Multi-point Bayesian active learning reliability analysis

Tong Zhou, Xujia Zhu, Tong Guo, You Dong, Michael Beer

Multi-point Bayesian active learning reliability analysis

Tong Zhou[a], Xujia Zhu[b], Tong Guo[c], You Dong[d,*], Michael Beer[e,f,g]

[a]*Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Hong Kong*
[b]*Laboratoire des signaux et systèmes, Université Paris-Saclay, CNRS, CentraleSupélec, Gif-sur-Yvette 91190, France*
[c]*School of Civil Engineering, Southeast University, Nanjing 210098, China*
[d]*Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hong Kong*
[e]*Institute for Risk and Reliability, Leibniz University Hannover, Hannover, 30167, Germany*
[f]*Institute for Risk and Reliability, University of Liverpool, Liverpool, L69 7ZF, United Kingdom*
[g]*International Joint Research Center for Resilient Infrastructure & International Joint Research Center for Engineering Reliability and Stochastic Mechanics, Tongji University, Shanghai, 200092, China*

Abstract

This manuscript presents a novel Bayesian active learning reliability method integrating both Bayesian failure probability estimation and Bayesian decision-theoretic multi-point enrichment process. First, an epistemic uncertainty measure called integrated margin probability (IMP) is proposed as an upper bound for the mean absolute deviation of failure probability estimated by Kriging. Then, adhering to the Bayesian decision theory, a look-ahead learning function called multi-point stepwise margin reduction (MSMR) is defined to quantify the possible reduction of IMP brought by adding a batch of new samples in expectation. The cost-effective implementation of MSMR-based multi-point enrichment process is conducted by three key workarounds: (a) Thanks to analytical tractability of the inner integral, the MSMR reduces to a single integral. (b) The remaining single integral in the MSMR is numerically computed with the rational truncation of the quadrature set. (c) A heuristic treatment of maximizing the MSMR is devised to fastly select a batch of best next points per iteration, where the prescribed or adaptive scheme is used to specify the batch size. The proposed method is tested on two benchmark examples and two dynamic reliability problems. The results indicate that the adaptive scheme in the MSMR gains a good balance between the computing resource consumption and the overall computational time. Then, the MSMR fairly outperforms those existing leaning functions and parallelization strategies in terms of the accuracy of failure probability estimate, the number of iterations, as well as the number of performance function evaluations, especially in complex dynamic reliability problems.

Keywords: Multi-point stepwise margin reduction, Bayesian active learning, Bayesian decision theory, Prescribed and adaptive schemes, Parallel computing, Reliability analysis

Nomenclature

AK-MCS	adaptive Kriging - Monte Carlo simulation	IPMR	integrated probability of misclassification reduction
ALR	active learning reliability	MCS	Monte Carlo simulation
BALR	Bayesian active learning reliability	MFWNP	modulated filtered white-noise process
CDF	cumulative distribution function	MSMR	multi-point stepwise margin reduction
CV	coefficient of variation	PABQ	parallel adaptive Bayesian quadrature
ED	experimental design	PDF	probability density function
EM	ensemble of metamodels	SMR	stepwise margin reduction
EMVR	expected margin volume reduction	SUR	stepwise uncertainty reduction
IMP	integrated margin probability	VAIS	variance-amplified importance sampling

1. Introduction

In probabilistic reliability analysis, a primary task is to compute the failure probability of an engineering system in the presence of various uncertainties associated with physical properties, environmental loads, and operating conditions, etc. This uncertainty can be modeled by a set of d random variables $\boldsymbol{X} = \{X_1, \ldots, X_d\} \in \mathbb{X} \subset \mathbb{R}^d$, with the joint probability density function (PDF) $f_{\boldsymbol{X}}(\boldsymbol{x})$. Without loss of generality, the $\boldsymbol{X}$ is

*Corresponding authors.
Email addresses: tong.ce.zhou@outlook.com (Tong Zhou), xujia.zhu@l2s.centralesupelec.fr (Xujia Zhu), guotong@seu.edu.cn (Tong Guo), you.dong@polyu.edu.hk (You Dong), beer@irz.uni-hannover.de (Michael Beer)

assumed to be a standard Gaussian vector. If other probabilistic distributions are considered, an iso-probabilistic transformation that maps the original space to the standard Gaussian space can be employed, e.g., Nataf or Rosenblatt transform. The state of the system can be defined by a so-called performance function $y = \mathcal{G}(\boldsymbol{x}) : \mathbb{R}^d \mapsto \mathbb{R}$, where the system is in a failed state when $\mathcal{G}(\boldsymbol{x}) \leq 0$. Then, the failure probability P_f can be defined as [1]

$$P_f = \mathbb{P}(\mathcal{G}(\boldsymbol{x}) \leq 0) = \int_{\mathbb{X}} \mathbb{1}_{\mathcal{G}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{1}$$

where $\mathbb{P}(\cdot)$ denotes the probability measure; the failure indicator function $\mathbb{1}_{\mathcal{G}}(\boldsymbol{x})$ is expressed as

$$\mathbb{1}_{\mathcal{G}}(\boldsymbol{x}) = \begin{cases} 1, & \mathcal{G}(\boldsymbol{x}) \leq 0, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

The analytical solution to Eq. (1) is practically intractable in most real-world scenarios due to the complex and implicit expression of $\mathcal{G}(\cdot)$. In recent decades, a variety of numerical reliability methods have been developed and can be generally divided into the following four categories. (a) *Stochastic simulation methods*, such as Monte Carlo simulation (MCS) [2], importance sampling [3], and subset simulation [4]; (b) *Analytical approximation methods*, e.g., first- and second-order reliability methods [5, 6]; (c) *Numerical integration-based methods*, such as moment methods [7, 8], and probability density evolution method [9, 10]; (d) *Surrogate model-assisted methods*, which consist in substituting the expensive-to-run performance function by a cheap-but-accurate surrogate model, and the failure probability can be then estimated by performing some classical reliability algorithms on the well-trained surrogate model. Commonly-used surrogate models in this context include polynomial chaos expansions [11, 12], Kriging [13], radial basis function [14], support vector regression [15], and ensemble of metamodels (EM) [16], to name just a few. Particularly, the active learning reliability (ALR) methods [17, 18], where surrogate models are used with an adaptive experimental design scheme, have received increasing attention during the last two decades. Specifically, guided by a well-defined learning function, a surrogate model can be refined by sequentially adding new performance function evaluations to its experimental design (ED); this sequential process is continued until it meets an appropriate convergence criterion related to the accuracy of failure probability estimate. Obviously, the main modules in the ALR framework comprise a surrogate model, a reliability estimation algorithm, a learning function, and a convergence criterion [18]. Among those existing surrogate models, Kriging is particularly prevalent, owing to its exact interpolation property and Bayesian interpretation ability [19]. Two pioneering contributions in this context are the efficient global reliability analysis [20] and the adaptive Kriging - Monte Carlo simulation (AK-MCS) [13]. The interested readers are referred to [17, 18] for a comprehensive review.

More recently, the Bayesian active learning reliability (BALR) methods [21], serving as one subcategory of the ALR framework, have surged in the literature. These methods consist in treating the Kriging-based failure probability estimation as a Bayesian inference problem, giving rise to the posterior statistics of failure probability, e.g., posterior mean and variance [21]. The posterior mean is a favorable estimate of failure probability, while the posterior variance can be viewed as an epistemic uncertainty measure of failure probability, due to only a limited number of performance function evaluations in the ED. Naturally, the posterior variance can be reduced by adding new training samples, introducing a framework to build learning functions. Unfortunately, the exact posterior variance is expressed as a double integral, whose numerical solution suffers from an intensive computational burden. In this way, several computationally cheap alternatives that only involve a single integral have been well developed, such as the upper bounding- [22], partial- [23], or semi- [24] posterior variance of failure probability. Then, learning functions are simply defined as the integrand of those epistemic uncertainty measures, whereby the best next point is selected as the one achieving the maximum learning function value. Typical learning functions include the upper-bound posterior variance contribution [22], the expected misclassification probability contribution [21, 24], and the left- or right-shifted contribution [23], etc.

Unlike those aforementioned learning functions, a family of look-ahead learning functions has recently emerged from the Bayesian decision-theoretic framework [25, 26]. They feature the ability to explicitly quantify the potential impact of adding a new point(s) on reducing those well-defined epistemic uncertainty measures. Different epistemic uncertainty measures give rise to distinct expressions of look-ahead learning functions. Typical ones comprise the stepwise uncertainty reduction (SUR) [25], the expected margin volume reduction (EMVR) [27], the expected integrated error reduction [28], the integrated probability of misclassification reduction (IPMR) [19], the SUR-Bichon criterion [29], and the stepwise margin reduction [30], etc. Note that a single-point enrichment process is often used with those learning functions, rendering the overall computational time of reliability analysis still very intensive.

To alleviate this drawback, the parallel (B)ALR methods have gained increasing popularity in the structural reliability community due to its potential to accelerate the overall computation and make the most of available computing facilities [31, 32]. This goal is often achieved by combining a single-point learning function with some additional multi-point enrichment strategies, whereby a batch of best next points can be added per iteration. Those existing multi-point enrichment strategies can be broadly categorized into the following four groups. (a)

The clustering strategy. This category is the most widely-used one in the literature, consisting of the K-means clustering techniques [33, 34, 35, 22] and the K-medoids clustering techniques [36, 37]. Basically, the centroids of K clusters of the candidate pool (weighted by the learning function values) are selected as the batch of K best next points per iteration. However, it remains challenging to specify a rational value of K. (b) *Pseudo learning function-based strategy* [38]. After selecting the 1-st new point via the initial learning function, the next new points are sequentially selected by the pseudo learning function. It approximates the real updated learning function via multiplying the initial one by an influence function of newly added points [24]. (c) *Kriging believer or constant liar strategy* [39]. Once the 1-st new point is selected via the initial learning function, the Kriging is updated by the 1-st new point and the predictive mean of Kriging on it [34, 40] or a constant fixed by the user [41]; then, the learning function is recomputed to select the 2-nd new point. This updating process is repeated K times to add a batch of K new points per iteration. (d) *EM-based strategy* [39]. The EM is a weighted average model that combines K diverse metamodels (or surrogate models). Then, every individual metamodel, along with the same learning function, could provide one new point, giving rise to a batch of K new points, at most, per iteration (removing duplicates). Obviously, the batch size per iteration is limited by the number of individual metamodels in the EM [16]. Although those existing (B)ALR methods assisted by various additional parallel enrichment strategies yield favorable empirical results, a theoretically sound and numerically simple approach is still desirable for practitioners.

In this study, we develop a new parallel Bayesian active learning reliability method by proposing a cost-effective look-ahead learning function called multi-point stepwise margin reduction (MSMR). Particularly, the multi-point enrichment process can be conducted based on the MSMR itself, without needing additional batch enrichment strategies. Besides, the batch size can be either prescribed or adaptively identified per iteration. The critical contributions of this study are three-fold.

- In the Bayesian inference of failure probability, the integrated margin probability (IMP) is proved as an upper bound of the mean absolute deviation of failure probability. In comparison with the exact posterior variance, the IMP is a much computationally cheaper measure of epistemic uncertainty of failure probability.
- The learning function MSMR is defined to quantify the expected gain on reducing the IMP induced by adding a batch of new samples, and it is formulated as a double integral. Analytical expression of the inner integral is obtainable, owing to the desirable form of IMP; then, the remaining outer integral is numerically computed by truncating the quadrature set.
- The multi-point enrichment process is cheaply conducted via the heuristic treatment of maximizing the MSMR, together with the truncation of candidate pool. Furthermore, both prescribed and adaptive schemes are devised to specify the size of batch of new samples added per iteration.

The remainder of this paper is organized as follows. Section 2 reviews the BALR methods. Section 3 is devoted to the proposed learning function MSMR. Then, multi-point Bayesian active learning reliability analysis based on the MSMR is elucidated in Section 4. The proposed approach is testified in Section 5. Finally, some concluding remarks are given in Section 6.

2. Preliminaries

This section provides an overview of the BALR methods. Section 2.1 reviews Bayesian inference of failure probability. Then, Section 2.2 discusses the existing learning functions and parallelization implementations, as well as their shortcomings.

2.1. Bayesian inference of failure probability

The BALR framework starts by considering the performance function $\mathcal{G}(\boldsymbol{x})$ as a sample path of Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x}) \sim \mathcal{GP}\left(\mu_n(\boldsymbol{x}), c_n(\boldsymbol{x}, \boldsymbol{x}')\right)$ in a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ (Appendix A). Then, the posterior distribution of the failure indicator function $\widehat{\mathbb{1}}_n(\cdot)$ and further the posterior statistics of the failure probability $\widehat{P}_{f,n}$ can be readily derived. In this way, both learning function and convergence criterion in the workflow of ALR can be built accordingly, as illustrated in Fig. 1.

Specifically, suppose that an ED $\mathcal{D}_n = \{\mathcal{X}_n, \mathcal{Y}_n\}$ of size n is provided, a Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ can be trained, with the posterior mean $\mu_n(\boldsymbol{x})$, variance $\sigma_n^2(\boldsymbol{x})$, and covariance $c_n(\boldsymbol{x}, \boldsymbol{x}')$ given by Eqs. (A.6), (A.7) and (A.8), respectively. The subscript n is a reminder of conditioning on $\mathcal{D}_n$. Then, by substituting $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ into Eq. (2), the failure indicator function $\widehat{\mathbb{1}}_n(\boldsymbol{x})$ is expressed as [21]

$$\widehat{\mathbb{1}}_n(\boldsymbol{x}) \sim \mathcal{GBP}\left(\mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x}), c_{\widehat{\mathbb{1}}_n}(\boldsymbol{x}, \boldsymbol{x}')\right), \tag{3}$$

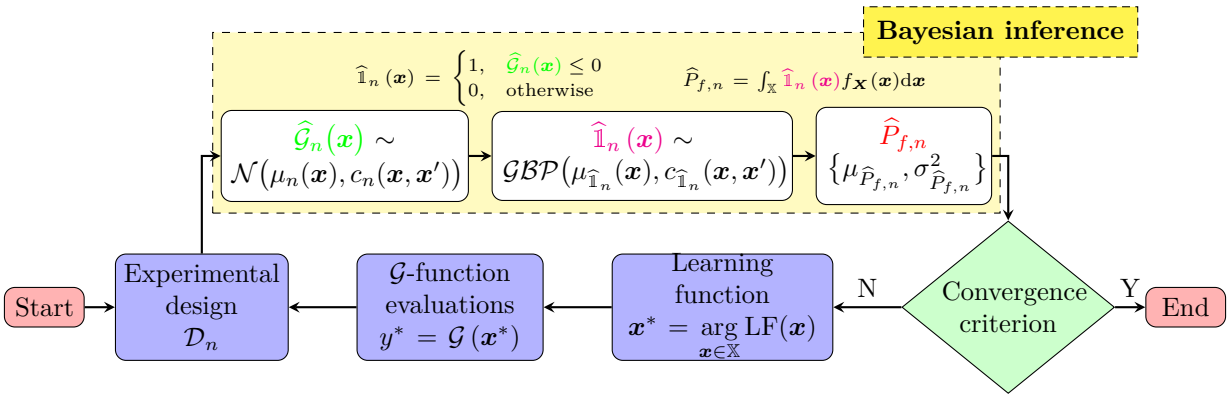


Figure 1: Basic workflow of Bayesian active learning reliability analysis

which follows a generalized Bernoulli process, with its mean $\mu_{\widehat{\mathbb{I}}_n}(\boldsymbol{x})$, variance $\sigma^2_{\widehat{\mathbb{I}}_n}(\boldsymbol{x})$, and covariance $c_{\widehat{\mathbb{I}}_n}(\boldsymbol{x}, \boldsymbol{x}')$, respectively, expressed as

$$\mu_{\widehat{\mathbb{I}}_n}(\boldsymbol{x}) = \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right), \tag{4}$$

$$\sigma^2_{\widehat{\mathbb{I}}_n}(\boldsymbol{x}) = \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right), \tag{5}$$

$$c_{\widehat{\mathbb{I}}_n}(\boldsymbol{x}, \boldsymbol{x}') = \Phi_2\left(\begin{bmatrix}0\\0\end{bmatrix}; \begin{bmatrix}\mu_n(\boldsymbol{x})\\ \mu_n(\boldsymbol{x}')\end{bmatrix}, \begin{bmatrix}\sigma_n^2(\boldsymbol{x}), & c_n(\boldsymbol{x}, \boldsymbol{x}')\\ c_n(\boldsymbol{x}', \boldsymbol{x}), & \sigma_n^2(\boldsymbol{x}')\end{bmatrix}\right) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\Phi\left(-\frac{\mu_n(\boldsymbol{x}')}{\sigma_n(\boldsymbol{x}')}\right), \tag{6}$$

where $\Phi(\cdot)$ is the cumulative distribution function (CDF) of a standard Gaussian random variable; $\Phi_2(\cdot; \boldsymbol{\mu}, \boldsymbol{C})$ is the joint CDF of a bivariate Gaussian random vector with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{C}$. Note that $\Phi_2(\cdot; \cdot, \cdot)$ has no closed-form expression and thus needs to be numerically computed.

Finally, substituting Eq. (3) into Eq. (1), the failure probability $\widehat{P}_{f,n}$ conditional on $\mathcal{D}_n$ is expressed as

$$\widehat{P}_{f,n} = \int_{\mathbb{X}} \widehat{\mathbb{I}}_n(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}. \tag{7}$$

Although the exact distribution of $\widehat{P}_{f,n}$ is intractable, its mean $\mu_{\widehat{P}_{f,n}}$ and variance $\sigma^2_{\widehat{P}_{f,n}}$ are given as [21]

$$\mu_{\widehat{P}_{f,n}} = \int_{\mathbb{X}} \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{8}$$

$$\sigma^2_{\widehat{P}_{f,n}} = \int_{\mathbb{X}}\int_{\mathbb{X}} \Phi_2\left(\begin{bmatrix}0\\0\end{bmatrix}; \begin{bmatrix}\mu_n(\boldsymbol{x})\\ \mu_n(\boldsymbol{x}')\end{bmatrix}, \begin{bmatrix}\sigma_n^2(\boldsymbol{x}), & c_n(\boldsymbol{x}, \boldsymbol{x}')\\ c_n(\boldsymbol{x}', \boldsymbol{x}), & \sigma_n^2(\boldsymbol{x}')\end{bmatrix}\right) f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}') \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}' - \mu^2_{\widehat{P}_{f,n}}, \tag{9}$$

respectively, which encapsulate our probabilistic belief regarding the P_f in light of the current ED $\mathcal{D}_n$.

Eqs. (8) and (9) generally have no analytical solution and thus need to be computed with some numerical integration methods. For example, the variance-amplified importance sampling (VAIS) expresses Eq. (8) as

$$\mu_{\widehat{P}_{f,n}} = \int_{\mathbb{X}} \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) \frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})} h_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \approx \frac{1}{Q_1}\sum_{q=1}^{Q_1}\left[\Phi\left(-\frac{\mu_n\left(\boldsymbol{x}^{(q)}\right)}{\sigma_n\left(\boldsymbol{x}^{(q)}\right)}\right)\frac{f_{\boldsymbol{X}}\left(\boldsymbol{x}^{(q)}\right)}{h_{\boldsymbol{X}}\left(\boldsymbol{x}^{(q)}\right)}\right] = \tilde{\mu}_{\widehat{P}_{f,n}}, \tag{10}$$

where $h_{\boldsymbol{X}}(\boldsymbol{x})$ denotes the importance sampling density function, and it usually takes the joint PDF of a d-variate Gaussian vector with mean vector $\boldsymbol{0}$ and covariance matrix $\alpha^2 \boldsymbol{I}_d$. The amplification coefficient α is set as 1.5 in this paper [21, 19]. $\mathcal{X}_{\mathrm{Q}_1} = \{\boldsymbol{x}^{(q)}, q = 1, \ldots, Q_1\}$ denotes a set of Q_1 samples drawn from $h_{\boldsymbol{X}}(\boldsymbol{x})$, and the Halton low-discrepancy sequence is used here. Further, the sample variance of $\tilde{\mu}_{\widehat{P}_{f,n}}$ is computed as $\mathrm{Var}\left[\tilde{\mu}_{\widehat{P}_{f,n}}\right] = \frac{1}{Q_1(Q_1-1)}\sum_{q=1}^{Q_1}\left[\Phi\left(-\frac{\mu_n(\boldsymbol{x}^{(q)})}{\sigma_n(\boldsymbol{x}^{(q)})}\right)\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})} - \tilde{\mu}_{\widehat{P}_{f,n}}\right]^2$, and the associated coefficient of variation (CV) is expressed as $\mathrm{CV}\left[\tilde{\mu}_{\widehat{P}_{f,n}}\right] = \sqrt{\mathrm{Var}\left[\tilde{\mu}_{\widehat{P}_{f,n}}\right]} \Big/ \tilde{\mu}_{\widehat{P}_{f,n}}$. Then, the sample size Q_1 in Eq. (10) is considered sufficient when $\mathrm{CV}\left[\tilde{\mu}_{\widehat{P}_{f,n}}\right] \le \varepsilon_{\mathrm{Q}}$, where the tolerance ε_{Q} can be specified as 5% here.

The $\mu_{\widehat{P}_{f,n}}$ is often taken as a favorable estimate of $\widehat{P}_{f,n}$, while the $\sigma^2_{\widehat{P}_{f,n}}$ serves as a rational measure of epistemic uncertainty of $\widehat{P}_{f,n}$, due to only a limited number of $\mathcal{G}$-function evaluations in the $\mathcal{D}_n$. Then, in the BALR (Fig. 1), the convergence criterion can be specified such that the $\sigma^2_{\widehat{P}_{f,n}}$ falls below a prescribed tolerance. To secure the precision of $\mu_{\widehat{P}_{f,n}}$, the $\sigma^2_{\widehat{P}_{f,n}}$ shall be reduced as much as possible by adding new samples. Hence,

the learning function can be designed to suggest where to make the next $\mathcal{G}$-function evaluation(s) to reduce the $\sigma^2_{\widehat{P}_{f,n}}$ at most. Obviously, both Bayesian inference of failure probability and the ingredients of the ALR are fairly fused in the BALR framework.

2.2. Existing learning functions, parallelization strategies, and the associated problems

Eq. (9) indicates that the $\sigma^2_{\widehat{P}_{f,n}}$ is computationally intensive due to the non-analytical form of $\Phi_2(\cdot;\cdot,\cdot)$. To alleviate this hurdle, some computationally cheap alternatives have been proposed in the literature. Then, the resulting learning functions and the associated parallelization strategies are listed in Table 1. However, those existing practices suffer from three potential bottlenecks as follows.

- The existing learning functions are mainly defined as the integrand of certain epistemic uncertainty measures, e.g., the PABQ, the PBALC3, and the SBALQ in Table 1. They design a single-point learning function based on a plain belief that the point with the maximum integrand value contributes the most to reducing the associated epistemic uncertainty measure. Essentially, this practice only considers the individual effect of adding a new sample itself, without explicitly accounting for its impact on any other point around it and further on reducing the epistemic uncertainty measure [26]. Hence, it is very challenging for those single-point learning functions to accurately quantify the expected gain of adding a new point on the reduction of the associated epistemic uncertainty measures.
- It is a common practice to combine a single-point learning function with some additional multi-point enrichment strategies to favor parallel computing. As the single-point learning function based on the integrand of the uncertainty measure serves as a basis, this practice still suffers from the first drawback. Moreover, this combination practice makes the analysis of reducing the overall uncertainty measure more complex. For example, although the clustering strategy used in the PABQ [22] could avoid the overlap between a batch of new samples, it is difficult to justify their optimality of reducing the associated epistemic uncertainty measures.
- The existing look-ahead learning functions could quantify the influence of adding a new point(s) on reducing certain epistemic uncertainty measures and are generally expressed as a single integral, such as the EMVR [27], the IPMR [19], and the SUR [42] in Table 1. However, they are faced with the lack of computationally cheap multi-point analytical expressions. For example, both the EMVR and the IPMR have difficulty in obtaining multi-point analytical expressions. In contrast, the SUR has the multi-point analytical expression but is a function of $\Phi_2(\cdot;\cdot,\cdot)$; see Eq. (F.2). Due to the necessity of repeatedly evaluating $\Phi_2(\cdot;\cdot,\cdot)$ in an element-wise manner, maximizing the SUR is very time-consuming.

Table 1: Existing epistemic uncertainty measures, learning functions, and parallelization strategies

Methods	Epistemic uncertainty metrics	Learning functions	Parallelization
PA-BFPL [21]	$\sigma^2_{\widehat{P}_{f,n}}$	$\Phi\left(-\frac{\lvert\mu_n(\boldsymbol{x})\rvert}{\sigma_n(\boldsymbol{x})}\right)$	Clustering
PABQ [22]	$\mathbb{E}_{\boldsymbol{X}}\left[\sigma_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right]$	$\sigma_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})$	Clustering
PBALC [23]	$\mathbb{E}_{\boldsymbol{X}}\left[\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}+b\right)-\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right]$	$\left[\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}+b\right)-\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right]$	-
SBALQ [24]	$\mathbb{E}_{\boldsymbol{X}}\left[\Phi\left(-\frac{\lvert\mu_n(\boldsymbol{x})\rvert}{\sigma_n(\boldsymbol{x})}\right)\right]$	$\Phi\left(-\frac{\lvert\mu_n(\boldsymbol{x})\rvert}{\sigma_n(\boldsymbol{x})}\right)$	Pseudo function
EMVR [27]	$\mathbb{E}_{\boldsymbol{X}}\left[\mathbb{1}[-\beta\sigma_n(\boldsymbol{x})\leq\mu_n(\boldsymbol{x})\leq\beta\sigma_n(\boldsymbol{x})]\right]$	$\mathbb{E}_{\boldsymbol{X}}\left[\begin{array}{l}\Phi\left(\frac{\sigma_n(\boldsymbol{x}_+)}{\lvert c_n(\boldsymbol{x},\boldsymbol{x}_+)\rvert}\left(\mu_n(\boldsymbol{x})+\beta\sigma_{n+1}(\boldsymbol{x})\right)\right)\\ -\Phi\left(\frac{\sigma_n(\boldsymbol{x}_+)}{\lvert c_n(\boldsymbol{x},\boldsymbol{x}_+)\rvert}\left(\mu_n(\boldsymbol{x})-\beta\sigma_{n+1}(\boldsymbol{x})\right)\right)\end{array}\right]$	-
IPMR [19]	$\mathbb{E}_{\boldsymbol{X}}\left[\Phi\left(-\frac{\lvert\mu_n(\boldsymbol{x})\rvert}{\sigma_n(\boldsymbol{x})}\right)\right]$	$\mathbb{E}_{\boldsymbol{X}}\left[\Phi\left(\frac{-\frac{\lvert\mu_n(\boldsymbol{x})\rvert}{\sigma_n(\boldsymbol{x})}}{\lvert\rho_n(\boldsymbol{x},\boldsymbol{x}_+)\rvert}\right)\right]$	-
SUR [42]	$\mathbb{E}_{\boldsymbol{X}}\left[\sigma^2_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right]$	Eq. (F.2)	Eq. (F.3)

To address the drawback of those existing methods, we will develop a theoretically sound and numerically simple multi-point learning function in Section 3.

3. The proposed multi-point look-ahead learning function

In the BALR, the sequential experimental design process essentially corresponds to sequential decision-making under uncertainty, which can be addressed from the Bayesian decision-theoretic perspective [43]. Section 3.1 reviews Bayesian decision theory in the context of reliability analysis, serving as a theoretical underpinning of the proposed learning function. Then, Section 3.2 presents the associated loss function considered in this

paper. Section 3.3 presents the basic definition of the resulting learning function, called multi-point stepwise margin reduction (MSMR). The computational challenges of the MSMR-based multi-point enrichment process are discussed in Sections 3.4 and 3.5. Finally, the associated workarounds are elucidated in Sections 3.6, 3.7, and 3.8.

3.1. A short reminder of Bayesian decision theory for reliability analysis

As shown in Fig. 1, Bayesian inference of failure probability allows reasoning about the uncertainty in the failure probability, in light of the available ED, via its posterior statistics (Eqs. (8) and (9)). Then, the BALR entails a sequence of decisions of where to perform the next $\mathcal{G}$-function evaluation under our posterior belief. More precisely, the decision policy is defined as $\underline{X}_N : \mathcal{G} \mapsto \underline{X}_N(\mathcal{G}) = (X_1(\mathcal{G}), \ldots, X_N(\mathcal{G})) \in \mathbb{X}^N$, and the corresponding estimate $\mu_{\widehat{P}_{f,N}}$ is defined in Eq. (8), where N is the budget of $\mathcal{G}$-function evaluations.

From the Bayesian decision-theoretic perspective, the best decision policy is the one achieving or getting close to the Bayes risk defined as [44]

$$R_{\mathrm{B}} = \inf_{\boldsymbol{X}_N} \mathbb{E}\left[\mathcal{L}\big(\mu_{\widehat{P}_{f,N}}, \widehat{P}_{f,N}\big)\right], \tag{11}$$

where $\inf\{\cdot\}$ denotes the infimum of a set; $\mathcal{L}\big(\mu_{\widehat{P}_{f,N}}, \widehat{P}_{f,N}\big)$ is a loss function reflecting the error of the final failure probability estimate $\mu_{\widehat{P}_{f,N}}$; $\mathbb{E}[\cdot]$ denotes the expectation with respect to all the uncertainty in the sequence of decisions.

The optimal policy for Eq. (11) can be formally gained via the *dynamic programming*, which unfortunately suffers from the "curse of dimensionality" [44, 19]. A sub-optimal but effective way is to build a one-step look-ahead policy as follows:

$$\boldsymbol{x}^{(n+1)} = \arg\min_{\boldsymbol{x}\in\mathcal{X}} \mathbb{E}\left[\mathcal{L}\big(\mu_{\widehat{P}_f}, \widehat{P}_f\big) \mid \mathcal{X}_n, \mathcal{Y}_n, \boldsymbol{x}\right] = \arg\min_{\boldsymbol{x}\in\mathcal{X}} \mathbb{E}_n\left[\mathcal{L}\big(\mu_{\widehat{P}_{f,n+1}}, \widehat{P}_{f,n+1}\big) \mid \boldsymbol{X}^{(n+1)} = \boldsymbol{x}\right]. \tag{12}$$

which essentially treats each decision as a termination decision.

Obviously, the Bayesian risk involved in Eq. (12) can be seen as a global measure of uncertainty of $\widehat{P}_f$. In this paper, we select the absolute error of $\mu_{\widehat{P}_f}$, i.e., $\left|\widehat{P}_f - \mu_{\widehat{P}_f}\right|$, as the loss function, thereby the learning function is generally expressed as

$$J_{n+1}(\boldsymbol{x}) = \mathbb{E}_n\left[\left|\widehat{P}_{f,n+1} - \mu_{\widehat{P}_{f,n+1}}\right| \mid \boldsymbol{X}^{(n+1)} = \boldsymbol{x}\right], \tag{13}$$

and the best next point $\boldsymbol{x}^{(n+1)}$ is then the one minimizing the $J_{n+1}(\boldsymbol{x})$.

However, due to no closed-form expression of Eq. (13), we attempt to explore an upper-bounding metric of Eq. (13) in Section 3.2. On this basis, a multi-point learning function will be defined in Section 3.3.

3.2. An epistemic uncertainty measure of failure probability

First, the mean failure domain $\widehat{\mathbb{F}}_n$ can be defined as

$$\widehat{\mathbb{F}}_n = \{\boldsymbol{x} \in \mathbb{X} : \mu_n(\boldsymbol{x}) \leq 0\}. \tag{14}$$

Then, given the Gaussian assumption of Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$, we introduce two auxiliary regions

$$\widehat{\mathbb{F}}_n^- = \{\boldsymbol{x} \in \mathbb{X} : \mu_n(\boldsymbol{x}) \leq 0 - \beta\sigma_n(\boldsymbol{x})\}, \tag{15}$$

$$\widehat{\mathbb{F}}_n^+ = \{\boldsymbol{x} \in \mathbb{X} : \mu_n(\boldsymbol{x}) \leq 0 + \beta\sigma_n(\boldsymbol{x})\}, \tag{16}$$

and there holds $\widehat{\mathbb{F}}_n^- \in \widehat{\mathbb{F}}_n \in \widehat{\mathbb{F}}_n^+$. The coefficient $\beta(>0)$ represents the confidence level for limit state estimation.

In this way, the margin between $\widehat{\mathbb{F}}_n^-$ and $\widehat{\mathbb{F}}_n^+$, denoted as $\widehat{\mathbb{M}}_n$, is given by [45]

$$\widehat{\mathbb{M}}_n = \{\boldsymbol{x} \in \mathbb{X} : -\beta\sigma_n(\boldsymbol{x}) \leq \mu_n(\boldsymbol{x}) \leq \beta\sigma_n(\boldsymbol{x})\}, \tag{17}$$

and the margin probability $\pi_n(\boldsymbol{x})$ can be then calculated as [30]

$$\pi_n(\boldsymbol{x}) = \mathbb{P}\left(\widehat{\mathcal{G}}_n(\boldsymbol{x}) \in \widehat{\mathbb{M}}_n\right) = \mathbb{P}\left(\widehat{\mathcal{G}}_n(\boldsymbol{x}) \in \widehat{\mathbb{F}}_n^+\right) - \mathbb{P}\left(\widehat{\mathcal{G}}_n(\boldsymbol{x}) \in \widehat{\mathbb{F}}_n^-\right) = \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} + \beta\right) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} - \beta\right), \tag{18}$$

which stands for the probability that the point $\boldsymbol{x}$ falls into the limit-state margin $\widehat{\mathbb{M}}_n$.

Proposition 1. *Define the integrated margin probability (IMP) as*

$$H_n = \mathbb{E}_{\boldsymbol{X}}\left[\pi_n(\boldsymbol{x})\right] = \int_{\mathbb{X}} \pi_n(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{19}$$

where $\mathbb{E}_{\boldsymbol{X}}[\cdot]$ *is the expectation with respect to* $\boldsymbol{X}$*. When* $\beta \geq \Phi^{-1}(\frac{5}{8}) \approx 0.3186$*, we have*

$$\mathbb{E}_n\left[\left|\widehat{P}_{f,n} - \mu_{\widehat{P}_{f,n}}\right|\right] \leq 2H_n, \tag{20}$$

where $\mathbb{E}_n[\cdot]$ *is the expectation with respect to Kriging* $\widehat{\mathcal{G}}_n(\boldsymbol{x})$*.*

The proof of Proposition 1 is given in Appendix B. Obviously, when $\beta = \Phi^{-1}(\frac{5}{8})$ in $\pi_n(\boldsymbol{x})$, the $2H_n$ is the upper bound of the mean absolute deviation of $\widehat{P}_{f,n}$. Then, when $H_n \to 0$, the $\mu_{\widehat{P}_{f,n}}$ converges to the actual failure probability P_f in expectation. Therefore, the IMP H_n can be viewed as an epistemic uncertainty measure of $\widehat{P}_{f,n}$.

Similar to the VAIS-based estimate of $\mu_{\widehat{P}_{f,n}}$ in Eq. (10), the H_n can be estimated as

$$H_n \approx \tilde{H}_n = \frac{1}{Q_2}\sum_{q=1}^{Q_2}\left[\pi_n\left(\boldsymbol{x}^{(q)}\right) \cdot \frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}\right], \tag{21}$$

where $\mathcal{X}_{\mathrm{Q}_2} = \left\{\boldsymbol{x}^{(q)}, q = 1, \ldots, Q_2\right\}$ denotes a set of Q_2 samples drawn from $h_{\boldsymbol{X}}(\boldsymbol{x})$. Then, the sample CV $\mathrm{CV}\left[\tilde{H}_n\right]$ can be computed readily, and the sample size Q_2 is considered sufficient when $\mathrm{CV}\left[\tilde{H}_n\right] \leq \varepsilon_{\mathrm{Q}}$, as depicted in Section 2.1.

3.3. Basic expression of the learning function MSMR

According to the IMP H_n defined in Eqs. (19) and (20), the Bayesian decision-theoretic learning function in Eq. (13) can be reformulated as

$$J_{n+1}(\boldsymbol{x}) = \mathbb{E}_n\left[H_{n+1} \mid \boldsymbol{X}^{(n+1)} = \boldsymbol{x}\right], \tag{22}$$

where H_{n+1} denotes the future IMP due to adding $\boldsymbol{x}$ to the current ED $\mathcal{D}_n$. Further, a multi-point enrichment scenario is considered here, where the potential impact of adding a batch of new samples on the IMP needs to be explored.

Specfically, assume that a batch of k new points and their $\mathcal{G}$-function responses $\left\{\mathcal{X}_k^+, \mathcal{Y}_k^+\right\} = \left\{\left(\boldsymbol{x}_+^{(r)}, y_+^{(r)}\right)\right\}_{r=1}^{k}$ are added into the current ED $\mathcal{D}_n$, the look-ahead posterior of Kriging is provided by the Kriging update formulas (Appendix C), with its look-ahead mean $\mu_{n+k}(\boldsymbol{x})$, variance $\sigma_{n+k}^2(\boldsymbol{x})$, and covariance $c_{n+k}(\boldsymbol{x}, \boldsymbol{x}')$ given by Eqs. (C.1), (C.2) and (C.3), respectively.

Then, the future IMP H_{n+k} due to the addition of $\left\{\mathcal{X}_k^+, \mathcal{Y}_k^+\right\}$ can be expressed as

$$H_{n+k}\left(\mathcal{X}_k^+, \mathcal{Y}_k^+\right) \coloneqq \mathbb{E}_{\boldsymbol{X}}\left[\pi_{n+k}\left(\boldsymbol{x}; \mathcal{X}_k^+, \mathcal{Y}_k^+\right)\right], \tag{23}$$

which is a function of both $\mathcal{X}_k^+$ and $\mathcal{Y}_k^+$. The $\pi_{n+k}\left(\boldsymbol{x}; \mathcal{X}_k^+, \mathcal{Y}_k^+\right)$ is defined as

$$\begin{aligned}
\pi_{n+k}\left(\boldsymbol{x}; \mathcal{X}_k^+, \mathcal{Y}_k^+\right) &= \Phi\left(\frac{\beta\sigma_{n+k}(\boldsymbol{x}) - \mu_{n+k}(\boldsymbol{x})}{\sigma_{n+k}(\boldsymbol{x})}\right) - \Phi\left(\frac{-\beta\sigma_{n+k}(\boldsymbol{x}) - \mu_{n+k}(\boldsymbol{x})}{\sigma_{n+k}(\boldsymbol{x})}\right), \\
&= \Phi\left(\frac{\beta\sigma_{n+k}(\boldsymbol{x}) - \mu_n(\boldsymbol{x})}{\sigma_{n+k}(\boldsymbol{x})} + \frac{-c_n(\boldsymbol{x}, \mathcal{X}_k^+)^\top \left(\mathcal{C}_k^+\right)^{-1}}{\sigma_{n+k}(\boldsymbol{x})}\left(\mathcal{Y}_k^+ - \mu_n(\mathcal{X}_k^+)\right)\right) \\
&\quad - \Phi\left(\frac{-\beta\sigma_{n+k}(\boldsymbol{x}) - \mu_n(\boldsymbol{x})}{\sigma_{n+k}(\boldsymbol{x})} + \frac{-c_n(\boldsymbol{x}, \mathcal{X}_k^+)^\top \left(\mathcal{C}_k^+\right)^{-1}}{\sigma_{n+k}(\boldsymbol{x})}\left(\mathcal{Y}_k^+ - \mu_n(\mathcal{X}_k^+)\right)\right), \\
&= \Phi\left(a_1(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \mathcal{U}_k^+\right) - \Phi\left(a_2(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \mathcal{U}_k^+\right),
\end{aligned} \tag{24}$$

where $a_1(\boldsymbol{x}) = \dfrac{\beta\sigma_{n+k}(\boldsymbol{x}) - \mu_n(\boldsymbol{x})}{\sigma_{n+k}(\boldsymbol{x})}$, $a_2(\boldsymbol{x}) = \dfrac{-\beta\sigma_{n+k}(\boldsymbol{x}) - \mu_n(\boldsymbol{x})}{\sigma_{n+k}(\boldsymbol{x})}$, and $\boldsymbol{b}(\boldsymbol{x}) = \dfrac{-\left(\mathcal{C}_k^+\right)^{-1} c_n(\boldsymbol{x}, \mathcal{X}_k^+)}{\sigma_{n+k}(\boldsymbol{x})}$ are three quantities independent of $\mathcal{Y}_k^+$; then, $\mathcal{U}_k^+ = \mathcal{Y}_k^+ - \mu_n\left(\mathcal{X}_k^+\right)$ is a vector of size $k \times 1$ depending on $\mathcal{Y}_k^+$.

Unfortunately, since the responses $\mathcal{Y}_k^+$ are not exactly known without evaluating the actual $\mathcal{G}$-function on $\mathcal{X}_k^+$, the $\pi_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+, \mathcal{Y}_k^+)$ in Eq. (24) and the $H_{n+k}(\mathcal{X}_k^+, \mathcal{Y}_k^+)$ in Eq. (23) are unknown at all. To address this

bottleneck, the responses $\mathcal{Y}_k^+$ are replaced by the Kriging predictions evaluated on $\mathcal{X}_k^+$, i.e., $\boldsymbol{Y}_k^+ = \{Y_+^{(r)} = \widehat{\mathcal{G}}_n(\boldsymbol{x}_+^{(r)})\}_{r=1}^k \sim \mathcal{N}_k\left(\mu_n(\mathcal{X}_k^+), \mathcal{C}_k^+\right)$. Then, Eq. (23) is reformulated as

$$\mathcal{H}_{n+k}(\mathcal{X}_k^+) = \mathbb{E}_{\boldsymbol{X}}\left[\Pi_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\right], \tag{25}$$

which becomes a function of $\mathcal{X}_k^+$ solely. The integrand $\Pi_{n+k}\left(\boldsymbol{x}; \mathcal{X}_k^+\right)$ is expressed as

$$\Pi_{n+k}\left(\boldsymbol{x}; \mathcal{X}_k^+\right) = \Phi\left(a_1(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \boldsymbol{U}_k^+\right) - \Phi\left(a_2(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \boldsymbol{U}_k^+\right), \tag{26}$$

which is very similar to Eq. (24), but the involved term $\mathcal{U}_k^+$ is replaced by $\boldsymbol{U}_k^+$, with $\boldsymbol{U}_k^+ = \boldsymbol{Y}_k^+ - \mu_n(\mathcal{X}_k^+) \sim \mathcal{N}_k\left(\boldsymbol{0}, \mathcal{C}_k^+\right)$ following a k-variate centered Gaussian distribution.

Finally, the potential gain on reducing the IMP, due to adding $\mathcal{X}_k^+$, can be expressed as

$$\Delta\mathcal{H}_{n+k}\left(\mathcal{X}_k^+\right) = H_n - \mathcal{H}_{n+k}\left(\mathcal{X}_k^+\right). \tag{27}$$

Since $\mathcal{H}_{n+k}(\mathcal{X}_k^+)$ is a random quantity through $\boldsymbol{U}_k^+$, the gain $\Delta\mathcal{H}_{n+k}(\mathcal{X}_k^+)$ is also a random variable, failing to be a deterministic criterion.

Analogous to Eq. (22), the learning function is defined by taking the expectation of $\Delta\mathcal{H}_{n+k}(\mathcal{X}_k^+)$, that is,

$$\begin{aligned}
\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+) &= \mathbb{E}_{\boldsymbol{U}_k^+}\left[\Delta\mathcal{H}_{n+k}\left(\mathcal{X}_k^+\right)\right], \\
&= H_n - \mathbb{E}_{\boldsymbol{U}_k^+}\left[\mathbb{E}_{\boldsymbol{X}}\left[\Pi_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\right]\right], \\
&= \mathbb{E}_{\boldsymbol{X}}\left[\pi_n(\boldsymbol{x})\right] - \mathbb{E}_{\boldsymbol{X}}\left[\mathbb{E}_{\boldsymbol{U}_k^+}\left[\Pi_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\right]\right], \\
&= \mathbb{E}_{\boldsymbol{X}}\left[\pi_n(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{U}_k^+}\left[\Pi_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\right]\right], \\
&= \mathbb{E}_{\boldsymbol{X}}\left[I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\right],
\end{aligned} \tag{28}$$

where $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+) = \pi_n(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{U}_k^+}\left[\Pi_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\right]$.

In this way, a batch of k best next points at this iteration is selected as the candidate batch achieving the maximum expected gain, i.e.,

$$\mathcal{X}_k^* = \left\{\boldsymbol{x}^{(n+1)}, \ldots, \boldsymbol{x}^{(n+k)}\right\} = \underset{\mathcal{X}_k^+ \in \mathcal{X}_\mathrm{C}}{\arg\max}\, \mathrm{MSMR}_{n+k}\left(\mathcal{X}_k^+\right), \tag{29}$$

where $\mathcal{X}_\mathrm{C} = \left\{\boldsymbol{x}^{(i)}, i = 1, \ldots, C\right\}$ is a candidate pool of size C. Obviously, the $\mathcal{X}_k^*$ is exactly the one minimizing the future IMP. Hence, Eqs. (22) and (28) are self-consistent.

As more and more batches of new points are added by Eq. (29), the IMP H_n is expected to be reduced step by step. Hence, the learning function in Eq. (28) is called multi-point stepwise margin reduction (MSMR) here.

3.4. Inner integral in the MSMR

Eq. (28) shows that $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$ involves two nested integrals. Then, the inner integral of $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$ is analytically derived as follows.

Proposition 2. *In* $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$*, the inner integral* $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ *can be analytically expressed as*

$$I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+) = \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} + \beta\right) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} - \beta\right) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} + \beta\frac{\sigma_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)}{\sigma_n(\boldsymbol{x})}\right) + \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} - \beta\frac{\sigma_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)}{\sigma_n(\boldsymbol{x})}\right), \tag{30}$$

where $\sigma_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ *acts as a reminder of conditioning on* $\mathcal{X}_k^+$*. As per Eq. (C.2), the term* $\frac{\sigma_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)}{\sigma_n(\boldsymbol{x})}$ *can be further expressed as*

$$\frac{\sigma_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)}{\sigma_n(\boldsymbol{x})} = \sqrt{1 - \frac{c_n(\boldsymbol{x}, \mathcal{X}_k^+)^\top (\mathcal{C}_k^+)^{-1} c_n(\boldsymbol{x}, \mathcal{X}_k^+)}{\sigma_n^2(\boldsymbol{x})}} = \sqrt{1 - \rho_{n+k}^2(\boldsymbol{x}, \mathcal{X}_k^+)} \geq 0, \tag{31}$$

where $\rho_{n+k}(\boldsymbol{x}, \mathcal{X}_k^+)$ *is given by*

$$\rho_{n+k}(\boldsymbol{x}, \mathcal{X}_k^+) = \frac{\sqrt{c_n(\boldsymbol{x}, \mathcal{X}_k^+)^\top (\mathcal{C}_k^+)^{-1} c_n(\boldsymbol{x}, \mathcal{X}_k^+)}}{\sigma_n(\boldsymbol{x})} \in [0, 1], \tag{32}$$

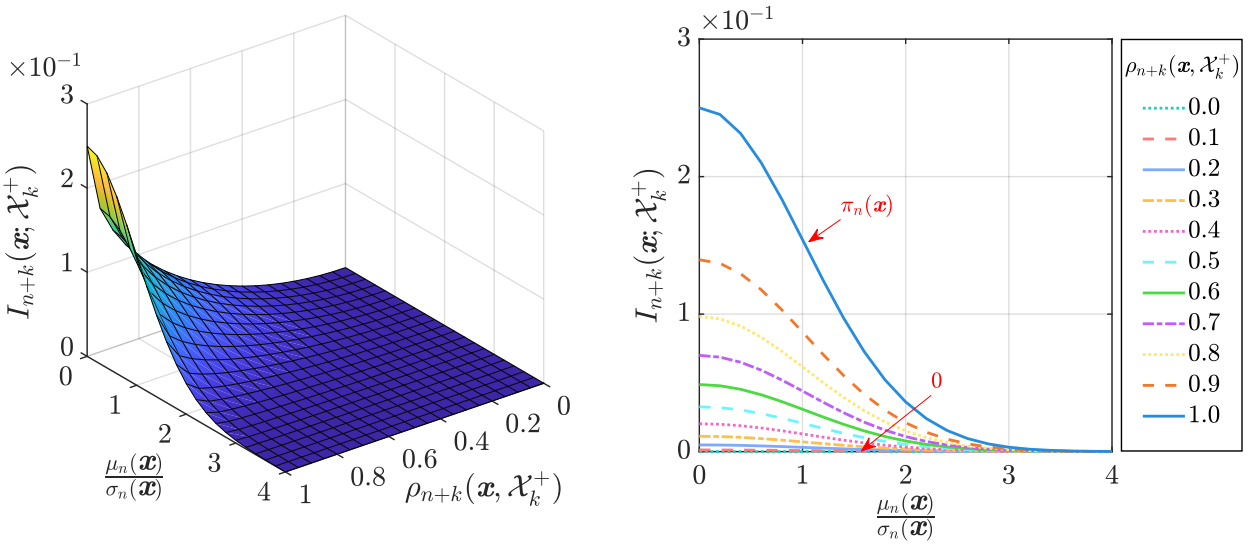
which can be interpreted as the correlation coefficient between $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ *and the space spanned by* $\widehat{\mathcal{G}}_n(\mathcal{X}_k^+)$*.*

Besides, the lower and upper bounds of $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ *are given as*

$$\begin{cases} I_{n+k}^\mathrm{L}(\boldsymbol{x}; \mathcal{X}_k^+) = 0, \\ I_{n+k}^\mathrm{U}(\boldsymbol{x}; \mathcal{X}_k^+) = \pi_n(\boldsymbol{x}). \end{cases} \tag{33}$$

The proof of Proposition 2 is provided in Appendix D. Then, the geometrical interpretation of $\rho_{n+k}(\boldsymbol{x}, \mathcal{X}_k^+)$ is given in Appendix E. Basically, the closer between $\boldsymbol{x}$ and $\mathcal{X}_k^+$, the greater the $\rho_{n+k}(\boldsymbol{x}, \mathcal{X}_k^+)$.

Eqs. (30) and (32) indicate that $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ is a function of both $\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}$ and $\rho_{n+k}(\boldsymbol{x}, \mathcal{X}_k^+)$. The former specifies the relative location of $\boldsymbol{x}$ in the input space, while the latter encodes the impact of adding a batch of new points $\mathcal{X}_k^+$. Given that $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ is an even function with respect to $\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}$, Fig. 2 only shows the behavior of $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ for $\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} > 0$. Obviously, $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ gets close to its upper bound $\pi_n(\boldsymbol{x})$ when $\rho_{n+k}(\boldsymbol{x}, \mathcal{X}_k^+) \to 1$. Further, $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ achieves its greatest value only when $\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} \to 0$ and $\rho_{n+k}(\boldsymbol{x}, \mathcal{X}_k^+) \to 1$.



(a) (b)

Figure 2: Illustration of the inner integral $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ in the MSMR

3.5. Outer integral in the MSMR

By substituting Eq. (30) into Eq. (28), $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$ reduces to a single integral, that is,

$$\begin{aligned}\mathrm{MSMR}_{n+k}\left(\mathcal{X}_k^+\right) &= \mathbb{E}_{\boldsymbol{X}}\left[I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\right], \\ &= \mathbb{E}_{\boldsymbol{X}}\left[\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} + \beta\right) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} - \beta\right) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} + \beta\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) + \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} - \beta\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right]\end{aligned} \tag{34}$$

which generally has no analytical expression. Besides, according to Eq. (33), there holds

$$0 \leq \mathrm{MSMR}_{n+k}(\mathcal{X}_k^+) \leq H_n, \tag{35}$$

which means that $\mathrm{MSMR}_{n+k}(\cdot)$ is non-negative but is always less than H_n.

Further, the VAIS-based estimation of $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$ is given by

$$\mathrm{MSMR}_{n+k}\left(\mathcal{X}_k^+\right) \approx \frac{1}{Q_3}\sum_{q=1}^{Q_3}\left[I_{n+k}\left(\boldsymbol{x}^{(q)}; \mathcal{X}_k^+\right)\frac{f_{\boldsymbol{X}}\left(\boldsymbol{x}^{(q)}\right)}{h_{\boldsymbol{X}}\left(\boldsymbol{x}^{(q)}\right)}\right], \tag{36}$$

where $\mathcal{X}_{\mathrm{Q}_3} = \left\{\boldsymbol{x}^{(q)}, q = 1, \ldots, Q_3\right\}$ denotes a set of Q_3 samples drawn from $h_{\boldsymbol{X}}(\boldsymbol{x})$.

In the BALR's workflow, a total of 3 single integrals need to be numerically computed by the VAIS at each iteration, i.e., the $\mu_{\widehat{P}_{f,n}}$ in Eq. (10), the H_n in Eq. (21), and the $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$ in Eq. (36). Unlike both $\mu_{\widehat{P}_{f,n}}$ and H_n that are only computed once per iteration, the computation of $\mathrm{MSMR}_{n+k}(\cdot)$ has to be repeated on all possible candidate batches in the $\mathcal{X}_{\mathrm{C}}$, i.e., $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+), \forall\, \mathcal{X}_k^+ \in \mathcal{X}_{\mathrm{C}}$. The large number of candidate batches makes it infeasible to specify different values of Q_3 for distinct batches $\mathcal{X}_k^+ \in \mathcal{X}_{\mathrm{C}}$, according to their respective CV values. Hence, the same value of Q_3 is preferable to $\mathrm{MSMR}_{n+k}(\cdot)$, regardless of different candidate batches $\mathcal{X}_k^+$.

To this end, the settings of the sizes of their quadrature sample sets, i.e., Q_1, Q_2, and Q_3, are specified as follows. First, let $Q_1 = Q_2 = Q$, where the Q is specified according to the following hybrid criterion

$$\left(\mathrm{CV}\left[\tilde{\mu}_{\widehat{P}_{f,n}}\right] \leq \varepsilon_{\mathrm{Q}}\right) \bigcap \left(\mathrm{CV}\left[\tilde{H}_n\right] \leq \varepsilon_{\mathrm{Q}}\right), \tag{37}$$

which will secure the integral precision of both $\tilde{\mu}_{\widehat{P}_{f,n}}$ and $\tilde{H}_n$ readily. Then, since the H_n is the upper bound of $\mathrm{MSMR}_{n+k}(\cdot)$ (Eq. (35)), the values of $\mathrm{MSMR}_{n+k}(\cdot)$ at those promising batches shall be very close to H_n. In this way, the size of quadrature point set sufficient for $\tilde{H}_n$ shall be also favorable for $\mathrm{MSMR}_{n+k}(\cdot)$, particularly for those promising candidate batches. Hence, it is desirable to specify that $Q_1 = Q_2 = Q_3 = Q$.

Further, the $\mathcal{X}_{\mathrm{Q}}$ will be taken as the candidate pool $\mathcal{X}_{\mathrm{C}}$ in Eq. (29). Then, selecting the best next batch $\mathcal{X}_k^*$ from the $\mathcal{X}_{\mathrm{C}}$ by maximizing $\mathrm{MSMR}_{n+k}(\cdot)$ may encounter the following two critical bottlenecks.

- *Computer memory issue.* According to Eq. (36), a single evaluation of $\mathrm{MSMR}_{n+k}(\cdot)$ entails computing $\left\{I_{n+k}\left(\boldsymbol{x}^{(q)}; \mathcal{X}_k^+\right)\right\}_{q=1}^{Q}$, or equivalently $\left\{\rho_{n+k}\left(\boldsymbol{x}^{(q)}, \mathcal{X}_k^+\right)\right\}_{q=1}^{Q}$. Then, according to Eq. (32), the following matrix operation is intrinsically involved

$$\left\{c_n(\boldsymbol{x}^{(q)}, \mathcal{X}_k^+)^{\top}(\mathcal{C}_k^+)^{-1}c_n(\boldsymbol{x}^{(q)}, \mathcal{X}_k^+)\right\}_{q=1}^{Q} = \mathrm{diag}\left(c_n\left(\mathcal{X}_{\mathrm{Q}}, \mathcal{X}_k^+\right)^{\top}\left(\mathcal{C}_k^+\right)^{-1}c_n\left(\mathcal{X}_{\mathrm{Q}}, \mathcal{X}_k^+\right)\right), \tag{38}$$

  where $\mathrm{diag}(\cdot)$ returns the diagonal elements of a matrix. Obviously, this matrix operation has to store a matrix of size $Q \times Q$. If the sample size Q is significant (e.g., $\mathcal{O}(10^{5-6})$), it will raise serious computer memory issue, which will be addressed in Section 3.6.

- *Vast size of candidate pool.* According to Eq. (29), maximizing $\mathrm{MSMR}_{n+k}(\cdot)$ entails iterating over all possible choices of $\mathcal{X}_k^+ \in \mathcal{X}_{\mathrm{C}}$, giving rise to a total of $\binom{C}{k}$ calls to $\mathrm{MSMR}_{n+k}(\cdot)$. For example, if $C = 2\times10^5$ and $k = 10$, there are $\binom{C}{k} = 2.82 \times 10^{46}$ possible choices, making it computationally intractable. This challenge will be handled in Sections 3.7 and 3.8.

3.6. Truncation of quadrature set in the MSMR

To avoid storing the large-sized matrix in Eq. (38), two effective workarounds are developed in turn as follows.

First, instead of direct matrix manipulation, Eq. (38) can be equivalently computed in the form of 'sum $(A. * (B \setminus A), 1)$' in MATLAB, where $\setminus$ denotes the matrix division; $.*$ denotes the element-wise multiplication; $\mathrm{sum}(\cdot, 1)$ returns the sum of each column in a matrix. In this way, the maximum size of matrix to store will reduce from $Q \times Q$ to $Q \times k$.

Second, to further reduce the maximum size of matrix, the quadrature set $\mathcal{X}_{\mathrm{Q}}$ in the numerical computation of $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$ can be truncated greatly. Obviously, it is natural to only retain those quadrature points with significant values of $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$ in Eq. (36). However, the $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ depends on the specific location of $\mathcal{X}_k^+$ and thus varies among the $\mathcal{X}_{\mathrm{C}}$. For simplicity, the truncation criterion for the $\mathcal{X}_{\mathrm{Q}}$ should avoid the presence of $\mathcal{X}_k^+$. Given that the $\pi_n(\boldsymbol{x})$ is the upper bound of $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ (Eq. (33)), the $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$ will be close to 0, if the $\pi_n(\boldsymbol{x})\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$ is very negligible. Hence, the quadrature point $\boldsymbol{x}$ with greater value of $\pi_n(\boldsymbol{x})\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$ has higher probability of gaining significant value of $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$.

In this way, the $\mathcal{X}_{\mathrm{Q}}$ can be truncated to a subset of $Q_{\mathrm{T}}(\ll Q)$ quadrature points with the greatest values of $\pi_n(\boldsymbol{x})\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$. First, the $\mathcal{X}_{\mathrm{Q}}$ is sorted in decreasing order of $\pi_n(\boldsymbol{x})\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$, resulting in $\left\{\boldsymbol{x}^{(q_j)}, j = 1, \ldots, Q\right\}$, where q_j is the index of quadrature point with the j-th greatest value. Then, the truncated quadrature point set $\mathcal{X}_{\mathrm{Q_T}}$ is obtained as

$$\mathcal{X}_{\mathrm{Q_T}} = \left\{\boldsymbol{x}^{(q_j)}, j = 1, \ldots, Q_{\mathrm{T}}\right\}, \tag{39}$$

with the size Q_{T} adaptively determined as

$$Q_{\mathrm{T}} = \inf\left\{\mathcal{Q} \in \mathbb{N} : \frac{\sum_{j=1}^{\mathcal{Q}} \pi_n(\boldsymbol{x}^{(q_j)})\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(q_j)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(q_j)})}}{\sum_{j=1}^{Q} \pi_n(\boldsymbol{x}^{(q_j)})\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(q_j)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(q_j)})}} \geq \lambda\right\}, \tag{40}$$

where the ratio λ is set as 0.99 here. Obviously, the Q_{T} will alter in each iteration of BALR.

Therefore, Eq. (36) can be further reduced to

$$\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+) \approx \frac{1}{Q}\sum_{j=1}^{Q_{\mathrm{T}}}\left[I_{n+k}\left(\boldsymbol{x}^{(q_j)}; \mathcal{X}_k^+\right)\frac{f_{\boldsymbol{X}}\left(\boldsymbol{x}^{(q_j)}\right)}{h_{\boldsymbol{X}}\left(\boldsymbol{x}^{(q_j)}\right)}\right], \tag{41}$$

where the maximum size of matrix to store further reduces from $Q \times k$ to $Q_{\mathrm{T}} \times k$.

For illustration, Fig. 3 presents the truncation of $\mathcal{X}_{\mathrm{Q}}$ in an iteration of BALR for the shear frame example (Section 5.3). First, the sample size Q of $\mathcal{X}_{\mathrm{Q}}$ is specified as 6×10^5 in this iteration, and only a small portion of them have significant values of $\pi_n(\boldsymbol{x})\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$ (see the left panel of Fig. 3). Then, according to Eq. (40), $Q_{\mathrm{T}} = 7560$, and $\frac{Q_{\mathrm{T}}}{Q} = 1.26\%$, as shown in the right panel of Fig. 3. Clearly, the truncation of $\mathcal{X}_{\mathrm{Q}}$ greatly facilitates a single computation of $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$.

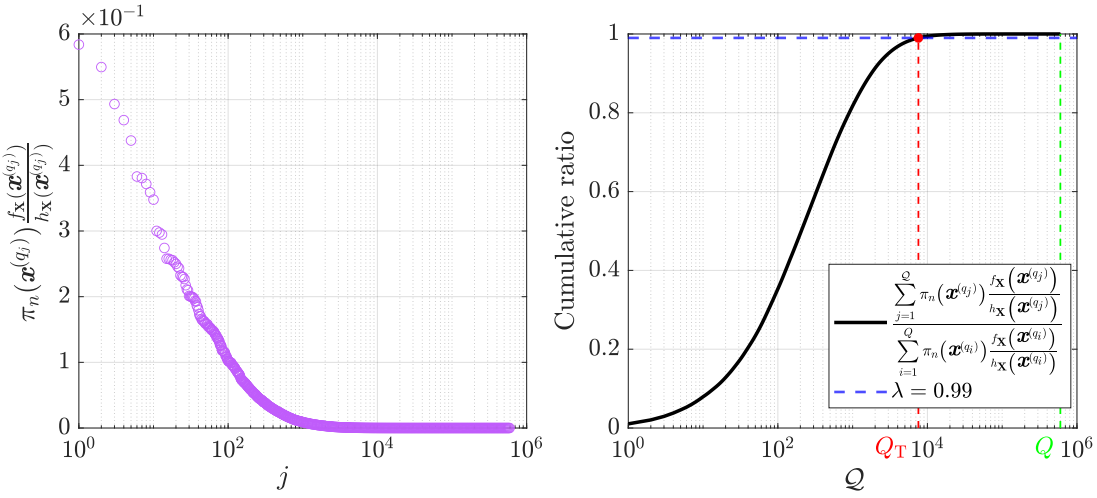


Figure 3: Illustration of truncating quadrature set $\mathcal{X}_{\mathrm{Q}}$ in the MSMR

3.7. Truncation of candidate pool in the MSMR

To deal with the massive candidate batches when maximizing the $\mathrm{MSMR}_{n+k}(\cdot)$ in Eq. (29), the candidate pool $\mathcal{X}_{\mathrm{C}}$ can be greatly truncated according to the MSMR's informed preference of some candidate batches over others in the input space.

Eq. (33) indicates that $\pi_n(\boldsymbol{x}^{(q)})\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}$ varies with the location of quadrature point $\boldsymbol{x}^{(q)}$, but it is always the upper bound of $I_{n+k}(\boldsymbol{x}^{(q)};\mathcal{X}_k^+)\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}$, no matter where the batch of k new samples $\mathcal{X}_k^+$ is. Then, Fig. 2 shows that $I_{n+k}(\boldsymbol{x}^{(q)};\mathcal{X}_k^+)\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}$ is very close to its upper bound $\pi_n(\boldsymbol{x}^{(q)})\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}$ on those quadrature points $\boldsymbol{x}^{(q)}$ around the $\mathcal{X}_k^+$ (i.e., $\rho_{n+k}(\boldsymbol{x}^{(q)},\mathcal{X}_k^+)\to 1$), but becomes almost 0 on the other quadrature points $\boldsymbol{x}^{(q)}$ away from the $\mathcal{X}_k^+$ (i.e., $\rho_{n+k}(\boldsymbol{x}^{(q)},\mathcal{X}_k^+)\to 0$). Here, to clarify the impact of $\mathcal{X}_k^+$ on the value of $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$, two different situations are discussed for comparison.

- If the $\mathcal{X}_k^+$ is placed in a local region featuring great value of $\pi_n(\boldsymbol{x})\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$, the resulting values of $I_{n+k}(\boldsymbol{x}^{(q)};\mathcal{X}_k^+)\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}$ will be very close to $\pi_n(\boldsymbol{x})\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$ and thus significant on those quadrature points $\boldsymbol{x}^{(q)}$ in this local region. Then, the values are negligible in other regions (see the boxes with cyan background in Table 2).
- If the $\mathcal{X}_k^+$ is placed in a local region with small value of $\pi_n(\boldsymbol{x})\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$, the corresponding values of $I_{n+k}(\boldsymbol{x}^{(q)};\mathcal{X}_k^+)\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(q)})}$ are close to $\pi_n(\boldsymbol{x})\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$ but very trivial in this local region. Hence, their values are almost negligible on those quadrature points $\boldsymbol{x}^{(q)}$ in the entire input space (see the boxes with yellow background in Table 2).

Obviously, the value of $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$ in the former case is much greater than that in the latter case. Hence, the promising candidate batches $\mathcal{X}_k^+$ are exactly those quadrature points with greatest values of $\pi_n(\boldsymbol{x})\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$.

Table 2: Characteristic of a promising candidate batch informed by the MSMR

$I_{n+k}(\boldsymbol{x};\mathcal{X}_k^+)\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$ / $\pi_n(\boldsymbol{x})\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$ / $\rho_{n+k}(\boldsymbol{x},\mathcal{X}_k^+)$	$\to 0$	Great
$\to 0$	$\to 0$	$\to 0$
$\to 1$	$\to 0$	Great

In this way, the candidate pool $\mathcal{X}_{\mathrm{C}}$ can be truncated to

$$\mathcal{X}_{\mathrm{C_T}} = \left\{\boldsymbol{x}^{(q_j)}, j = 1,\ldots,C_{\mathrm{T}}\right\}, \tag{42}$$

where the size C_{T} is set as 10^3 here, which will be justified in Section 5.

Then, Eq. (29) can be further reduced to

$$\mathcal{X}_k^* = \underset{\mathcal{X}_k^+\in\mathcal{X}_{\mathrm{C_T}}}{\arg\max}\,\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+), \tag{43}$$

which only involves $\binom{C_{\mathrm{T}}}{k}$ possible candidate batches but is still unaffordable to some extent. This will be further addressed in Section 3.8.

3.8. Heuristic MSMR-based multi-point enrichment process

A heuristic treatment of maximizing MSMR in Eq. (43) is devised here, owing to the attractive ability of $\mathrm{MSMR}_{n+k}(\cdot)$ to work directly with a batch of k new samples $\mathcal{X}_k^+$. The basic idea is to sequentially select a batch of k new samples, rather than selecting them once per iteration.

Specifically, when $k=1$, the $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$ in Eq. (41) reduces to

$$\mathrm{MSMR}_{n+1}\left(\boldsymbol{x}_+\right) \approx \frac{1}{Q}\sum_{j=1}^{Q_\mathrm{T}} I_{n+1}\left(\boldsymbol{x}^{(q_j)};\boldsymbol{x}_+\right)\frac{f_{\boldsymbol{X}}\left(\boldsymbol{x}^{(q_j)}\right)}{h_{\boldsymbol{X}}\left(\boldsymbol{x}^{(q_j)}\right)}, \tag{44}$$

where $I_{n+1}(\cdot;\boldsymbol{x}_+)$ reminds that it is a function of a candidate point $\boldsymbol{x}_+\in\mathcal{X}_{\mathrm{C_T}}$ solely. In this regard, the 1-st best next point $\boldsymbol{x}^{(n+1)}$ is selected as

$$\boldsymbol{x}^{(n+1)} = \underset{\boldsymbol{x}_+\in\mathcal{X}_{\mathrm{C_T}}}{\arg\max}\,\mathrm{MSMR}_{n+1}(\boldsymbol{x}_+), \tag{45}$$

then, $\mathcal{X}_{\mathrm{C_T}} = \mathcal{X}_{\mathrm{C_T}}\setminus\left\{\boldsymbol{x}^{(n+1)}\right\}$, and $\mathcal{X}_1^* = \left\{\boldsymbol{x}^{(n+1)}\right\}$.

When $k\geq 2$, assume that the former $(k-1)$ best next points $\mathcal{X}_{k-1}^* = \left\{\boldsymbol{x}^{(n+1)},\ldots,\boldsymbol{x}^{(n+k-1)}\right\}$ have been selected and are viewed as fixed arguments, the $\mathrm{MSMR}_{n+k}(\mathcal{X}_k^+)$ in Eq. (41) is recast as

$$\mathrm{MSMR}_{n+k}\left(\mathcal{X}_{k-1}^*,\boldsymbol{x}_+\right) \approx \frac{1}{Q}\sum_{j=1}^{Q_\mathrm{T}} I_{n+k}\left(\boldsymbol{x}^{(q_j)};\mathcal{X}_{k-1}^*,\boldsymbol{x}_+\right)\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(q_j)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(q_j)})}, \tag{46}$$

where $I_{n+k}(\cdot;\mathcal{X}_{k-1}^*,\boldsymbol{x}_+)$ reminds that it is also a function of a candidate point $\boldsymbol{x}_+\in\mathcal{X}_{\mathrm{C_T}}$ solely. In this regard, the k-th best next point $\boldsymbol{x}^{(n+k)}$ can be selected as

$$\boldsymbol{x}^{(n+k)} = \underset{\boldsymbol{x}_+\in\mathcal{X}_{\mathrm{C_T}}}{\arg\max}\,\mathrm{MSMR}_{n+k}\left(\mathcal{X}_{k-1}^*,\boldsymbol{x}_+\right), \tag{47}$$

then, $\mathcal{X}_{\mathrm{C_T}} = \mathcal{X}_{\mathrm{C_T}}\setminus\mathcal{X}_{k-1}^*$, and $\mathcal{X}_k^* = \mathcal{X}_{k-1}^*\bigcup\left\{\boldsymbol{x}^{(n+k)}\right\}$.

In this way, Eq. (43) reduces to K consecutive runs of single-point selection in Eqs. (45) and (47). Then, the number of candidate choices reduces from $\binom{C_\mathrm{T}}{k}$ to $C_\mathrm{T}\times K$ in total, much computationally cheaper to handle.

Until now, when to terminate the sequential increase of k in Eq. (46), or equivalently how to determine the final size K of batch of new samples per iteration, has not been discussed yet. Here, two different schemes of specifying the K value are developed here.

- *Prescribed scheme.* The batch size K is prescribed as a fixed value; then, according to Eqs. (45) and (47), sequentially increasing k until K will result in $\mathcal{X}_K^* = \left\{\boldsymbol{x}^{(n+k)}, k=1,\ldots,K\right\}$ readily.
- *Adaptive scheme.* The batch size K is automatically identified per iteration. First, Eqs. (45) and (47) indicate that the expected gain of adding the k-th best next point $\boldsymbol{x}^{(n+k)}$ can be expressed as

$$G_{n+k} = \begin{cases}\mathrm{MSMR}_{n+k}\left(\mathcal{X}_k^*\right) - \mathrm{MSMR}_{n+k-1}\left(\mathcal{X}_{k-1}^*\right), & k\geq 2,\\ \mathrm{MSMR}_{n+1}\left(\mathcal{X}_1^*\right), & k=1,\end{cases} \tag{48}$$

  which is essentially the expected reduction of IMP due to adding $\boldsymbol{x}^{(n+k)}$ in the batch. As k increases, the G_{n+k} reduces gradually. Then, if the G_{n+k} itself or the ratio $\frac{G_{n+k}}{G_{n+1}}$ becomes very small, adding $\boldsymbol{x}^{(n+k)}$ does not exert sufficient gain, and the sequential increase of k can be stopped. Therefore, the K value can be determined as

$$K = \inf\left\{G_{n+k}\leq 0\bigcup\frac{G_{n+k}}{G_{n+1}}\leq\varepsilon_\mathrm{G}\bigcup k\leq n_\mathrm{core}\right\}, \tag{49}$$

  where the tolerance ε_G is set as 0.3; n_core is the number of available CPU cores. The condition $G_{n+k}\leq 0$ should be satisfied twice in a row for reassurance.

Algorithm 1 provides the flowchart of MSMR-based multi-point enrichment process in a single iteration. This flowchart involves a total of $K\times C_\mathrm{T}\times Q_\mathrm{T}$ calls to its integrand $I_{n+k}(\cdot;\cdot)$. Thanks to analytical tractability of $I_{n+k}(\cdot;\cdot)$ (Eq. (30)), the computation of $I_{n+k}(\cdot;\cdot)$ on the $\mathcal{X}_{\mathrm{Q_T}}$ (Lines 8-9 in Algorithm 1) can be conducted via the vectorization technique in MATLAB, as exemplified by the $I_{n+1}(\mathcal{X}_{\mathrm{Q_T}};\boldsymbol{x}_+^{(i)})$ in Fig. 4(a). As a result, the number of element-wise evaluations of integrand reduces from $K\times C_\mathrm{T}\times Q_\mathrm{T}$ to $K\times C_\mathrm{T}$ in Algorithm 1 (the "for-loop" in Lines 7-11), and they can be readily performed in parallel.

Algorithm 1 MSMR-based multi-point enrichment process at an iteration

Input: The Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$, the K value in the *prescribed scheme*, or the ε_{G} value in the *adaptive scheme*.

Specify the size Q of the quadrature point set $\mathcal{X}_{\mathrm{Q}}$ according to both $\mathrm{CV}\big[\tilde{\mu}_{\widehat{P}_{f,n}}\big]$ and $\mathrm{CV}\big[\tilde{H}_n\big]$. ▷ Eqs. (10), (21) and (37)

The Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ provides means $\big\{\mu_n(\boldsymbol{x}^{(q)})\big\}_{q=1}^{Q}$ and variances $\big\{\sigma_n^2(\boldsymbol{x}^{(q)})\big\}_{q=1}^{Q}$ at $\mathcal{X}_{\mathrm{Q}}$. ▷ Eqs. (A.6), (A.7)

Obtain the truncated quadrature set $\mathcal{X}_{\mathrm{Q_T}} = \big\{\boldsymbol{x}^{(q_j)}\big\}_{j=1}^{Q_{\mathrm{T}}}$ from the $\mathcal{X}_{\mathrm{Q}}$. ▷ Eq. (40)

Obtain the truncated candidate pool $\mathcal{X}_{\mathrm{C_T}} = \big\{\boldsymbol{x}_{+}^{(i)}\big\}_{i=1}^{C_{\mathrm{T}}}$ from the $\mathcal{X}_{\mathrm{Q}}$. ▷ Eq. (42)

Set $k = 1$, and $\mathcal{X}_0^* = \{\}$.

while true **do**

  for $i = 1, \ldots, C_{\mathrm{T}}$, **do**

    Compute $\big\{\rho_{n+k}\big(\boldsymbol{x}^{(q_j)}; \mathcal{X}_{k-1}^*, \boldsymbol{x}_{+}^{(i)}\big)\big\}_{j=1}^{Q_{\mathrm{T}}}$ on all quadrature points in the $\mathcal{X}_{\mathrm{Q_T}}$. ▷ Eq. (32)

    Compute $\big\{I_{n+k}\big(\boldsymbol{x}^{(q_j)}; \mathcal{X}_{k-1}^*, \boldsymbol{x}_{+}^{(i)}\big)\big\}_{j=1}^{Q_{\mathrm{T}}}$ on all quadrature points in the $\mathcal{X}_{\mathrm{Q_T}}$. ▷ Eq. (30)

    Compute $\mathrm{MSMR}_{n+k}\big(\mathcal{X}_{k-1}^*, \boldsymbol{x}_{+}^{(i)}\big)$. ▷ Eqs. (44) or (46)

  end for

  Select the k-th best next point $\boldsymbol{x}^{(n+k)}$ from $\mathcal{X}_{\mathrm{C_T}}$ according to $\mathrm{MSMR}_{n+k}(\cdot)$. ▷ Eqs. (45) or (47)

  Identify the batch size K in the *adaptive scheme*. ▷ Eqs. (48) and (49)

  if $k \geq K$ **then**

    Break;

  else

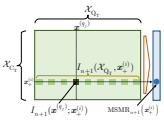
    Update: $\mathcal{X}_k^* = \mathcal{X}_{k-1}^* \bigcup \big\{\boldsymbol{x}^{(n+k)}\big\}$, $\mathcal{X}_{\mathrm{C_T}} = \mathcal{X}_{\mathrm{C_T}} \setminus \big\{\boldsymbol{x}^{(n+k)}\big\}$, and $k = k + 1$.

  end if

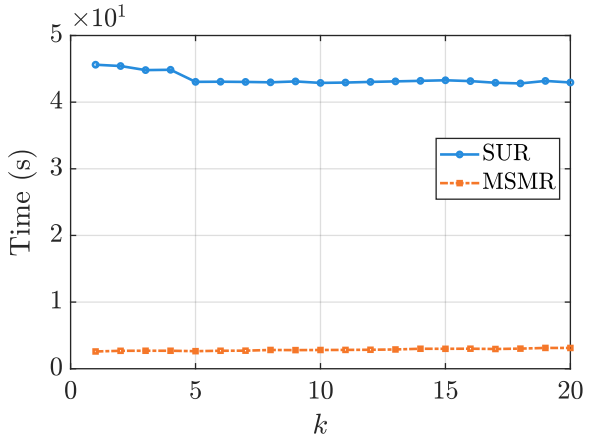
end while

Output: The batch of K best next samples $\mathcal{X}_K^* = \big\{\boldsymbol{x}^{(n+1)}, \ldots, \boldsymbol{x}^{(n+K)}\big\}$ obtained at this iteration.

Remark 1. *The existing learning function SUR [25, 42] is outlined in Appendix F for comparison. Note that the settings of both quadrature set and candidate pool for the SUR were not detailed in [42]. Hence, the truncation of quadrature set and candidate pool, as well as the heuristic maximization of learning function, developed in this study can be readily customized to the SUR, with just replacing $\pi_n(\boldsymbol{x})$ by $\sigma^2_{\widehat{\mathbb{I}}_n}(\boldsymbol{x})$ in the three workarounds. Then, the SUR-based multi-point enrichment process can be conducted in a similar way to the MSMR (Algorithm 1). Similarly, there are also a total of $K \times C_{\mathrm{T}} \times Q_{\mathrm{T}}$ evaluations of its integrand in the SUR. However, all those evaluations have to be conducted in an element-wise manner, due to the presence of $\Phi_2(\cdot;\cdot,\cdot)$. Hence, the SUR-based multi-point enrichment process will be very time-consuming, even resorting to parallel computing.*



(a) computation of $\mathrm{MSMR}_{n+1}(\boldsymbol{x}_{+}^{(i)})$



(b) comparison of running time in an iteration

Figure 4: Comparison between the proposed MSMR and the existing SUR

For illustration, Fig. 4(b) shows a comparison of running time between the MSMR and the SUR in an iteration, where $Q_{\mathrm{T}} = 2 \times 10^5$, $C_{\mathrm{T}} = 10^3$, and $K = 20$. An Intel i9-14900KF CPU processor with 3.2 GHz, 64G RAM, and 20 available CPU cores is utilized. The running time of MSMR at each loop is nearly 2 seconds, giving rise to the total running time of 45 seconds in an iteration. By comparison, the SUR entails almost 43

seconds per loop, giving rise to a total of 857 seconds in an iteration. Clearly, the MSMR needs much less running time than the SUR in an iteration.

4. Multi-point Bayesian active learning reliability analysis based on the MSMR

The basic flowchart of MSMR-based multi-point Bayesian active learning reliability method is shown in Fig. 5. The main steps are summarized as follows.

Step 1 **Generation of the initial ED**

The initial ED prefers to be as space-filling as possible, thereby gaining a fair Kriging in the initial stage. To this end, the initial ED is generated following the so-called "four-sigma" rule [46]. First, the sampling domain $\mathbb{X}_{\mathrm{s}}$ is defined by

$$\mathbb{X}_{\mathrm{s}} = \prod_{l=1}^{d} [-4, 4]\,. \tag{50}$$

Then, the Latin centroidal Voronoi tessellation method [47] is taken to generate a set of uniform samples in the $\mathbb{X}_{\mathrm{s}}$, denoted as $\mathcal{X}_{n_0} = \left\{\boldsymbol{x}^{(i)}, i = 1, \ldots, n_0\right\}$ of size $n_0 = \max(10, d+1)$. Perform $\mathcal{G}$-function evaluations on the $\mathcal{X}_{n_0}$ to obtain $\mathcal{Y}_{n_0} = \left\{y^{(i)} = \widehat{\mathcal{G}}_n\left(\boldsymbol{x}^{(i)}\right), i = 1, \ldots, n_0\right\}$. Finally, the initial ED is collected as $\mathcal{D}_{n_0} = \{\mathcal{X}_{n_0}, \mathcal{Y}_{n_0}\}$, and let $n = n_0$;

Step 2 **Kriging calibration**

Training a Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ based on the current ED $\mathcal{D}_n$ provides the posterior mean $\mu_n(\boldsymbol{x})$, variance $\sigma_n(\boldsymbol{x})$, and covariance $c_n(\boldsymbol{x}, \boldsymbol{x}')$; see Eqs. (A.6), (A.7) and (A.8).

Step 3 **Failure probability estimation**

According to Eqs. (10) and (21), the $\mu_{\widehat{P}_{f,n}}$ and H_n are numerically computed based on the VAIS. Then, the quadrature point set $\mathcal{X}_{\mathrm{Q}}$ is taken as the candidate pool $\mathcal{X}_{\mathrm{C}}$ in Step 5.

To facilitate numerical implementation, the $\mathcal{X}_{\mathrm{Q}}$ in each iteration can be sequentially specified as follows. First, the final quadrature set in the last iteration is taken as the initial quadrature set $\mathcal{X}_{\mathrm{Q}}^{\mathrm{i}}$ of size Q_{i} in the current iteration. Then, if both $\mathrm{CV}\left[\tilde{\mu}_{\widehat{P}_{f,n}}\right]$ and $\mathrm{CV}\left[\tilde{H}_n\right]$ meet Eq. (37), the final value of Q in this iteration is exactly equal to Q_{i}; otherwise, sequentially add sets of Q_{a} new sample points until satisfying Eq. (37), and the final sample size in this iteration is equal to $Q = Q_{\mathrm{i}} + Q_{\mathrm{a}} \times n_{\mathrm{a}}$, where n_{a} is the number of steps to need. Here, set $Q_{\mathrm{i}} = 2 \times 10^5$ in the first iteration, and $Q_{\mathrm{a}} = 2 \times 10^5$ in all iterations.

Generally, after several initial iterations, the $\mathcal{X}_{\mathrm{Q}}$ remains unchanged during the remaining iterations. Hence, this practice avoids frequently altering the $\mathcal{X}_{\mathrm{Q}}$ to a great extent.

Step 4 **Check of convergence criterion**

A hybrid convergence criterion combining two distinct ones is deployed here. First, recall that the H_n is an upper-bounding metric about the mean absolute deviation of $\mu_{\widehat{P}_{f,n}}$, the $\frac{\tilde{H}_n}{\tilde{\mu}_{\widehat{P}_{f,n}}}$ is naturally a favorable metric for the precision of $\tilde{\mu}_{\widehat{P}_{f,n}}$. Here, the relative decrease of $\frac{\tilde{H}_n}{\tilde{\mu}_{\widehat{P}_{f,n}}}$, rather than absolute one, is used to define the convergence criterion, that is,

$$\Delta_H = \frac{\frac{\tilde{H}_n}{\tilde{\mu}_{\widehat{P}_{f,n}}}}{\max\limits_{i \le n}\left(\frac{\tilde{H}_i}{\tilde{\mu}_{\widehat{P}_{f,i}}}\right)} \le \varepsilon_H, \tag{51}$$

where the tolerance ε_H is set as 0.3 (resp., 0.5) in the static (resp., dynamic) reliability problems.

Second, the stabilization of $\tilde{\mu}_{\widehat{P}_{f,n}}$ can be also taken as a convergence criterion, that is,

$$\Delta_{\widehat{P}_f} = \frac{\left|\tilde{\mu}_{\widehat{P}_{f,n}} - \tilde{\mu}_{\widehat{P}_{f,n-1}}\right|}{\tilde{\mu}_{\widehat{P}_{f,n-1}}} \le \varepsilon_{\mathrm{stab}}, \tag{52}$$

where the tolerance $\varepsilon_{\mathrm{stab}}$ is set as 0.1 here.

According to Eqs. (51) and (52), the hybrid convergence criterion is given by

$$\left(\Delta_H \le \varepsilon_H\right) \bigcap \left(\Delta_{P_f} \le \varepsilon_{\mathrm{stab}}\right). \tag{53}$$

Then, if Eq. (53) is met in 2 consecutive iterations, skip to Step 7, otherwise, continue to Step 5.

Step 5 **MSMR-based multi-point selection process**

According to Algorithm 1, select a batch of K best next samples $\mathcal{X}_K^* = \left\{\boldsymbol{x}^{(n+k)}, k = 1, \ldots, K\right\}$ from the candidate pool $\mathcal{X}_{\mathrm{C}}$ via the MSMR, with the batch size K either prescribed or adaptively identified.

Step 6 **Enrichment of $\mathcal{G}$-function evaluations**
Perform $\mathcal{G}$-function evaluations on $\mathcal{X}_K^*$ in parallel, yielding $\mathcal{Y}_K^* = \left\{y^{(n+k)}, k = 1, \ldots, K\right\}$. Then, conduct the following updates: $\mathcal{D}_{n+K} = \mathcal{D}_n \bigcup \{\mathcal{X}_K^*, \mathcal{Y}_K^*\}$, $n = n + K$, and return to Step 2.

Step 7 **End of the algorithm**
The final failure probability estimate $\mu_{\widehat{P}_{f,n}}$, the total number of iterations n_{iter}, and the total number of $\mathcal{G}$-function evaluation n_{eval} are recorded in this algorithm.

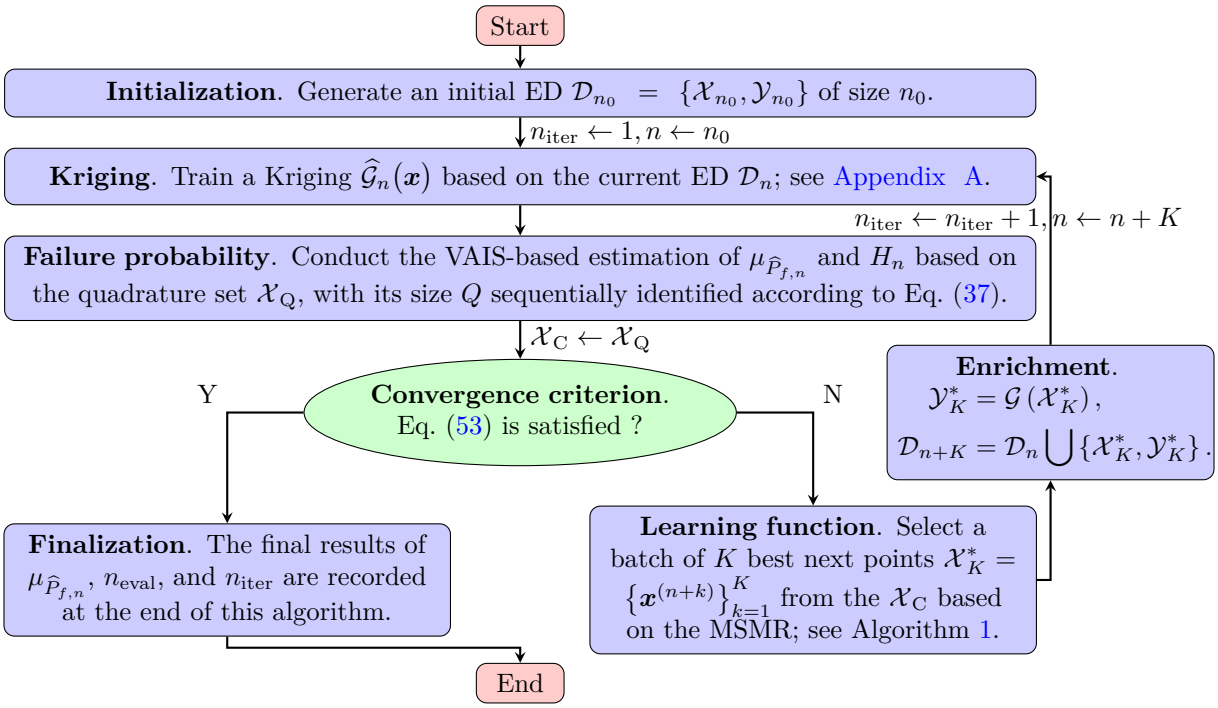


Figure 5: MSMR-based parallel Bayesian active learning reliability analysis

Remark 2. *The total computational time T_{c} in Fig. 5 can be roughly computed as $T_{\text{c}} = t_{\text{c}} \times n_{\text{iter}}$, where t_{c} is the computational time for an iteration, and n_{iter} is the total number of iterations. Then, the t_{c} can be basically divided into two parts: (a) The $\mathcal{G}$-function evaluation time t_{g} (Step 6); (b) The running time t_{a} of MSMR-based multi-point enrichment process, comprising the training of Kriging, the computation of failure probability, and the evaluation of MSMR (resp., Steps 2, 3 and 5). Basically, the t_{g} depends on the complexity of $\mathcal{G}$-function, while the t_{a} is related to the batch size K. If the t_{g} is far greater than t_{a}, the t_{c} will be dominated by t_{g}, and the advantage of reducing n_{iter} is readily converted to that of T_{c}; otherwise, the advantage of T_{c} will be governed by both $t_{\text{a}}/t_{\text{c}}$ and n_{iter}. This will be exemplified in Section 5.*

5. Illustrative examples

Four examples are investigated to test the performance of the proposed MSMR. The former two are the well-known benchmark examples, while the latter two are the actual engineering problems. The MCS is conducted to provide the reference failure probability $\widehat{P}_f^{\text{MCS}}$.

In the MSMR, both the prescribed scheme (K = 5, 10, 15, and 20) and the adaptive scheme ($n_{\text{core}} = 20$) are conducted. For comparison, several existing parallel (B)ALR methods are performed, including the PABQ [22], the ALR module in UQLab [18], and the SUR [42]. The default settings are considered in the ALR module in UQLab [18]; then, the settings of SUR are referred to Remark 1. In those reliability methods, the maximum value of n_{eval} is set as 300 due to computational cost considerations. Besides, the results from other existing reliability methods available in the literature are also provided.

All the (B)ALR methods are repeated 10 times to account for the randomness from both initial ED and sampling. Four performance metrics are recorded, i.e., $\mu_{\widehat{P}_f}$, n_{iter}, n_{eval}, and the relative error $\delta_{\widehat{P}_f} = \left|\mu_{\widehat{P}_f} - \widehat{P}_f^{\text{MCS}}\right| / \widehat{P}_f^{\text{MCS}} \times 100\%$. Then, the average of the four performance metrics, as well as the CV of $\mu_{\widehat{P}_{f,n}}$, in the 10 repetitions are computed for each example. Besides, the average of the total computational time T_{c} of different reliability methods are also recorded in the two engineering problems for comparison purposes.

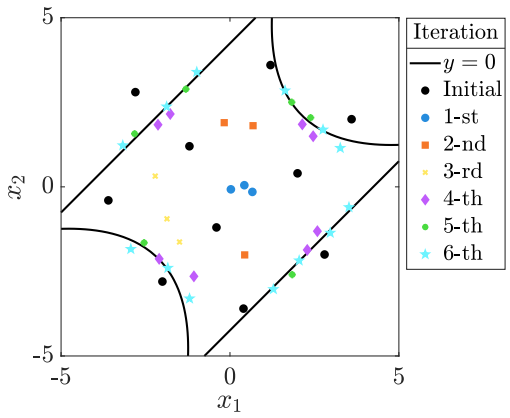
5.1. Four-branch function

The first example considers a four-branch function, which is a typical benchmark in the reliability literature [13, 18, 17]. The performance function is expressed as

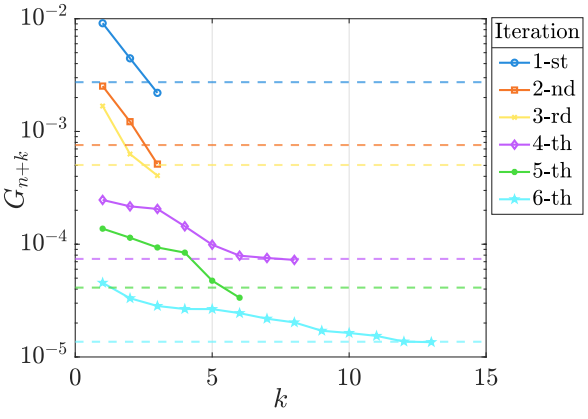
$$\mathcal{G}(\boldsymbol{X}) = \min \left\{ \begin{array}{c} a + 0.1(X_1 - X_2)^2 - \dfrac{X_1 + X_2}{\sqrt{2}} \\ a + 0.1(X_1 - X_2)^2 + \dfrac{X_1 + X_2}{\sqrt{2}} \\ (X_1 - X_2) + \dfrac{b}{\sqrt{2}} \\ (X_2 - X_1) + \dfrac{b}{\sqrt{2}} \end{array} \right\}, \tag{54}$$

where $\boldsymbol{X} = \{X_1, X_2\}$ is a vector of two independent standard Gaussian variables; the two constants a and b are set as 3 and 6, respectively.

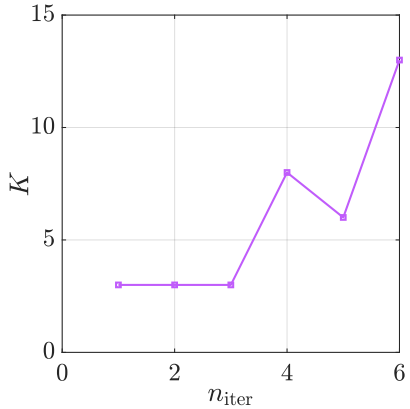
Fig. 6 presents one run of the proposed MSMR (adaptive scheme) for the typical four-branch function. The convergence of this algorithm is achieved with 6 iterations of multi-point enrichment. As marked as black circles in Fig. 6(a), the initial training samples are scattered in the input space. Then, the first 3 batches of new samples are far away from the limit state, due to the relatively poor accuracy of Kriging. By contrast, the remaining 3 batches of new samples are located near the actual limit state. Fig. 6(b) shows the expected gain G_{n+k} achieved by the k-th best next point $\boldsymbol{x}^{(n+k)}$ in different batches. The values of G_{n+k} generally decrease with the increasing of k and then determine the final batch size K per iteration. As illustrated in Fig. 6(c), the batch sizes K identified in the latter iterations are greater than those in the former ones. Obviously, thanks to the adaptive scheme, the MSMR avoids adding some 'useless' new samples when faced with a poor Kriging in the initial stage. Figs. 6(d) and 6(e) show the performances of the hybrid convergence criterion (Eq. (53)). Clearly, the IMP H_n is dramatically reduced by sequentially adding batches of new samples, and a favorable accuracy of failure probability estimate $\mu_{\widehat{P}_f}$ is finally gained.



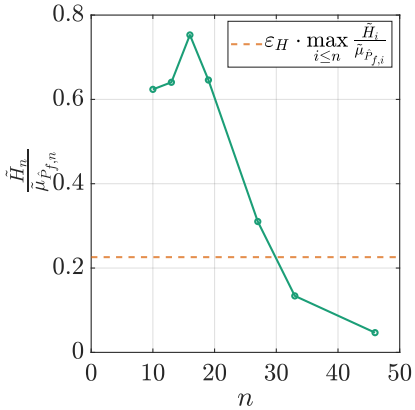
(a) sample distribution at each iteration



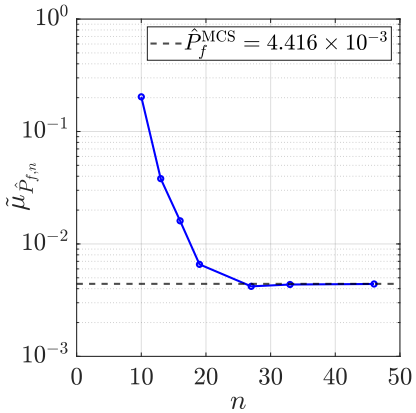
(b) the expected gain G_{n+k} of $\boldsymbol{x}^{(n+k)}$ at each iteration



(c) batch size K per iteration



(d) convergence of H_n



(e) convergence of $\mu_{\widehat{P}_f}$

Figure 6: The MSMR (adaptive scheme) for the four-branch function

Table 3 gives a comparison between different parallel (B)ALR methods for the four-branch function. Overall, all the reliability methods provide favorable estimates of P_f. Then, as the prescribed batch size K increases, the number of iterations n_{iter} reduces, but the number of $\mathcal{G}$-function evaluations n_{eval} increases. The MSMR (prescribed scheme) needs comparable iterations to the existing PABQ. Then, in comparison with the prescribed scheme ($K = 20$) needing 104 $\mathcal{G}$-function evaluations in average, the adaptive scheme avoids adding some unfavorable new samples, with only needing 47.5 $\mathcal{G}$-function evaluations in average.

Table 3: Reliability results in the four-branch function

Method		$\mathbb{E}[n_{\text{iter}}]$	$\mathbb{E}[n_{\text{eval}}]$	$\mathbb{E}\left[\mu_{\widehat{P}_f}\right]$	$\text{CV}\left[\mu_{\widehat{P}_f}\right]$	$\mathbb{E}\left[\delta_{\widehat{P}_f}\right]$	References
MCS	-	-	10^6	4.416×10^{-3}	-	-	[13]
RBIK	$K = 6$	17.7	110	4.429×10^{-3}	0.07 %	-	[36]
	$K = 8$	13.7	111.9	4.428×10^{-3}	0.07 %	-	[36]
	$K = 10$	12.1	121.4	4.429×10^{-3}	0.07 %	-	[36]
PABQ	$K = 6$	6.6	43.6	4.440×10^{-3}	2.53 %	-	[22]
	$K = 10$	5.2	52	4.400×10^{-3}	2.22 %	-	[22]
	$K = 15$	4.65	64.75	4.440×10^{-3}	1.35 %	-	[22]
AK-KB^n	$K = 3$	22.5	74.6	4.419×10^{-3}	-	-	[48]
	$K = 6$	11.5	73.1	4.411×10^{-3}	-	-	[48]
P-AK-MCS	$K = 4$	15.6	70.4	4.490×10^{-3}	-	-	[49]
	$K = 8$	8.8	74.4	4.560×10^{-3}	-	-	[49]
ALR in UQLab	$K = 5$	20.3	106.5	4.571×10^{-3}	1.42 %	3.51 %	-
	$K = 10$	10.2	102	4.581×10^{-3}	2.22 %	3.74 %	-
	$K = 15$	8	115	4.541×10^{-3}	2.49 %	2.99 %	-
	$K = 20$	6.9	128	4.539×10^{-3}	1.78 %	2.79 %	-
MSMR	$K = 5$	7.1	40.5	4.352×10^{-3}	2.14 %	2.23 %	-
	$K = 10$	6.5	65	4.410×10^{-3}	1.23 %	0.78 %	-
	$K = 15$	6	85	4.397×10^{-3}	0.68 %	0.43 %	-
	$K = 20$	5.7	104	4.412×10^{-3}	1.15 %	0.99 %	-
	Adaptive	7.1	47.5	4.394×10^{-3}	0.66 %	0.74 %	-

5.2. Static reliability analysis of a planar truss

The second example addresses static reliability analysis of a planar truss under vertical loads. This is also a typical benchmark in the reliability literature [12, 33, 17]. Fig. 7 shows that this truss consists of 23 bars and 13 nodes, with the vertical concentrated loads applied on the upper nodes. A total of 10 independent random variables are considered, i.e., $\{E_1, A_1, E_2, A_2, P_1, \ldots, P_6\}$, where E_1, A_1 denote the Young's modulus and cross-sectional area of horizontal bars, respectively; E_2, A_2 denote the Young's modulus and cross-sectional area of diagonal bars, respectively; $P_1, \ldots, P_6$ represent the vertical concentrated loads. Table 4 lists the statistical information of those random variables in the truss.

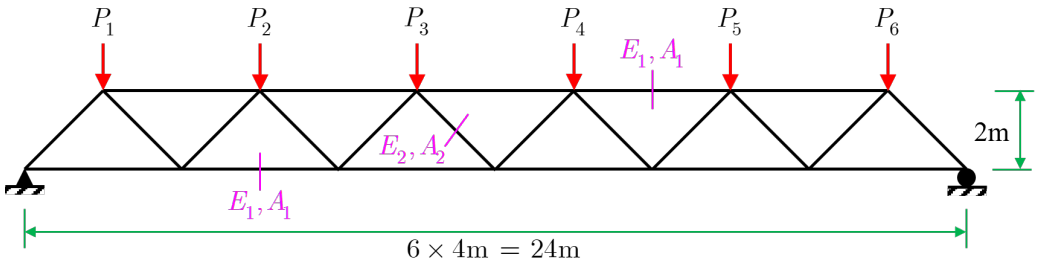


Figure 7: A planar truss under vertical loads

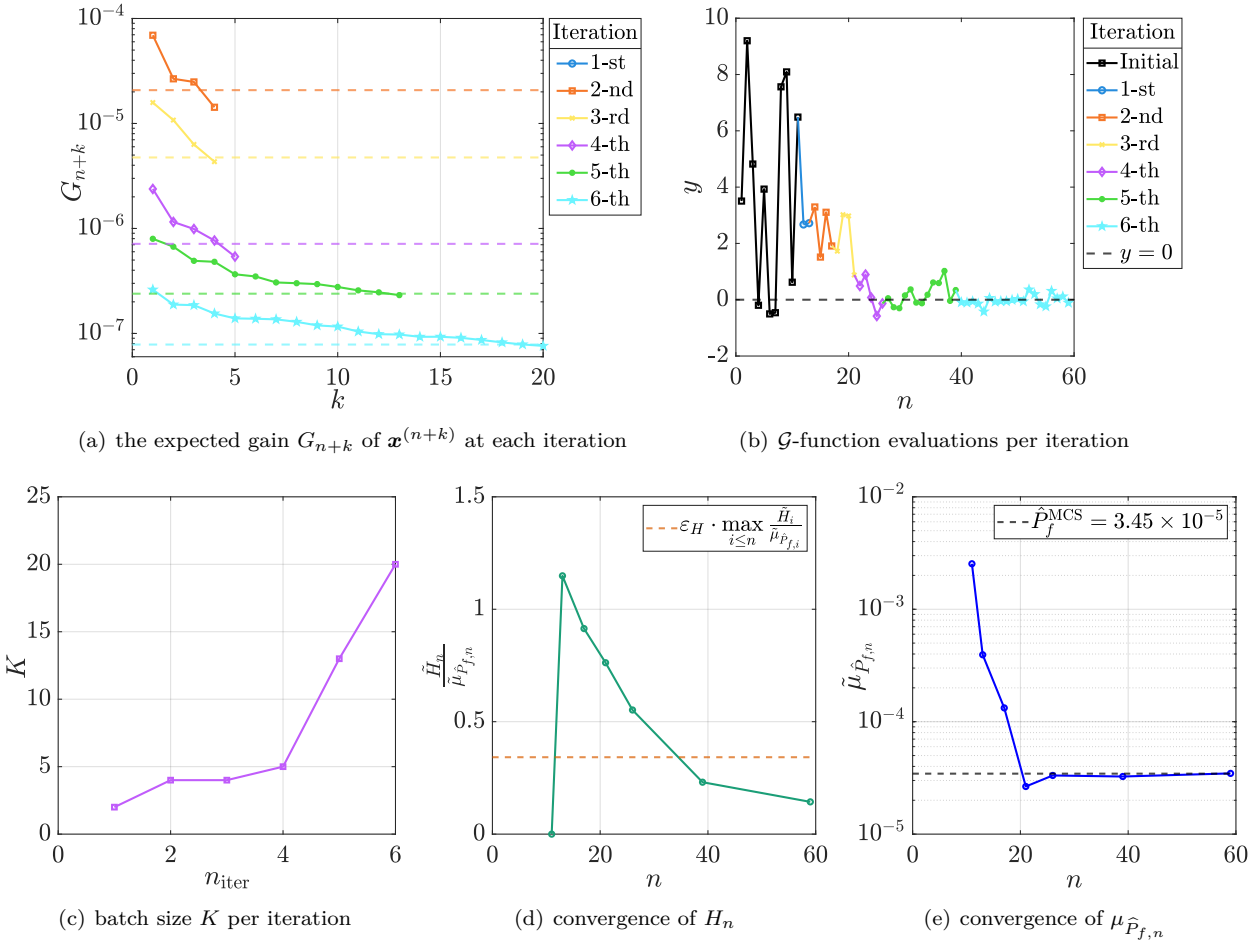
Table 4: Statistical information of random variables in the truss example [33]

Variable	Unit	Distribution	Mean	Standard deviation
E_1, E_2	Pa	Lognormal	2.1×10^{11}	2.1×10^{10}
A_1	m^2	Lognormal	2.0×10^{-3}	2.0×10^{-4}
A_2	m^2	Lognormal	1.0×10^{-3}	1.0×10^{-4}
$P_1, \ldots, P_6$	N	Gumbel	5×10^4	7.5×10^3

Of interest is the vertical mid-span deflection $U(\boldsymbol{x})$ of the truss, which is computed with an in-house finite-element analysis code developed in the MATLAB environment. The maximum allowable deflection is set as 14 cm. Therefore, the performance function is expressed as

$$\mathcal{G}(\boldsymbol{x}) = 14 - U(\boldsymbol{x}). \tag{55}$$

Fig. 8 illustrates one run of the proposed MSMR (adaptive scheme) in the planar truss example. This algorithm achieves its convergence with 6 iterations of multi-point enrichment. Informed by the MSMR, the expected gain G_{n+k} brought by the k-th best next sample $\boldsymbol{x}^{(n+k)}$ at each iteration is shown in Fig. 8(a). Due to the inferior accuracy of the initial Kriging model, the G_{n+k} equals 0 in the first iteration, which is hence invisible in the logarithmic scale. Then, unlike the first 3 iterations, the batches of new samples added in the latter iterations are grossly located in the vicinity of limit state, and the corresponding batch size K increases significantly, as shown in Figs. 8(b) and 8(c). Clearly, this behavior avoids selecting some 'useless' new samples in the initial stage and then reduces the error of failure probability estimate with fewer iterations in the latter stage. As a result, the IMP H_n is reduced remarkably, and the $\mu_{\widehat{P}_{f,n}}$ gradually converges to the reference value; see Figs. 8(d) and 8(e).



(a) the expected gain G_{n+k} of $\boldsymbol{x}^{(n+k)}$ at each iteration

(b) $\mathcal{G}$-function evaluations per iteration

(c) batch size K per iteration

(d) convergence of H_n

(e) convergence of $\mu_{\widehat{P}_{f,n}}$

Figure 8: The MSMR (adaptive scheme) for the truss example

Table 5 gives a comparison of different parallel (B)ALR methods for the planar truss example. In comparison with the existing AK-MCS, the ALR module in UQLab, and the PABQ, the proposed MSMR (prescribed scheme) needs much fewer iterations, whilst yielding better accuracy of $\mu_{\widehat{P}_f}$, i.e., smaller values of relative error $\delta_{\widehat{P}_f}$. Although the n_{eval} value needed by the existing look-ahead learning function IPMR is comparable to that of MSMR, the multi-point version of IPMR is still unavailable, failing to favor parallel computing by itself. In comparison with the prescribed scheme ($K = 20$), the adaptive scheme of MSMR needs much smaller value of n_{eval}, say 79.3%, but a slight increase of n_{iter}.

5.3. A lumped-mass shear frame under stochastic ground motion excitation

The third example tackles with a 10-story lumped-mass shear frame with random structural parameters subjected to fully nonstationary stochastic seismic excitation. Fig. 9(a) shows the basic geometric information

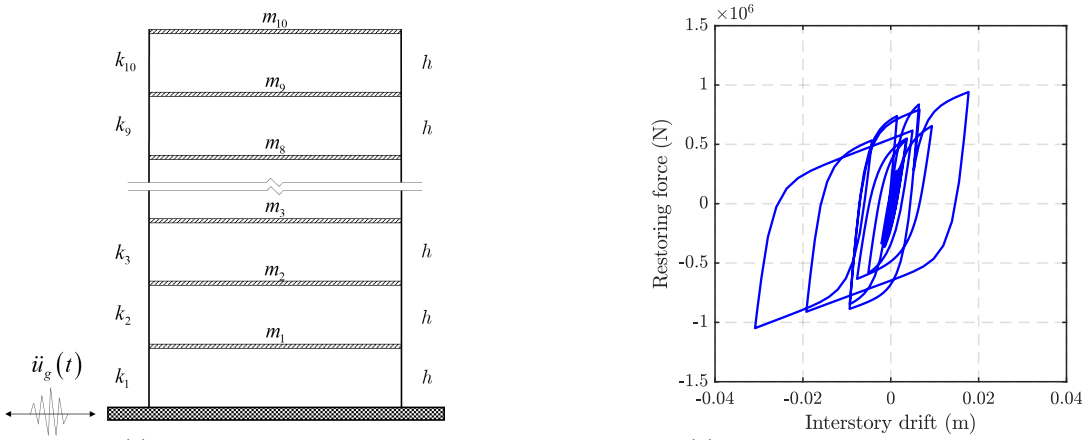
Table 5: Reliability results in the planar truss example

Method		$\mathbb{E}[n_{\text{iter}}]$	$\mathbb{E}[n_{\text{eval}}]$	$\mathbb{E}\left[\mu_{\widehat{P}_f}\right]$	$\text{CV}\left[\mu_{\widehat{P}_f}\right]$	$\mathbb{E}\left[\delta_{\widehat{P}_f}\right]$	References
MCS	-	-	2×10^6	3.45×10^{-5}	-	-	[12]
AK-MCS	$K = 1$	-	122	3.7×10^{-5}	-	-	[33]
	$K = 6$	28	174	3.4×10^{-5}	-	-	[33]
IPMR	$K = 1$	-	45.2	3.447×10^{-5}	-	-	[19]
ALR in UQLab	$K = 5$	13.7	74.5	3.519×10^{-5}	4.83 %	4.51 %	-
	$K = 10$	9.6	97	3.538×10^{-5}	4.24 %	4.17 %	-
	$K = 15$	7.1	102.5	3.558×10^{-5}	3.56 %	4.24 %	-
	$K = 20$	6.4	119	3.535×10^{-5}	5.14 %	4.39 %	-
PABQ	$K = 5$	21.3	111.5	3.125×10^{-5}	6.24 %	9.43 %	-
	$K = 10$	10.1	101	2.932×10^{-5}	2.69 %	15.01 %	-
	$K = 15$	7.9	113.5	3.065×10^{-5}	6.20 %	11.17 %	-
	$K = 20$	6.3	116	2.991×10^{-5}	4.16 %	13.31 %	-
MSMR	$K = 5$	8.1	46.5	3.442×10^{-5}	1.64 %	1.23 %	-
	$K = 10$	6.3	64	3.436×10^{-5}	1.91 %	1.62 %	-
	$K = 15$	5.1	72.5	3.438×10^{-5}	1.31 %	1.04 %	-
	$K = 20$	5	91	3.427×10^{-5}	1.14 %	1.13 %	-
	Adaptive	7.8	72.2	3.427×10^{-5}	1.59 %	1.45 %	-

of this frame, with the height of each floor 3600 mm. The equation of motion for this frame is given by [50]

$$\boldsymbol{M}\ddot{\boldsymbol{U}}(t) + \boldsymbol{C}\dot{\boldsymbol{U}}(t) + \boldsymbol{f}(U, \dot{U}, t) = -\boldsymbol{M}\boldsymbol{I}\ddot{u}_g(t), \tag{56}$$

where $\ddot{\boldsymbol{U}}(t)$, $\dot{\boldsymbol{U}}(t)$, and $\boldsymbol{U}(t)$ denote the 10×1 vectors of acceleration, velocity, and displacement of this system, respectively; $\boldsymbol{M}$, $\boldsymbol{C}$ and $\boldsymbol{f}$ denote the 10×10 mass matrix, the 10×10 damping matrix, and the 10×1 vector of restoring force of this system, respectively; $\boldsymbol{I}$ is a 10×1 vector with elements all 1; $\ddot{u}_g(t)$ represents the accelerogram of the earthquake excitation.



(a) geometric layout (b) hysteretic curve of the first floor

Figure 9: A 10-story lumped-mass shear frame

The mass matrix $\boldsymbol{M} = \text{diag}(m_1 \ldots, m_{10})$ is a diagonal matrix, with the lumped masses from bottom to top floor are taken as deterministic values: $m_1 = 3.478, m_2 = 3.225, m_3 = 2.887, m_4 = 2.667, m_5 =, \cdots, = m_{10} = 2.558\,(\times 10^5 \text{kg})$ [10]. The stiffness matrix $\boldsymbol{K}$ is assembled by the inter-story stiffness values from bottom to top floor. They are considered as independent random variables, with the statistical information listed in Table 6. Rayleigh damping is considered such that $\boldsymbol{C} = a\boldsymbol{M} + b\boldsymbol{K}$, with $a = 0.2904\text{s}^{-1}$ and $b = 0.0066\text{s}$. The inter-story relationship between restoring force and drift is described by the Bouc-Wen model [51], and the 13 related parameters are valued as: $\alpha_{\text{BW}} = 0.04, A_{\text{BW}} = 1, n_{\text{BW}} = 1, \beta_{\text{BW}} = 320, \gamma_{\text{BW}} = 150, \delta_{\upsilon_{\text{BW}}} = 2000, \delta_{\eta_{\text{BW}}} = 2000, p_{\text{BW}} = 1000, q_{\text{BW}} = 0.25, \psi_{\text{BW}} = 0.05, \delta_{\psi_{\text{BW}}} = 5, \lambda_{\text{BW}} = 0.5, \zeta_{\text{BW}} = 0.99$ [52]. Fig. 9(b) presents the typical hysteretic curve of the first floor, where strong non-linearity is observed.

A fully non-stationary stochastic ground motion model called the modulated filtered white-noise process (MFWNP) [53, 54, 55] is adopted to generate seismic excitation $\ddot{u}_g(t)$. It has the advantage of separating temporal and spectral non-stationary characteristics of the process. Specifically, the $\ddot{u}_g(t; \Omega)$ is expressed as [53]

$$\ddot{u}_g(t; \Omega) = q\,(t; \Omega) \left\{ \frac{1}{\sigma_h(t)} \int_0^t h\,[t - \tau, \boldsymbol{\lambda}(\tau; \Omega)]\, w(\tau) \mathrm{d}\tau \right\}, \tag{57}$$

Table 6: Initial inter-story stiffness from bottom to top floor ($\times 10^2$N/mm) [10]

Stiffness	K_1	K_2	K_3	K_4	$K_5 - K_{10}$
Distribution	Lognormal	Lognormal	Lognormal	Lognormal	Lognormal
Mean	1.962	1.875	1.758	1.754	1.662
COV	0.1	0.1	0.1	0.1	0.1

where $q(t;\Omega)$ is a time-modulation function; $w(\tau)$ is a Gaussian white-noise process; $h[t-\tau, \boldsymbol{\lambda}(\tau;\Omega)]$ is the impulse-response function of a time-varying filter with parameters $\boldsymbol{\lambda}(\tau;\Omega)$; $\sigma_h^2(t) = \int_0^t h^2[t-\tau, \boldsymbol{\lambda}(\tau;\Omega)]\, w(\tau)\mathrm{d}\tau$ denotes the variance of the filtered white-noise process. In this way, the temporal non-stationarity is totally determined by the modulation function $q(t;\Omega)$, while the spectral non-stationarity is defined by the time-variant filter $h[t-\lambda, \boldsymbol{\lambda}(\tau;\Omega)]$.

The 'Gamma' modulation function is adopted as [54]

$$q(t;\Omega) = \alpha_1 t^{\alpha_2 - 1} \exp\left(-\frac{t}{\alpha_3}\right), \tag{58}$$

where α_1 controls the intensity, α_2 controls the shape; α_3 controls the duration of ground motion. The three parameters can be determined from the expected Arias intensity I_a, the effective duration D_{5-95} (i.e., the time interval between the instants when 5% and 95% of I_a are reached), and the t_{mid} (i.e., the instant when 45% of I_a is reached). More details can refer to [54].

The impulse response function $h[t-\tau, \boldsymbol{\lambda}(\tau;\Omega)]$ of the time-variant filter takes the pseudo-acceleration response of a single-degree-of-freedom linear oscillator, that is, [54]

$$h[t-\tau, \boldsymbol{\lambda}(\tau;\Omega)] = \begin{cases} \frac{\omega_f(\tau)}{\sqrt{1-\zeta_f^2(\tau)}} \cdot \exp\left[-\zeta_f(\tau)\omega_f(\tau)(t-\tau)\right] \cdot \sin\left[\omega_f(\tau)\sqrt{1-\zeta_f^2(\tau)}(t-\tau)\right], & t \geq \tau, \\ 0, & \text{otherwise}, \end{cases} \tag{59}$$

where $\boldsymbol{\lambda}(\tau;\Omega) = [\omega_f(\tau), \zeta_f(\tau)]$, with the circular frequency $\omega_f(\tau)$ and damping ratio $\zeta_f(\tau)$ expressed as [54]

$$\omega_f(\tau) = \omega_{\mathrm{mid}} + \omega'(\tau - t_{\mathrm{mid}}), \tag{60}$$

$$\zeta_f(\tau) = \zeta_f, \tag{61}$$

where ω_{mid} is the frequency at the time instant t_{mid}; ω' is the rate of change of the frequency with time; ζ_f is a time-invariant damping ratio. Hence, $\Theta = \{I_a, D_{5-95}, t_{\mathrm{mid}}, \omega_{\mathrm{mid}}, \omega', \zeta_f\}$. Since the ω' has little effect on the structural responses, it is viewed as a constant [56]. Then, statistical information of the remaining 5 parameters is listed in Table 7.

Table 7: Statistical information of random parameters in the stochastic ground motion model [56]

Variables	Units	Description	Distribution	Parameters[a]	
I_a	m/s	Arias intensity	Lognormal	1.9	0.3
D_{5-95}	s	Time interval of 5% - 95% of I_a	Lognormal	2.21	0.23
t_{mid}	s	Time instant at 45% of I_a	Lognormal	1.698	0.21
$\omega_{\mathrm{mid}}/2\pi$	Hz	Filter frequncy at t_{mid}	Uniform	2.8	4.8
ζ_f	-	Filter damping ratio	Uniform	0.25	0.45

[a] Parameters: the mean and standard deviation of the natural logarithm of a lognormal variable; the lower and upper bounds of a uniform variable.

Finally, to secure zero residual velocity and displacement and yield reliable spectral response at long periods, the accelerogram $\ddot{u}_g(t;\Omega)$ in Eq. (57) is passed through a high-pass filter. Here, the Batterworth filter with the cutoff frequency $f_c = 0.5\pi$ is employed. Besides, an identical white-noise process is considered in the generation of seismic accelerograms. Fig. 10 presents the time histories of three typical seismic accelerograms generated by the MFWNP model, where both temporal and spectral non-stationary characteristics are well observed. Additionally, zero residual displacement is fairly observed at the end of seismic accelerogram.

To summarize, a total of 15 random variables are considered in this stochastic dynamic system, i.e., $\{K_1, K_2, \ldots, K_{10}, I_a, D_{5-95}, t_{\mathrm{mid}}, \omega_{\mathrm{mid}}, \zeta_f\}$. Then, of interest are the inter-story drifts $U_i(\boldsymbol{X}, t), i = 1, \ldots, 10$, between the i-th and $(i-1)$-th floor of this frame, which are computed with an in-house finite-element analysis code in MATLAB. The maximum allowable inter-story drift is taken as $72 = 3600 \times \frac{1}{50}$ mm. In this regard, the *system* failure probability P_f is defined as

$$P_f = \mathbb{P}\left(\bigcup_{i=1}^{10} (\exists t \in [0, 20\mathrm{s}], 72 - |U_i(\boldsymbol{X}, t)| \leq 0)\right). \tag{62}$$

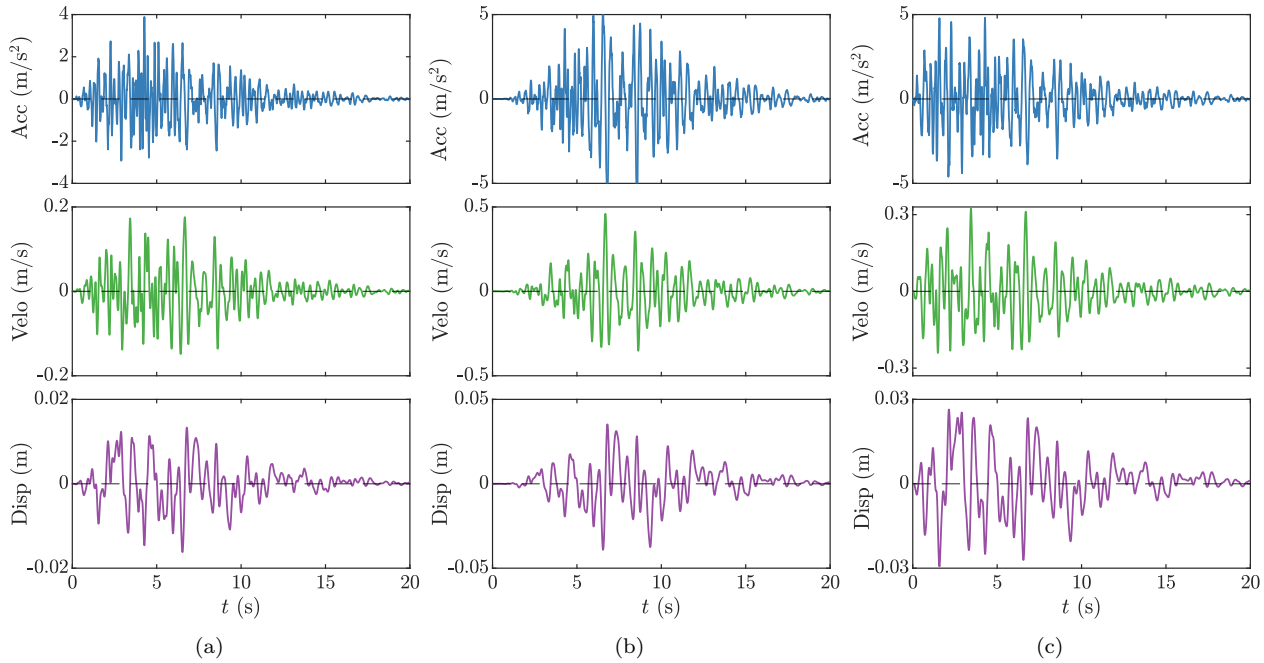


Figure 10: Three typical fully non-stationary seismic accelerograms

Then, Eq. (62) can be rewritten as the standard form in Eq. (1), with the $\mathcal{G}$-function expressed as [57]

$$\mathcal{G}(\boldsymbol{x}) = 72 - \max_{1\le i\le 10}\left(\max_{t\in[0,20\mathrm{s}]}|U_i(\boldsymbol{x},t)|\right). \tag{63}$$

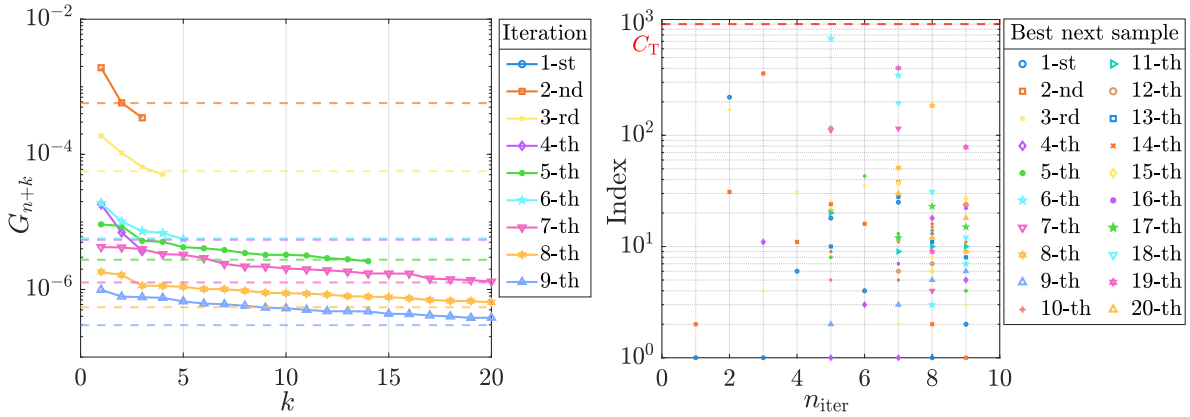
Fig. 11 shows a single run of the proposed MSMR (adaptive scheme) for the shear frame example. A total of 9 iterations of multi-point enrichment are needed when this algorithm is converged, as shown in Fig. 11(a). In comparison with the first 3 iterations, the batch sizes K in the latter iterations increase significantly and even reach the number of available CPU cores n_{core}; see Fig. 11(c). If more CPU cores are available, the batch sizes at the latter iterations will be further increased, and the average value of K is expected to rise to some extent. To justify the rationality of $C_{\mathrm{T}} = 10^3$ in the truncation of candidate pool (Eq. (42)), Fig. 11(b) shows the indexes of those best next samples selected from the $\mathcal{X}_{\mathrm{C_T}}$ in the 9 iterations, and most of the indexes are less than 500. Hence, it is reassuring to select the $\mathcal{X}_k^*$ from the $\mathcal{X}_{\mathrm{C_T}}$, rather than from the entire $\mathcal{X}_{\mathrm{C}}$. Finally, Figs. 11(d) and 11(e) show that when the hybrid convergence criterion is satisfied, a good agreement between $\mu_{\widehat{P}_{f,n}}$ and $\widehat{P}_f^{\mathrm{MCS}}$ is gained.

Table 8 gives a comparison of different parallel (B)ALR methods for the shear frame example, where the values of T_{c} are also listed for comparison. Both the ALR module in UQLab and the PABQ are not converged when the n_{eval} exceeds 300. By comparison, both the SUR and the MSMR achieve their convergence with fewer iterations. Since one run of finite element analysis of this shear frame is fast (<1 second), the T_{c} values of those reliability methods are mainly dominated by their running time of multi-point enrichment process (see Remark 2). Hence, although the n_{iter} needed by the SUR is comparable to that of the MSMR, the T_{c} of the SUR is far greater than that of the MSMR (see Remark 1).

5.4. Seismic reliability analysis of a planar reinforced concrete frame

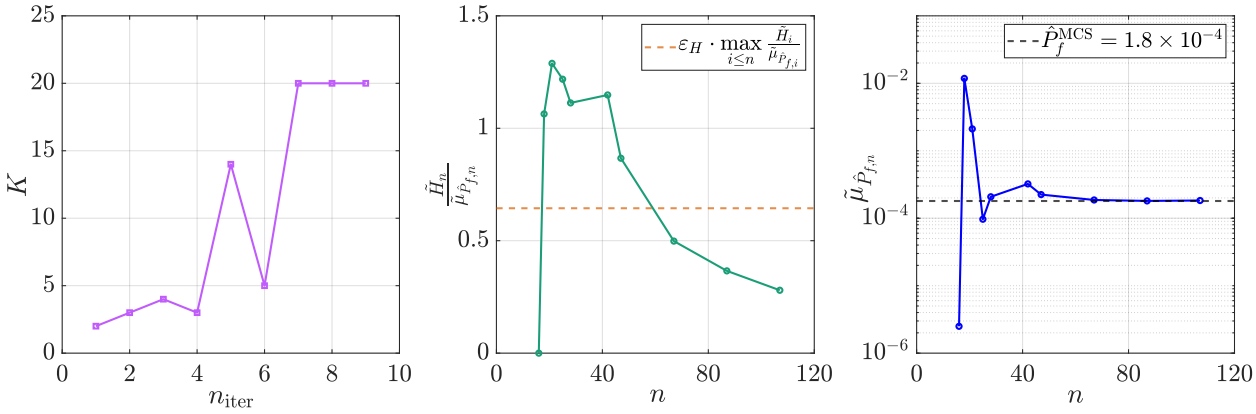
The final example addresses a 3-bay, 6-storey planar reinforced concrete frame under fully nonstationary stochastic seismic excitation. The basic geometric dimensions and reinforcement details of those members are sketched in Fig. 12. Finite element analysis of this structure is performed using the OpenSees software [58]. Both beams and columns are modeled by the force-based elements with fiber-discretized cross section. The uniaxial constitutive relationships of concrete and rebar are described by the Concrete-01 and Steel-01 models, respectively. The thickness of concrete slabs at each floor is 100 mm, and their weights are applied on those beams at each floor. Rayleigh damping is adopted, with the damping ratio 5%. The physical parameters associated with concrete and rebars are viewed as random variables, with their statistical information listed in Table 9. The MFWNP model (Section 5.3) is utilized to generate seismic accelerograms, and the associated parameters are given in Table 7. Hence, this dynamic system consists of a total of 16 random variables, i.e., $\{f_{\mathrm{cc}}, \epsilon_{\mathrm{cc}}, f_{\mathrm{cu}}, \epsilon_{\mathrm{cu}}, f_{\mathrm{c}}, \epsilon_{\mathrm{c}}, f_{\mathrm{u}}, \epsilon_{\mathrm{u}}, f_{\mathrm{y}}, E_0, b, I_a, D_{5-95}, t_{\mathrm{mid}}, \omega_{\mathrm{mid}}, \zeta_f\}$.

Figs. 13(a) and 13(b) illustrate the stress-strain curves of concrete and rebar at the end section of the leftmost bottom column, respectively. Then, Fig. 13(c) presents the hysteretic curve of the first floor. Obviously, strong nonlinearity in both material- and structure-levels are witnessed.



(a) the expected gain G_{n+k} of $\boldsymbol{x}^{(n+k)}$ per iteration

(b) the indexes of best next samples per iteration



(c) batch size K per iteration

(d) convergence of H_n

(e) convergence of $\mu_{\hat{P}_{f,n}}$

Figure 11: The MSMR (adaptive scheme) for the shear frame example

Table 8: Reliability results in the shear frame example

Method		$\mathbb{E}[n_{\mathrm{iter}}]$	$\mathbb{E}[n_{\mathrm{eval}}]$	$\mathbb{E}\left[\mu_{\hat{P}_f}\right]$	$\mathrm{CV}\left[\mu_{\hat{P}_f}\right]$	$\mathbb{E}\left[\delta_{\hat{P}_f}\right]$	$\mathbb{E}[T_{\mathrm{c}}]$ (s)
MCS	-	-	3×10^6	1.8×10^{-4}	4.31 %	-	5.02×10^4
ALR in UQLab	$K = 5$	> 58	> 300	1.832×10^{-4}	3.87 %	3.68 %	> 675.3
	$K = 10$	> 30	> 300	1.836×10^{-4}	3.83 %	3.16 %	> 415.6
	$K = 15$	> 20	> 300	1.788×10^{-4}	4.10 %	3.17 %	> 312.9
	$K = 20$	> 16	> 300	1.797×10^{-4}	4.05 %	1.79 %	> 283.4
PABQ	$K = 5$	> 58	> 300	1.214×10^{-4}	36.51 %	34.87 %	$> 2.80 \times 10^3$
	$K = 10$	> 30	> 300	1.516×10^{-4}	22.17 %	21.34 %	$> 1.53 \times 10^3$
	$K = 15$	> 20	> 300	1.781×10^{-4}	17.38 %	13.86 %	$> 1.17 \times 10^3$
	$K = 20$	> 16	> 300	1.783×10^{-4}	32.77 %	25.83 %	$> 7.89 \times 10^2$
SUR	$K = 5$	12.8	75	1.825×10^{-4}	5.67 %	4.06 %	2.51×10^3
	$K = 10$	8.1	87	1.816×10^{-4}	6.81 %	5.47 %	2.17×10^3
	$K = 15$	8	121	1.822×10^{-4}	1.97 %	1.66 %	3.49×10^3
	$K = 20$	7	136	1.822×10^{-4}	2.15 %	2.14 %	4.20×10^3
MSMR	$K = 5$	11.8	70	1.792×10^{-4}	3.60 %	2.11 %	254.8
	$K = 10$	8	86	1.816×10^{-4}	3.34 %	1.77 %	242.3
	$K = 15$	7.5	113.5	1.789×10^{-4}	4.66 %	4.02 %	305.3
	$K = 20$	7	136	1.793×10^{-4}	2.87 %	1.35 %	337.8
	Adaptive	9	107	1.786×10^{-4}	3.62 %	3.92 %	266.1

Of interest are the inter-story drifts $U_i(\boldsymbol{x}, t), i = 1, \ldots, 6$, between the i-th and $(i-1)$-th floor of this frame. The failure threshold is set as $66 = 3300 \times 1/50$ mm. In this way, the *system* failure probability is defined as

$$P_f = \mathbb{P}\left(\bigcup_{i=1}^{6} \left(\exists t \in [0, 20\mathrm{s}], 66 - |U_i(\boldsymbol{X}, t)| \le 0\right)\right), \tag{64}$$

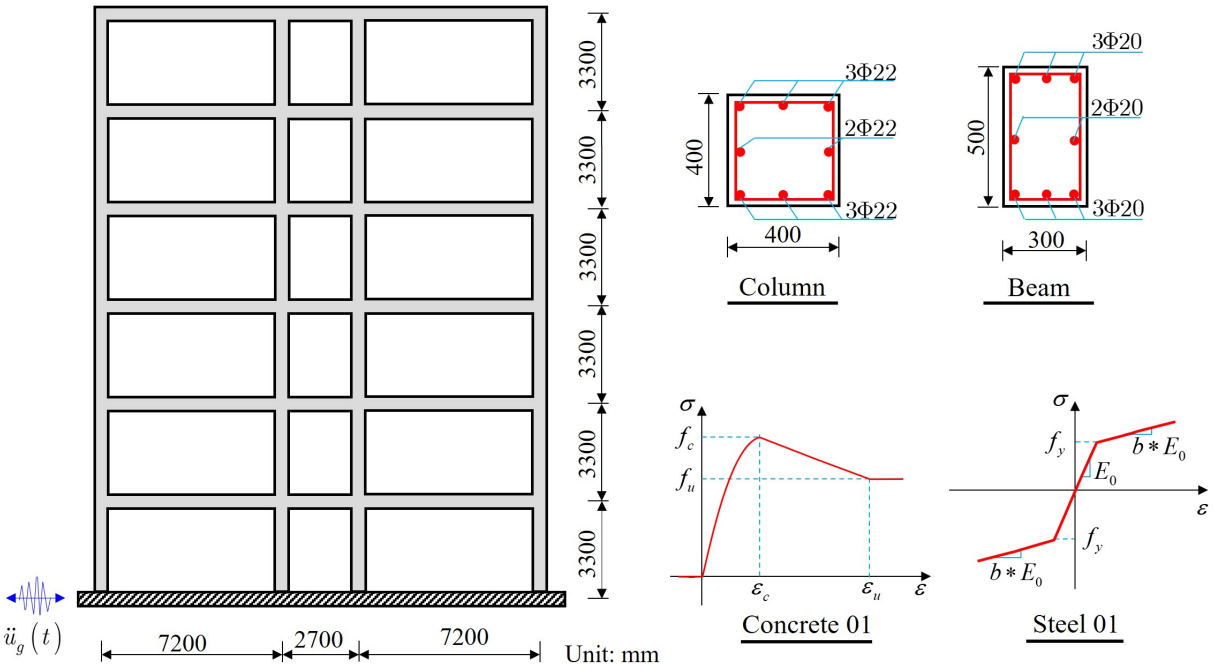


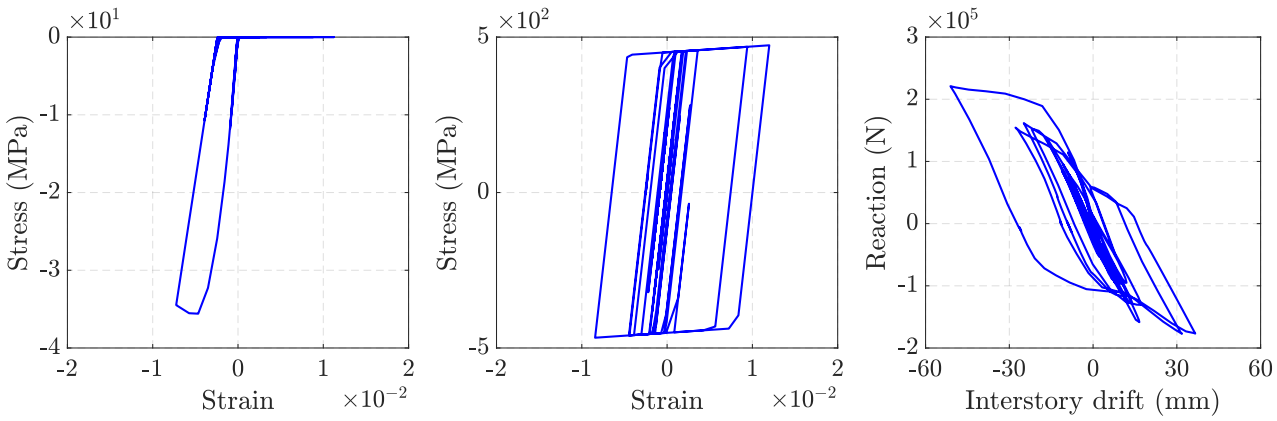
Figure 12: A planar reinforced concrete frame

Table 9: Statistical information of random variables in the reinforced concrete frame

Variable	Unit	Description	Distribution	Mean	COV
f_{cc}	MPa	Maximum strength of confined concrete	Lognormal	35	0.1
ϵ_{cc}	-	Strain at maximum strength of confined concrete	Lognormal	0.005	0.05
f_{cu}	MPa	Crushing strength of confined concrete	Lognormal	25	0.1
ϵ_{cu}	-	Strain at crushing strength of confined concrete	Lognormal	0.02	0.05
f_{c}	MPa	Maximum strength of unconfined concrete	Lognormal	27	0.1
ϵ_{c}	-	Strain at maximum strength of unconfined concrete	Lognormal	0.002	0.05
f_{u}	MPa	Crushing strength of unconfined concrete	Lognormal	10	0.1
ϵ_{u}	-	Strain at crushing strength of unconfined concrete	Lognormal	0.006	0.05
f_{y}	MPa	Yield strength of rebar	Lognormal	400	0.1
E_0	GPa	Initial Young's modulus of rebar	Lognormal	200	0.1
b	-	Strain-hardening ratio of rebar	Lognormal	0.007	0.05

which can be then rewritten as the standard form in Eq. (1), with the $\mathcal{G}$-function expressed as

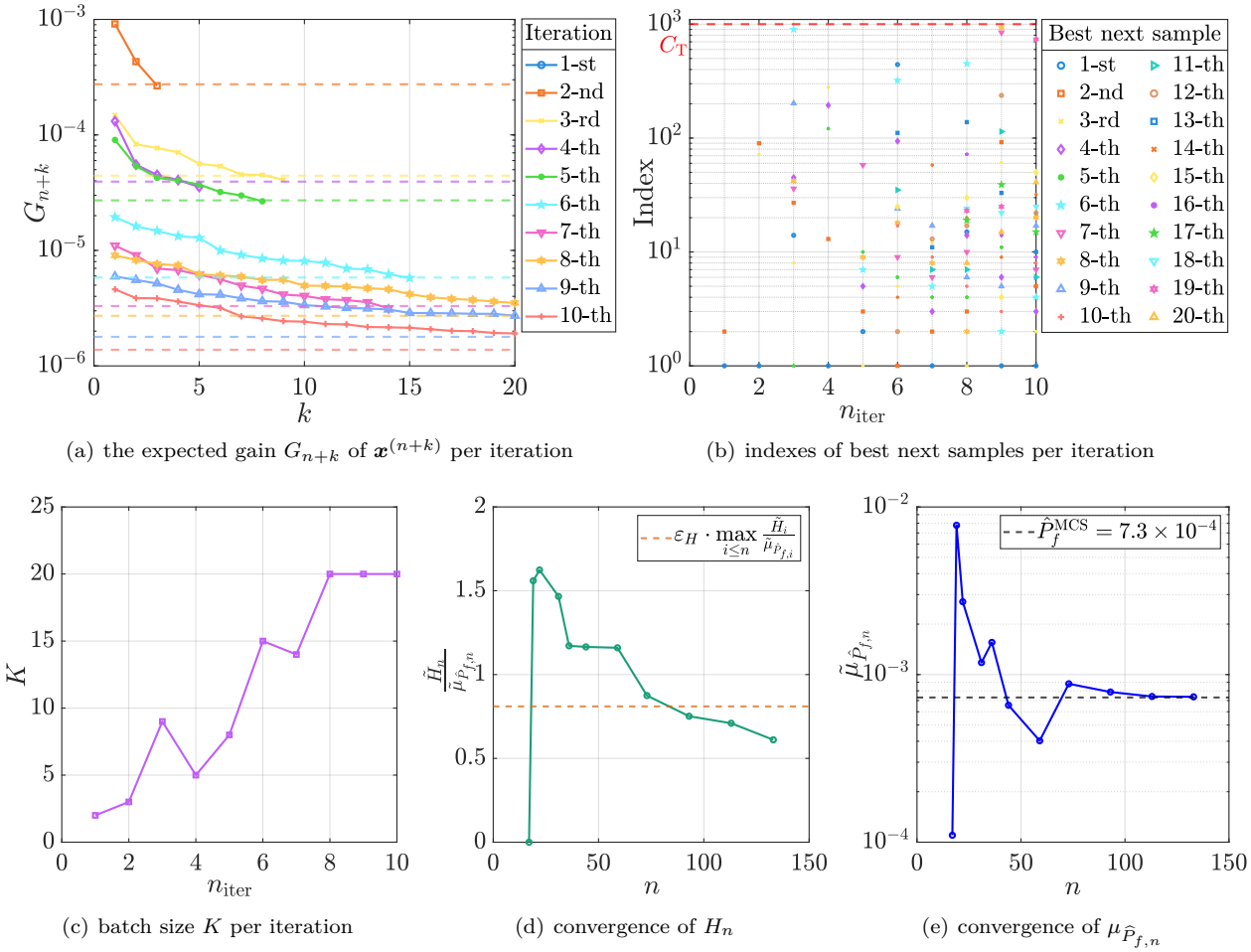
$$\mathcal{G}(\boldsymbol{x}) = 66 - \max_{1\le i\le 6}\left(\max_{t\in[0,20\mathrm{s}]}|U_i(\boldsymbol{x},t)|\right). \tag{65}$$



(a) stress-strain curve of concrete (b) stress-strain curve of rebar (c) hysteretic curve of the bottom column

Figure 13: Nonlinear behaviors of the reinforced concrete frame

Fig. 14 shows a single run of the proposed MSMR (adaptive scheme) for the reinforced concrete frame example. A total of 10 iterations of multi-point enrichment are needed by the convergence of this algorithm; see Fig. 14(a). After the first several iterations, the batch sizes K increase significantly and even reach the n_{core} in the latter iterations; see Fig. 14(c). Besides, Fig. 14(b) shows that the indexes of those best next points selected by the MSMR are mostly smaller than $C_{\text{T}} = 10^3$. Hence, the practice of truncating the $\mathcal{X}_{\text{C}}$ to the $\mathcal{X}_{\text{C}_{\text{T}}}$ in Eq. (42) is very favorable and efficient. Further, Figs. 14(d) and 14(e) indicate that the hybrid convergence criterion in Eq. (53) effectively secures the robust convergence of this algorithm.



(a) the expected gain G_{n+k} of $\boldsymbol{x}^{(n+k)}$ per iteration

(b) indexes of best next samples per iteration

(c) batch size K per iteration

(d) convergence of H_n

(e) convergence of $\mu_{\hat{P}_{f,n}}$

Figure 14: The MSMR (adaptive scheme) for the reinforced concrete frame example

Table 10 gives a comparison between different parallel (B)ALR methods for the reinforced concrete frame example. Both the ALR module in UQLab and the PABQ are not converged when the n_{eval} reaches 300. Meanwhile, they produce significant values of relative error $\delta_{\hat{P}_f}$. By contrast, both the SUR and the SMR achieve better accuracy of $\mu_{\hat{P}_f}$ with much fewer iterations, say only 50% - 65%. Further, in comparison with the SUR, the MSMR needs slightly fewer iterations and much less computational time, thanks to avoiding the presence of $\Phi_2(\cdot;\cdot,\cdot)$ in the expression of MSMR (see Remark 1). Further, in comparison with the prescribed scheme ($K = 20$), much smaller n_{iter} but slightly greater n_{iter} is observed in the adaptive scheme of MSMR.

Additionally, the advantage of MSMR reducing n_{iter} does not give rise to a very remarkable reduction of T_{c}, when compared with the ALR module in UQLab. This is due to that a single run of finite element analysis of this reinforced concrete frame takes around 10 seconds; then, the running time of MSMR per iteration is still greater than that of the ALR module in UQLab. Of course, when more computationally intensive reliability problems are considered, the superior advantage of MSMR in terms of T_{c} will be manifested readily.

5.5. Discussions

To probe into the difference between the prescribed scheme and the adaptive scheme in the MSMR, Fig. 15 summarizes the results of MSMR in the four examples. Then, two critical findings are discussed here.

- In the prescribed scheme, the n_{iter} decreases as the increasing of the prescribed batch size K, but the decreasing trend gradually slows down, as plotted as the solid lines in Fig. 15. This implies that when K is significant, some 'useless' new samples are prone to be added due to the inferior accuracy of Kriging

Table 10: Reliability results in the reinforced concrete frame example

Method		$\mathbb{E}[n_{\text{iter}}]$	$\mathbb{E}[n_{\text{eval}}]$	$\mathbb{E}\left[\mu_{\widehat{P}_f}\right]$	$\text{CV}\left[\mu_{\widehat{P}_f}\right]$	$\mathbb{E}\left[\delta_{\widehat{P}_f}\right]$	$\mathbb{E}[T_{\text{c}}]$ (s)
MCS	-	-	4×10^5	7.3×10^{-4}	5.85 %	-	6.14×10^5
ALR in UQLab	$K = 5$	> 58	> 300	6.423×10^{-4}	4.74 %	12.02 %	> 1173.7
	$K = 10$	> 30	> 300	6.728×10^{-4}	17.01 %	13.91 %	> 882.5
	$K = 15$	> 20	> 300	6.471×10^{-4}	9.33 %	11.36 %	> 892.7
	$K = 20$	> 16	> 300	6.246×10^{-4}	9.47 %	15.03 %	> 947.3
PABQ	$K = 5$	> 58	> 300	2.803×10^{-4}	60.37 %	61.61 %	$> 3.23 \times 10^3$
	$K = 10$	> 30	> 300	3.300×10^{-4}	50.96 %	56.10 %	$> 1.83 \times 10^3$
	$K = 15$	> 20	> 300	4.161×10^{-4}	37.15 %	43.07 %	$> 1.54 \times 10^3$
	$K = 20$	> 16	> 300	3.743×10^{-4}	46.08 %	51.81 %	$> 1.34 \times 10^3$
SUR	$K = 5$	29.9	161.5	7.545×10^{-4}	7.72 %	7.44 %	6.51×10^3
	$K = 10$	17.5	182	7.420×10^{-4}	9.14 %	6.89 %	6.89×10^3
	$K = 15$	13.7	207.5	7.737×10^{-4}	5.28 %	6.45 %	8.01×10^3
	$K = 20$	13.1	259	7.654×10^{-4}	3.41 %	5.21 %	9.42×10^3
MSMR	$K = 5$	21.3	118.5	7.487×10^{-4}	4.44 %	4.28 %	656.1
	$K = 10$	12.1	128	7.333×10^{-4}	3.63 %	2.84 %	635.7
	$K = 15$	11.7	177.5	7.372×10^{-4}	6.92 %	4.77 %	790.3
	$K = 20$	10.1	199	7.420×10^{-4}	3.06 %	3.08 %	902.6
	Adaptive	10.9	147.6	7.365×10^{-4}	5.30 %	3.45 %	626.5

in initial iterations. Besides, when the decrease of n_{iter} becomes stalled, the T_{c} of MSMR will grow to some extent due to the running time of multi-point enrichment process increasing with the batch size K; see Tables 8 and 10. Obviously, the turning point (from fast to slow decrease of n_{iter}) may offer a good balance between n_{call} (resp., computing resource consumption) and n_{iter} (resp., total computational time). However, such a 'balanced' value of K can not known a priori. Hence, it is unwise to blindly increase the K in the prescribed scheme.

- In the adaptive scheme, the average batch size K varies with the reliability problem at hand, as plotted as dashed lines in Fig. 15. Basically, if the $\mathcal{G}$-function is more complex, the average K will increase to some extent. More importantly, the average K in the adaptive scheme is very close to the so-called 'balanced' batch size in the prescribed scheme. For example, the 'balanced' K value in the prescribed scheme falls in the range between 10 and 15 in Example 4; then, the average K value in the adaptive scheme exactly lies in this range. Hence, the adaptive scheme itself could achieve a good balance between the computing resource consumption and the total computational time to some extent.

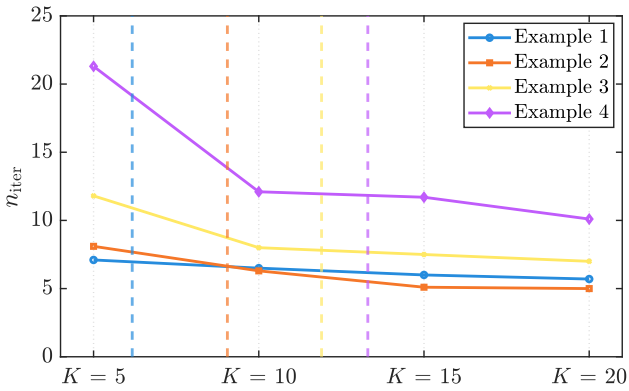


Figure 15: Results of MSMR (both prescribed and adaptive schemes) in the four examples

Besides, Tables 8 and 10 indicate that the proposed MSMR has significant advantage over the existing ALR module in UQLab and the PABQ in terms of n_{iter} and n_{eval}. However, the MSMR only shows slightly better, or comparable, performances of n_{iter} and n_{eval} to the existing SUR. The reasons are given as follows.

First, both the IMP H_n defined in the MSMR (Eq. (19)) and the uncertainty measure H_n^{S} defined in the SUR (Eq. (F.1)) are the upper bounds of the mean absolute deviation of failure probability estimated by Kriging. According to Fig. B.16, the IMP H_n is slightly greater than the H_n^{S}. Hence, comparable values of n_{iter} and n_{eval} are achieved in the MSMR and the SUR. However, in comparison with the SUR, the major contribution of MSMR lies in that it is analytically expressed as a function of $\Phi(\cdot)$, avoiding $\Phi_2(\cdot;\cdot,\cdot)$ which is

more computationally intensive, and thus entailing much less running time, as shown in Remark 1 and Fig. 4(b). Besides, the adaptive scheme in the MSMR could further reduce the n_{eval} to some extent.

Recall that the number of iteration n_{iter} and further the total computational time T_{c} are more concerned in the parallel active learning reliability analysis. Hence, considering the two aforementioned aspects, the proposed MSMR is very advantageous in terms of T_{c}, particularly when expensive-to-evaluate reliability problems are addressed.

6. Concluding remarks

An efficient parallel Bayesian active learning reliability method is proposed based on devising a multi-point look-ahead learning function MSMR. The MSMR is defined from Bayesian decision theory and is expressed as a double integral. Then, the MSMR-based multi-point enrichment process is efficiently conducted. The efficacy of the proposed approach is verified on four examples. Comparisons are made against several existing learning functions and parallelization strategies. Some concluding remarks are given as follows.

- The IMP is proved as an upper-bounding metric for the mean absolute deviation of Kriging-based failure probability estimation. Then, the resulting learning function MSMR enables explicitly quantifying the expected gain of adding a batch of new points on improving the accuracy of failure probability.
- The MSMR-based multi-point enrichment process is cheaply implemented by three compatible workarounds, i.e., the truncation of quadrature set, the truncation of candidate pool, and the heuristic treatment of maximizing the MSMR. Actually, these three workarounds provide a unified paradigm and can be readily customized to other look-ahead learning functions.
- Unlike those common parallel active learning reliability methods, the MSMR-based multi-point enrichment process is based on the learning function itself, without resorting to any additional parallelization strategy. Hence, it is more theoretically sound and numerically elegant.
- Both prescribed and adaptive schemes are devised in the MSMR to specify the batch size per iteration. Notably, the adaptive scheme could gain a fair trade-off between the computing resource assumption and the total computational time to some extent.

It is admitted that the running time of MSMR per iteration is slightly greater than that of performance function itself in those numerical examples. Hence, the advantage of MSMR is not very remarkable in terms of the total computational time. To highlight such advantage, the verification of MSMR in more expensive-to-run reliability problems will be investigated in the near future.

CRediT authorship contribution statement

Tong Zhou: Conceptualization, Formal analysis, Methodology, Software, Validation, Visualization, Writing - original draft. **Xujia Zhu**: Conceptualization, Methodology, Writing -review & editing. **Tong Guo**: Writing - review & editing. **You Dong**: Funding acquisition, Writing -review & editing. **Michael Beer**: Writing - review & editing.

Conflict of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability

Data will be made available on request.

Acknowledgements

The support of the National Natural Science Foundation of China (Grant Nos. 52408222, 52078448), the Research Grants Council of Hong Kong (Grant Nos. PolyU 15225722 and PolyU 15221521), and Université Paris-Saclay (Welcome Research Package No. WRP394) are highly appreciated.

Appendix A. Kriging

The Kriging assumes the response of $\mathcal{G}(\boldsymbol{x})$ as one realization of a Gaussian process, expressed as [59]

$$\mathcal{G}(\boldsymbol{x}) \approx \widehat{\mathcal{G}}_n(\boldsymbol{x}) = \boldsymbol{\beta}^\top \boldsymbol{f}(\boldsymbol{x}) + \sigma^2 W(\boldsymbol{x}), \tag{A.1}$$

where the trend function $\boldsymbol{\beta}^\top \boldsymbol{f}(\boldsymbol{x}) = \beta_0 + \sum_{l=1}^d \beta_l x_l$, with $\boldsymbol{f}(\boldsymbol{x}) = \{1, x_1, \ldots, x_d\}$ a set of basis functions and $\boldsymbol{\beta} = \{\beta_l, l = 0, \ldots, d\}$ a set of unknown coefficients. Then, σ^2 is the variance of Gaussian process; $W(\boldsymbol{x})$ is a zero-mean, unit-variance Gaussian process described by a correlation function $R(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta})$. The Matern-5/2 kernel function is adopted as [59]

$$R(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}) = \prod_{l=1}^d \left(1 + \sqrt{5}\frac{|x_l - x_l'|}{\theta_l} + \frac{5}{3}\left(\frac{|x_l - x_l'|}{\theta_l}\right)^2\right) \exp\left(-\sqrt{5}\frac{|x_l - x_l'|}{\theta_l}\right), \tag{A.2}$$

where the kernel parameters $\boldsymbol{\theta} = \{\theta_l > 0, l = 1, \ldots, d\}$.

When providing an ED $\mathcal{D}_n = \{\mathcal{X}_n, \mathcal{Y}_n\}$ of size n, both $\boldsymbol{\beta}$ and σ^2 are estimated as

$$\widehat{\boldsymbol{\beta}} = \left(\boldsymbol{F}^\top \boldsymbol{R}^{-1} \boldsymbol{F}\right)^{-1} \boldsymbol{F}^\top \boldsymbol{R}^{-1} \mathcal{Y}_n, \tag{A.3}$$

$$\widehat{\sigma}^2 = \frac{1}{n}(\mathcal{Y}_n - \boldsymbol{F}\boldsymbol{\beta})^\top \boldsymbol{R}^{-1}(\mathcal{Y}_n - \boldsymbol{F}\boldsymbol{\beta}), \tag{A.4}$$

where $\boldsymbol{F} \coloneqq \left(f_j\left(\boldsymbol{x}^{(i)}\right)\right)_{1 \le i \le n, 1 \le j \le d+1}$ is the information matrix; $\boldsymbol{R} \coloneqq \left(R\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}; \boldsymbol{\theta}\right)\right)_{1 \le i,j \le n}$ is the correlation matrix between all points in $\mathcal{X}_n$. Then, the $\boldsymbol{\theta}$ can be estimated as [59]

$$\widehat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta} \in \Theta}{\arg\min}\, \widehat{\sigma}^2 |\boldsymbol{R}|^{\frac{1}{n}}, \tag{A.5}$$

where Θ is the support of $\boldsymbol{\theta}$.

Finally, the Kriging prediction $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ conditional on $\mathcal{D}_n$ is still a Gaussian process, i.e., $\widehat{\mathcal{G}}_n(\boldsymbol{x}) \sim \mathcal{GP}(\mu_n(\cdot), c_n(\cdot,\cdot))$, with the posterior mean $\mu_n(\boldsymbol{x})$, variance $\sigma_n^2(\boldsymbol{x})$, and covariance $c_n(\boldsymbol{x}, \boldsymbol{x}')$ expressed as [59]

$$\mu_n(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x})^\top \widehat{\boldsymbol{\beta}} + \boldsymbol{r}(\boldsymbol{x})^\top \boldsymbol{R}^{-1}\left(\mathcal{Y}_n - \boldsymbol{F}\widehat{\boldsymbol{\beta}}\right), \tag{A.6}$$

$$\sigma_n^2(\boldsymbol{x}) = \widehat{\sigma}^2\left(1 - \boldsymbol{r}(\boldsymbol{x})^\top \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) + \boldsymbol{u}(\boldsymbol{x})^\top \left(\boldsymbol{F}^\top \boldsymbol{R}^{-1} \boldsymbol{F}\right)^{-1} \boldsymbol{u}(\boldsymbol{x})\right), \tag{A.7}$$

$$c_n(\boldsymbol{x}, \boldsymbol{x}') = \widehat{\sigma}^2\left(R(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{r}(\boldsymbol{x})^\top \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}') + \boldsymbol{u}(\boldsymbol{x})^\top \left(\boldsymbol{F}^\top \boldsymbol{R}^{-1} \boldsymbol{F}\right)^{-1} \boldsymbol{u}(\boldsymbol{x}')\right), \tag{A.8}$$

where $\boldsymbol{r}(\boldsymbol{x}) = \left[R\left(\boldsymbol{x}, \boldsymbol{x}^{(1)}\right), \ldots, R\left(\boldsymbol{x}, \boldsymbol{x}^{(n)}\right)\right]^\top$ and $\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{F}^\top \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x})$.

Appendix B. Proof of Proposition 1

Proof. First, the mean absolute deviation of $\widehat{P}_{f,n}$ satisfies the following inequality

$$\begin{aligned}
\mathbb{E}_n\left[\left|\widehat{P}_{f,n} - \mu_{\widehat{P}_{f,n}}\right|\right] &= \mathbb{E}_n\left[\left|\int_{\mathbb{X}} \widehat{\mathbb{1}}_n(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \int_{\mathbb{X}} \mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}\right|\right], \\
&= \mathbb{E}_n\left[\left|\int_{\mathbb{X}} \left(\widehat{\mathbb{1}}_n(\boldsymbol{x}) - \mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}\right|\right], \\
&\le \mathbb{E}_n\left[\int_{\mathbb{X}} \left|\widehat{\mathbb{1}}_n(\boldsymbol{x}) - \mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right| f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}\right], \\
&= \int_{\mathbb{X}} \mathbb{E}_n\left[\left|\widehat{\mathbb{1}}_n(\boldsymbol{x}) - \mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right|\right] f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \\
&= \mathbb{E}_{\boldsymbol{X}}\left[\mathbb{E}_n\left[\left|\widehat{\mathbb{1}}_n(\boldsymbol{x}) - \mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right|\right]\right].
\end{aligned} \tag{B.1}$$

Given that

$$\left|\widehat{\mathbb{1}}_n(\boldsymbol{x}) - \mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right| = \begin{cases} \left|1 - \mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right|, & \widehat{\mathcal{G}}_n(\boldsymbol{x}) \le 0 \\ \left|0 - \mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right|, & \text{otherwise} \end{cases} = \begin{cases} \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right), & \widehat{\mathcal{G}}_n(\boldsymbol{x}) \le 0, \\ \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right), & \text{otherwise}, \end{cases} \tag{B.2}$$

the $\mathbb{E}_n\left[\left|\widehat{\mathbb{1}}_n(\boldsymbol{x})-\mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right|\right]$ is formulated as

$$\begin{aligned}\mathbb{E}_n\left[\left|\widehat{\mathbb{1}}_n(\boldsymbol{x})-\mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right|\right] &= \int_{-\infty}^{0}\Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)f_{\widehat{Y}}(\widehat{y})\mathrm{d}\widehat{y}+\int_{0}^{+\infty}\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)f_{\widehat{Y}}(\widehat{y})\mathrm{d}\widehat{y},\\ &= \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)+\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right),\\ &= 2\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right),\\ &= 2\sigma^2_{\widehat{\mathbb{1}}_n}(\boldsymbol{x}).\end{aligned} \tag{B.3}$$

Substitute Eq. (B.3) into Eq. (B.1), resulting in

$$\mathbb{E}_n\left[\left|\widehat{P}_{f,n}-\mu_{\widehat{P}_{f,n}}\right|\right]\leq 2\mathbb{E}_{\boldsymbol{X}}\left[\sigma^2_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right]. \tag{B.4}$$

Based on Eq. (B.4), we look for the value β so that

$$\pi_n(\boldsymbol{x})=\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}+\beta\right)-\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}-\beta\right)\geq\sigma^2_{\widehat{\mathbb{1}}_n}(\boldsymbol{x}), \tag{B.5}$$

which would then provide an upper bound using H_n for the mean absolute deviation of $\widehat{P}_{f,n}$ in Eq. (B.1).

To this end, we study the properties of the function

$$h_\beta(t)=\Phi(t+\beta)-\Phi(t-\beta)-\Phi(t)(1-\Phi(t)), \tag{B.6}$$

where we have $\pi_n(\boldsymbol{x})-\sigma_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})=h_\beta\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)$.

Taking the derivative of Eq. (B.6) with respect to t gives

$$\begin{aligned}h'_\beta(t) &= \phi(t+\beta)-\phi(t-\beta)-\phi(t)\left(1-2\Phi(t)\right),\\ &= \frac{\exp(-t^2/2)}{\sqrt{2\pi}}\left(\exp(-\beta^2/2)\left(\exp(-\beta t)-\exp(\beta t)\right)-\left(1-2\Phi(t)\right)\right).\end{aligned} \tag{B.7}$$

Therefore, the sign of $h'_\beta(t)$ only depends on the sign of

$$J_\beta(t)=\exp(-\beta^2/2)\left(\exp(-\beta t)-\exp(\beta t)\right)-(1-2\Phi(t)). \tag{B.8}$$

To study the sign property of $J_\beta(t)$, we take its derivative with respect to t, i.e.,

$$J'_\beta(t)=2\phi(t)-\beta\exp(-\beta^2/2)\left(\exp(\beta t)+\exp(-\beta t)\right). \tag{B.9}$$

$\phi(t)$ is increasing for $t<0$ and decreasing for $t>0$, while $\exp(\beta t)+\exp(-\beta t)$ is decreasing for $t<0$ and increasing for $t>0$. Therefore, $J'_\beta(t)$ is increasing for $t<0$ and decreasing for $t>0$, and thus its maximum is reached at $t=0$, i.e.,

$$J'_\beta(t)\leq J'_\beta(0)=\sqrt{\frac{2}{\pi}}-2\beta\exp(-\beta^2/2). \tag{B.10}$$

Taking the derivative of $L(\beta)=2\beta\exp(-\beta^2/2)$, we have $L'(\beta)=2(1-\beta^2)\exp(-\beta^2/2)$. This implies that L is increasing for values $\beta<1$ and decreasing for $\beta>1$, so its maximum is give by $L(1)=2\exp(-0.5)>\sqrt{\frac{2}{\pi}}$. Since L is a continuous function and $L(0)=\lim_{\beta\to+\infty}L(t)=0<\sqrt{\frac{2}{\pi}}$, the intermediate value theorem guarantees that the equation $L(\beta)=\sqrt{\frac{2}{\pi}}$ has two roots $0<\beta_1<1<\beta_2$.

In the following development, we will study the properties of J'_β, J_β, and h_β for $\beta<\beta_1$. First, it is trivial that $J'_\beta(0)>0$. Furthermore, because $\lim_{t\to\infty}J'_\beta(t)=-\infty$ and J'_β is a continuous even function, $J'_\beta(0)=0$ has two roots, which are denoted as $-t_1$ and t_1 with $t_1>0$. Hence, J'_β and J_β have the following properties:

$$J'_\beta(t)\begin{cases}<0, \text{for } t<-t_1, & \Rightarrow J_\beta \text{ is decreasing in } (-\infty,-t_1);\\ >0, \text{for } -t_1<t<t_1, & \Rightarrow J_\beta \text{ is increasing in } (-t_1,t_1);\\ <0, \text{for } t>t_1, & \Rightarrow J_\beta \text{ is decreasing in } (t_1,+\infty).\end{cases} \tag{B.11}$$

In addition, given that $J_\beta(0) = 0$, J_β is a continuous odd function, and $\lim_{t\to-\infty} J_\beta(t) = +\infty$, $J_\beta(t) = 0$ has three roots $-t_2$, 0, and t_2 with $t_2 > 0$. As a result, we have

$$J_\beta(t) \begin{cases} > 0, \text{for } t < -t_2, & \Rightarrow h_\beta \text{ is increasing in } (-\infty, -t_2); \\ < 0, \text{for } -t_2 < t < 0, & \Rightarrow h_\beta \text{ is decreasing in } (-t_2, 0); \\ > 0, \text{for } 0 < t < t_2, & \Rightarrow h_\beta \text{ is increasing in } (0, t_2); \\ < 0, \text{for } t > t_2, & \Rightarrow h_\beta \text{ is decreasing in } (t_2, +\infty). \end{cases} \tag{B.12}$$

Therefore, $t = 0$ is the only local minimum of h_β. Because $\lim_{t\to\infty} h_\beta(t) = 0$, $h_\beta(t) \geq 0$ for all $t \in \mathbb{R}$ if and only if $h_\beta(0) \geq 0$. Following the definition of h_β in Eq. (B.6), we obtain

$$h_\beta(0) = \Phi(\beta) - \Phi(-\beta) - \frac{1}{4} = 2\Phi(\beta) - \frac{5}{4}. \tag{B.13}$$

Consequently, $h_\beta(0) \geq 0 \Leftrightarrow \Phi(\beta) \geq \frac{5}{8} \Leftrightarrow \beta \geq \beta_0 = \Phi^{-1}(\frac{5}{8})$. Note that these results are achieved for the case where $\beta < \beta_1$ (this is the condition upon which the derivation leads to the results). Numerically, it could be easily verified that $J'_{\beta_0}(0)$ in Eq. (B.10) is greater than 0 and $\beta_0 < 1$. As a result, $\beta_0 < \beta_1$, and thus $h_{\beta_0}(t) \geq 0$ for all $t \in \mathbb{R}$.

Because of the monotonicity of $h_\beta(t)$ (in β) and β_0 satisfying the required inequality with the equality being reached for $t = 0$, β_0 is the minimum admissible value of β, and for all $\beta \geq \beta_0$,

$$\pi_n(\boldsymbol{x}) - \sigma_{\widehat{\mathbb{1}}_n}(\boldsymbol{x}) = h_\beta\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) \geq 0, \tag{B.14}$$

which gives

$$H_n = \mathbb{E}_{\boldsymbol{X}}\left[\pi_n(\boldsymbol{x})\right] \geq \mathbb{E}_{\boldsymbol{X}}\left[\sigma^2_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right]. \tag{B.15}$$

Fig. B.16 shows a comparison between $\pi_n(\boldsymbol{x})$ and $\sigma^2_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})$ with $\beta = \Phi^{-1}(\frac{5}{8})$. Obviously, $\pi_n(\boldsymbol{x})$ is always greater than $\sigma^2_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})$, regardless of $\boldsymbol{x}$.

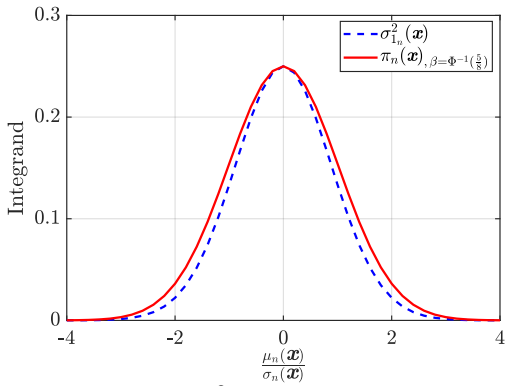


Figure B.16: Comparison of $\sigma^2_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})$ and the margin probability $\pi_n(\boldsymbol{x})$

Finally, combining Eqs. (B.4) and (B.15) proves the inequality in Proposition 1, i.e.,

$$\mathbb{E}_n\left[\left|\widehat{P}_{f,n} - \mu_{\widehat{P}_{f,n}}\right|\right] \leq 2H_n \tag{B.16}$$

□

Appendix C. Multi-point Kriging update formulas

A Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ can be trained from an ED $\mathcal{D}_n = \{\mathcal{X}_n, \mathcal{Y}_n\}$ of size n; see Appendix A. Then, denote $\mathcal{X}_k^+ = \{\boldsymbol{x}_+^{(1)}, \ldots, \boldsymbol{x}_+^{(k)}\}$ and $\mathcal{Y}_k^+ = \{y_+^{(1)}, \ldots, y_+^{(k)}\}$ as a batch of $k(\geq 1)$ new points and their $\mathcal{G}$-function responses. When the $\mathcal{D}_n$ is enriched with $\{\mathcal{X}_k^+, \mathcal{Y}_k^+\}$, the look-ahead posteriors of Kriging are expressed as [60]

$$\mu_{n+k}(\boldsymbol{x}) = \mu_n(\boldsymbol{x}) + c_n(\boldsymbol{x}, \mathcal{X}_k^+)^\top (\mathcal{C}_k^+)^{-1}\left(\mathcal{Y}_k^+ - \mu_n(\mathcal{X}_k^+)\right), \tag{C.1}$$

$$\sigma^2_{n+k}(\boldsymbol{x}) = \sigma^2_n(\boldsymbol{x}) - c_n(\boldsymbol{x}, \mathcal{X}_k^+)^\top (\mathcal{C}_k^+)^{-1} c_n(\boldsymbol{x}, \mathcal{X}_k^+), \tag{C.2}$$

$$c_{n+k}(\boldsymbol{x}, \boldsymbol{x}') = c_n(\boldsymbol{x}, \boldsymbol{x}') - c_n(\boldsymbol{x}, \mathcal{X}_k^+)^\top (\mathcal{C}_k^+)^{-1} c_n(\boldsymbol{x}', \mathcal{X}_k^+), \tag{C.3}$$

where $c_n\left(\boldsymbol{x}, \mathcal{X}_k^+\right) \coloneqq \left[c_n\left(\boldsymbol{x}, \boldsymbol{x}_+^{(1)}\right), \ldots, c_n\left(\boldsymbol{x}, \boldsymbol{x}_+^{(k)}\right)\right]^\top$ is a $k \times 1$ vector of covariances between $\boldsymbol{x}$ and $\mathcal{X}_k^+$; $\mathcal{C}_k^+ \coloneqq \left[c_n\left(\boldsymbol{x}_+^{(i)}, \boldsymbol{x}_+^{(j)}\right)\right]_{1 \leq i,j \leq k}$ is a $k \times k$ matrix of covariances at $\mathcal{X}_k^+$; $\mu_n(\mathcal{X}_k^+) \coloneqq \left[\mu_n\left(\boldsymbol{x}_+^{(1)}\right), \ldots, \mu_n\left(\boldsymbol{x}_+^{(k)}\right)\right]^\top$ is a $k \times 1$ vector of posterior means at $\mathcal{X}_k^+$.

Obviously, the look-ahead posterior of Kriging can be directly obtained from some matrix manipulations on the current posteriors, without any optimization of Kriging parameters. Therefore, unlike the retraining of Kriging, this update process is very computationally cheap.

Appendix D. Proof of Proposition 2

Proof. In the $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ in Eq. (28), the term $\mathbb{E}_{\boldsymbol{U}_k^+}\left[\Pi_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\right]$ is derived as

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{U}_k^+}\left[\Pi_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\right] \\
&= \int_{\mathbb{R}^k} \pi_{n+k}\left(\boldsymbol{x}; \mathcal{X}_k^+, \mathcal{U}_k^+\right) f_{\boldsymbol{U}_k^+}(\mathcal{U}_k^+) \mathrm{d}\mathcal{U}_k^+, \\
&= \underbrace{\int_{\mathbb{R}^k} \Phi\left(a_1(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \mathcal{U}_k^+\right) f_{\boldsymbol{U}_k^+}(\mathcal{U}_k^+) \mathrm{d}\mathcal{U}_k^+}_{\text{ⓐ}} - \underbrace{\int_{\mathbb{R}^k} \Phi\left(a_2(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \mathcal{U}_k^+\right) f_{\boldsymbol{U}_k^+}(\mathcal{U}_k^+) \mathrm{d}\mathcal{U}_k^+}_{\text{ⓑ}},
\end{aligned} \tag{D.1}$$

where $f_{\boldsymbol{U}_k^+}(\mathcal{U}_k^+)$ is the joint PDF of $\boldsymbol{U}_k^+$.

By the definition of $\Phi(\cdot)$, the integrand in the component ⓐ can be expressed as

$$\Phi\left(a_1(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \mathcal{U}_k^+\right) = \mathbb{P}\left(\Gamma \leq a_1(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \mathcal{U}_k^+\right) = \mathbb{P}\left(\Gamma \leq a_1(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \boldsymbol{U}_k^+ | \boldsymbol{U}_k^+ = \mathcal{U}_k^+\right), \tag{D.2}$$

where $\Gamma \sim \mathcal{N}(0,1)$ is a standard Gaussian variable independent of $\boldsymbol{U}_k^+$. Then, according to the law of total probability, the component ⓐ can be expressed as

$$\text{ⓐ} = \int_{\mathbb{R}^k} \mathbb{P}\left(\Gamma \leq a_1(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \boldsymbol{U}_k^+ | \boldsymbol{U}_k^+ = \mathcal{U}_k^+\right) f_{\boldsymbol{U}_k^+}(\mathcal{U}_k^+) \mathrm{d}\mathcal{U}_k^+ = \mathbb{P}\left(\Gamma \leq a_1(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \boldsymbol{U}_k^+\right), \tag{D.3}$$

then, the component ⓑ can be obtained similarly.

In this way, Eq. (D.1) is rewritten as

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{U}_k^+}\left[\Pi_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\right] &= \mathbb{P}\left(\Gamma \leq a_1(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \boldsymbol{U}_k^+\right) - \mathbb{P}\left(\Gamma \leq a_2(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})^\top \boldsymbol{U}_k^+\right), \\
&= \mathbb{P}\left(\Gamma - \boldsymbol{b}(\boldsymbol{x})^\top \boldsymbol{U}_k^+ \leq a_1(\boldsymbol{x})\right) - \mathbb{P}\left(\Gamma - \boldsymbol{b}(\boldsymbol{x})^\top \boldsymbol{U}_k^+ \leq a_2(\boldsymbol{x})\right), \\
&= \mathbb{P}\left(\Lambda \leq a_1(\boldsymbol{x})\right) - \mathbb{P}\left(\Lambda \leq a_2(\boldsymbol{x})\right),
\end{aligned} \tag{D.4}$$

where $\Lambda = \Gamma - \boldsymbol{b}(\boldsymbol{x})^\top \boldsymbol{U}_k^+$ is a Gaussian variable, with its mean μ_Λ and variance σ_Λ^2 expressed as

$$\begin{cases}
\mu_\Lambda = 0 - \boldsymbol{b}(\boldsymbol{x})^\top \boldsymbol{0} = 0, \\
\sigma_\Lambda^2 = 1 + \boldsymbol{b}(\boldsymbol{x})^\top \mathcal{C}_k^+ \boldsymbol{b}(\boldsymbol{x}) = 1 + \dfrac{-c_n(\boldsymbol{x}, \mathcal{X}_k^+)^\top \left(\mathcal{C}_k^+\right)^{-1}}{\sigma_{n+k}(\boldsymbol{x})} \mathcal{C}_k^+ \dfrac{-\left(\mathcal{C}_k^+\right)^{-1} c_n(\boldsymbol{x}, \mathcal{X}_k^+)}{\sigma_{n+k}(\boldsymbol{x})} = \dfrac{\sigma_n^2(\boldsymbol{x})}{\sigma_{n+k}^2(\boldsymbol{x})}.
\end{cases} \tag{D.5}$$

Hence, Eq. (D.4) can be further given as

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{U}_k^+}\left[\Pi_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)\right] &= \Phi\left(\frac{a_1(\boldsymbol{x}) - 0}{\frac{\sigma_n(\boldsymbol{x})}{\sigma_{n+k}(\boldsymbol{x})}}\right) - \Phi\left(\frac{a_2(\boldsymbol{x}) - 0}{\frac{\sigma_n(\boldsymbol{x})}{\sigma_{n+k}(\boldsymbol{x})}}\right), \\
&= \Phi\left(\frac{\beta\sigma_{n+k}(\boldsymbol{x}) - \mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) - \Phi\left(\frac{-\beta\sigma_{n+k}(\boldsymbol{x}) - \mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right), \\
&= \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} + \beta\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} - \beta\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right).
\end{aligned} \tag{D.6}$$

Combining Eqs. (D.6) and (28) proves the analytical expression of $I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+)$ in Eq. (30), i.e.,

$$I_{n+k}(\boldsymbol{x}; \mathcal{X}_k^+) = \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} + \beta\right) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} - \beta\right) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} + \beta\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) + \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})} - \beta\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right). \tag{D.7}$$

Furthermore, the partial derivative of $I_{n+k}(\boldsymbol{x};\mathcal{X}_k^+)$ with respect to $\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}$ is given as

$$\begin{aligned}\frac{\partial I_{n+k}(\boldsymbol{x};\mathcal{X}_k^+)}{\partial \frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}} &= -\phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}+\beta\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\beta+\phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}-\beta\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)(-\beta),\\ &= -\beta\left[\phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}+\beta\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)+\phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}-\beta\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right],\\ &< 0,\end{aligned} \tag{D.8}$$

where $\phi(\cdot)$ is the standard Gaussian PDF.

Eq. (D.8) implies that $I_{n+k}(\boldsymbol{x};\mathcal{X}_k^+)$ is a monotonously decreasing function with respect to $\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}$. When $\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}=0$, $I_{n+k}(\boldsymbol{x};\mathcal{X}_k^+)=\pi_n(\boldsymbol{x})$; when $\frac{\sigma_{n+k}(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}=1$, $I_{n+k}(\boldsymbol{x};\mathcal{X}_k^+)=0$. Hence, the lower and upper bounds of $I_{n+k}(\boldsymbol{x};\mathcal{X}_k^+)$ in Eq. (33) can be proved. □

Appendix E. Geometrical interpretation of $\rho_{n+k}(\boldsymbol{x},\mathcal{X}_k^+)$

Let $(\Omega,\mathcal{F},\mathbb{P})$ the space of random variables of finite variance. Then, assume that U (resp. V) is a random variable with mean μ_U (resp. μ_V) and variance σ_U^2 (resp. σ_V^2), and

$$\langle U,V\rangle=\mathbb{E}[UV] \tag{E.1}$$

is an inner product. In this way, the correlation coefficient between U and V can be expressed as

$$\mathrm{Corr}[U,V]=\frac{\mathbb{E}[(U-\mu_U)(V-\mu_V)]}{\sigma_U\sigma_V}=\frac{\langle U-\mu_U,V-\mu_V\rangle}{\|U-\mu_U\|\cdot\|V-\mu_V\|}=\cos(\theta), \tag{E.2}$$

where $\|\cdot\|=\sqrt{\langle\cdot,\cdot\rangle}$ denotes the Euclidean norm. Hence, the correlation coefficient of two random variables can be interpreted as the cosine of the angle between them after centralization.

Denote $Y=\widehat{\mathcal{G}}_n(\boldsymbol{x})$ and $\boldsymbol{Y}_k^+=\{\widehat{\mathcal{G}}_n(\boldsymbol{x}_+^{(1)}),\ldots,\widehat{\mathcal{G}}_n(\boldsymbol{x}_+^{(k)})\}$ as the Kriging prediction at point $\boldsymbol{x}$ and the vector of Kriging predictions at $\mathcal{X}_k^+$, respectively. Then, according to the Gaussian property of Kriging, there exists

$$\begin{bmatrix}\bar{Y}\\ \bar{\boldsymbol{Y}}_k^+\end{bmatrix}=\begin{bmatrix}Y-\mu_n(\boldsymbol{x})\\ \boldsymbol{Y}_k^+-\mu_n(\mathcal{X}_k^+)\end{bmatrix}\sim\mathcal{N}_{k+1}\left(\begin{bmatrix}0\\0\end{bmatrix},\begin{bmatrix}\sigma_n^2(\boldsymbol{x}) & c_n(\boldsymbol{x},\mathcal{X}_k^+)^\top\\ c_n(\boldsymbol{x},\mathcal{X}_k^+) & \mathcal{C}_k^+\end{bmatrix}\right). \tag{E.3}$$

Then, the projection of $\bar{Y}$ to the vector space spanned by $\bar{\boldsymbol{Y}}_k^+$ can be assumed as

$$proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})=\boldsymbol{\alpha}^\top\bar{\boldsymbol{Y}}_k^+, \tag{E.4}$$

where $\boldsymbol{\alpha}$ is a vector of projection coefficients. The orthogonality between $\bar{Y}-proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})$ and $\bar{\boldsymbol{Y}}_k^+$ is equivalent to their independence. Hence, $\bar{Y}-proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})$ is independent of (and further uncorrelated to) $\bar{\boldsymbol{Y}}_k^+$, i.e.,

$$\mathrm{Cov}\left[\bar{Y}-proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y}),\bar{\boldsymbol{Y}}_k^+\right]=0\Leftrightarrow\mathrm{Cov}\left[\bar{Y},\bar{\boldsymbol{Y}}_k^+\right]-\mathrm{Cov}\left[\boldsymbol{\alpha}^\top\bar{\boldsymbol{Y}}_k^+,\bar{\boldsymbol{Y}}_k^+\right]=0\Leftrightarrow c_n(\boldsymbol{x},\mathcal{X}_k^+)^\top-\boldsymbol{\alpha}^\top\mathcal{C}_k^+=0, \tag{E.5}$$

resulting in $\boldsymbol{\alpha}=(\mathcal{C}_k^+)^{-1}c_n(\boldsymbol{x},\mathcal{X}_k^+)$. In this way, the orthogonal projection of $\bar{Y}$ to the vector space spanned by $\bar{\boldsymbol{Y}}_k^+$ is expressed as $proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})=c_n(\boldsymbol{x},\mathcal{X}_k^+)^\top(\mathcal{C}_k^+)^{-1}\bar{\boldsymbol{Y}}_k^+$.

Further, the mean and variance of $proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})$ are expressed as

$$\begin{cases}\mathbb{E}\left[proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})\right]=\boldsymbol{\alpha}^\top\mathbb{E}\left[\bar{\boldsymbol{Y}}_k^+\right]=\boldsymbol{\alpha}^\top\mathbf{0}=0,\\ \mathrm{Var}\left[proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})\right]=\boldsymbol{\alpha}^\top\mathcal{C}_k^+\boldsymbol{\alpha}=c_n(\boldsymbol{x},\mathcal{X}_k^+)^\top(\mathcal{C}_k^+)^{-1}c_n(\boldsymbol{x},\mathcal{X}_k^+).\end{cases} \tag{E.6}$$

Besides, the covariance between $\bar{Y}$ and $proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})$ is expressed as

$$\mathrm{Cov}\left[\bar{Y},proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})\right]=c_n(\boldsymbol{x},\mathcal{X}_k^+)^\top(\mathcal{C}_k^+)^{-1}\mathrm{Cov}\left[\bar{Y},\bar{\boldsymbol{Y}}_k^+\right]=c_n(\boldsymbol{x},\mathcal{X}_k^+)^\top(\mathcal{C}_k^+)^{-1}c_n(\boldsymbol{x},\mathcal{X}_k^+). \tag{E.7}$$

Hence, the correlation coefficient between $\bar{Y}$ and $proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})$ is given by

$$\begin{aligned}\mathrm{Corr}\left[\bar{Y},proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})\right]&=\frac{\mathrm{Cov}\left[\bar{Y},proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})\right]}{\sqrt{\mathrm{Var}\left[\bar{Y}\right]}\sqrt{\mathrm{Var}\left[proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})\right]}}=\frac{\sqrt{c_n(\boldsymbol{x},\mathcal{X}_k^+)^\top\left(\mathcal{C}_k^+\right)^{-1}c_n(\boldsymbol{x},\mathcal{X}_k^+)}}{\sigma_n(\boldsymbol{x})},\\ &=\rho_{n+k}\left(\boldsymbol{x},\mathcal{X}_k^+\right)\geq 0.\end{aligned} \tag{E.8}$$

Since both $\bar{Y}$ and $proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})$ are centered variables, $\mathrm{Corr}\left[\bar{Y},proj_{\bar{\boldsymbol{Y}}_k^+}(\bar{Y})\right]$ is directly the cosine of the angle between them, as per Eq. (E.2). Hence, $\rho_{n+k}(\boldsymbol{x},\mathcal{X}_k^+)$ can be interpreted as the cosine of the angle between $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ and the vector space spanned by $\widehat{\mathcal{G}}_n(\mathcal{X}_k^+)$. Generally, the closer between $\boldsymbol{x}$ and $\mathcal{X}_k^+$ is, the greater $\rho_n(\boldsymbol{x},\mathcal{X}_k^+)$ is.

Appendix F. Stepwise uncertainty reduction (SUR)

An existing learning function SUR [25, 42] is outlined here for comparison. In the SUR, a measure of epistemic uncertainty of failure probability is expressed as

$$H_n^{\mathrm{S}} = \mathbb{E}_{\boldsymbol{X}}\left[\sigma^2_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})\right] = \mathbb{E}_{\boldsymbol{X}}\left[\Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right], \tag{F.1}$$

which is also an upper bound of the mean absolute deviation of $\widehat{P}_{f,n}$, as proved in Eq. (B.4).

Then, by quantifying the effect of adding a batch of k new samples on the reduction of H_n^{S}, the SUR is finally expressed as

$$\mathrm{SUR}_{n+k}\left(\mathcal{X}_k^+\right) = H_n^{\mathrm{S}} - \mathbb{E}_{\boldsymbol{X}}\left[\Phi_2\left(\begin{bmatrix}\frac{\mu_n(\boldsymbol{x})}{\sigma_{n+k}(\boldsymbol{x})}\\ -\frac{\mu_n(\boldsymbol{x})}{\sigma_{n+k}(\boldsymbol{x})}\end{bmatrix};\begin{bmatrix}0\\0\end{bmatrix},\begin{bmatrix}\frac{\sigma_n^2(\boldsymbol{x})}{\sigma_{n+k}^2(\boldsymbol{x})} & 1-\frac{\sigma_n^2(\boldsymbol{x})}{\sigma_{n+k}^2(\boldsymbol{x})}\\ 1-\frac{\sigma_n^2(\boldsymbol{x})}{\sigma_{n+k}^2(\boldsymbol{x})} & \frac{\sigma_n^2(\boldsymbol{x})}{\sigma_{n+k}^2(\boldsymbol{x})}\end{bmatrix}\right)\right]. \tag{F.2}$$

Finally, a batch of k best next points $\mathcal{X}_k^*$ can be selected as

$$\mathcal{X}_k^* = \underset{\mathcal{X}_k^+\in\mathcal{X}_{\mathrm{C}}}{\arg\max}\,\mathrm{SUR}_{n+k}\left(\mathcal{X}_k^+\right). \tag{F.3}$$

For more details about the SUR, refer to [25, 42].

References

[1] R. E. Melchers, A. T. Beck, Structural reliability analysis and prediction, John wiley & sons, 2018.
[2] C. Song, R. Kawai, Monte carlo and variance reduction methods for structural reliability analysis: A comprehensive review, Probabilistic Engineering Mechanics 73 (2023). doi:10.1016/j.probengmech.2023.103479.
[3] S. Engelund, R. Rackwitz, A benchmark study on importance sampling techniques in structural reliability, Structural Safety 12 (4) (1993) 255–276. doi:10.1016/0167-4730(93)90056-7.
[4] S.-K. Au, J. Beck, Estimation of small failure probabilities in high dimensions by subset simulation, Probabilistic Engineering Mechanics 16 (4) (2001) 263–277. doi:10.1016/S0266-8920(01)00019-4.
[5] A. M. Hasofer, N. C. Lind, Exact and invariant second-moment code format., ASCE Journal of Engineering Mechanics 100 (1) (1974) 111–121. doi:10.1061/JMCEA3.0001848.
[6] K. Breitung, Asymptotic approximations for multinormal integrals, Journal of Engineering Mechanics 110 (3) (1984) 357–366. doi:10.1061/(ASCE)0733-9399(1984)110:3(357).
[7] Y.-G. Zhao, Z.-H. Lu, Structural reliability: approaches from perspectives of statistical moments, John Wiley & Sons, 2021.
[8] T. Zhou, Y. Peng, Adaptive bayesian quadrature based statistical moments estimation for structural reliability analysis, Reliability Engineering and System Safety 198 (2020). doi:10.1016/j.ress.2020.106902.
[9] J. Li, J. Chen, The principle of preservation of probability and the generalized density evolution equation, Structural Safety 30 (1) (2008) 65–77. doi:10.1016/j.strusafe.2006.08.001.
[10] T. Zhou, Y. Peng, A two-stage point selection strategy for probability density evolution method-based reliability analysis, Structural and Multidisciplinary Optimization 65 (5) (2022). doi:10.1007/s00158-022-03244-7.
[11] R. G. Ghanem, P. D. Spanos, Stochastic finite elements: a spectral approach, Springer-Verlag Inc, 1991.
[12] G. Blatman, B. Sudret, An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis, Probabilistic Engineering Mechanics 25 (2) (2010) 183 197. doi:10.1016/j.probengmech.2009.10.003.
[13] B. Echard, N. Gayton, M. Lemaire, Ak-mcs: An active learning reliability method combining kriging and monte carlo simulation, Structural Safety 33 (2) (2011) 145–154. doi:10.1016/j.strusafe.2011.01.002.
[14] T. Zhou, Y. Peng, J. Li, An efficient reliability method combining adaptive global metamodel and probability density evolution method, Mechanical Systems and Signal Processing 131 (2019) 592–616. doi:10.1016/j.ymssp.2019.06.009.
[15] T. Zhou, Y. Peng, An active-learning reliability method based on support vector regression and cross validation, Computers and Structures 276 (2023). doi:10.1016/j.compstruc.2022.106943.
[16] T. Zhou, Y. Peng, Ensemble of metamodels-assisted probability density evolution method for structural reliability analysis, Reliability Engineering and System Safety 228 (2022). doi:10.1016/j.ress.2022.108778.
[17] R. Teixeira, M. Nogal, A. O'Connor, Adaptive approaches in metamodel-based reliability analysis: A review, Structural Safety 89 (2021). doi:10.1016/j.strusafe.2020.102019.
[18] M. Moustapha, S. Marelli, B. Sudret, Active learning for structural reliability: Survey, general framework and benchmark, Structural Safety 96 (2022). doi:10.1016/j.strusafe.2021.102174.
[19] T. Zhou, T. Guo, C. Dang, M. Beer, Bayesian reinforcement learning reliability analysis, Computer Methods in Applied Mechanics and Engineering 424 (2024). doi:10.1016/j.cma.2024.116902.
[20] B. Bichon, M. Eldred, L. Swiler, S. Mahadevan, J. McFarland, Efficient global reliability analysis for nonlinear implicit performance functions, AIAA Journal 46 (10) (2008) 2459–2468. doi:10.2514/1.34321.
[21] C. Dang, M. A. Valdebenito, M. G. Faes, P. Wei, M. Beer, Structural reliability analysis: A bayesian perspective, Structural Safety 99 (2022). doi:10.1016/j.strusafe.2022.102259.
[22] C. Dang, P. Wei, M. Faes, M. Valdebenito, M. Beer, Parallel adaptive bayesian quadrature for rare event estimation, Reliability Engineering and System Safety 225 (2022). doi:10.1016/j.ress.2022.108621.
[23] C. Dang, M. G. Faes, M. A. Valdebenito, P. Wei, M. Beer, Partially bayesian active learning cubature for structural reliability analysis with extremely small failure probabilities, Computer Methods in Applied Mechanics and Engineering 422 (2024). doi:10.1016/j.cma.2024.116828.
[24] C. Dang, M. Beer, Semi-bayesian active learning quadrature for estimating extremely low failure probabilities, Reliability Engineering and System Safety 246 (2024). doi:10.1016/j.ress.2024.110052.

[25] J. Bect, D. Ginsbourger, L. Li, V. Picheny, E. Vazquez, Sequential design of computer experiments for the estimation of a probability of failure, Statistics and Computing 22 (3) (2012) 773–793. doi:10.1007/s11222-011-9241-4.
[26] P. Pei, T. Zhou, One-step look-ahead policy for active learning reliability analysis, Reliability Engineering and System Safety 236 (2023). doi:10.1016/j.ress.2023.109312.
[27] T. Zhou, T. Guo, Y. Dong, Y. Peng, Polynomial chaos kriging-based structural reliability analysis via the expected margin volume reduction, Computers and Structures 287 (2023). doi:10.1016/j.compstruc.2023.107117.
[28] P. Wei, Y. Zheng, J. Fu, Y. Xu, W. Gao, An expected integrated error reduction function for accelerating bayesian active learning of failure probability, Reliability Engineering and System Safety 231 (2023). doi:10.1016/j.ress.2022.108971.
[29] C. Duhamel, C. Helbert, M. Munoz Zuniga, C. Prieur, D. Sinoquet, A sur version of the bichon criterion for excursion set estimation, Statistics and Computing 33 (2) (2023). doi:10.1007/s11222-023-10208-4.
[30] T. Zhou, T. Guo, Y. Dong, F. Yang, D. M. Frangopol, Look-ahead active learning reliability analysis based on stepwise margin reduction, Reliability Engineering and System Safety 243 (2024). doi:10.1016/j.ress.2023.109830.
[31] R. T. Haftka, D. Villanueva, A. Chaudhuri, Parallel surrogate-assisted global optimization with expensive functions a survey, Structural and Multidisciplinary Optimization 54 (1) (2016) 3 13. doi:10.1007/s00158-016-1432-3.
[32] T. Zhou, T. Guo, C. Dang, L. Jia, Y. Dong, Parallel active learning reliability analysis: A multi-point look-ahead paradigm, Computer Methods in Applied Mechanics and Engineering 434 (2025) 117524. doi:https://doi.org/10.1016/j.cma.2024.117524.
[33] R. Schöbi, B. Sudret, S. Marelli, Rare event estimation using polynomial-chaos kriging, ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering 3 (2) (2017). doi:10.1061/AJRUA6.0000870.
[34] Z. Wen, H. Pei, H. Liu, Z. Yue, A sequential kriging reliability analysis method with characteristics of adaptive sampling regions and parallelizability, Reliability Engineering and System Safety 153 (2016) 170 179. doi:10.1016/j.ress.2016.05.002.
[35] H. Zhan, N.-C. Xiao, Y. Ji, An adaptive parallel learning dependent kriging model for small failure probability problems, Reliability Engineering and System Safety 222 (2022). doi:10.1016/j.ress.2022.108403.
[36] Z. Chen, G. Li, J. He, Z. Yang, J. Wang, A new parallel adaptive structural reliability analysis method based on importance sampling and k-medoids clustering, Reliability Engineering and System Safety 218 (2022). doi:10.1016/j.ress.2021.108124.
[37] G. Li, T. Wang, Z. Chen, J. He, X. Wang, X. Du, Rbik-ss: A parallel adaptive structural reliability analysis method for rare failure events, Reliability Engineering and System Safety 239 (2023). doi:10.1016/j.ress.2023.109513.
[38] D. Zhan, J. Qian, Y. Cheng, Pseudo expected improvement criterion for parallel ego algorithm, Journal of Global Optimization 68 (3) (2017) 641 662. doi:10.1007/s10898-016-0484-7.
[39] F. A. Viana, R. T. Haftka, L. T. Watson, Sequential sampling for contour estimation with concurrent function evaluations, Structural and Multidisciplinary Optimization 45 (4) (2012) 615 618. doi:10.1007/s00158-011-0733-9.
[40] F. Yang, R. Kang, Q. Liu, C. Shen, R. Du, F. Zhang, A new active learning method for reliability analysis based on local optimization and adaptive parallelization strategy, Probabilistic Engineering Mechanics 75 (2024). doi:10.1016/j.probengmech.2023.103572.
[41] Y. Meng, D. Zhang, B. Shi, D. Wang, F. Wang, An active learning kriging model with approximating parallel strategy for structural reliability analysis, Reliability Engineering and System Safety 247 (2024). doi:10.1016/j.ress.2024.110098.
[42] C. Chevalier, D. Ginsbourger, J. Bect, E. Vazquez, V. Picheny, Y. Richet, Fast parallel kriging-based stepwise uncertainty reduction with application to the identification of an excursion set, Technometrics 56 (4) (2014) 455–465. doi:10.1080/00401706.2013.860918.
[43] C. Agrell, K. Dahl, Sequential bayesian optimal experimental design for structural reliability analysis, Statistics and Computing 31 (3) (2021). doi:10.1007/s11222-021-10000-2.
[44] C. Dimitrakakis, R. Ortner, Decision making under uncertainty and reinforcement learning, Springer, 2018.
[45] V. Dubourg, B. Sudret, J.-M. Bourinet, Reliability-based design optimization using kriging surrogates and subset simulation, Structural and Multidisciplinary Optimization 44 (5) (2011) 673 690. doi:10.1007/s00158-011-0653-8.
[46] Q. Pan, D. Dias, An efficient reliability method combining adaptive support vector machine and monte carlo simulation, Structural Safety 67 (2017) 85–95. doi:10.1016/j.strusafe.2017.04.006.
[47] Y.-Z. Ma, M. Liu, H. Nan, H.-S. Li, Z.-Z. Zhao, A novel hybrid adaptive scheme for kriging-based reliability estimation a comparative study, Applied Mathematical Modelling 108 (2022) 1–26. doi:10.1016/j.apm.2022.03.015.
[48] J. Wang, Z. Cao, G. Xu, J. Yang, A. Kareem, An adaptive kriging method based on k-means clustering and sampling in n-ball for structural reliability analysis, Engineering Computations (Swansea, Wales) 40 (2) (2023) 378 410. doi:10.1108/EC-12-2021-0705.
[49] Z. Zhao, Z.-H. Lu, Y.-G. Zhao, P-ak-mcs: Parallel ak-mcs method for structural reliability analysis, Probabilistic Engineering Mechanics 75 (2024). doi:10.1016/j.probengmech.2023.103573.
[50] T. Zhou, Y. Peng, A new active-learning function for adaptive polynomial-chaos kriging probability density evolution method, Applied Mathematical Modelling 106 (2022) 86–99. doi:10.1016/j.apm.2022.01.030.
[51] F. Ma, H. Zhang, A. Bockstedte, G. Foliente, P. Paevere, Parameter analysis of the differential model of hysteresis, Journal of Applied Mechanics, Transactions ASME 71 (3) (2004) 342–349. doi:10.1115/1.1668082.
[52] T. Zhou, S. Marelli, B. Sudret, Y. Peng, Ak-pdemi: A failure-informed enrichment algorithm for improving the ak-pdem in reliability analysis, Mechanical Systems and Signal Processing 180 (2022). doi:10.1016/j.ymssp.2022.109435.
[53] S. Rezaeian, A. Der Kiureghian, A stochastic ground motion model with separable temporal and spectral nonstationarities, Earthquake Engineering and Structural Dynamics 37 (13) (2008) 1565 1584. doi:10.1002/eqe.831.
[54] S. Rezaeian, A. D. Kiureghian, Simulation of synthetic ground motions for specified earthquake and site characteristics, Earthquake Engineering and Structural Dynamics 39 (10) (2010) 1155 1180. doi:10.1002/eqe.997.
[55] T. Zhou, A.-Q. Li, Stochastic modeling and synthesis of near-fault forward-directivity ground motions, KSCE Journal of Civil Engineering 24 (2020) 483–498. doi:10.1007/s12205-020-0390-x.
[56] C. Nardin, O. S. Bursi, F. Paolacci, A. Pavese, G. Quinci, Experimental performance of a multi-storey braced frame structure with non-structural industrial components subjected to synthetic ground motions, Earthquake Engineering and Structural Dynamics 51 (9) (2022) 2113 2136. doi:10.1002/eqe.3656.
[57] T. Zhou, T. Guo, Y. Dong, Y. Peng, Structural reliability analysis based on probability density evolution method and stepwise truncated variance reduction, Probabilistic Engineering Mechanics 75 (2024). doi:10.1016/j.probengmech.2024.103580.
[58] F. McKenna, M. Scott, G. Fenves, Nonlinear finite-element analysis software architecture using object composition, Journal of Computing in Civil Engineering 24 (1) (2010) 95–107. doi:10.1061/(ASCE)CP.1943-5487.0000002.
[59] C. Lataniotis, S. Marelli, B. Sudret, The gaussian process modeling module in uqlab, Journal of Soft Computing in Civil Engineering 2 (3) (2018) 91 116. doi:10.22115/SCCE.2018.129323.1062.
[60] C. Chevalier, D. Ginsbourger, X. Emery, Corrected kriging update formulae for batch-sequential data assimilation, Lecture Notes in Earth System Sciences 0 (2014) 119–122. doi:10.1007/978-3-642-32408-6_29.