SEQUENTIAL ACTIVE LEARNING OF LOW-DIMENSIONAL MODEL REPRESENTATIONS FOR RELIABILITY ANALYSIS*

MAX EHRE†, IASON PAPAIOANNOU†, BRUNO SUDRET‡, AND DANIEL STRAUB†

Abstract. To date, the analysis of high-dimensional, computationally expensive engineering models remains a difficult challenge in risk and reliability engineering. We use a combination of dimensionality reduction and surrogate modelling termed partial least squares-driven polynomial chaos expansion (PLS-PCE) to render such problems feasible. Standalone surrogate models typically perform poorly for reliability analysis. Therefore, in a previous work, we have used PLS-PCEs to reconstruct the intermediate densities of a sequential importance sampling approach to reliability analysis. Here, we extend this approach with an active learning procedure that allows for improved error control at each importance sampling level. To this end, we formulate an estimate of the combined estimation error for both the subspace identified in the dimension reduction step and surrogate model constructed therein. With this, it is possible to adapt the training set so as to optimally learn the subspace representation and the surrogate model constructed therein. The approach is gradient-free and thus can be directly applied to black box-type models. We demonstrate the performance of this approach with a series of low- (2 dimensions) to high- (869 dimensions) dimensional example problems featuring a number of well-known caveats for reliability methods besides high dimensions and expensive computational models: strongly nonlinear limit-state functions, multiple relevant failure regions and small probabilities of failure.

Key words. Reliability Analysis, Rare event simulation, PLS-PCE, Dimensionality reduction, Active learning, Sequential importance sampling

AMS subject classifications. 62L99, 62P30, 62J02, 65C05

1. Introduction and previous work.

An important challenge in the design, analysis and maintenance of engineering systems is the management of the associated uncertainties. It is common practice to analyse engineering systems by employing computational models that aim at representing the physical processes relevant to the system in consideration. These computational models take the form of an input-output mapping. Therein, uncertainty is represented by equipping the model input with an appropriate probabilistic model. Undesirable system responses are defined through a limit-state function (LSF). Reliability analysis is concerned with quantifying the probability of failure, which can be expressed as a d-fold integral of the input probability mass over the failure domain defined by non-positive values of the LSF, where d is the number of uncertain model inputs (see Section 2). In engineering, target failure probabilities are typically small; hence, reliability analysis requires the estimation of rare event probabilities. Reliability analysis approaches can be categorized into approximation (e.g. the first- and second-order reliability methods FORM and SORM [66, 27, 18]) and simulation methods. If the LSF is only weakly nonlinear and the input dimension of the model is moderate, FORM and SORM perform well even for small failure probabilities. The simplest simulation method is the Monte Carlo method [54]. The Monte Carlo method performs well independent of the problem input dimension, however its performance deteriorates as the failure probability decreases if the computational budget is fixed. Various techniques such as importance sampling (IS) [13, 24, 2] and line-sampling [30, 39] have been proposed to mitigate this dependence on the magnitude of the failure probability. More recently, sequential MC methods such as subset simulation [3] and IS-based sequential methods [41, 42, 83, 61, 68, 60] have been used successfully to efficiently solve high-dimensional reliability problems with small failure probabilties. If the computational model is expensive and a hierarchy of increasingly coarse and cheap models is accessible, multilevel and multi-fidelity [63] MC methods can help alleviate computational cost by performing most model evaluations on the cheaper models (e.g., a discretized differential equation with coarser resolution). In [79], multilevel MC is combined with subset simulation and recently [82] have introduced multilevel sequential IS based on the sequential IS approach in [61]. All of the above-mentioned approaches are designed to work with the probabilistic computational model directly. However, often this model encompasses a numerical solver for (sets of) partial differential equations such that a model evaluation is computationally expensive.

*Submitted to the editors DATE.

Funding: This project was supported by the German Research Foundation (DFG) through Grant STR 1140/6-1 under SPP 1886.

†Engineering Risk Analysis Group, Technical University of Munich (max.ehre@tum.de,iason.papaioannou@tum.de,straub@tum.de).

‡Chair of Risk, Safety and Uncertainty Quantification, ETH Zürich (sudret@ibk.eth.ch).

This has increasingly lead researchers to turn towards surrogate model-based reliability methods. Such methods attempt to approximate the expensive computational model with a cheap surrogate model, whose coefficients are identified based on a set of original model evaluations: the training set. [25] uses a polynomial response surface method for performing reliability analysis as early as 1989. [28] proposes an improved version of the response surface method. Since then, a variety of surrogate modelling techniques has been applied in the context of reliability analysis such as artificial neural networks [57, 34, 71], support vector machines [33, 12, 11], Gaussian process regression-based models [22, 21] and projection to polynomial bases including polynomial chaos expansions (PCE) [47, 45, 44, 73] and low-rank tensor approximations [38].

Static, *global* surrogate models suffer from a decrease in accuracy in the tails of the model response distribution such that they are of limited use for reliability analysis. In this context, *static* refers to surrogate models that are constructed based on a fixed training set and *global* refers to surrogate models that are trained and evaluated on the entire input space (as opposed to locally con- and re-fined models). Thus, one can distinguish two strategies to overcome this limitation:

- *Locality*: Surrogate models are coupled with sequential sampling techniques which serve to focus the training set and accuracy in the relevant regions around the failure hypersurface [56, 12, 11, 6, 58].
- *Adaptivity* (in the training set): The training set is augmented with points that are most informative with respect to the failure probability estimate according to an 'in-fill criterion'. The refined surrogate model is then used to estimate the probability of failure with a sampling method and a large number of cheap samples. Such procedures are summarized under the terms active learning (AL) or optimal experimental design. AL in combination with crude Monte Carlo have been applied in reliability-based optimization and reliability analysis in [22, 53, 8, 65]. [71] investigates the performance of splines and neural networks in combination with directional sampling and IS and [21, 14] combine Gaussian process models with IS. [70] proposes a crude Monte Carlo procedure relying on a Gaussian process surrogate model with PCE-based mean trend (PCE-Kriging) along with a novel termination criterion for the AL.

Often, both AL and sequential sampling techniques are combined using various combinations of in-fill criteria and sequential sampling techniques such as adaptive IS [5] and subset simulation [12, 32, 6, 11]. [52] turns away from surrogate models that naturally provide a measure of prediction uncertainty such as Gaussian processes or support vector machines and demonstrate how an AL algorithm can be realized with PCE using a bootstrap estimator of the PCE prediction uncertainty.

In spite of a plethora of existing approaches to surrogate-assisted reliability analysis, the literature on high-dimensional problems ($d \geq 100$) in this context is scarce. [36, 46] propose to perform reliability analysis with a static, global Kriging model constructed in a low-dimensional linear subspace of the original model input space, which is identified by the active subspaces method [16] and autoencoders, respectively. Both [36, 46] apply their methods to moderate-dimensional problems with up to $d = 20$ and $d = 40$ input variables, respectively. [55] uses sliced inverse regression to identify a linear low-dimensional subspace and construct a static, global PCE in this space based on which they perform reliability analysis directly. [89] develops these ideas further by combining the active subspace-Kriging model with an AL approach and applies this combination to a high-dimensional analytical problem of $d = 300$ that posesses a perfectly linear low-dimensional structure.

In this work, we propose an importance sampler based on a dimensionality-reducing surrogate model termed partial least squares-driven PCE (PLS-PCE) [59] to efficiently solve high-dimensional reliability problems with underlying computationally expensive, nonlinear models and small target probabilities ($\mathcal{O}(10^{-9})$). Similar to sliced inverse regression and active subspaces, PLS-PCE achieves dimensionality reduction by identifying a low-dimensional linear subspace of the original input space. Our method is based on [58] but introduces AL to refine the PLS-PCE approximation in each sequence of the IS procedure. In [58], PLS-PCE models are reconstructed in each level of a sequential importance sampling (SIS) scheme that is used to gradually shift the importance density towards the optimal importance density. In this work, we augment this approach with two novel contributions to rare event simulation of computationally expensive, potentially (but not necessarily) high-dimensional and nonlinear models:

1. We demonstrate how to perform active learning with PCE models by deriving an in-fill criterion from large-sample properties of the PCE coefficient estimates.

2. We use projection to linear subspaces to construct efficient surrogate models for high-dimensional problems and include the subspace estimation error in the in-fill criterion. This means, we are not only learning the surrogate model but also the subspace itself.

Using AL in the context of PLS-PCE-based SIS provides effective error control and benefits from the local confinement of the learning procedure of each subspace/surrogate model combination to the support of the current importance density. Constructing local variance estimates for polynomial models in the way we propose here creates new possibilities to design goal-oriented surrogate modelling approaches that are driven by adaptive sampling based on such models (where so far, Gaussian processes were the dominant tool).

In Section 2, we set up the reliability problem and discuss the crude Monte Carlo sampler of the probability of failure. Section 3 reviews IS and a variant of SIS [61] that is at the base of our approach. Section 4 introduces PLS-PCE models and their construction. Subsection 5.2 details the theoretical foundations of active learning of PLS-PCE models within SIS and summarizes our approach. In Section 6, we present comprehensive investigations of the method's performance in two engineering examples and provide a detailed discussion of the results. Conclusions are given in Section 7.

2. Reliability analysis. Consider a system represented by the computational model $\mathcal{Y} : \mathbb{D}_{\boldsymbol{X}} \to \mathbb{R}$ with d-dimensional continuous random input vector $\boldsymbol{X} : \Omega \to \mathbb{D}_{\boldsymbol{X}} \subseteq \mathbb{R}^d$, where Ω is the sample space of $\boldsymbol{X}$ and by $F_{\boldsymbol{X}}(\boldsymbol{x})$, we denote its joint cumulative distribution function (CDF). $\mathcal{Y}$ maps to the system response $Y = \mathcal{Y}(\boldsymbol{x})$ with the model input $\boldsymbol{x} \in \mathbb{D}_{\boldsymbol{X}}$. Based on the response Y, unacceptable system states are defined by means of the limit-state function (LSF) $\tilde{g}(Y)$. Defining $g(\boldsymbol{x}) = \tilde{g} \circ \mathcal{Y}(\boldsymbol{x})$ and introducing the convention

$$g(\boldsymbol{x}) = \begin{cases} \leq 0, \text{Failure} \\ > 0, \text{Safety}, \end{cases}$$

the failure event of the system is defined as $\mathrm{F} = \{\boldsymbol{x} \in \mathbb{D}_{\boldsymbol{X}} : g(\boldsymbol{x}) \leq 0\}$. The probability of failure is given by [19]

$$p = \mathbb{P}(\mathrm{F}) = \int_{\mathbb{D}_{\boldsymbol{X}}} \mathrm{I}[g(\boldsymbol{x}) \leq 0] f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = \mathbb{E}_{f_{\boldsymbol{X}}} \left[\mathrm{I}(g(\boldsymbol{X}) \leq 0)\right], \tag{2.1}$$

where $f_{\boldsymbol{X}}(\boldsymbol{x}) = \partial^d F / (\partial x_1 \dots \partial x_d)|_{\boldsymbol{x}}$ is the joint probability density function (PDF) of $\boldsymbol{X}$ and the indicator function $\mathrm{I}[\cdot]$ equals 1 if the condition in the argument is true and 0 otherwise. Without loss of generality, one may formulate an equivalent reliability problem with respect to the standard-normal probability space using the random vector $\boldsymbol{U} : \Omega \to \mathbb{R}^d$. Given an isoprobabilistic transformation $T : \mathbb{D}_{\boldsymbol{X}} \to \mathbb{R}^d$, such that $\boldsymbol{U} = T(\boldsymbol{X})$, see, e.g., [29, 48], and defining $G(\boldsymbol{U}) = g(T^{-1}(\boldsymbol{U}))$, one can write (2.1) as

$$p = \int_{\mathbb{R}^d} \mathrm{I}[G(\boldsymbol{u}) \leq 0] \varphi_d(\boldsymbol{u}) \, \mathrm{d}\boldsymbol{u} = \mathbb{E}_{\varphi_d} \left[\mathrm{I}(G(\boldsymbol{U}) \leq 0)\right], \tag{2.2}$$

where φ_d denotes the d-dimensional independent standard-normal PDF. The crude Monte Carlo estimate of (2.2) is

$$\widehat{p}_{\mathrm{MC}} = \frac{1}{n} \sum_{k=1}^{n} \mathrm{I}[G(\boldsymbol{u}^k) \leq 0], \quad \boldsymbol{u}^k \overset{i.i.d.}{\sim} \varphi_d, \tag{2.3}$$

where $\boldsymbol{u}^k \overset{i.i.d.}{\sim} \varphi_d$ means that $\{\boldsymbol{u}^k\}_{k=1}^n$ are n samples that are independent and identically distributed according to φ_d. This estimate is unbiased and has coefficient of variation (CoV)

$$\delta_{\mathrm{MC}} = \sqrt{\frac{1-p}{np}}. \tag{2.4}$$

The number of samples required to compute $\widehat{p}_{\mathrm{MC}}$ at a prescribed CoV δ_0 reads

$$n_0 = \frac{1-p}{\delta_0^2 p} \overset{p \ll 1}{\approx} \frac{1}{\delta_0^2 p}. \tag{2.5}$$

Therefore, crude Monte Carlo is inefficient for estimating rare event probabilities as, by definition, $p \ll 1$ and thus n_0 becomes large.

3. Sequential importance sampling for rare event estimation. Variance reduction techniques can be used to reduce the CoV of the probability estimate at a fixed budget of samples compared to crude Monte Carlo. One of the most commonly used variance reduction methods is the IS method. Let h be a density, such that $h(\boldsymbol{u}) > 0$ whenever $G(\boldsymbol{u}) \leq 0$. Then, one can rewrite (2.2)

$$p = \int_{\mathbb{R}^d} \mathrm{I}(G(\boldsymbol{u}) \leq 0) \frac{\overbrace{\varphi_d(\boldsymbol{u})}^{\omega(\boldsymbol{u})}}{h(\boldsymbol{u})} h(\boldsymbol{u}) \, \mathrm{d}\boldsymbol{u} = \mathbb{E}_h \left[\mathrm{I}(G(\boldsymbol{U}) \leq 0) \omega(\boldsymbol{U}) \right], \tag{3.1}$$

which leads to the (unbiased) importance sampling estimator

$$\widehat{p}_{\mathrm{IS}} = \frac{1}{n} \sum_{k=1}^{n} \mathrm{I}[G(\boldsymbol{u}^k) \leq 0] \omega(\boldsymbol{u}^k), \quad \boldsymbol{u}^k \overset{i.i.d.}{\sim} h. \tag{3.2}$$

The efficiency of IS depends intimately on the choice of the IS density h and numerous techniques to construct it have been put forward. There exists an optimal importance density h^* in the sense that it leads to $\mathbb{V}[\widehat{p}_{\mathrm{IS}}] = 0$:

$$h^*(\boldsymbol{u}) = \frac{1}{p} \mathrm{I}[G(\boldsymbol{u}) \leq 0] \varphi_d(\boldsymbol{u}). \tag{3.3}$$

While this result is not immediately useful in estimating p as it requires knowledge of p, it can be used to guide the selection of a suitable IS function h.

The SIS method proposed in [61] selects the IS density sequentially starting from a known distribution h_0 that is easy to sample from. It relies on a sequence of distributions $\{h_i(\boldsymbol{u})\}_{i=0}^{M}$,

$$h_i(\boldsymbol{u}) = \frac{\eta_i(\boldsymbol{u})}{p_i}, \; i = 1, \ldots, M, \tag{3.4}$$

where $\{\eta_i(\boldsymbol{u})\}_{i=0}^{M}$ are non-normalized versions of $\{h_i(\boldsymbol{u})\}_{i=0}^{M}$ and $\{p_i\}_{i=0}^{M}$ are the respective normalizing constants. The goal is to arrive at h_M, which is sufficiently close to h^* based on some criterion, and perform importance sampling with h_M. To this end, it is necessary to estimate p_M and obtain samples from h_M. Based on the likelihood ratio of two succeeding non-normalized distributions $\omega_i(\boldsymbol{u}) = \eta_i(\boldsymbol{u}) / \eta_{i-1}(\boldsymbol{u})$, we have

$$s_i = \frac{p_i}{p_{i-1}} = \int_{\mathbb{R}^d} \frac{\eta_i(\boldsymbol{u})}{\eta_{i-1}(\boldsymbol{u})} h_{i-1}(\boldsymbol{u}) \, \mathrm{d}\boldsymbol{u} = \mathbb{E}_{h_{i-1}} \left[\omega_i(\boldsymbol{u}) \right]. \tag{3.5}$$

Therefore, an estimate of p_M is given by

$$\widehat{p}_M = \prod_{i=1}^{M} \widehat{s}_i \text{ with } \widehat{s}_i = \frac{1}{n} \sum_{k=1}^{n} \omega_i\left(\boldsymbol{u}^k\right), \quad \boldsymbol{u}^k \overset{i.i.d.}{\sim} h_{i-1}. \tag{3.6}$$

Samples from h_i can be obtained using Markov Chain Monte Carlo (MCMC) methods given samples from h_{i-1}. More precisely, [61] proposes a resample-move scheme in which Markov chain seeds are obtained as samples from h_{i-1} that are then reweighted (resampled with weights) according to $\omega_i(\boldsymbol{u})$. In this way, the seed samples are already approximately distributed according to the stationary distribution of the Markov chain h_i and long burn-in periods can be avoided. We adopt an adaptive conditional MCMC sampler (aCS) to perform the move step due to its robust performance in high-dimensional settings. Details can be found in [61].

The h_i are chosen as smooth approximations of h^* using the standard-normal CDF $\Phi(\cdot)$ (compare Fig. 1):

$$h_i(\boldsymbol{u}) = \frac{1}{p_i} \Phi\left(-\frac{G(\boldsymbol{u})}{\sigma_i}\right) \varphi_d(\boldsymbol{u}) = \frac{1}{p_i} \eta_i(\boldsymbol{u}), \tag{3.7}$$

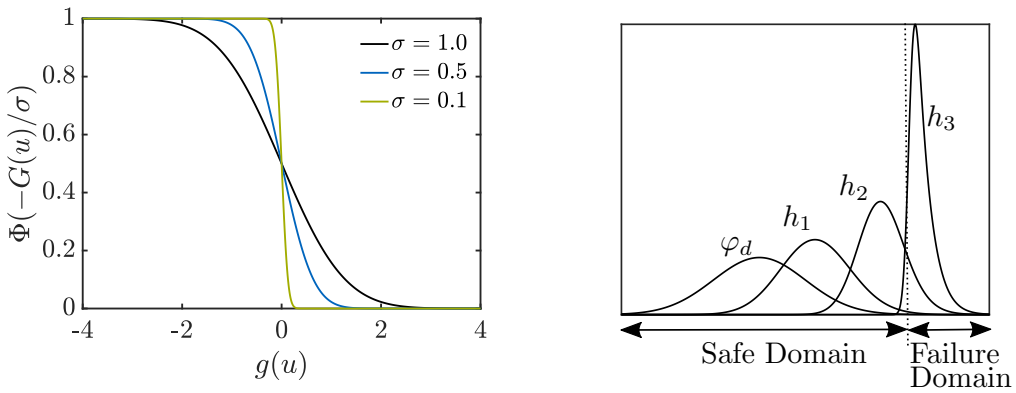


Fig. 1: Smooth approximations to the indicator function $I(g(\boldsymbol{u}) \leq 0)$ (left) and importance densities $h_i(\boldsymbol{u}) \propto \Phi(-G(\boldsymbol{u})/\sigma_i)\,\varphi_d(\boldsymbol{u})$ based on this approximation (right).

where $p_i = \mathbb{E}_{\varphi_d}[\Phi(-G(\boldsymbol{U})/\sigma_i)]$ is a normalizing constant and σ_i is the smoothing parameter. Prescribing $\sigma_0 > \sigma_1 > \cdots > \sigma_M$ ensures that the sequence $\{h_i(\boldsymbol{u})\}_{i=0}^M$ approaches h^*. In each level, to avoid degeneration of the weights ω_i (meaning ω_i assuming values close to 0 at all current samples), $h_{i-1}(\boldsymbol{u})$ and $h_i(\boldsymbol{u})$ cannot be too different in the sense that they share no support regions on which both have considerable probability mass. This is avoided by prescribing an upper bound for the estimated coefficient of variation of the weights $\widehat{\delta}_{w,i} = \widehat{\mathbb{COV}}[\omega_i(\boldsymbol{U})]$, which provides a criterion for determining σ_i:

$$\sigma_i = \underset{\sigma \in [0, \sigma_{i-1}]}{\arg\min} \left(\widehat{\delta}_{\omega,i}(\sigma) - \delta_{\text{target}} \right)^2. \tag{3.8}$$

[61] recommends $\delta_{\text{target}} = 1.5$. The algorithm terminates when h_i is close enough to h^* in the sense that

$$\widehat{\mathbb{COV}}\left[\frac{h^*(\boldsymbol{U})}{h_i(\boldsymbol{U})}\right] = \widehat{\mathbb{COV}}\left[\frac{\varphi_d(\boldsymbol{U})\mathrm{I}(G(\boldsymbol{U}) \leq 0)}{\varphi_d(\boldsymbol{U})\Phi(-G(\boldsymbol{U})/\sigma_i)}\right] = \widehat{\mathbb{COV}}\left[\frac{\mathrm{I}(G(\boldsymbol{U}) \leq 0)}{\Phi(-G(\boldsymbol{u})/\sigma_i)}\right] \leq \delta_{\text{target}}. \tag{3.9}$$

The final estimate of $\mathbb{P}(\mathrm{F})$ reads

$$\widehat{p}_{\text{SIS}} = \widehat{p}_M \widehat{\mathbb{E}}_{\varphi_d}\left[\frac{\mathrm{I}(G(\boldsymbol{U}) \leq 0)}{\eta_M(\boldsymbol{U})}\right] = \left(\prod_{i=1}^M \widehat{s}_i\right) \frac{1}{n} \sum_{k=1}^n \frac{\mathrm{I}(G(\boldsymbol{u}^k) \leq 0)}{\Phi(-G(\boldsymbol{u}^k)/\sigma_M)}, \quad \boldsymbol{u}^k \overset{i.i.d.}{\sim} h_M. \tag{3.10}$$

Algorithm 3.1 summarizes the complete SIS-aCS procedure.

4. Partial least squares-based polynomial chaos expansions.

4.1. Polynomial Chaos Expansions. Polynomial chaos expansions (PCEs) are a tool for forward modelling the relationship between an input $\boldsymbol{X}$ and an output $Y = \mathcal{Y}(\boldsymbol{X})$. With $\mathcal{H}$, we denote the Hilbert space of functions that are square-integrable with respect to $f_{\boldsymbol{X}}$, i.e., $\{v : \mathbb{E}_{f_{\boldsymbol{X}}}[v(\boldsymbol{X})^2] < \infty\}$. $\mathcal{H}$ admits an inner product of two functions $v, w \in \mathcal{H}$:

$$\langle v, w \rangle_{\mathcal{H}} = \mathbb{E}_{f_{\boldsymbol{X}}(\boldsymbol{x})}[v(\boldsymbol{X})w(\boldsymbol{X})] = \int_{\mathbb{R}^d} v(\boldsymbol{x})w(\boldsymbol{x})f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}. \tag{4.1}$$

Let $\{v_j(\boldsymbol{X}), j \in \mathbb{N}\}$ be a complete and orthonormal basis of $\mathcal{H}$ so that $\langle v_j, v_\ell \rangle_{\mathcal{H}} = \delta_{j\ell}$ and let $\mathcal{Y} \in \mathcal{H}$. Then,

$$\mathcal{Y}(\boldsymbol{X}) = \sum_{j=0}^{\infty} b_j v_j(\boldsymbol{X}), \tag{4.2}$$

where the coefficients b_j are defined by projecting $\mathcal{Y}$ on the basis:

$$b_j = \langle \mathcal{Y}, v_j \rangle_{\mathcal{H}}, \quad j \in \mathbb{N}. \tag{4.3}$$

Algorithm 3.1 SIS-aCS [61]

Input LSF $G(\boldsymbol{u})$, target CoV δ_{target}, samples per level n, input dimension d, burn-in period b, max. iterations $i_{\max}$

Set $i = 0$, $\sigma_0 = \infty$, $h_0(\boldsymbol{u}) = \varphi_d(\boldsymbol{u})$

Sample $\mathbf{U}_0 = \{\boldsymbol{u}^k, k = 1, \dots, n\} \in \mathbb{R}^{n \times d}$ ▷ $\boldsymbol{u}^k \overset{i.i.d.}{\sim} h_0(\boldsymbol{u})$

Compute $\mathbf{G}_0 = G(\mathbf{U}_0) \in \mathbb{R}^{n \times 1}$

for $i \leftarrow 1, i_{\max}$ **do**

  $i \leftarrow i + 1$

  Compute σ_i according to (3.8)

  Compute weights $\boldsymbol{\omega}_i = \{\Phi[-\mathbf{G}_{i-1}/\sigma_i]/\Phi[-\mathbf{G}_{i-1}/\sigma_{i-1}], k = 1, \dots, n\} \in \mathbb{R}^{n \times 1}$

  Compute $\widehat{s}_i$ according to (3.6).

  $\mathbf{U}_{i-1} \leftarrow$ draw weighted resample from $\mathbf{U}_{i-1}$ with weights $\boldsymbol{\omega}_i$ ▷ sample with replacement

  $(\mathbf{U}_i, \mathbf{G}_i) =$ MCMC-aCS($\mathbf{U}_{i-1}, \mathbf{G}_{i-1}, b$) ▷ Details on MCMC-aCS in [61]

  if (3.9) **then**

    break

Set $M \leftarrow i$

Estimate failure probability $\widehat{p}_{\text{SIS}} = \left(\prod_{i=1}^{M} \widehat{s}_i\right) \frac{1}{n} \sum_{k=1}^{n} \frac{\mathrm{I}(\mathrm{G}_M^k \leq 0)}{\Phi(-\mathrm{G}_M^k/\sigma_M)}$ ▷ (3.10)

return $\mathbf{U}_M, \mathbf{G}_M, \widehat{p}_{\text{SIS}}$.

Since $\mathcal{Y} \in \mathcal{H}$, the truncation

$$\widehat{\mathcal{Y}}_n(\boldsymbol{X}) = \sum_{j=0}^{n} b_j v_j(\boldsymbol{X}) \tag{4.4}$$

asymptotically converges to $\mathcal{Y}$ as $n \to \infty$ in the mean square sense. [87] demonstrates how to construct complete orthonormal bases of $\mathcal{H}$ as polynomial families for various standard input distribution types. In particular, if $F_{\boldsymbol{X}}(\boldsymbol{x}) = \Phi_d(\boldsymbol{x})$, where Φ_d denotes the d-variate independent standard-normal CDF, the tensorized, normalized probabilist's Hermite polynomials

$$\Psi_{\boldsymbol{k}}(\boldsymbol{U}) = \prod_{i=1}^{d} \psi_{k_j}(U_j) \tag{4.5}$$

form a complete orthonormal basis of $\mathcal{H}$. $\{\psi_j(U), j \in \mathbb{N}\}$ are the univariate, normalized (probabilist's) Hermite polynomials and $\boldsymbol{k} = (k_1, \dots, k_d) \in \mathbb{N}^d$. By means of the isoprobabilistic transformation $T : \boldsymbol{X} \to \boldsymbol{U}$ introduced in the previous section, we define PCEs in standard-normal space for the remainder of the paper. The PCE of maximum total order p reads

$$\widehat{\mathcal{Y}}_p(\boldsymbol{U}) = \sum_{|\boldsymbol{k}| \leq p} b_{\boldsymbol{k}} \Psi_{\boldsymbol{k}}(\boldsymbol{U}). \tag{4.6}$$

The total number of basis functions in the PCE, P, depends on the input dimension d and the maximum total polynomial order p:

$$P = \binom{d+p}{p}. \tag{4.7}$$

The projection in (4.3) can be transformed into an equivalent ordinary least squares (OLS) problem [7]. PCEs become computationally intractable if d is large, i.e., they cannot be used for problems with high-dimensional input due to the sheer number of basis functions and corresponding coefficients. In particular, the computation is rendered infeasible by the necessary number of operations to compute the set of P multi-indices and the necessary number of model evaluations to obtain meaningful estimates of the coefficients. Solution strategies to overcome these limitations (at least partially) include a hyperbolic truncation of the

index set (this means to replace the condition on the ℓ_1-norm in (4.6), $|\boldsymbol{k}| \leq p$, with one on a general ℓ_q-norm of $|\boldsymbol{k}|_\alpha = (\sum_{i=1}^d p_i^q)^{1/q} \leq p$ with $q < 1$) or enforcing a maximum interaction order (i.e., a maximum number of non-zero entries in $\boldsymbol{k}$) [9]. These approaches result in more parsimonious models and allow for PCEs to be applied in higher-dimensional problems, however do so at the cost of decreased model expressivity. Sparsity-inducing solvers have been proposed to relax the dimensionality constraint imposed by the size of the regression problem. Approaches may be based on a variety of solvers for the ℓ_1-regularized least squares problem such as least-angle regression (LARS) that is used for PCEs in [10], compressive sensing [88] and orthogonal matching pursuit [62, 76, 20] as well as sparse Bayesian learning methods [75, 35, 69, 78]. For a comprehensive overview, the reader is referred to the recent literature review and benchmark study [51, 50].

4.2. Basis adaptation via partial least squares. In order to obtain a parsimonious yet expressive model, we turn to low-dimensional model representations rather than sparse solutions to the full-dimensional model. To achieve this, the PCE representation is rotated onto a new basis defined by the variables $\boldsymbol{Z} = \mathbf{Q}^{\mathrm{T}}\boldsymbol{U}$, where $\mathbf{Q} \in \mathbb{R}^{d\times d}$ and $\mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \mathbf{I}$, with $\mathbf{I}$ denoting the identity matrix. This has first been proposed in [74]. The PCE with respect to the novel basis reads

$$\widehat{\mathcal{Y}}_p^{\mathbf{Q}}(\boldsymbol{U}) = \sum_{|\boldsymbol{k}|\leq p} a_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}(\boldsymbol{Z}) = \sum_{|\boldsymbol{k}|\leq p} a_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}\left(\mathbf{Q}^{\mathrm{T}}\boldsymbol{U}\right). \tag{4.8}$$

With $\boldsymbol{U}$ a standard-normal random vector and $\mathbf{Q}$ an orthogonal matrix, $\boldsymbol{Z}$ is a standard-normal random vector. Therefore, both original and transformed input space possess the same PCE basis, namely the probabilist's Hermite polynomials. Merely, a new set of coefficients $a_{\boldsymbol{k}}$ enters the formulation in the adapted basis. The columns of $\mathbf{Q}$ define linear combinations of the original input. We seek to choose $\mathbf{Q}$ such that most of the relevant information to construct an accurate surrogate $\mathcal{Y}$ is captured in the first m directions, where $m < d$ leads to dimensionality reduction. We retain only these first m columns of $\mathbf{Q}$ in the matrix $\mathbf{Q}_m$ and define a corresponding PCE of reduced dimension as

$$\widehat{\mathcal{Y}}_p^{\mathbf{Q}_m}(\boldsymbol{U}) = \sum_{|\boldsymbol{k}|\leq p} a_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}\left(\mathbf{Q}_m^{\mathrm{T}}\boldsymbol{U}\right), \tag{4.9}$$

where $\boldsymbol{k} \in \mathbb{N}^m$. [74] computes the basis adaptation $\mathbf{Q}_m$ by evaluating first- or second-order PCE coefficients only with a sparse-grid numerical quadrature. [77] couples this approach with compressive sensing to simultaneously identify $\mathbf{Q}_m$ and the PCE coefficients in the subspace. In [59], we show that important directions can be identified efficiently based on a set of original function evaluations via partial least squares (PLS).

PLS establishes a linear relationship between variables $\boldsymbol{U}$ and Y based on $n_{\mathcal{E}}$ observations of both quantities [85]. By $\mathbf{U}_{\mathcal{E}} \in \mathbb{R}^{n_{\mathcal{E}}\times d}$, we denote the matrix of $n_{\mathcal{E}}$ observations of $\boldsymbol{U}$ and by $\mathbf{Y}_{\mathcal{E}} \in \mathbb{R}^{n_{\mathcal{E}}\times 1}$ we denote the corresponding vector of scalar responses. PLS sequentially identifies m latent components $\{\boldsymbol{t}_j\}_{j=1}^m$, where $\boldsymbol{t}_j \in \mathbb{R}^{n_{\mathcal{E}}\times 1}$ such that they have maximum covariance with $\mathbf{Y}_{\mathcal{E}}$. After determining each $\boldsymbol{t}_j$, PLS assumes a linear relationship between $\boldsymbol{t}_j$ and $\mathbf{Y}_{\mathcal{E}}$ and evaluates the corresponding coefficient a_j of $\boldsymbol{t}_j$ by OLS. After each iteration, the matrices $\mathbf{U}_{\mathcal{E}}$ and $\mathbf{Y}_{\mathcal{E}}$ are deflated by the contribution of the j-th PLS-component. Components are extracted until a certain error criterion is met, which can be formulated, e.g., through the norm of the residual response vector or via cross-validation. Dimensionality-reducing regression methods such als PLS-based regression are known to shrink the regression coefficients towards zero to produce biased estimates in exchange for reducing the estimator variances (bias-variance-tradeoff). In this way, these dimensionality-reducing methods are able to produce smaller overall mean squared estimation errors. (see, e.g., [17] for PLS).

The nonlinear version of PLS in turn relaxes the assumption of a linear relationship between latent component and the response. A number of nonlinear PLS algorithms have been proposed [67]. Here we employ the approach of Refs. [84, 4] that introduces an additional loop into the algorithm for running a Newton-Raphson procedure iterating between the current latent component and the response. Ultimately, we are interested in computing the orthogonal transformation matrix $\boldsymbol{Q}_m$ in (4.9). PLS produces two different matrices $\boldsymbol{R}$ and $\boldsymbol{W}$ that are suitable to this end, which motivates two different flavors of PLS-PCE. In PLS-PCE-R as proposed in [59] (see Subsection 4.3), each nonlinear relationship between the $\{\boldsymbol{t}_j\}_{j=1}^m$ and the response is modelled as a univariate PCE. The coefficients of these univarate PCEs are computed simultaneously with the latent structure and the resulting model is a sum of univariate PCEs. Alternatively, the univariate PCEs

are discarded after the PLS-PCE algorithm terminates and a multivariate (sparse) PCE is constructed in the subspace formed by the so-called weights $\{\boldsymbol{w}_j\}_{j=1}^m$ leading to PLS-PCE-W (see Subsection 4.4).

4.3. PLS-PCE-R. PLS-PCE-R identifies m latent components and for each component, it returns the direction $\boldsymbol{r}_j$ and the univariate PCE along this direction. The univariate PCEs are defined by their polynomial orders $\{q_j\}_{j=1}^m$ and the associated coefficient vectors $\{\boldsymbol{a}_j\}_{j=1}^m$. The polynomial order is identified with leave-one-out cross validation [15]. For each (j-th) latent component, the nonlinear PLS iteration is repeated for different polynomial orders and q_j is chosen as the order minimizing the leave-one-out error. The PLS-PCE-R model reads

$$\widehat{\mathcal{Y}}(\boldsymbol{u}) = \widehat{a}_0 + \sum_{j=1}^{m} \left(\widehat{\boldsymbol{a}}_j^{q_j}\right)^{\mathrm{T}} \boldsymbol{\psi}_{q_j}\left[\boldsymbol{r}_j^{\mathrm{T}}\left(\boldsymbol{u}-\boldsymbol{\mu}_{\mathbf{U}}\right)\right], \tag{4.10}$$

where $\widehat{a}_0 = \widehat{\mathbb{E}}[\mathbf{Y}]$, $\boldsymbol{\psi}_{q_j}(\boldsymbol{U})$ is a vector function assembling the evaluations of the one-dimensional Hermite polynomials up to order q_j and $\boldsymbol{\mu}_{\mathbf{U}}$ is the columnwise sample mean of $\mathbf{U}_{\mathcal{E}}$. The model structure is illustrated in Fig. 2. The PLS directions $\boldsymbol{r}_j$ can be evaluated in terms of the PLS weights $\boldsymbol{w}_j$ and loads $\boldsymbol{p}_j$ through the following recursive relation [31]

$$\begin{aligned} \boldsymbol{r}_1 &= \boldsymbol{w}_1 \\ \boldsymbol{r}_j &= \boldsymbol{w}_j - \boldsymbol{r}_{j-1}\left(\boldsymbol{p}_{j-1}^{\mathrm{T}}\boldsymbol{w}_j\right). \end{aligned} \tag{4.11}$$

$\mathbf{R} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_m] \in \mathbb{R}^{d\times m}$ is a matrix collecting all PLS directions. $\mathbf{R}$ is not necessarily orthogonal, i.e., in general $\mathbf{R}^{\mathrm{T}}\mathbf{R} \neq \mathbf{I}$. However, in [59] it is shown that $\mathbf{R}^{\mathrm{T}}\mathbf{R} \approx \mathbf{I}$ when $n_{\mathcal{E}}$ is large and $\mathbf{U}_{\mathcal{E}}^{\mathrm{T}}\mathbf{U}_{\mathcal{E}}$ is diagonal, which is the case if $\mathbf{U}_{\mathcal{E}}$ is drawn from φ_d. In this case, (4.10) is equivalent to a PCE of the form (4.9), where only main effects in the latent components are considered.

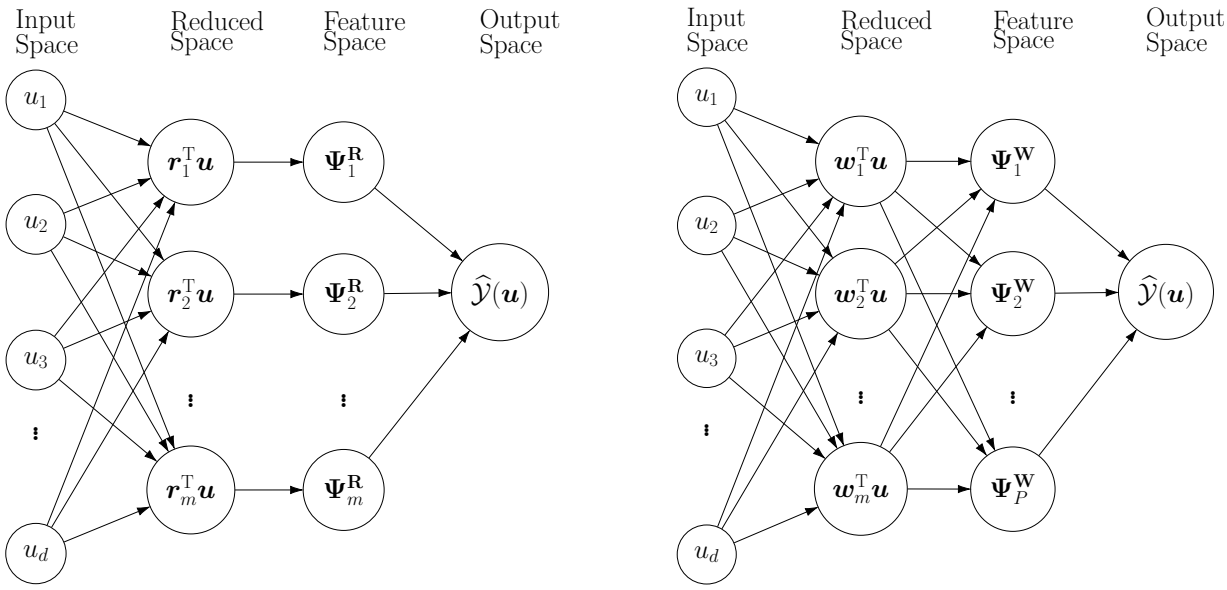
4.4. PLS-PCE-W. PLS-PCE-W defines $\mathbf{W}$ as basis of the subspace rather than $\mathbf{R}$, where $\mathbf{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m] \in \mathbb{R}^{d\times m}$. Within linear PLS, the columns of $\mathbf{W}$ form an orthogonal basis. Within nonlinear PLS, the Newton-Raphson step may introduce deviations from orthogonality, which are however negligible in all tested examples. The univariate PCEs obtained through the Newton-Raphson step will be optimal with respect to $\mathbf{R}$, not $\mathbf{W}$. Thus, in PLS-PCE-W these univariate polynomials are discarded once $\mathbf{W}$ is identified and a multivariate (sparse) PCE is constructed in the subspace defined by $\mathbf{W}$ using least-angle regression and a hyperbolic truncation scheme for the multivariate PCE basis as proposed by [10]. In this way PLS-PCE-W achieves more flexibility compared to PLS-PCE-R by including interactions of the latent components in exchange for a departure from optimality in the match between latent component and surrogate model. In analogy to (4.9), the PLS-PCE-W model reads

$$\widehat{\mathcal{Y}}(\boldsymbol{u}) = \widehat{a}_0 + \sum_{\boldsymbol{k}\in\boldsymbol{\alpha}} \widehat{a}_{\boldsymbol{k}}\boldsymbol{\Psi}_{\boldsymbol{k}}\left[\mathbf{W}^{\mathrm{T}}\left(\boldsymbol{u}-\boldsymbol{\mu}_{\mathbf{U}}\right)\right], \tag{4.12}$$

where $\boldsymbol{\alpha} \in \mathbb{N}^{P\times d}$ is the multi-index set, which indicates the polynomial orders of the d univariate polynomials in each of the P multivariate polynomials as obtained with LARS. Both PLS-PCE-R and PLS-PCE-W are summarized in Algorithm 4.1. In the following, we will use the PLS-PCE-W model as we observed a superior performance for this model compared to PLS-PCE-R models in the context of the proposed approach.

5. Learning PLS-PCE models in each SIS level.

5.1. The sequential subspace importance sampler. We recently proposed to reconstruct low-dimensional PLS-PCE-W models in each level of SIS to improve the tractability of high-dimensional reliability analysis with computationally expensive models [58]. We term this approach sequential subspace importance sampling or SSIS. The efficiency of SIS benefits from surrogate modelling through a considerable reduction of required model evaluations. The PLS-PCE model alone, being a global surrogate model, is a relatively limited tool for reliability analysis. Combining it with SIS provides the means to sequentially move the training set towards relevant regions in the input space and thereby renders difficult reliability problems accessible to surrogate modelling. At the i-th SSIS level, a new *local* training set is sampled from the current importance density h_i through a resampling step on the N available samples from h_i. The new *local* training set is appended to the *global* training set comprising earlier designs from levels 1 through $i-1$. Based on the



(a) PLS-PCE-R (b) PLS-PCE-W

Fig. 2: Structure of two different PLS-PCE models, where $\boldsymbol{\Psi}_j^{\mathbf{W}} = \boldsymbol{\Psi}_{\boldsymbol{\alpha}_j}$ as defined in (4.12) and $\boldsymbol{\Psi}_j^{\mathbf{R}} = (\widehat{\boldsymbol{a}}_j^{q_j})^{\mathrm{T}}\boldsymbol{\psi}_{q_j}$ as seen from (4.10). Essential differences exist in the choice of the reduced space basis (layer 2) and the modelling of cross-terms when mapping from reduced to feature space (layers 2 & 3) with PLS-PCE-W (b).

updated *global* training set, a new PLS-PCE model is constructed and SIS is rerun for $i+1$ levels from h_0 to obtain samples for the next *local* training set. Due to this restart, it is sensible to let previously used local training sets remain in the global training set such that the i-th surrogate model accurately predicts the LSF output along the entire path of samples moving from the nominal distribution h_0 to h_i. The restart itself incurs no additional LSF evaluations and serves to stabilize the method: Without restart, the computation of σ_{i+1} according to (3.8) is based on two different surrogate models: the most recent model constructed in level i appears in the numerator of the sample CoV of the weights and the model constructed in level $i-1$ appears in the denominator. These models may however be too different from one another to admit a solution in (3.8), i.e., to achieve the prescribed CoV δ_{target} between two subsequent IS densities.

In an additional step, before propagating the intermediate importance density to the next level of the SSIS algorithm, we introduce AL. This ensures a prescribed surrogate model accuracy in regions of high probability mass of the current sampling density. In turn, this refined surrogate model is used to propagate samples to the next level. When the underlying SIS algorithm reaches convergence, a final AL procedure, performed over samples of the final importance density, ensures that the probability of failure is estimated with a surrogate model that captures the failure hypersurface well. This approach is termed adaptive sequential subspace importance sampling or ASSIS.

Active learning has emerged in the late 1980s as a subfield of machine learning [72] and was known in the statistical theory of regression as optimal experimental design since the early 1970s [26]. At its heart is the idea that supervised learning algorithms can perform better if allowed to choose their training data. We consider a 'pool-based sampling' variant of active learning, in which a large pool of unlabeled data points are made available to the algorithm. Within SIS, one has n samples from h_i available in the i-th level. The algorithm then selects n_{add} points that are labeled (i.e. for which the LSF is evaluated) and added to the training set based on a measure of information gain. This measure typically takes the form of a learning

Algorithm 4.1 PCE-driven PLS algorithm [59]

Input Input matrix $\mathbf{U}_{\mathcal{E}}$ and output vector $\mathbf{Y}_{\mathcal{E}}$, maximum polynomial order p

Set $\mathbf{E} = \mathbf{U}_{\mathcal{E}} - \boldsymbol{\mu}_{\mathbf{U}}$, $\mathbf{F} = \mathbf{Y}_{\mathcal{E}} - \boldsymbol{\mu}_{\mathbf{Y}}$, $\epsilon_w = 10^{-3}$, $\epsilon_y = 10^{-3}$, $j = 1$
repeat
  Compute weight $\boldsymbol{w}_j^0 = \mathbf{E}^{\mathrm{T}}\mathbf{F}/\|\mathbf{E}^{\mathrm{T}}\mathbf{F}\|$
  for $q \leftarrow 1, p$ **do**
    Set $\boldsymbol{w}_j^q = \boldsymbol{w}_j^0$
    repeat
      Compute score $\boldsymbol{t}_j^q = \mathbf{E}\boldsymbol{w}_j^q$
      Fit a 1D PCE of order q $\widehat{\boldsymbol{a}}_j^q \leftarrow$ fit $\left[\mathbf{F} = (\boldsymbol{a}_j^q)^{\mathrm{T}}\boldsymbol{\psi}_q(\boldsymbol{t}_j^q) + \boldsymbol{\epsilon}\right]$
      Set $\widehat{\mathcal{M}}_j^q(t) = (\widehat{\boldsymbol{a}}_j^q)^{\mathrm{T}}\boldsymbol{\psi}_q(\boldsymbol{t}_j^q)(t)$
      Compute the error $\boldsymbol{e} = \mathbf{F} - (\widehat{\boldsymbol{a}}_j^q)^T\boldsymbol{\psi}_q(\boldsymbol{t}_j^q)$
      Compute $\Delta\boldsymbol{w}_j^q = (\mathbf{A}^{\mathrm{T}}\mathbf{A})^{-1}\mathbf{A}^{\mathrm{T}}\boldsymbol{e}$ with $\mathbf{A} = \nabla_{\boldsymbol{w}}(\widehat{\boldsymbol{a}}_j^q)^{\mathrm{T}}\boldsymbol{\psi}_q(\mathbf{E}\boldsymbol{w})$
      Set $\boldsymbol{w}_j^q \leftarrow \boldsymbol{w}_j^q + \Delta\boldsymbol{w}_j^q$
      Normalize $\boldsymbol{w}_j^q \leftarrow \boldsymbol{w}_j^q/\|\boldsymbol{w}_j^q\|$
    until $\|\Delta\boldsymbol{w}_j^q\|$ is smaller than ϵ_w
    Evaluate the relative leave-one-out error ϵ_{LOO}^q as in [10]
  Set $\{q_j, \widehat{\boldsymbol{a}}_j^{q_j}, \boldsymbol{w}_j^{q_j}\}$ as the triple $\{q, \widehat{\boldsymbol{a}}_j^q, \boldsymbol{w}_j^q\}$ with the smallest ϵ_{LOO}^q
  Compute score: $\boldsymbol{t}_j^{q_j} = \mathbf{E}\boldsymbol{w}_j^{q_j}$
  Compute load: $\boldsymbol{p}_j^{q_j} = \mathbf{E}^{\mathrm{T}}\boldsymbol{t}_j^{q_j}/((\boldsymbol{t}_j^{q_j})^{\mathrm{T}}\boldsymbol{t}_j^{q_j})$
  Deflate: $\mathbf{E} \leftarrow \mathbf{E} - \boldsymbol{t}_j^{q_j}(\boldsymbol{p}_j^{q_j})^{\mathrm{T}}$, $\mathbf{F} \leftarrow \mathbf{F} - (\widehat{\boldsymbol{a}}_j^{q_j})^{\mathrm{T}}\boldsymbol{\psi}_{q_j}(\boldsymbol{t}_j^{q_j})$
  $j \leftarrow j + 1$
until change in $\|\mathbf{F}\|$ is smaller than ϵ_y
Compute $\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_m]$ according to (4.11) ▷ For the R-based version of PLS-PCE
Build $\widehat{\mathcal{Y}}(\boldsymbol{u})$ according to (4.10)
Gather $\mathbf{W} = [\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_m]$ ▷ For the W-based version of PLS-PCE
Build $\widehat{\mathcal{Y}}(\boldsymbol{u})$ according to (4.12) and [10]
return $\mathbf{R}/\mathbf{W}$, $\widehat{\mathcal{Y}}(\boldsymbol{u})$

function $\mathcal{L}$ that is maximized over the sample pool to perform selection. The learning function employed in the context of SSIS is discussed in Subsection 5.2.

The probability of failure estimator for SSIS/ASSIS is analogous to (3.10) with the difference that SIS is performed with an LSF approximation $\widehat{G}$ that is based on the final surrogate model:

$$\widehat{p} = \left(\prod_{i=1}^{M} \widehat{s}_i\right) \frac{1}{n}\sum_{k=1}^{n} \frac{\mathrm{I}(\widehat{G}\left(\boldsymbol{u}^k\right) \leq 0)\varphi_d\left(\boldsymbol{u}^k\right)}{\eta_M\left(\boldsymbol{u}^k\right)}, \quad \boldsymbol{u}^k \overset{i.i.d.}{\sim} h_M. \tag{5.1}$$

The ratio of normalizing constants $\{\widehat{s}_i\}_{i=1}^M$ are estimated as

$$\widehat{s}_i = \frac{1}{n}\sum_{k=1}^{n} \widehat{\omega}_i\left(\boldsymbol{u}^k\right) = \frac{1}{n}\sum_{k=1}^{n} \frac{\Phi(-\widehat{G}\left(\boldsymbol{u}^k\right)/\sigma_i)}{\Phi(-\widehat{G}\left(\boldsymbol{u}^k\right)/\sigma_{i-1})}, \quad \boldsymbol{u}^k \overset{i.i.d.}{\sim} h_i. \tag{5.2}$$

The SSIS/ASSIS algorithms are stopped based on a similar criterion as for SIS given in (3.9):

$$\widehat{\mathbb{COV}}\left[\frac{\mathrm{I}(\widehat{G}(\boldsymbol{U}) \leq 0)}{\Phi(-\widehat{G}\left(\boldsymbol{U}\right)/\sigma_i)}\right] \leq \delta_{\text{target}}. \tag{5.3}$$

Fig. 3 depicts flow diagrams of the SSIS and ASSIS algorithms.

5.2. Active learning of low-dimensional model representations. In the context of SSIS, the learning function $\mathcal{L}$ should express the prediction uncertainty at each sample of the current IS density for a given PLS-PCE-W surrogate. This prediction uncertainty is due to the estimation of both the subspace and the surrogate model with a finite-sized training set. We describe this uncertainty with the variance of the LSF based on the surrogate model conditional on $\boldsymbol{u}$, $\mathbb{V}[\widehat{G}|\boldsymbol{U}=\boldsymbol{u}]$. Note that, whenever the distribution with respect to which $\mathbb{E}[\cdot]$ or $\mathbb{V}[\cdot]$ are evaluated is not made explicit as a subscript, it is implicitly assumed as the distribution of the argument. For example, $\mathbb{V}[\widehat{G}|\boldsymbol{U}=\boldsymbol{u}] = \mathbb{V}_{f_{\widehat{G}|\boldsymbol{u}}}[\widehat{G}|\boldsymbol{U}=\boldsymbol{u}]$.

Let $\boldsymbol{\xi}_0 = \boldsymbol{a} \in \mathbb{R}^{P\times 1}$ and $\boldsymbol{\xi}_j = \boldsymbol{w}_j \in \mathbb{R}^{d\times 1}$, $j = 1,\dots,m$, such that $\boldsymbol{\xi} = [\boldsymbol{\xi}_0^{\mathrm{T}}, \boldsymbol{\xi}_1^{\mathrm{T}} \dots, \boldsymbol{\xi}_m^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{(md+P)\times 1}$ is the collection of all $md+P$ model parameters. Further, let $\boldsymbol{\xi}^\star$ denote their corresponding point estimates returned by Algorithm 4.1. The first-order expansion of $\widehat{\mathbb{V}}[\widehat{G}|\boldsymbol{u}]$ around $\boldsymbol{\xi}^\star$ reads

$$\widehat{\sigma}^2_{\widehat{G}}(\boldsymbol{u}) = \widehat{\mathbb{V}}[\widehat{G}|\boldsymbol{u}] \approx \left[\frac{\partial \widehat{G}}{\partial \boldsymbol{\xi}}\right]^{\mathrm{T}}_{\boldsymbol{\xi}=\boldsymbol{\xi}^\star} \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\xi}\boldsymbol{\xi}} \left[\frac{\partial \widehat{G}}{\partial \boldsymbol{\xi}}\right]_{\boldsymbol{\xi}=\boldsymbol{\xi}^\star}, \tag{5.4}$$

where $\widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\xi}\boldsymbol{\xi}}$ is an estimate of the parameter covariance matrix. Next, we neglect the pairwise cross-covariance of PCE coefficients $\boldsymbol{a}$ and the subspace components $\boldsymbol{w}_j$ and consider

$$\widehat{\sigma}^2_{\widehat{G}}(\boldsymbol{u}) = \widehat{\mathbb{V}}[\widehat{G}|\boldsymbol{u}] \approx \sum_{j=0}^{m} \left[\frac{\partial \widehat{G}(\boldsymbol{u},\boldsymbol{\xi})}{\partial \boldsymbol{\xi}_j}\right]^{\mathrm{T}}_{\boldsymbol{\xi}_j=\boldsymbol{\xi}_j^\star} \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\xi}_j\boldsymbol{\xi}_j} \left[\frac{\partial \widehat{G}(\boldsymbol{u},\boldsymbol{\xi})}{\partial \boldsymbol{\xi}_j}\right]_{\boldsymbol{\xi}_j=\boldsymbol{\xi}^\star} \tag{5.5}$$

This significantly reduces the number of $\boldsymbol{\Sigma}_{\boldsymbol{\xi}\boldsymbol{\xi}}$-entries that have to be estimated, namely from $P^2+2Pmd+m^2d^2$ to P^2+md^2. More importantly, the coefficients of the PCE, $\boldsymbol{\xi}_0$, are obtained with linear regression while the subspace, $\{\boldsymbol{\xi}_j\}_{j=1}^m$, is obtained in the inner loop of Algorithm 4.1 with nonlinear regression. Due to this sequential estimation of the $\{\boldsymbol{\xi}_j\}_{j=0}^m$, there is no straightforward way of obtaining an estimate of the full covariance matrix. In particular, we are not aware of such an estimate for the parameters of nonlinear PLS. Hence, this simplification is not only convenient but also necessary in practice. We do observe, however, that the off-diagonal elements of the estimated component-wise cross-covariance matrices $\widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\xi}_j\boldsymbol{\xi}_j}$ are several orders of magnitude smaller compared to the the main diagonal elements. This indicates that the model uncertainty estimate is dominated by parameter variances. In fact, in a more radical approach that remains unexplored in this work, one may consider parameter variances only (i.e., only $P+md$ entries of the full covariance matrix are retained). Such an approach is, e.g., used in [64]. Under some regularity conditions, the estimator $\boldsymbol{\xi}_j^*$ is consistent [86] and converges in distribution to a multivariate Gaussian distribution with mean $\boldsymbol{\xi}_j$ and covariance $\boldsymbol{\Sigma}_{\boldsymbol{\xi}_j\boldsymbol{\xi}_j}$. In analogy with linear regression, an estimate of $\boldsymbol{\Sigma}_{\boldsymbol{\xi}_j\boldsymbol{\xi}_j}$ is given through

$$\widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\xi}_j\boldsymbol{\xi}_j} = \widehat{\sigma}^2_\epsilon \left(\mathbf{A}_j^{\mathrm{T}}\mathbf{A}_j\right)^{-1} \tag{5.6}$$

with

$$\mathbf{A}_j = \left[\frac{\partial \widehat{\mathcal{Y}}(\boldsymbol{u},\boldsymbol{\xi})}{\partial \boldsymbol{\xi}_j}\right]_{\substack{\boldsymbol{\xi}=\boldsymbol{\xi}^\star \\ \boldsymbol{u}=\mathbf{U}_\mathcal{E}}} \in \mathbb{R}^{n_\mathcal{E}\times d} \quad \text{and} \quad \widehat{\sigma}^2_\epsilon = \frac{1}{n_\mathcal{E}-md-P}\sum_{k=1}^{n_\mathcal{E}} \left[\mathbf{Y}^k_\mathcal{E} - \widehat{\mathcal{Y}}(\mathbf{U}^k_\mathcal{E})\right]^2. \tag{5.7}$$

$\widehat{\sigma}^2_\epsilon$ is the standard estimator for the error variance of the surrogate model. $\mathbf{A}_j$ is the gradient of the surrogate model $\mathcal{Y}$ with respect to the model parameters evaluated at each of the $n_\mathcal{E}$ points in the training set $\mathbf{U}_\mathcal{E}$. $\mathbf{A}_0$ is merely the design matrix and does not require the computation of any derivatives. Note that computing the gradients $\{\mathbf{A}_j\}_{j=0}^m$ does not require any model evaluations. For $j=0$, it is

$$\frac{\partial \widehat{\mathcal{Y}}(\boldsymbol{u},\boldsymbol{\xi})}{\partial \boldsymbol{\xi}_0} = \left[\boldsymbol{\Psi}_i\left(\mathbf{W}^{\mathrm{T}}(\boldsymbol{u}-\boldsymbol{\mu}_\mathbf{U})\right)\right]_{i=1}^{P-1} \quad \text{with} \quad \mathbf{W} = [\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \dots, \boldsymbol{\xi}_m]. \tag{5.8}$$

For $j > 0$ and recalling $\boldsymbol{z} = \mathbf{W}^{\mathrm{T}}(\boldsymbol{u} - \boldsymbol{\mu}_{\mathbf{U}})$, we have

$$\begin{aligned}\frac{\partial \Psi_{\boldsymbol{k}}(\boldsymbol{z})}{\partial \boldsymbol{\xi}_j} &= \frac{\partial}{\partial \boldsymbol{w}_j}\Psi_{\boldsymbol{k}}(\mathbf{W}^{\mathrm{T}}(\boldsymbol{u} - \boldsymbol{\mu}_{\mathbf{U}})) \\ &= (\boldsymbol{u} - \boldsymbol{\mu}_{\mathbf{U}})\,\frac{\partial \Psi_{\boldsymbol{k}}(z_j)}{\partial z_j} \\ &= (\boldsymbol{u} - \boldsymbol{\mu}_{\mathbf{U}})\left(\prod_{\substack{i=1\\ i\neq j}}^{m}\psi_{k_i}(\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{u})\right)\frac{\partial \psi_{k_j}(\boldsymbol{w}_j^{\mathrm{T}}\boldsymbol{u})}{\partial z_j} \\ &= (\boldsymbol{u} - \boldsymbol{\mu}_{\mathbf{U}})\left(\prod_{\substack{i=1\\ i\neq j}}^{m}\psi_{k_i}(\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{u})\right)\sqrt{k_j}\psi_{k_j-1}(\boldsymbol{w}_j^{\mathrm{T}}\boldsymbol{u}).\end{aligned} \tag{5.9}$$

In the last equality, we have used the following expression for derivatives of univariate normalized Hermite polynomials:

$$\frac{\mathrm{d}\psi_n(x)}{\mathrm{d}x} = \sqrt{n}\psi_{n-1}(x). \tag{5.10}$$

$\partial \widehat{\mathcal{Y}}(\boldsymbol{u}, \boldsymbol{\xi})/\partial \boldsymbol{\xi}_j$ for $j > 0$ follows as

$$\frac{\partial \widehat{\mathcal{Y}}(\boldsymbol{u}, \boldsymbol{\xi})}{\partial \boldsymbol{\xi}_j} = \frac{\partial \widehat{\mathcal{Y}}(\boldsymbol{z})}{\partial \boldsymbol{w}_j} = \sum_{\boldsymbol{k}\in\boldsymbol{\alpha}} \widehat{a}_{\boldsymbol{k}}\frac{\partial \Psi_{\boldsymbol{k}}(\boldsymbol{z})}{\partial \boldsymbol{\xi}_j}, \qquad j > 0. \tag{5.11}$$

The partial derivative $\partial \widehat{G}/\partial \boldsymbol{\xi}_j$ in (5.5) can be evaluated using the chain rule of differentiation, which yields

$$\frac{\partial \widehat{G}}{\partial \boldsymbol{\xi}_j} = \frac{\partial \widehat{G}}{\partial \widehat{\mathcal{Y}}}\frac{\partial \widehat{\mathcal{Y}}}{\partial \boldsymbol{\xi}_j}. \tag{5.12}$$

The first term on the right-hand side is typically easy to compute and often equals ± 1 (the sign is irrelevant as the gradient enters the quadratic form in (5.5)) if the LSF returns the difference between the model output and a prescribed threshold. In this case, the first factor on the right-hand side of (5.12) drops out. If, however, the LSF is not continuously differentiable with respect to the model, we may construct a surrogate model of G directly by using a training set containing LSF evaluations rather than model evaluations in Algorithm 4.1. The second term on the right-hand side can be obtained reusing the gradients from the $\mathbf{A}_j$ in (5.7) that — in this case — are not evaluated at the training set and thus are functions of $\boldsymbol{u}$.

When setting up the learning function, there is a distinction to be made between an intermediate SIS level and the final SIS level: In the intermediate level, the goal is to accurately estimate the ratios of normalizing constants and to propagate the samples to the next level. In the final level, the goal is to build the probability of failure estimator and thus to accurately approximate the true limit-state hypersurface. With this in mind, the learning functions for adapting the surrogate models in levels $i = 1, \ldots, M$, and after the final level are readily stated as

$$\mathcal{L}_G(\boldsymbol{u}) = \begin{cases}\sigma_{\widehat{G}}(\boldsymbol{u}), & \text{intermediate SIS level}\\ \sigma_{\widehat{G}}(\boldsymbol{u})/|\widehat{G}(\boldsymbol{u})|, & \text{after final SIS level.}\end{cases} \tag{5.13}$$

After the final level, SIS has converged and we are using samples from the final biasing density h_M to refit a surrogate model that captures the failure hypersurface well. The learning function in this case is defined in the spirit of the learning function put forward in [22]. The denominator penalizes samples whose image under $\widehat{G}$ is far away from 0 assuming that therefore they are themselves far away from the failure hypersurface. Such samples are unlikely to be misclassified as safe if located in the failure domain or vice versa. In all previous levels of SIS, there is no failure hypersurface to be approximated but only importance weights and

the resulting ratio of normalizing constants. Here, the denominator in the learning function is dropped as there is no benefit to penalizing samples with large absolute image values under $\widehat{G}$.

In each AL iteration, the pool is searched for one or several points maximizing $\mathcal{L}(\boldsymbol{u})$. If $n_{\text{add}} > 1$ new points are added per AL iteration, the current sample pool is transformed to the low-dimensional subspace defined by $\mathbf{W}$ in order to identify n_{add} clusters (e.g., with k-means). Clustering in the subspace circumvents the performance deterioration most clustering methods experience in high dimensions [40]. The point maximising (5.13) in each cluster is added to the training set. In this way, the algorithm avoids a local concentration of the training set in a single region and is also able to handle problems with multiple disconnected failure domains as long as these are contained in the subspace.

The active learning is terminated based on the maximum local standard deviation relative to the target average in the intermediate levels or based on the relative change of the probability of failure estimate after the final level:

$$\left\{ \begin{array}{ll} \max\limits_{k=1,\dots,n} \left(\frac{\sigma_{\widehat{G}}(\boldsymbol{u}_k)}{\widehat{\mathbb{E}}[\widehat{G}(\boldsymbol{U})]} \right) \leq \epsilon_{\text{AL}}, & \text{intermediate SIS level} \\ \frac{\widehat{p}-\widehat{p}_{\text{last}}}{\widehat{p}} \leq \epsilon_{\text{AL}}, & \text{after final SIS level} \end{array} \right\}, \tag{5.14}$$

where appropriate choices for ϵ_{AL} lie in $[10^{-2}, 10^{-1}]$. $\widehat{p}$ and $\widehat{p}_{\text{last}}$ denote the probability of failure estimate based on the current and the last training set within the AL loop. The probability of failure is estimated with a surrogate model-based run of SIS-aCS in each AL iteration. This causes no additional cost in terms of original model evaluations and ensures a reliable evaluation of the criterion even for extremely small failure probabilities. The active learning procedure is detailed in Algorithm 5.1 and the complete method is detailed in Algorithm 5.2.

Algorithm 5.1 Active Learning

Input LSF $G(\boldsymbol{u})$, AL error level ϵ_{AL}, # of AL clusters n_{add}, Polynomial order p, training set $\{\mathbf{U}_{\mathcal{E}}, \mathbf{G}_{\mathcal{E}}\}$,
    Sample pool $\mathbf{U}_{\text{pool}}$

while *true* **do** ▷ Active learning loop
    Run $[\mathbf{W}, \widehat{G}] = \text{PLS-PCE}(\mathbf{U}_{\mathcal{E}}, \mathbf{G}_{\mathcal{E}}, p, \text{'W'})$ ▷ Algorithm 4.1
    if (5.14) **then**
        break
    Identify n_{add} clusters among $\mathbf{U}_{\text{pool}}\mathbf{W}$ ▷ Clustering performed in the subspace defined by $\mathbf{W}$
    for each cluster **do**
        $\mathbf{U}_{\text{cluster}} = \{\mathbf{u} \in \mathbf{U}_{\text{pool}} : \mathbf{u} \in \text{cluster}\}$
        Evaluate $\mathbf{u}^\star = \text{argmax}[\mathcal{L}(\mathbf{U}_{\text{cluster}})]$ according to (5.5)–(5.7), (5.12), and (5.13).
        Append $\mathbf{U}_{\mathcal{E}} \leftarrow [\mathbf{U}_{\mathcal{E}}, \mathbf{u}^\star]$
        Append $\mathbf{G}_{\mathcal{E}} \leftarrow [\mathbf{G}_{\mathcal{E}}, G(\mathbf{u}^\star)]$
        Remove $\mathbf{u}^\star$ from $\mathbf{U}_{\text{pool}}$
return $\mathbf{U}_{\mathcal{E}}$, $\mathbf{G}_{\mathcal{E}}$, $\widehat{G}$.

6. Numerical experiments.

6.1. Error measures. In the following, we examine a series of examples of low to high input dimensionality characterized by varying degrees of nonlinearity of the LSF and varying number of disconnected failure regions. The computational cost of each approach is measured with the total number of required calls to the underlying computational model. The accuracy of the estimator is measured in terms of relative bias and CoV

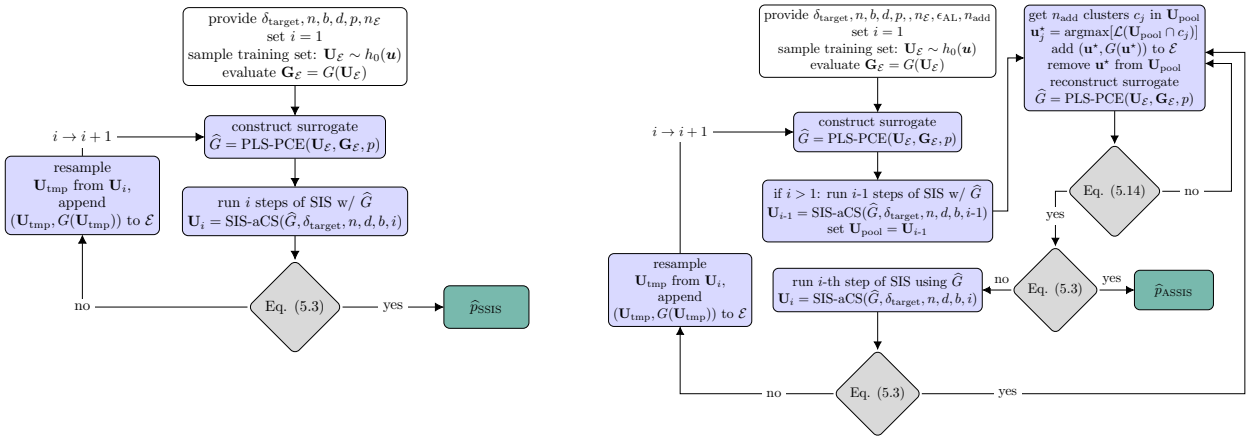
$$\text{relative Bias} = \frac{p - \mathbb{E}[\widehat{p}]}{p} \tag{6.1}$$

$$\text{CoV} = \frac{\sqrt{\mathbb{V}[\widehat{p}]}}{\mathbb{E}[\widehat{p}]}, \tag{6.2}$$

Algorithm 5.2 ASSIS (with PLS-PCE-W)

Input LSF $G(\boldsymbol{u})$, Target CoV δ_{target}, Samples per level n, Input dimension d, training set size $n_{\mathcal{E}}$, AL error
    level ϵ_{AL}, # of AL clusters n_{add}, Polynomial order p,

Set $i = 0$, $\sigma_i = \infty$, $h_i(\boldsymbol{u}) = \varphi_d(\boldsymbol{u})$
Initialize $\mathbf{U}_{\mathcal{E}} = [\cdot]$, $\mathbf{G}_{\mathcal{E}} = [\cdot]$
Sample $\mathbf{U}_0 = \{\boldsymbol{u}^k\}_{k=1}^n \in \mathbb{R}^{n \times d}$ ▷ $\boldsymbol{u}^k \overset{i.i.d.}{\sim} h_i(\boldsymbol{u})$
while *true* **do** ▷ Sequential importance sampling loop
    $i \leftarrow i + 1$
    Sample $\mathbf{U}_{\text{tmp}} = \{\boldsymbol{u}^k\}_{k=1}^{n_{\mathcal{E}}} \in \mathbb{R}^{n_{\mathcal{E}} \times d}$ ▷ $\boldsymbol{u}^k \overset{i.i.d.}{\sim} h_i(\boldsymbol{u})$
    Compute $\mathbf{G}_{\text{tmp}} = G(\mathbf{U}_{\text{tmp}}) \in \mathbb{R}^{n_{\mathcal{E}} \times 1}$
    Append $\mathbf{U}_{\mathcal{E}} \leftarrow [\mathbf{U}_{\mathcal{E}}, \mathbf{U}_{\text{tmp}}]$
    Append $\mathbf{G}_{\mathcal{E}} \leftarrow [\mathbf{G}_{\mathcal{E}}, \mathbf{G}_{\text{tmp}}]$
    if $i > 1$ **then**
        Run $\widehat{G} = \text{PLS-PCE}(\mathbf{U}_{\mathcal{E}}, \mathbf{G}_{\mathcal{E}}, p, \text{'W'})$ ▷ Algorithm 4.1
        Run $\mathbf{U}_{i-1}, \mathbf{G}_{i-1} = \text{SIS-aCS}(\widehat{G}, \delta_{\text{target}}, n, d, i-1)$ ▷ Algorithm 3.1
    Run $\mathbf{U}_{\mathcal{E}}, \mathbf{G}_{\mathcal{E}}, \widehat{G} = \text{Active Learning}(G(\boldsymbol{u}), \epsilon_{\text{AL}}, n_{\text{add}}, p, \mathbf{U}_{\mathcal{E}}, \mathbf{G}_{\mathcal{E}}, \mathbf{U}_{i-1})$ ▷ Algorithm 5.1
    Compute $\mathbf{G}_{i-1} = \widehat{G}(\mathbf{U}_{i-1}) \in \mathbb{R}^{n \times 1}$
    Compute σ_i according to (3.8)
    Compute $\widehat{\omega}_i$ and $\widehat{s}_i$ according to (5.2)
    $\mathbf{U}_{i-1}, \mathbf{G}_{i-1} \leftarrow$ resample from $\mathbf{U}_{i-1}, \mathbf{G}_{i-1}$ with weights $\widehat{\omega}_i(\mathbf{U}_{i-1})$ ▷ sample with replacement
    Run $\mathbf{U}_i, \mathbf{G}_i = \text{SIS-aCS}(\mathbf{U}_{i-1}, \mathbf{G}_{i-1})$ ▷ Perform a single MCMC step
    if (5.3) **then**
        Set $M \leftarrow i$
        Run $\mathbf{U}_{\mathcal{E}}, \mathbf{G}_{\mathcal{E}}, \widehat{G} = \text{Active Learning}(G(\boldsymbol{u}), \epsilon_{\text{AL}}, n_{\text{add}}, p, \mathbf{U}_{\mathcal{E}}, \mathbf{G}_{\mathcal{E}}, \mathbf{U}_{i-1})$ ▷ Algorithm 5.1
        break
Run $(\mathbf{U}_M, \mathbf{G}_M, \widehat{p}_{\text{ASSIS}}) = \text{SIS-aCS}(\widehat{G}_M, \delta_{\text{target}}, n, d, M)$ ▷ Algorithm 3.1
return $M, \mathbf{U}_M, \mathbf{G}_M, \widehat{p}_{\text{ASSIS}}$.



(a) SSIS (b) ASSIS

Fig. 3: Comparison of SIS-PLS-PCE with (right) and without (left) active learning.

Table 1: Low- to medium-dimensional investigated benchmark problems.

Problem	Failure probability	Inputs	Input Variables	Properties	References
Hat	$1.037 \cdot 10^{-4}$	2	standard-normal	Strongly nonlinear	[70]
Cantilever	$3.94 \cdot 10^{-6}$	2	Gaussian	Strongly nonlinear	[6]
4-Branch (acc. to [6])	$5.60 \cdot 10^{-9}$	2	standard-normal	Multiple failure regions; extremely rare event	[6, 81]
Borehole ($276.7 \frac{m^3}{\text{year}}$)	$1 \cdot 10^{-5}$	8	Log-normal, Uniform	Strongly nonlinear, No underlying low-dimensional structure	[1]
Truss (0.12m)	$1.6 \cdot 10^{-3}$	10	Log-normal, Gumbel	mildly nonlinear	[43]
Rare Truss (0.18m)	$1.02 \cdot 10^{-8}$	10	Log-normal, Gumbel	Extremely rare event; nonlinear	[43] (modified)
Quadratic ($\kappa = 5$)	$6.62 \cdot 10^{-6}$	10	standard-normal	Strongly nonlinear; Underlying low-dimensional structure	[24, 80]
Quadratic ($\kappa = 5$)	$6.62 \cdot 10^{-6}$	100	standard-normal	Strongly nonlinear; Underlying low-dimensional structure	[24, 80]

where p is the known exact probability of failure or a reference solution computed with a large number of samples as reported in the corresponding references in Table 1. Further, we compute the relative root mean squared error (RMSE) of the probability of any failure estimate $\widehat{p}$, which combines bias and variability of the estimator as

$$\text{relative RMSE} = \sqrt{\frac{\mathbb{E}[(p-\widehat{p})^2]}{p^2}} = \sqrt{\text{relative Bias}^2 + \left(\frac{\mathbb{E}[\widehat{p}]}{p}\right)^2 \text{CoV}^2} \tag{6.3}$$

The expectation and variance operators in the above equations are approximated by repeating each analysis 100 times. Additionally, the relative estimation error is defined as

$$\text{relative error} = \frac{\widehat{p}}{p}. \tag{6.4}$$

6.2. Low- and medium-dimensional examples. The subspace importance sampler is designed to tackle high-dimensional problems, yet its performance should not deteriorate as the problem dimension decreases. We first investigate its performance in eight examplatory problems with dimension $2 \leq d \leq 100$. We demonstrate how both SSIS and ASSIS cope with multiple failure domains, strong nonlinearities and extremely small target failure probabilities. In the interest of brevity, the examples are listed in Table 1 along with the problem dimension, target probability of failure and key characteristics of the problem. The references provided in Table 1 may be consulted for detailed descriptions of the problem setups.

We solve the example problems with SIS-aCS with $n = 2 \cdot 10^3$ samples per level and a burn-in period of $b = 5$ samples within each MCMC chain. As suggested in [61], we choose $\delta_{\text{target}} = 1.5$ for the exit criterion (3.9) for SIS-aCS as well as our surrogate-based samplers. We compare this reference to SSIS and ASSIS for which we use an initial sample size of $n_{\mathcal{E}} = 5d$. All underlying PLS-PCE-W models are computed with a maximum number of subspace directions of $m = 10$ and a maximum total polynomial degree of $|q|_{\ell_q} \leq 7$, where $q = 0.75$. To achieve a fair comparison between ASSIS and SSIS, we first run ASSIS and then SSIS with $n_{\mathcal{E}}$ for the latter chosen such that both methods use an approximately equal number of LSF evaluations. For both SSIS and ASSIS, we choose $n = 10^4$ with a burn-in period of $b = 30$. For ASSIS, we set $\epsilon_{\text{AL}} = 0.1$.

Within SSIS/ASSIS many samples per level and long burn-in periods are affordable as sampling is performed with the surrogate model. For ASSIS we select $n_{\text{add}} = 1$ unless prior knowledge of the problem structure suggests otherwise (the only exception in the set of examples considered here is the 4-branch function for which we select $n_{\text{add}} = 4$ as it features four relevant failure regions in the input space). Fig. 4 displays the performance of SIS, SSIS and ASSIS for the examples in Table 1 in terms of the error measures defined in (6.1)–(6.3) and the total number of LSF evaluations (with the original model).

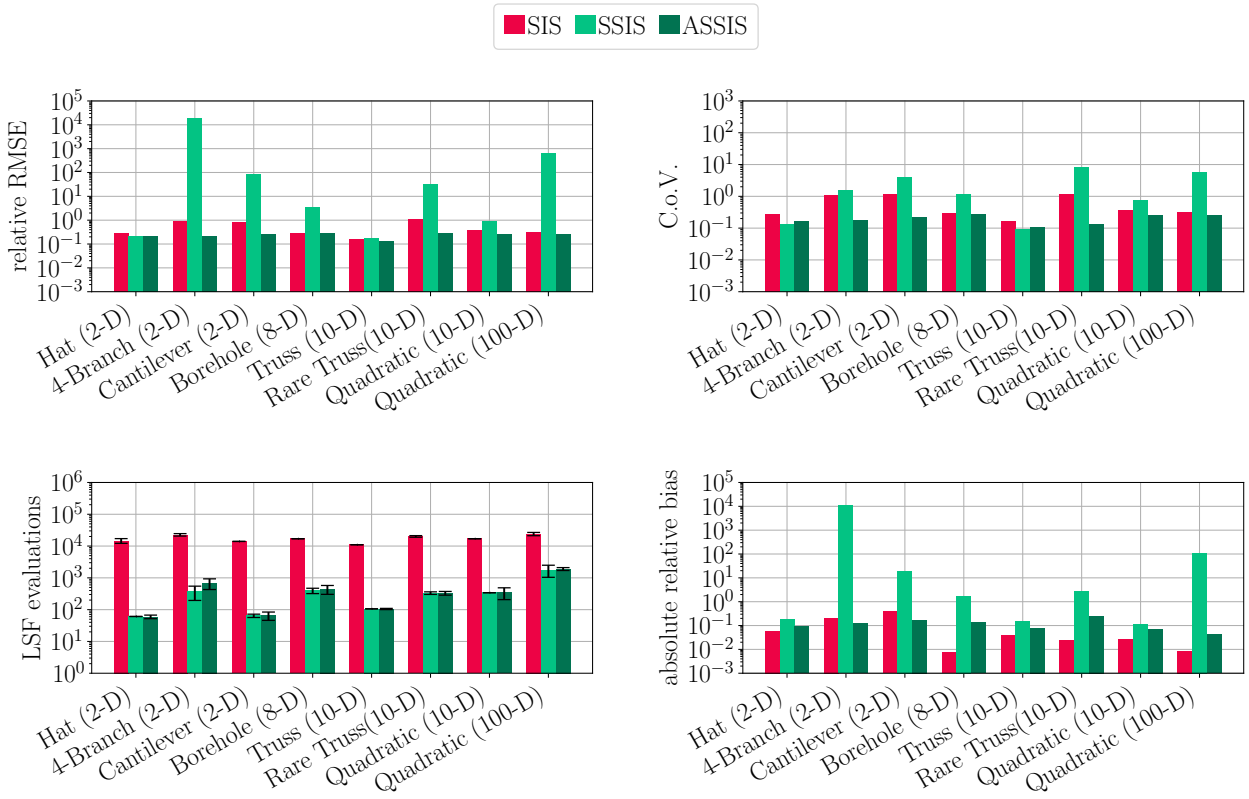


Fig. 4: Low- and medium-dimensional examples: accuracy and cost comparison. Cost error bars include ± 2 standard deviations.

For all showcased examples, ASSIS yields equally or more accurate estimates compared to SSIS at equal cost. It also either matches or outperforms SIS at significantly reduced costs. Except for the easiest problems, i.e., those featuring well-behaved (truss) or low-dimensional (2D hat) LSFs associated with comparatively large failure probabilities, the in-level adaptivity of ASSIS leads to significant bias correction (Fig. 4, bottom right) and variance reduction (Fig. 4, top right).

[61] discusses the choice of the MCMC sampler for SIS and find that aCS as employed here is outperformed by a Gaussian mixture proposal in low-dimensional problems, while the latter is the preferred choice as the problem dimension grows. Our method is designed for the solution of high-dimensional reliability problems and we thus consistently use aCS.

Comparing the truss and the rare truss models, the additional number of SIS levels required in the solution of the latter evidently leads to a deterioration of the SSIS estimate (Fig. 4, top left). This is due to single runs (less than 10 %) among the 100 repetitions in which the sampled training sets lead to extreme outliers in the failure probability estimates (Fig. 5). While this effect vanishes when increasing the number of samples in the training set, ASSIS offers a more cost-effective alternative to avoid such outliers by actively

learning an informative augmentation of adverse training sets. In this way, subspace identification and surrogate modelling errors cannot propagate and accumulate across the levels of SIS as they are controlled by the AL procedure. In fact, the phenomenon of rather rare but all the more severe outliers deteriorating the error mean and variability is a problem SSIS is facing not only in the rare truss example but also in the cantilever and both quadratic examples. Conversely, it is seen that in the 4-branch example, SSIS consistently and considerably overestimates the probability of failure while ASSIS captures the probability of failure rather well.

The two quadratic LSF models with 10 and 100 input dimensions demonstrate how the required num-

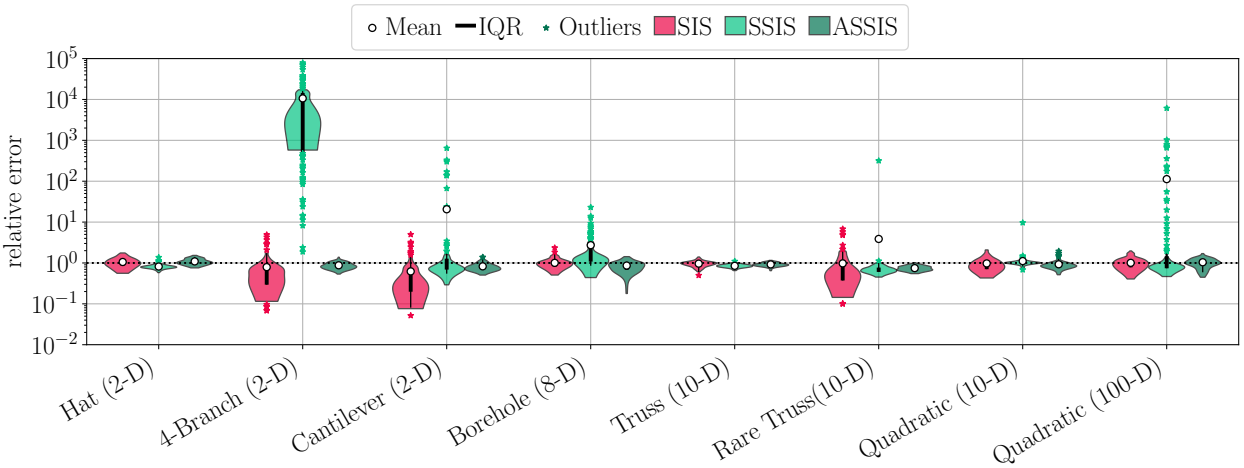


Fig. 5: Low- and medium-dimensional examples: violin plots of the relative error along with means, interquartile ranges (IQR) and outliers. For the sake of clarity, kernel density estimates are computed after excluding outliers based on the relative distance to the data median.

ber of LSF evaluations depends on the problem dimension in both surrogate-based approaches. This is due to the fact that the PLS-PCE model requires at least d (often more) samples to identify a suitable subspace. Thus, as described above, we choose $n_{\mathcal{E}}$ as a multiple of d. Since the surrogate-free version of SIS-aCS does not possess such a dependence on a problem dimension at all, the ratio of computational cost associated with SIS and ASSIS decreases as d increases. This observation also indicates that if d grows large enough, SIS-aCS will outperform any surrogate-based approach. This is expected for cases with $d = \mathcal{O}(10^5)$ and above; therefore, this observation is of little practical relevance for most engineering models, where ASSIS will likely be the most cost-effective choice.

6.3. High-dimensional example: Steel plate. We consider a modified version of the example given in [80, 49], which consists of a low-carbon steel plate of length 0.32 m, width 0.32 m, thickness $t = 0.01$ m, and a hole of radius 0.02 m located at the center. The Poisson ratio is set to $\nu = 0.29$ and the density of the plate is $\rho = 7850$ kg/m^3. The horizontal and vertical displacements are constrained at the left edge. The plate is subjected to a random surface load that acts on the right narrow plate side. The load is modelled as a log-normal random variable with mean $\mu_q = 60$ MPa and $\sigma_q = 12$ MPa. The Young's modulus $E(x, y)$ is considered uncertain and spatially variable. It is described by a homogeneous random field with lognormal marginal distribution, mean value $\mu_E = 2 \times 10^5$ MPa and standard deviation $\sigma_E = 3 \times 10^4$ MPa. The autocorrelation function of the underlying Gaussian field $\ln E$ is modeled by the isotropic exponential model

$$\rho_{\ln E}(\Delta x, \Delta y) = \exp\left\{-\frac{\sqrt{\Delta x^2 + \Delta y^2}}{l_E}\right\} \tag{6.5}$$

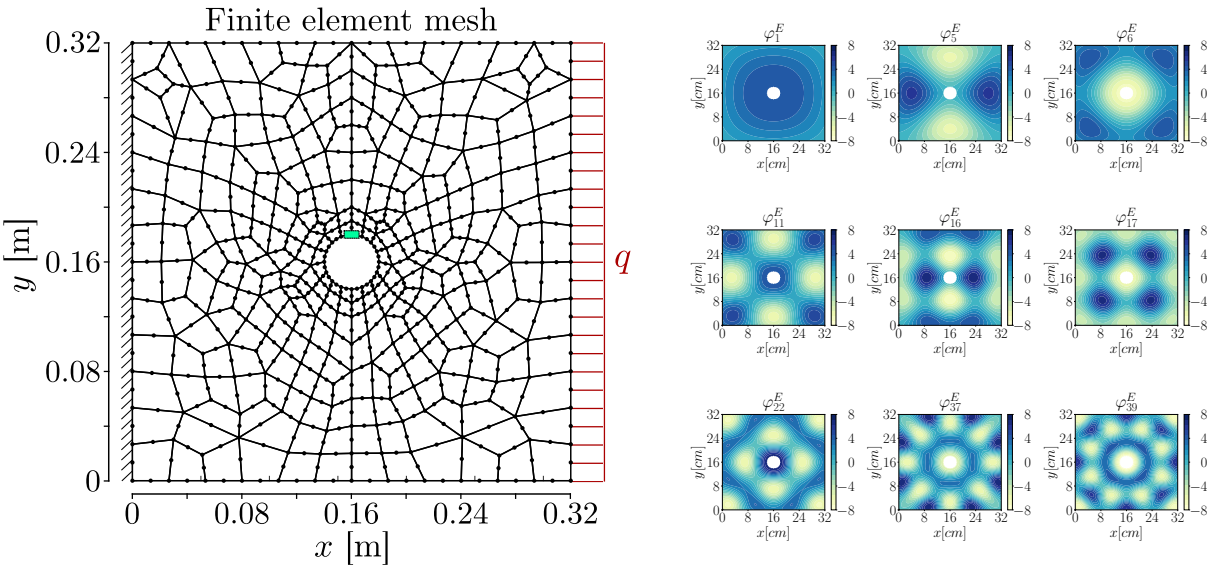


Fig. 6: Left: FE-mesh of 2D-plate model with control node of the first principal stress σ_1 .

with correlation length $l_{\ln E} = 0.04$m. The Gaussian random field $\ln E$ is discretized by a Karhunen-Loève-expansion (KLE) with $d_E = 868$, which yields a mean error variance of 7.5% and reads

$$E(x,y) = \exp\left\{\mu_{\ln E} + \sigma_{\ln E} \sum_{i=1}^{d_E} \sqrt{\lambda_i^E}\varphi_i^E(x,y)\xi_i\right\}. \tag{6.6}$$

$\mu_{\ln E}$ and $\sigma_{\ln E}$ are the parameters of the log-normal marginal distribution of E, $\{\lambda_i^q, \varphi_i^E\}$ are the eigenpairs of the correlation kernel in (6.5) and $\boldsymbol{\xi} \in \mathbb{R}^{d\times 1}$ is a standard-normal random vector. The most influential eigenfunctions (based on a global output-oriented sensitivity analysis of the plate model performed in [23]) are shown in Fig. 6 on the right.

The stress $(\boldsymbol{\sigma}(x,y) = [\sigma_x(x,y), \sigma_y(x,y), \tau_{xy}(x,y)]^T)$, strain $(\boldsymbol{\epsilon}(x,y) = [\epsilon_x(x,y), \epsilon_y(x,y), \gamma_{xy}(x,y)]^T)$ and displacement $(\mathbf{u}(x,y) = [u_x(x,y), u_y(x,y)]^T)$ fields of the plate are given through elasticity theory, namely the Cauchy-Navier equations [37]. Given the configuration of the plate, the model can be simplified under the plane stress hypothesis, which yields

$$G(x,y)\nabla^2\mathbf{u}(x,y) + \frac{E(x,y)}{2(1-\nu)}\nabla(\nabla\cdot\mathbf{u}(x,y)) + \mathbf{b} = 0. \tag{6.7}$$

Therein, $G(x,y) := E(x,y)/(2(1+\nu))$ is the shear modulus, and $\mathbf{b} = [b_x, b_y]^T$ is the vector of body forces acting on the plate. (6.7) is discretized with a finite-element method. That is, the spatial domain of the plate is discretized into 282 eight-noded quadrilateral elements, as shown in Fig. 6. In a grid independence study, the plate's probability of failure was found to slightly increase with decreasing mesh element size, which is likely due to the reduction of averaging effects when integrating higher-order KL-terms. However, for the purpose of testing ASSIS, the model is sufficiently accurate and features two important properties: 1. It possesses a low-dimensional structure that can be exploited with dimensionality-reducing surrogates. 2. It is truly high-dimensional in the sense that the solution does not only depend on a small subset of the input variables (i.e., the low-dimensional structure is not a trivial subspace of the original input space). The LSF is defined by means of a threshold for the the first principal plane stress

$$\sigma_1 = 0.5(\sigma_x + \sigma_y) + \sqrt{[0.5(\sigma_x + \sigma_y)]^2 + \tau_{xy}^2}$$

Table 2: Accuracy and cost of SIS, SSIS & ASSIS for the plate example based on 100 repetitions of the analysis. The reference $p_{\text{ref}} = 4.23 \cdot 10^{-6}$ is computed with 100 repeated runs of subset simulation with 10^4 samples per level with CoV = 0.0119 for the mean estimate.

Method	$\mathbb{E}[p]$	relative RMSE	CoV	relative bias	avg. # LSF evaluations
SIS-aCS	$3.88 \cdot 10^{-6}$	0.576	0.625	0.083	17000
SSIS	$3.99 \cdot 10^{-6}$	0.061	0.021	0.058	1300
ASSIS	$4.10 \cdot 10^{-6}$	0.036	0.021	0.030	1318

evaluated at node 11 (see green marker Fig. 6, left). Node 11 indicates a location where maximum plane stresses occur frequently in this example. The LSF reads

$$g(\boldsymbol{U}) = \sigma_{\text{threshold}} - \sigma_1(\boldsymbol{U}), \tag{6.8}$$

where $\sigma_{\text{threshold}} = 450$ MPa. The target probability of failure is determined to $p = 4.23 \cdot 10^{-6}$ with CoV = 0.0119 as the average of 100 repeated runs of subset simulation [3] with 10^4 samples per level.

SIS-aCS is run with $n = 2 \cdot 10^3$ samples per level and a burn-in period of $b = 5$ samples within each MCMC chain. SSIS and ASSIS are run with $n = 10^5$ samples per SIS level, a burn-in period $b = 30$ and an AL threshold of $\epsilon_{\text{AL}} = 0.1$. In the first level $n_{\mathcal{E}} = 900$ and in each additional level only $n_{\mathcal{E}} = 100$ samples are added in the initial sampling phase. Table 2 lists the average estimated probabilities of failure along with error measures and average number of required LSF evaluations. It is seen that both SSIS and ASSIS alleviate computational cost by more than an order of magnitude while at the same time reducing the relative RMSE by at least an order of magnitude. The decomposition of the RMSE in CoV and relative bias reveals that this is mostly due to variance reduction as SIS-aCS already yields a small bias.

A parameter study of important 'tweakable' parameters of ASSIS is depicted in Fig. 7. Parameters that are not subject to a parametric study are chosen as above, with the exception of $n = 10^4$ instead of $n = 10^5$. The estimation error and computational cost of ASSIS is analyzed for varying active learning threshold ϵ_{AL}, number of samples in the training set $n_{\mathcal{E}}$, the number of samples per SIS level n and the target CoV δ_{target} used for the SIS procedure. The scaling of 10% between the initial training set and all subsequent training samples is kept constant.

The parameters ϵ_{AL} and $n_{\mathcal{E}}$ describe the behaviour of the surrogate modelling and active learning procedures while n and δ_{target} describe SIS itself. Fig. 7 shows that increasing the target coefficient of variation leads to a reduced number of levels in the SIS procedure, which is directly associated with a reduction in computational cost. The reduction is relatively small here as most of the samples are added in the first level. By design, the number of required samples remains unaffected by varying the number of samples per SIS level, while the estimation error depends reciprocally on it. Conversely, and also by design, the computational cost depends monotonically on the choice of $n_{\mathcal{E}}$. If a majority of the used original LSF evaluations are added during an AL procedure, this relationship may be nonlinear. For the plate example, however, the initially drawn training samples at each level makes up for the majority of used original LSF evaluations, hence the linear dependency. The estimation errors decrease slightly with increasing training set size, although the effect is limited as high accuracy is already achieved with the first training set of the lowest investigated size.The fact that the subspace does not change significantly with increasing SIS level leaves little to be learned by adding more LSF evaluations to the training set. This is also the reason for the competitive performance of SSIS in this example. The estimation errors (as well as the computational cost in this case) remain unaffected by varying AL thresholds ϵ_{AL}, which is in line with the observation that a large fraction of the computational budget is spent on sampling the initial training set rather than the AL-based training set augmentation.

7. Concluding remarks. This paper proposes a method for the cost-efficient solution of high-dimensional reliability problems. We build on a recently introduced dimensionality reducing surrogate modelling technique termed partial least squares-driven polynomial chaos expansion (PLS-PCE) [59] and previous work, in

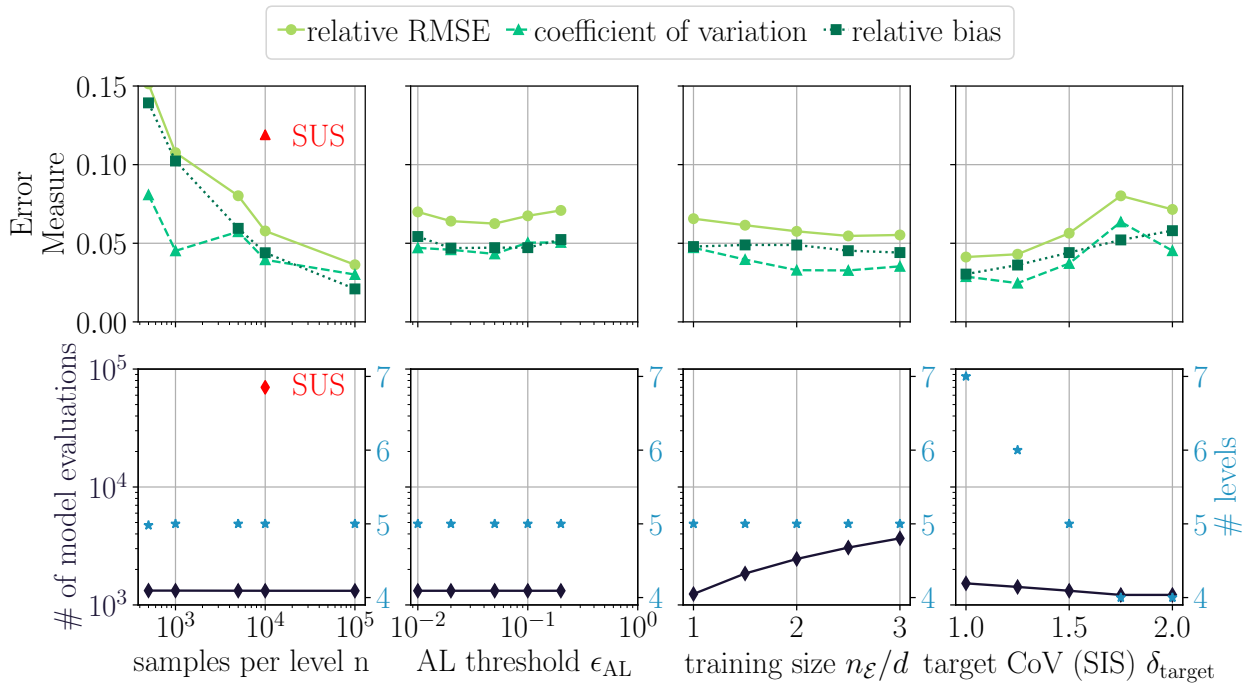


Fig. 7: Steel plate reliability using ASSIS: parameter influence studies. Top: Error measures as defined in (6.1)–(6.3) for ASSIS (green lines w/ markers). Bottom: Computational cost in terms of total number of limit-state function evaluations with the true computational model (left y-axis; black solid lines with diamond markers) and number of SIS levels to convergence (right y-axis; blue star markers). Top left: CoV of a subset simulation reference run with $n = 10^4$ samples per level (red triangle marker). Bottom left: total number of required limit-state function evaluations of a subset simulation reference run with $n = 10^4$ samples per level (red triangle marker).

which we use PLS-PCE surrogates to reconstruct biasing densities within a sequential importance sampling scheme [58] (sequential subspace importance sampling: SSIS). We refine this approach by devising an active learning procedure in each SIS level, which serves to effectively control the estimation error introduced by the surrogate-based importance density reconstructions. The learning procedure, i.e., the selection of new points for the training set , is driven by an estimate of both the subspace and surrogate model estimation error. This criterion can be generally used in polynomial chaos expansion-based active learning procedures.

We showcase the performance of SSIS and ASSIS in nine example applications with input dimensionality ranging from $d = 2$ to 869. The examples feature different typical caveats for reliability methods such as multiple failure domains, strongly nonlinear limit-state functions and extremely small target probabilities of failure. Depending on the example, we achieve a cost reduction of one to over two orders of magnitude with ASSIS compared to the reference method (sequential importance sampling with the original model) at equal or lower estimation errors. It is shown that SSIS is susceptible to the randomness of the initial training set occasionally producing outliers if the training set is adverse. The active learning procedure (ASSIS) remedies this drawback and stabilizes the estimator by augmenting potentially adverse training sets with informative additional samples.

The million dollar question, as with any surrogate model, is on the method's ability to generalize. Certainly, there exist examples that do not possess a suitable linear subspace as required by PLS-PCE modelling. Further, cases of model misspecification may arise if the computational model cannot be represented with PCEs (e.g., if it is a rational function). Then, the probability of failure estimate produced by ASSIS will be neither consistent nor unbiased. However, by means of coupling PLS-PCE with sequential importance

sampling, both requirements are relaxed somewhat as only a locally accurate surrogate model is required to propagate samples from one intermediate biasing density to the next. Hence, ASSIS can still be expected to perform well if the computational model may be represented in terms of a sequence of local linear subspaces on which the model can be approximated well with polynomials. Relaxing the orthogonality or even the linearity assumption on the latent space transformation likely bears potential to improve the performance of dimensionality-reduced PCEs. Doing so will require methods to track the appropriate PCE basis upon determining the law of the transformed input random vector (as these will not be standard-normal if the latent space transformation is no longer subject to the orthogonality constraint).

8. Acknowledgment. This project was supported by the German Research Foundation (DFG) through Grant STR 1140/6-1 under SPP 1886.

REFERENCES

[1] J. An and A. Owen, *Quasi-regression*, Journal of Complexity, 17 (2001), pp. 588 – 607.
[2] S. Au and J. Beck, *A new adaptive importance sampling scheme for reliability calculations*, Structural Safety, 21 (1999), pp. 135 – 158.
[3] S.-K. Au and J. L. Beck, *Estimation of small failure probabilities in high dimensions by subset simulation*, Probabilistic Engineering Mechanics, 16 (2001), pp. 263–277, https://doi.org/https://doi.org/10.1016/S0266-8920(01)00019-4, https://www.sciencedirect.com/science/article/pii/S0266892001000194.
[4] G. Baffi, E. Martin, and A. Morris, *Non-linear projection to latent structures revisited (the neural network PLS algorithm)*, Computers & Chemical Engineering, 23 (1999), pp. 1293–1307.
[5] M. Balesdent, J. Morio, and J. Marzat, *Kriging-based adaptive importance sampling algorithms for rare event estimation*, Structural Safety, 44 (2013), pp. 1 – 10, https://doi.org/https://doi.org/10.1016/j.strusafe.2013.04.001, http://www.sciencedirect.com/science/article/pii/S0167473013000350.
[6] J. Bect, L. Li, and E. Vazquez, *Bayesian subset simulation*, SIAM/ASA Journal on Uncertainty Quantification, 5 (2017), pp. 762–786, https://doi.org/10.1137/16M1078276, https://doi.org/10.1137/16M1078276, https://arxiv.org/abs/https://doi.org/10.1137/16M1078276.
[7] M. Berveiller, B. Sudret, and M. Lemaire, *Stochastic finite element: a non intrusive approach by regression*, European Journal of Computational Mechanics, 15 (2006), pp. 81–92.
[8] B. J. Bichon, M. S. Eldred, L. P. Swiler, S. Mahadevan, and J. M. McFarland, *Efficient global reliability analysis for nonlinear implicit performance functions*, AIAA Journal, 46 (2008), pp. 2459–2468, https://doi.org/10.2514/1.34321, https://doi.org/10.2514/1.34321, https://arxiv.org/abs/https://doi.org/10.2514/1.34321.
[9] G. Blatman and B. Sudret, *Sparse polynomial chaos expansions and adaptive stochastic finite elements using a regression approach*, Comptes Rendus Mécanique, 336 (2008), pp. 518–523, https://doi.org/https://doi.org/10.1016/j.crme.2008.02.013, https://www.sciencedirect.com/science/article/pii/S1631072108000582.
[10] G. Blatman and B. Sudret, *Adaptive sparse polynomial chaos expansion based on least-angle regression*, Journal of Computational Physics, 230 (2011), pp. 2345 – 2367, https://doi.org/https://doi.org/10.1016/j.jcp.2010.12.021.
[11] J.-M. Bourinet, *Rare-event probability estimation with adaptive support vector regression surrogates*, Reliability Engineering & System Safety, 150 (2016), pp. 210 – 221, https://doi.org/https://doi.org/10.1016/j.ress.2016.01.023, http://www.sciencedirect.com/science/article/pii/S0951832016000387.
[12] J.-M. Bourinet, F. Deheeger, and M. Lemaire, *Assessing small failure probabilities by combined subset simulation and support vector machines*, Structural Safety, 33 (2011), pp. 343 – 353, https://doi.org/http://dx.doi.org/10.1016/j.strusafe.2011.06.001, http://www.sciencedirect.com/science/article/pii/S0167473011000555.
[13] C. G. Bucher, *Adaptive sampling — an iterative fast Monte Carlo procedure*, Structural Safety, 5 (1988), pp. 119 – 126.
[14] F. Cadini, F. Santos, and E. Zio, *An improved adaptive kriging-based importance technique for sampling multiple failure regions of low probability*, Reliability Engineering & System Safety, 131 (2014), pp. 109 – 117, https://doi.org/https://doi.org/10.1016/j.ress.2014.06.023, http://www.sciencedirect.com/science/article/pii/S0951832014001537.
[15] O. Chapelle, V. Vapnik, and Y. Bengio, *Model selection for small sample regression*, Machine Learning, 48 (2002), pp. 9–23.
[16] P. G. Constantine, E. Dow, and Q. Wang, *Active subspace methods in theory and practice: Applications to Kriging surfaces*, SIAM Journal on Scientific Computing, 36 (2014), pp. A1500–A1524, https://doi.org/10.1137/130916138.
[17] S. De Jong, *Pls shrinks*, Journal of Chemometrics, 9 (1995), pp. 323–326, https://doi.org/https://doi.org/10.1002/cem.1180090406.
[18] A. Der Kiureghian, *First-and second-order reliability methods*, in Engineering Design Reliability Handbook, E. Nikolaidis, D. M. Ghiocel, and S. Singhal, eds., CRC Press, Boca Raton, FL, 2005, ch. 14.
[19] O. Ditlevsen and H. O. Madsen, *Structural reliability methods*, John Wiley & Sons Ltd, 1996.
[20] A. Doostan and H. Owhadi, *A non-adapted sparse approximation of PDEs with stochastic inputs*, Journal of Computational Physics, 230 (2011), pp. 3015–3034.
[21] V. Dubourg, B. Sudret, and F. Deheeger, *Metamodel-based importance sampling for structural reliability analysis*, Probabilistic Engineering Mechanics, 33 (2013), pp. 47 – 57, https://doi.org/http://dx.doi.org/10.1016/j.probengmech.2013.02.002, http://www.sciencedirect.com/science/article/pii/S0266892013000222.
[22] B. Echard, N. Gayton, and M. Lemaire, *AK-MCS: An active learning reliability method combining Kriging and Monte Carlo simulation*, Structural Safety, 33 (2011), pp. 145 – 154, https://doi.org/http://dx.doi.org/10.1016/j.strusafe.2011.01.002, http://www.sciencedirect.com/science/article/pii/S0167473011000038.

[23] M. Ehre, I. Papaioannou, and D. Straub, *Global sensitivity analysis in high dimensions with PLS-PCE*, Reliability Engineering & System Safety, 198 (2020), p. 106861, https://doi.org/https://doi.org/10.1016/j.ress.2020.106861, http://www.sciencedirect.com/science/article/pii/S0951832019314140.

[24] S. Engelund and R. Rackwitz, *A benchmark study on importance sampling techniques in structural reliability*, Structural Safety, 12 (1993), pp. 255 – 276.

[25] L. Faravelli, *Response surface approach for reliability analysis*, Journal of Engineering Mechanics, 115 (1989), pp. 2763–2781, https://doi.org/10.1061/(ASCE)0733-9399(1989)115:12(2763), http://ascelibrary.org/doi/abs/10.1061/%28ASCE%290733-9399%281989%29115%3A12%282763%29.

[26] V. Fedorov, *Theory of Optimal Experiments*, Probability and Mathematical Statistics, Academic Press, 01 1972, https://books.google.de/books?id=PpUIwgEACAAJ.

[27] B. Fiessler, R. Rackwitz, and H.-J. Neumann, *Quadratic limit states in structural reliability*, Journal of the Engineering Mechanics Division, 105 (1979), pp. 661–676.

[28] X. Guan and R. Melchers, *Effect of response surface parameter variation on structural reliability estimates*, Structural Safety, 23 (2001), pp. 429 – 444, https://doi.org/http://dx.doi.org/10.1016/S0167-4730(02)00013-9, http://www.sciencedirect.com/science/article/pii/S0167473002000139.

[29] M. Hohenbichler and R. Rackwitz, *Non-normal dependent vectors in structural safety*, Journal of the Engineering Mechanics Division, 107 (1981), pp. 1227–1238.

[30] M. Hohenbichler and R. Rackwitz, *Improvement of second-order reliability estimates by importance sampling*, Journal of Engineering Mechanics, 114 (1988), pp. 2195–2199.

[31] A. Höskuldsson, *PLS regression methods*, Journal of Chemometrics, 2 (1988), pp. 211–228.

[32] X. Huang, J. Chen, and H. Zhu, *Assessing small failure probabilities by AK–SS: An active learning method combining Kriging and subset simulation*, Structural Safety, 59 (2016), pp. 86 – 95, https://doi.org/https://doi.org/10.1016/j.strusafe.2015.12.003, http://www.sciencedirect.com/science/article/pii/S0167473016000035.

[33] J. E. Hurtado, *Filtered importance sampling with support vector margin: a powerful method for structural reliability analysis*, Structural Safety, 29 (2007), pp. 2–15.

[34] J. E. Hurtado and D. A. Alvarez, *Neural-network-based reliability analysis: a comparative study*, Computer Methods in Applied Mechanics and Engineering, 191 (2001), pp. 113–132.

[35] S. Ji, Y. Xue, and L. Carin, *Bayesian compressive sensing*, IEEE Transactions on Signal Processing, 56 (2008), pp. 2346–2356.

[36] Z. Jiang and J. Li, *High dimensional structural reliability with dimension reduction*, Structural Safety, 69 (2017), pp. 35 – 46, https://doi.org/https://doi.org/10.1016/j.strusafe.2017.07.007, http://www.sciencedirect.com/science/article/pii/S0167473016302077.

[37] C. Johnson, *Numerical solution of partial differential equations by the finite element method*, Dover Publications, 2009.

[38] K. Konakli and B. Sudret, *Reliability analysis of high-dimensional models using low-rank tensor approximations*, Probabilistic Engineering Mechanics, 46 (2016), pp. 18 – 36.

[39] P. Koutsourelakis, H. Pradlwarter, and G. Schuëller, *Reliability of structures in high dimensions, part I: algorithms and applications*, Probabilistic Engineering Mechanics, 19 (2004), pp. 409 – 417.

[40] H. Kriegel, P. Kröger, and A. Zimek, *Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering*, ACM Trans. Knowl. Discov. Data, 3 (2009), pp. 1:1–1:58.

[41] D. P. Kroese, R. Y. Rubinstein, and P. W. Glynn, *Chapter 2 - The cross-entropy method for estimation*, in Handbook of Statistics: Machine Learning: Theory and Applications, vol. 31 of Handbook of Statistics, Elsevier, 2013, pp. 19 – 34.

[42] N. Kurtz and J. Song, *Cross-entropy-based adaptive importance sampling using Gaussian mixture*, Structural Safety, 42 (2013), pp. 35 – 44.

[43] S. H. Lee and B. M. Kwak, *Response surface augmented moment method for efficient reliability analysis*, Structural Safety, 28 (2006), pp. 261 – 272.

[44] J. Li, J. Li, and D. Xiu, *An efficient surrogate-based method for computing rare failure probability*, Journal of Computational Physics, 230 (2011), pp. 8683 – 8697, https://doi.org/http://dx.doi.org/10.1016/j.jcp.2011.08.008, http://www.sciencedirect.com/science/article/pii/S0021999111004803.

[45] J. Li and D. Xiu, *Evaluation of failure probability via surrogate models*, Journal of Computational Physics, 229 (2010), pp. 8966 – 8980, https://doi.org/http://dx.doi.org/10.1016/j.jcp.2010.08.022, http://www.sciencedirect.com/science/article/pii/S0021999110004699.

[46] M. Li and Z. Wang, *Deep learning for high-dimensional reliability analysis*, Mechanical Systems and Signal Processing, 139 (2020), p. 106399, https://doi.org/https://doi.org/10.1016/j.ymssp.2019.106399, http://www.sciencedirect.com/science/article/pii/S088832701930620X.

[47] R. Li and R. Ghanem, *Adaptive polynomial chaos expansions applied to statistics of extremes in nonlinear random vibration*, Probabilistic Engineering Mechanics, 13 (1998), pp. 125 – 136, https://doi.org/https://doi.org/10.1016/S0266-8920(97)00020-9.

[48] P.-L. Liu and A. Der Kiureghian, *Multivariate distribution models with prescribed marginals and covariances*, Probabilistic Engineering Mechanics, 1 (1986), pp. 105–112, https://doi.org/https://doi.org/10.1016/0266-8920(86)90033-0, https://www.sciencedirect.com/science/article/pii/0266892086900330.

[49] P.-L. Liu and K.-G. Liu, *Selection of random field mesh in finite element reliability analysis*, Journal of Engineering Mechanics, 119 (1993), pp. 667–680.

[50] N. Lüthen, S. Marelli, and B. Sudret, *A benchmark of basis-adaptive sparse polynomial chaos expansions for engineering regression problems*, 2021, https://arxiv.org/abs/2009.04800.

[51] N. Lüthen, S. Marelli, and B. Sudret, *Sparse polynomial chaos expansions: Literature survey and benchmark*, 2021, https://arxiv.org/abs/2002.01290.

[52] S. Marelli and B. Sudret, *An active-learning algorithm that combines sparse polynomial chaos expansions and bootstrap*

for structural reliability analysis, Structural Safety, 75 (2018), pp. 67 – 74.

[53] J. Oakley, *Estimating percentiles of uncertain computer code outputs*, Journal of the Royal Statistical Society: Series C (Applied Statistics), 53 (2004), pp. 83–93, https://doi.org/10.1046/j.0035-9254.2003.05044.x, https://rss.onlinelibrary.wiley.com/doi/abs/10.1046/j.0035-9254.2003.05044.x, https://arxiv.org/abs/https://rss.onlinelibrary.wiley.com/doi/pdf/10.1046/j.0035-9254.2003.05044.x.

[54] A. B. Owen, *Monte Carlo theory, methods and examples*, 2013.

[55] Q. Pan and D. Dias, *Sliced inverse regression-based sparse polynomial chaos expansions for reliability analysis in high dimensions*, Reliability Engineering & System Safety, 167 (2017), pp. 484 – 493, https://doi.org/https://doi.org/10.1016/j.ress.2017.06.026, http://www.sciencedirect.com/science/article/pii/S095183201630864X. Special Section: Applications of Probabilistic Graphical Models in Dependability, Diagnosis and Prognosis.

[56] V. Papadopoulos, D. G. Giovanis, N. D. Lagaros, and M. Papadrakakis, *Accelerated subset simulation with neural networks for reliability analysis*, Computer Methods in Applied Mechanics and Engineering, 223 (2012), pp. 70 – 80, https://doi.org/http://dx.doi.org/10.1016/j.cma.2012.02.013, http://www.sciencedirect.com/science/article/pii/S0045782512000552.

[57] M. Papadrakakis, V. Papadopoulos, and N. D. Lagaros, *Structural reliability analysis of elastic-plastic structures using neural networks and Monte Carlo simulation*, Computer Methods in Applied Mechanics and Engineering, 136 (1996), pp. 145–163.

[58] I. Papaioannou, M. Ehre, and D. Straub, *Efficient PCE representations for reliability analysis in high dimensions*, in Proceedings of the 19th working conference of the IFIP Working Group 7.5 on Reliability and Optimization of Structural Systems, J. Song, ed., ETH Zürich, 2018.

[59] I. Papaioannou, M. Ehre, and D. Straub, *PLS-based adaptation for efficient PCE representation in high dimensions*, Journal of Computational Physics, 387 (2019), pp. 186 – 204.

[60] I. Papaioannou, S. Geyer, and D. Straub, *Improved cross entropy-based importance sampling with a flexible mixture model*, Reliability Engineering & System Safety, 191 (2019), p. 106564.

[61] I. Papaioannou, C. Papadimitriou, and D. Straub, *Sequential importance sampling for structural reliability analysis*, Structural Safety, 62 (2016), pp. 66 – 75, https://doi.org/http://dx.doi.org/10.1016/j.strusafe.2016.06.002, http://www.sciencedirect.com/science/article/pii/S0167473016300169.

[62] Y. C. Pati, R. Rezaiifar, Y. C. P. R. Rezaiifar, and P. S. Krishnaprasad, *Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition*, in Proceedings of the 27th Annual Asilomar Conference on Signals, Systems, and Computers (1993), 1993, pp. 40–44.

[63] B. Peherstorfer, B. Kramer, and K. Willcox, *Multifidelity preconditioning of the cross-entropy method for rare event simulation and failure probability estimation*, SIAM/ASA Journal on Uncertainty Quantification, 6 (2018), pp. 737–761.

[64] G. Perrin, *Adaptive calibration of a computer code with time-series output*, Reliability Engineering & System Safety, 196 (2020), p. 106728, https://doi.org/https://doi.org/10.1016/j.ress.2019.106728, https://www.sciencedirect.com/science/article/pii/S0951832018311232.

[65] V. Picheny, D. Ginsbourger, O. Roustant, R. T. Haftka, and N.-H. Kim, *Adaptive Designs of Experiments for Accurate Approximation of a Target Region*, Journal of Mechanical Design, 132 (2010), https://doi.org/10.1115/1.4001873, https://doi.org/10.1115/1.4001873, https://arxiv.org/abs/https://asmedigitalcollection.asme.org/mechanicaldesign/article-pdf/132/7/071008/5807381/071008_1.pdf.

[66] R. Rackwitz and B. Fiessler, *Structural reliability under combined random load sequences*, Computers & Structures, 9 (1978), pp. 489 – 494, https://doi.org/http://dx.doi.org/10.1016/0045-7949(78)90046-9, http://www.sciencedirect.com/science/article/pii/0045794978900469.

[67] R. Rosipal, *Nonlinear partial least squares: An overview*, Chemoinformatics and Advanced Machine Learning Perspectives: Complex Computational Methods and Collaborative Techniques, (2010), https://doi.org/10.4018/978-1-61520-911-8.ch009.

[68] R. Y. Rubinstein and D. P. Kroese, *Simulation and the Monte Carlo Method*, Wiley Publishing, 3rd ed., 2017.

[69] K. Sargsyan, C. Safta, H. Najm, B. J. Debusschere, D. Ricciuto, and P. Thornton, *Dimensionality reduction for complex models via Bayesian compressive sensing*, International Journal for Uncertainty Quantification, 4 (2014), pp. 63–93.

[70] R. Schöbi, B. Sudret, and S. Marelli, *Rare event estimation using polynomial-chaos Kriging*, ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering, 3 (2017), p. D4016002, https://doi.org/10.1061/AJRUA6.0000870.

[71] L. Schueremans and D. V. Gemert, *Benefit of splines and neural networks in simulation based structural reliability analysis*, Structural Safety, 27 (2005), pp. 246 – 261, https://doi.org/http://dx.doi.org/10.1016/j.strusafe.2004.11.001, http://www.sciencedirect.com/science/article/pii/S0167473004000529.

[72] B. Settles, *Active learning literature survey*, Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[73] B. Sudret, G. Blatman, and M. Berveiller, *Response surfaces based on polynomial chaos expansions*, Construction reliability: Safety, Variability and Sustainability, (2013), pp. 147–167.

[74] R. Tipireddy and R. Ghanem, *Basis adaptation in homogeneous chaos spaces*, Journal of Computational Physics, 259 (2014), pp. 304 – 317, https://doi.org/https://doi.org/10.1016/j.jcp.2013.12.009.

[75] M. E. Tipping, *Sparse Bayesian learning and the relevance vector machine*, J. Mach. Learn. Res., 1 (2001), p. 211–244.

[76] J. A. Tropp and A. C. Gilbert, *Signal recovery from random measurements via orthogonal matching pursuit*, IEEE Trans. Inf. Theor., 53 (2007), p. 4655–4666.

[77] P. Tsilifis, X. Huan, C. Safta, K. Sargsyan, G. Lacaze, J. C. Oefelein, H. N. Najm, and R. G. Ghanem, *Compressive sensing adaptation for polynomial chaos expansions*, Journal of Computational Physics, 380 (2019), pp. 29 – 47.

[78] P. Tsilifis, I. Papaioannou, D. Straub, and F. Nobile, *Sparse polynomial chaos expansions using variational relevance*

vector machines, Journal of Computational Physics, 416 (2020), p. 109498, https://doi.org/https://doi.org/10.1016/j.jcp.2020.109498, http://www.sciencedirect.com/science/article/pii/S0021999120302722.

[79] E. Ullmann and I. Papaioannou, *Multilevel estimation of rare events*, SIAM/ASA Journal on Uncertainty Quantification, 3 (2015), pp. 922–953, https://doi.org/10.1137/140992953, https://doi.org/10.1137/140992953, https://arxiv.org/abs/https://doi.org/10.1137/140992953.

[80] F. Uribe, I. Papaioannou, Y. M. Marzouk, and D. Straub, *Cross-entropy-based importance sampling with failure-informed dimension reduction for rare event simulation*, SIAM/ASA Journal on Uncertainty Quantification, 9 (2021), pp. 818–847, https://doi.org/10.1137/20M1344585.

[81] P. Waarts, *Structural Reliability Using Finite Element Analysis: An Appraisal of DARS: Directional Adaptive Response Surface Sampling*, PhD thesis, 2000.

[82] F. Wagner, J. Latz, I. Papaioannou, and E. Ullmann, *Multilevel sequential importance sampling for rare event estimation*, SIAM Journal on Scientific Computing, 42 (2020), pp. A2062–A2087, https://doi.org/10.1137/19M1289601.

[83] Z. Wang and J. Song, *Cross-entropy-based adaptive importance sampling using von Mises-Fisher mixture for high dimensional reliability analysis*, Structural Safety, 59 (2016), pp. 42–52.

[84] S. Wold, N. Kettaneh-Wold, and B. Skagerberg, *Nonlinear PLS modeling*, Chemometrics and Intelligent Laboratory Systems, 7 (1989), pp. 53 – 65. Proceedings of the First Scandinavian Symposium on Chemometrics.

[85] S. Wold, A. Ruhe, H. Wold, and W. Dunn III, *The collinearity problem in linear regression. The partial least squares (PLS) approach to generalized inverses*, SIAM Journal on Scientific and Statistical Computing, 5 (1984), pp. 735–743.

[86] C.-F. Wu, *Asymptotic Theory of Nonlinear Least Squares Estimation*, The Annals of Statistics, 9 (1981), pp. 501 – 513, https://doi.org/10.1214/aos/1176345455.

[87] D. Xiu and G. E. Karniadakis, *The Wiener–Askey polynomial chaos for stochastic differential equations*, SIAM Journal on Scientific Computing, 24 (2002), pp. 619–644.

[88] L. Yan, L. Guo, and D. Xiu, *Stochastic collocation algorithms using l1-minimization*, International Journal for Uncertainty Quantification, 2 (2012), pp. 279–293.

[89] T. Zhou and Y. Peng, *Structural reliability analysis via dimension reduction, adaptive sampling, and Monte Carlo simulation*, Structural and Multidisciplinary Optimization, (2020), https://doi.org/10.1007/s00158-020-02633-0, https://doi.org/10.1007/s00158-020-02633-0.