented in details. The reader is referred to the mentioned references for further details on the other problems.

Table 3: Summary of the benchmark problems (#01 to #11 are from (Rozsas and Slobbe, 2019). #19 & #20 are based on finite element models.)

Problem	Dimension	$P_{f,\text{ref}}$	Reference
01 (TNO RP 14)	5	$7.69 \, 10^{-4}$	Rozsas and Slobbe (2019)
02 (TNO RP 24)	2	$2.90 \, 10^{-3}$	Rozsas and Slobbe (2019)
03 (TNO RP 28)	2	$1.31 \, 10^{-7}$	Rozsas and Slobbe (2019)
04 (TNO RP 31)	2	$3.20 \, 10^{-3}$	Rozsas and Slobbe (2019)
05 (TNO RP 38)	7	$8.20 \cdot 10^{-3}$	Rozsas and Slobbe (2019)
06 (TNO RP 53)	2	$3.14 \cdot 10^{-2}$	Rozsas and Slobbe (2019)
07 (TNO RP 54)	20	$9.79 \cdot 10^{-4}$	Rozsas and Slobbe (2019)
08 (TNO RP 63)	100	$3.77 \cdot 10^{-4}$	Rozsas and Slobbe (2019)
09 (TNO RP 75)	2	$9.80 \cdot 10^{-3}$	Rozsas and Slobbe (2019)
10 (TNO RP 107)	10	$2.85 \cdot 10^{-7}$	Rozsas and Slobbe (2019)
11 (TNO RP 111)	2	$7.83 \cdot 10^{-7}$	Rozsas and Slobbe (2019)
12 (4-branch series)	2	$3.85 \cdot 10^{-4}$	Echard et al. (2011)
13 (Hat function)	2	$4.40 \cdot 10^{-3}$	Schöbi et al. (2016)
14 (Damped oscillator)	8	$4.80 \cdot 10^{-3}$	Der Kiureghian and de Stefano (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	$2.00 \cdot 10^{-3}$	
18 (VNL function)	40	$1.40 \cdot 10^{-3}$	Bichon et al. (2008); Sadoughi et al. (2017)
19 (Transmission tower 1)	11	$5.76 \cdot 10^{-4}$	See Section B.1.1
20 (Transmission tower 2)	9	$6.27 \cdot 10^{-4}$	See Section B.1.1

B.1.1 Transmission tower

The transmission tower example is originally developped within this paper. It is a three-dimensional finite element model consisting of 51 nodes and 172 bars. The bars are split

in four groups, each characterized by their cross-sectional areas (A_1 to A_4) and constitutive materials Young's moduli (E_1 to E_4). The truss is subjected at its tip to a horizontal wind load F whose deviation α from the lateral axis of the hands is random. At the extremity of the hands two vertical loads due to cables weight are added. All these parameters are random and described in Table 4.

The finite element analysis is carried out in MATLAB and failure is assumed:

- for problem #19, when the displacement at the tip of the tower is larger than 0.07 m and;

- for problem #20, when the maximum stress in any of the bars is larger than the yield stress f_y defined in Table 4.
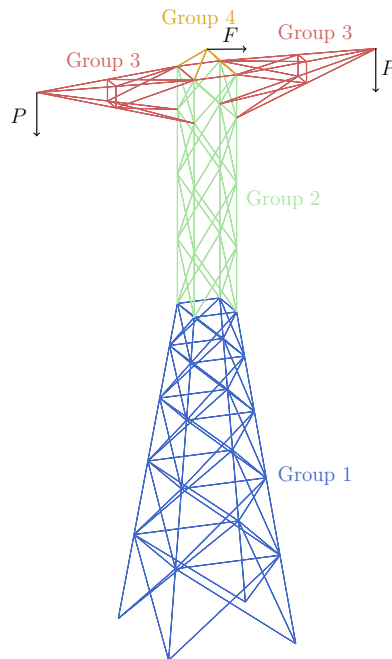

Figure 18: Illustration of the transmission tower used for problems #19 and #20.

Table 4: Probabilistic model for the transmission tower problems.

Parameter	Distribution	Mean	C.o.V
A_1 (m²)	Gaussian	10^{-3}	0.05
A_2 (m²)	Gaussian	10^{-3}	0.05
A_3 (m²)	Gaussian	$5 \cdot 10^{-3}$	0.05
A_4 (m²)	Gaussian	$5 \cdot 10^{-3}$	0.05
E_1 (GPa)[a]	Gaussian	210	0.05
E_2 (GPa)[a]	Gaussian	210	0.05
E_3 (GPa)[a]	Gaussian	210	0.05
E_4 (GPa)[a]	Gaussian	210	0.05
F (N)	Gumbel	$3.5 \cdot 10^4$	0.30
P (GPa)	Gumbel	10^4	0.30
α (degrees)	Uniform[b]	-30	30
f_y (MPa)[c]	Lognormal	355	0.2

[a] For problem #20, $E = E_1 = E_2 = E_3 = E_4$;

[b] The values in the next columns are parameters of the
Uniform distribution (minimum and maximum);

[c] Only used for Problem #20.

B.2 Algorithmic settings

B.2.1 Surrogate models

Kriging Kriging also known as Gaussian process modelling considers the model to approximate as a realization of a stochastic Gaussian process made up of two parts (Santner et al., 2003; Rasmussen and Williams, 2006):

$$\mathcal{M}\left(\boldsymbol{x}\right) = \boldsymbol{f}^T\left(\boldsymbol{x}\right)\boldsymbol{\beta} + \sigma^2 \boldsymbol{Z}\left(\boldsymbol{x}\right), \tag{6}$$

where $\boldsymbol{f}\left(\boldsymbol{x}\right)$ is a vector of regressors with their corresponding coefficients $\boldsymbol{\beta}$, σ^2 is the process constant variance and $\boldsymbol{Z}\left(\boldsymbol{x}\right)$ a zero-mean, unit-variance stationnary Gaussian process.

This parametric form is calibrated by learning over an experimental design and the prediction for any unknown sample is given by the following analytical formula:

$$\mu_{\widehat{g}}\left(\boldsymbol{x}\right) = \boldsymbol{f}^T\left(\boldsymbol{x}\right)\widehat{\boldsymbol{\beta}} + \boldsymbol{r}^T\left(\boldsymbol{x}\right)\boldsymbol{R}^{-1}\left(\boldsymbol{y} - \boldsymbol{F}^T\widehat{\boldsymbol{\beta}}\right), \tag{7}$$

where $\boldsymbol{f}^T\left(\boldsymbol{x}\right)\widehat{\boldsymbol{\beta}}$ is a polynomial trend calibrated through least-square regression, R is a parametric auto-correlation matrix, $\boldsymbol{r}\left(\boldsymbol{x}\right)$ is a vector of cross-correlations between the point $\boldsymbol{x}$ and the experimental design points and $\boldsymbol{F}$ is the so-called observation matrix. Interestingly, Kriging not only allows for prediction but can also provide a measure of its own accuracy

through the following variance:

$$\sigma_{\hat{g}}^2(\boldsymbol{x}) = \sigma^2 \left(1 - \boldsymbol{r}^T(\boldsymbol{x}) \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) + \boldsymbol{u}^T(\boldsymbol{x}) \left(\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{F} \right)^{-1} \boldsymbol{u}(\boldsymbol{x}) \right) \tag{8}$$

where $\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x})$.

In this benchmark, we consider ordinary Kriging, meaning that the trend is an unknown constant as it is a common practice in GP modelling. Furthermore, we consider an anisotropic Gaussian auto-correlation function, the parameters of which are calibrated using maximum likehood estimation. The optimization is carried out using a genetic algorithm whose results are refined by a gradient-based solver.

The implementations in the Kriging module (Lataniotis et al., 2017) of UQLab (Marelli and Sudret, 2014) were used for the applications in this paper.

Polynomial chaos expansions Polynomial chaos expansions result from a spectral expansion of a random variable $\boldsymbol{Y}$ onto a set of orthonormal polynomials (Xiu and Karni-adakis, 2002):

$$Y = \mathcal{M}(\boldsymbol{X}) = \sum_{\alpha \in \mathbb{N}^M} y_\alpha \Psi_\alpha(\boldsymbol{X}) \tag{9}$$

where Ψ_α are a set of multivariate polynomials orthornormal with respect to $f_{\boldsymbol{X}}$ and y_α are the coefficients representing the coordinates of the individual components of Ψ_α indexed by $\alpha \in \mathcal{N}^M$.

Building a PCE approximation requires two main steps. The first is to truncate the infinite expansions into a finite sum, which is achieved in this work using hyperbolic trunca-tion (q-norm with $q = 0.75$) (See Blatman and Sudret (2010) for details). Furthermore, the maximum degree is limited to 20 and the interactions to the second order. The second step is to estimate the coefficients, which is achieved here using a regularized least-square min-imization problem whose formulation allows for sparsity in the PC expansion. Practically, this problem is solved by the hybrid least-angle regression algorithm (LARS, Efron et al. (2004)) as proposed in Blatman and Sudret (2011).

The implementations in the PCE module (Marelli and Sudret, 2017) of UQLab (Marelli and Sudret, 2014) were used for the applications in this paper.

PC-Kriging PC-Kriging is a metamodellling technique obtained by combining polyno-mial chaos expansions and Kriging. More specifically, a PC-Kriging model is simply a uni-versal Kriging model whose trend is a set of orthonormal polynomials (Schöbi et al., 2016):

$$\mathcal{M}(\boldsymbol{x}) = \sum_{\alpha \in \mathcal{A}} y_\alpha \Psi_\alpha(\boldsymbol{x}) + \sigma^2 \boldsymbol{Z}(\boldsymbol{x}), \tag{10}$$

where $\mathcal{A}$ is a finite set of multi-indices.

The calibration of the PC-Kriging model is carried out sequentially: first the terms of the (sparse) polynomial trend are detected with LARS then the Kriging metamodel is fitted (both the trend coefficients and the hyperparameters of the covariance kernel). The same

algorithms and settings as in the two previous paragraphs are used. The only difference is the maximum polynomial degree, which is set here equal to 3.

The implementations in the PC-Kriging module (Schöbi et al., 2017) of UQLAB (Marelli and Sudret, 2014) were used for the applications in this paper.

B.2.2   Reliability analysis

Monte Carlo simulation   Monte Carlo simulation is a direct integration of Eq. 1 by sampling the probability density function $f_{\boldsymbol{X}}$. Given a population of size N, the MC estimate of P_f reads:

$$\widehat{P}_f = \sum_{k=1}^{N} \mathbf{1}_{\{\boldsymbol{x}:g(\boldsymbol{x})\leq 0\}}\left(\boldsymbol{x}^{(k)}\right) = \frac{N_{\text{fail}}}{N}, \tag{11}$$

where N_{fail} is the number of failed samples. The only parameter to calibrate here is the sample size. In this work, the simulation is carried out in batches of size 10^5 until the coefficient of variation of the estimate:

$$CoV = \sqrt{\frac{1 - \widehat{P}_f}{N\widehat{P}_f}} \tag{12}$$

is smaller than a threshold arbitrarily set to 0.025 or until the maximum sample size of 10^7 is reached.

Importance sampling   Importance sampling is a variance reduction simulation technique where the samples are generated following a proposal distribution rather than the original random variables PDF. The associated failure probability estimate therefore reads:

$$\widehat{P}_f = \frac{1}{N}\sum_{k=1}^{N} \mathbf{1}_{\{\boldsymbol{x}:g(\boldsymbol{x})\leq 0\}}\left(\boldsymbol{x}^{(k)}\right) \frac{f_{\boldsymbol{X}}\left(\boldsymbol{x}^{(k)}\right)}{h\left(\boldsymbol{x}^{(k)}\right)}, \tag{13}$$

where the set $\left\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\right\}$ is sampled following the distribution $h(\boldsymbol{x})$.

In this work, the proposal distribution is simply a standard Gaussian centered around the most probable failure point as estimated using FORM. The maximum sample size is set to 10^4.

Subset simulation   Subset simulation is another popular variance reduction technique obtained by solving a series of reliability problems with a relatively large target failure probability. Considering a set of nested events $\mathcal{D}_1 \supset \mathcal{D}_2, \ldots, \supset \mathcal{D}_m = \mathcal{D}_f$ defined such that $\mathcal{D}_k = \{\boldsymbol{x} : g(\boldsymbol{x}) \leq t_k\}$ with $t_1 > t_2 >, \ldots, t_m = 0$, the failure probability can be recast as:

$$P_f = \mathbb{P}\left(\mathcal{D}_f\right) = \mathbb{P}\left(\cap_{k=1}^m \mathcal{D}_k\right) = \mathbb{P}\left(\mathcal{D}_1\right) \prod_{i=1}^{m-1} \mathbb{P}\left(\mathcal{D}_{i+1}|\mathcal{D}_i\right) \tag{14}$$

While the initial failure probability $\mathbb{P}(\mathcal{D}_1)$ is estimated using crude Monte Carlo simulation, the remaining conditional failure probabilities are computed using Markov Chain Monte Carlo. The intermediate thresholds are set on-the-fly such that the $\mathbb{P}(\mathcal{D}_{i+1}|\mathcal{D}_i) = p_0$ is large

enough. Traditionnally p_0 is set to 0.1 but in this work, we consider $p_0 = 0.25$ and a batch size of 10^5 for each subset level. This overkill setting allows us to better explore the random input space and results in a relatively small coefficient of variation of $\widehat{P}_f$.

B.2.3   Learning functions

In this work, we consider three different learning functions. The first two are associated to Kriging and PC-Kriging while the third one is only used with PCE. They respectively read:

Deviation number

$$U\left(\boldsymbol{x}\right) = \frac{|\mu_{\widehat{g}}\left(\boldsymbol{x}\right)|}{\sigma_{\widehat{g}}\left(\boldsymbol{x}\right)} \tag{15}$$

Expected feasibility function

$$
\begin{aligned}
EFF(\boldsymbol{x}) = \mu_{\widehat{g}}(\boldsymbol{x}) &\left[2\Phi\left(\frac{-\mu_{\widehat{g}}(\boldsymbol{x})}{\sigma_{\widehat{g}}(\boldsymbol{x})}\right) - \Phi\left(\frac{-\epsilon - \mu_{\widehat{g}}(\boldsymbol{x})}{\sigma_{\widehat{g}}(\boldsymbol{x})}\right) - \Phi\left(\frac{\epsilon - \mu_{\widehat{g}}(\boldsymbol{x})}{\sigma_{\widehat{g}}(\boldsymbol{x})}\right)\right] \\
&- \sigma_{\widehat{g}}(\boldsymbol{x})\left[2\varphi\left(\frac{-\mu_{\widehat{g}}(\boldsymbol{x})}{\sigma_{\widehat{g}}(\boldsymbol{x})}\right) - \varphi\left(\frac{-\epsilon - \mu_{\widehat{g}}(\boldsymbol{x})}{\sigma_{\widehat{g}}(\boldsymbol{x})}\right) - \varphi\left(\frac{\epsilon - \mu_{\widehat{g}}(\boldsymbol{x})}{\sigma_{\widehat{g}}(\boldsymbol{x})}\right)\right] \\
&+ \epsilon\left[\Phi\left(\frac{\epsilon - \mu_{\widehat{g}}(\boldsymbol{x})}{\sigma_{\widehat{g}}(\boldsymbol{x})}\right) - \Phi\left(\frac{-\epsilon - \mu_{\widehat{g}}(\boldsymbol{x})}{\sigma_{\widehat{g}}(\boldsymbol{x})}\right)\right], \quad \tag{16}
\end{aligned}
$$

where $\epsilon = 2\sigma_{\widehat{g}}(\boldsymbol{x})$ and ϕ and Φ are respectively the probability density function (PDF) and cumulative distribution function (CDF) of a Gaussian random variable.

Fraction of bootstrap replicates   This learning function, developed by Marelli and Sudret (2016), is based on bootstrap replicates of a PCE model. First, B PCE models are built using B new experimental designs $\left\{\mathcal{E}^{(b)}, b = 1, \dots, B\right\}$ where each new ED is constructed by drawing samples with replacement from $\mathcal{E}$. Following this procedure, the learning function reads:

$$U_{\text{FBR}}(\boldsymbol{x}) = \frac{|B_s\left(\boldsymbol{x}\right) - B_f\left(\boldsymbol{x}\right)|}{B}, \tag{17}$$

where $B_s\left(\boldsymbol{x}\right)$ and $B_f\left(\boldsymbol{x}\right) \in [0,, \dots,, B]$ are respectively the number of safe and failed PC-bootstrap replicates at the point $\boldsymbol{x}$.

No specific algorithmic settings are considered for these learning functions. Considering a candidate pool made up of samples drawn within the previous iterations of the reliability algorithm, the next point to add to the ED is simply chosen by selecting the one that minimizes U (or U_{FBR}) and maximizes EFF.