Highlights

Bayesian reinforcement learning reliability analysis

Tong Zhou, Tong Guo, Chao Dang, Michael Beer

- A reliability method is proposed from the reinforcement learning perspective.
- Sequential experimental design is interpreted as a finite-horizon Markov decision process (MDP).
- Reward function in the MDP is defined in terms of the integrated probability of misclassification.
- A learning function called integrated probability of misclassification reduction (IPMR) is proposed.
- Fast IPMR-based sequential experimental design is conducted by three effective workarounds.

Bayesian reinforcement learning reliability analysis

Tong Zhou[a], Tong Guo[b,*], Chao Dang[c], Michael Beer[c,d,e]

[a]*Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hong Kong, China*
[b]*School of Civil Engineering, Southeast University, Nanjing 211189, China*
[c]*Institute for Risk and Reliability, Leibniz University Hannover, Hannover, 30167, Germany*
[d]*Institute for Risk and Reliability, University of Liverpool, Liverpool, L69 7ZF, United Kingdom*
[e]*International Joint Research Center for Resilient Infrastructure & International Joint Research Center for Engineering Reliability and Stochastic Mechanics, Tongji University, Shanghai, 200092, China*

Abstract

A Bayesian reinforcement learning reliability method that combines Bayesian inference for the failure probability estimation and reinforcement learning-guided sequential experimental design is proposed. The reliability-oriented sequential experimental design is framed as a finite-horizon Markov decision process (MDP), with the associated utility function defined by a measure of epistemic uncertainty about Kriging-estimated failure probability, referred to as integrated probability of misclassification (IPM). On this basis, a one-step Bayes optimal learning function termed integrated probability of misclassification reduction (IPMR), along with a compatible convergence criterion, is defined. Three effective strategies are implemented to accelerate IPMR-informed sequential experimental design: (i) Analytical derivation of the inner expectation in IPMR, simplifying it to a single expectation. (ii) Substitution of IPMR with its upper bound IPMR^{U} to avoid element-wise computation of its integrand. (iii) Rational pruning of both quadrature set and candidate pool in IPMR^{U} to alleviate computer memory constraint. The efficacy of the proposed approach is demonstrated on two benchmark examples and two numerical examples. Results indicate that IPMR^{U} facilitates a much more rapid reduction of IPM compared to other existing learning functions, while requiring much less computational time than IPMR itself. Therefore, the proposed reliability method offers a substantial advantage in both computational efficiency and accuracy, especially in complex dynamic reliability problems.

Keywords: Reinforcement learning, One-step Bayes optimal learning function, Integrated probability of misclassification reduction, Bayesian inference, Reliability analysis

Nomenclature

AK-SS	adaptive Krigng-subset simulation
ALR	Active learning reliability
BALR	Bayesian active learning reliability
BRLR	Bayesian reinforcement learning reliability
CDF	cumulative distribution function
COV	coefficient of variation
ED	experimental design
EIER	expected integrated error reduction
IPM	integrated probability of misclassification
IPMR	integrated probability of misclassification reduction
IS	importance sampling
LIF	least improvement function
MCS	Monte Carlo simulation
MDP	Markov decision process
PDF	probability density function
PM	probability of misclassification
REIF	reliability-based expected improvement function
RLCB	reliability-based lower confidence bounding
SS	subset simulation
SUR	stepwise uncertainty reduction
VAIS	variance-amplified importance sampling

1. Introduction

Engineering systems are inherently subject to uncertainties in the physical properties, external loads and operating conditions. Structural reliability analysis seeks to quantify the impact of these uncertainties by

*Corresponding author.
Email addresses: tong.ce.zhou@outlook.com (Tong Zhou), guotong@seu.edu.cn (Tong Guo), chao.dang@irz.uni-hannover.de (Chao Dang), beer@irz.uni-hannover.de (Michael Beer)

computing the associated probability of failure of those systems concerning some predefined limit states, and it is paramount to the design, assessment and maintenance of complex engineering systems [1].

In a probabilistic framework, the randomness associated with an engineering system is typically represented by a d-dimensional vector of random variables $\boldsymbol{Z} = \{Z_1, \ldots, Z_d\} \subset \mathbb{Z} \in \mathbb{R}^d$, with a known joint probability density function (PDF) $f_{\boldsymbol{Z}}(\boldsymbol{z})$. Then, the state of system is encoded by a performance function $y = G(\boldsymbol{z})$, relying on the output of expensive-to-evaluate computational models, such as high-fidelity finite-element models. Conventionally, the system is deemed in a failed configuration when $G(\boldsymbol{z}) < 0$, and the boundary between safe and failure domains is known as the limit state surface. The failure probability P_f is then defined as [1]

$$P_f = \mathbb{P}\left(G(\boldsymbol{Z}) \leq 0\right) = \int_{\mathbb{Z}} \mathbb{1}\left(\boldsymbol{z}\right) f_{\boldsymbol{Z}}(\boldsymbol{z}) \mathrm{d}z \tag{1}$$

where $\mathbb{P}\left(\cdot\right)$ denotes the probability operator; and $\mathbb{1}\left(\cdot\right)$ is the failure indicator function given by

$$\mathbb{1}\left(\boldsymbol{z}\right) = \begin{cases} 1, & G(\boldsymbol{z}) \leq 0 \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

In most real-world scenarios, analytical solution to Eq. (1) is generally intractable. Consequently, researchers have developed a host of numerical reliability analysis methods in the literature, which often entail the repeated evaluation of performance function. A reliability method will be computationally efficient if it only requires a minimal number of performance function evaluations. Basically, existing reliability methods can be categorized into four groups. (i) Analytical approximation methods, e.g., first- and second-order reliability methods [2]; (ii) Sampling methods, e.g., Monte Carlo simulation (MCS) [3], importance sampling (IS) [4], directional sampling [5], subset simulation (SS) [6] and line sampling [7]; (iii) Numerical integration methods, e.g., moment methods [8] and probability density evolution method [9, 10]; (iv) Active learning reliability (ALR) methods, where two seminal contributions are efficient global reliability analysis [11] and adaptive Kriging Monte Carlo simulation [12].

In the past decade, the ALR methods [13] have gained increasing popularity due to higher efficiency compared to the aforementioned categories. The core of the ALR methods lies in replacing the computationally-expensive performance function with a well-calibrated surrogate model, which is inexpensive to evaluate. In particular, the calibration of surrogate model and the estimation of failure probability are iteratively performed within the sequential experimental design process. Then, the accuracy of the surrogate-estimated failure probability is progressively improved until a relevant convergence criterion is met. For a more comprehensive review, interested readers may refer to [13, 14].

Commonly-used surrogate models in this context include Kriging [12], support vector regression [15], polynomial chaos expansion [16, 17], radial basis function [18], and ensemble of metamodels [19]. Kriging is arguably the most popular one, due to its Bayesian interpretation and uncertainty quantification capability. This advantage makes it well-suited for sequential experimental design. Therefore, we restrict our attention to Kriging. The combination of Kriging with MCS [12, 20], IS [21], SS [22], or probability density evolution method [23, 24] have been extensively explored in the literature. Notably, a Bayesian inference framework for Kriging-based failure probability estimation was recently developed in [25, 26, 27], where both the posterior mean and (upper-bound or exact) posterior variance of Kriging-estimated failure probability are derived. The former is considered as a desired estimate of failure probability, while the latter serves as a measure of epistemic uncertainty about Kriging-estimated failure probability, due to the limited training samples [28, 29]. This epistemic uncertainty measure can be reduced by sequentially adding informative training samples, and the sequential experimental design is terminated when this uncertainty measure falls below a predefined tolerance. This sub-category is collectively referred to as Bayesian active learning reliability (BALR) method.

Sequential experimental design [30] is arguably the most distinctive feature of the ALR methods, involving a sequence of decisions on where to make the next performance function evaluation(s) based on the available data. This sequential process is often achieved by specifying a learning function that assigns a score to each candidate point in the input space commensurate with its propensity for aiding the reliability analysis task. In the combination of Kriging and those simulation methods, common learning functions include the expected feasibility function [11], U function [12], reliability-based lower confidence bounding (RLCB) [31], reliability-based expected improvement function (REIF) [32], and others. A notable feature of these learning functions is that their expressions are generally defined in terms of the posterior mean and variance of Kriging. This indicates that they essentially balance the proximity of posterior mean of Kriging to the limit state surface and the posterior variance of Kriging in a heuristic manner [33]. In the BALR methods, the measure of epistemic uncertainty about failure probability can be defined in terms of its upper-bound posterior variance [26] or expected misclassification probability [27], all expressed as integrals. Then,

their integrands are simply used as the corresponding learning functions, with typical ones including the upper-bound posterior variance contribution [26] or expected misclassification probability contribution [27]. However, the new point featuring the greatest integrand does not necessarily lead to the biggest reduction of the corresponding epistemic uncertainty measure.

Essentially, sequential experimental design involves solving a problem of sequential decision-making under uncertainty. Owing to the Bayesian nature of Kriging, this process exactly fits within a Bayesian decision-theoretic framework [28], where the optimal experimental design policy is built on maximizing specific utility functions in expectation. The utility function reflects preferences over different sample locations for the reliability analysis task. However, most of the aforementioned learning functions may fall short of realizing the promise of an optimal policy [33]. Although they are computationally efficient and yield reasonable empirical results, they may leave substantial room for improvement. Reinforcement learning [34] is a class of theoretically-sound and principled methods for finding an optimal policy for sequential decision-making. This is achieved by an agent learning a policy to maximize its expected cumulative reward function through interaction with the environment. Recently, it has demonstrated promising results in diverse fields, including robot control [35], Bayesian optimal experimental design [36], Bayesian optimization [37], and maintenance planning [38]. Importantly, this approach has also been explored in the reliability analysis task [39]; however, the associated learning function significantly deviates from the form of an expected accumulative reward function. To the best of the authors' knowledge, a genuine implementation of reinforcement learning-guided reliability method has not been attempted before.

In this work, a Bayesian reinforcement learning reliability (BRLR) method is developed. On one hand, Bayesian inference for the failure probability estimation is conducted to propagate and quantify the associated epistemic uncertainty. On the other hand, reliability-oriented sequential experimental design is built from the reinforcement learning perspective, with the aim of maximally reducing this epistemic uncertainty per iteration. The primary contributions of this study are summarized as follows.

- Sequential experimental design is framed as a finite-horizon Markov decision process (MDP) in the reinforcement learning framework with a Bayesian decision-theoretic setting. This allows gaining the theoretically-optimal sampling policy through dynamic programming.
- The integrated probability of misclassification (IPM) is proved to be the upper bound for the absolute relative error of Kriging-estimated failure probability. Hence, it can serve as a measure of epistemic uncertainty about failure probability estimation.
- Reward function in the MDP is specified in terms of IPM, leading to a one-step Bayes optimal learning function termed integrated probability of misclassification reduction (IPMR) and a compatible hybrid convergence criterion.
- Cost-effective IPMR-based sequential experimental design is conducted through three critical workarounds. First, the inner expectation in IPMR is analytically derived, reducing it to a single integral; then, IPMR is substituted with its computationally-cheap upper bound $\mathrm{IPMR}^{\mathrm{U}}$. Second, the pruning of quadrature set for $\mathrm{IPMR}^{\mathrm{U}}$ is conducted by exploring the locality of its integrand. Third, the candidate pool is pruned based on the preference of $\mathrm{IPMR}^{\mathrm{U}}$ over different candidate samples.

The rest of this paper is organized as follows. Section 2 provides a review of basic concepts and outlines the primary objective of this study. Section 3 devotes to developing the reinforcement learning-guided learning function IPMR. Then, Section 4 details the workflow of the proposed BRLR. The efficacy of the proposed approach is demonstrated through four examples in Section 5. Finally, Section 6 presents concluding remarks.

2. Preliminaries

Given that the proposed BRLR method is set up in the standard normal space, the input random vector $\boldsymbol{Z} \sim f_{\boldsymbol{Z}}(\boldsymbol{z})$ is transformed into the standard normal vector $\boldsymbol{X} = \{X_1, \ldots, X_d\} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_d)$. The corresponding isoprobabilistic transform $\mathcal{T}$ is defined as

$$\boldsymbol{X} = \mathcal{T}^{-1}(\boldsymbol{Z}) \tag{3}$$

where $\mathcal{T}$ could be, for instance, Nataf or Rosenblatt transforms. Then, the original performance function in Eq. (1) can be reformulated as

$$\mathcal{G}(\boldsymbol{X}) = G(\mathcal{T}(\boldsymbol{X})) \tag{4}$$

where $\mathcal{G} = G \circ \mathcal{T}$ represents a performance function evaluated in the standard normal space.

Section 2.1 provides an overview of the fundamentals of ALR. Section 2.2 attempts to frame the sequential experimental design as a finite-horizon MDP in the reinforcement learning framework. Finally, Section 2.3 outlines the primary objective of this study.

2.1. Active learning reliability analysis

Fig. 1 illustrates the general flowchart of the ALR methods. The five main ingredients, i.e., the experimental design (ED) $\mathcal{D}_n = \{\mathcal{X}_n, \mathcal{Y}_n\} = \left\{(\boldsymbol{x}^{(i)}, y^{(i)})\right\}_{i=1}^{n}$, the surrogate model $\widehat{\mathcal{G}}_n(\boldsymbol{x})$, the failure probability estimation $\widehat{P}_{f,n}$, the stopping condition and the learning function, are sequentially assembled. This iterative process forms a closed loop that enables feedback and, thus, progressively refining the accuracy of $\widehat{P}_{f,n}$.

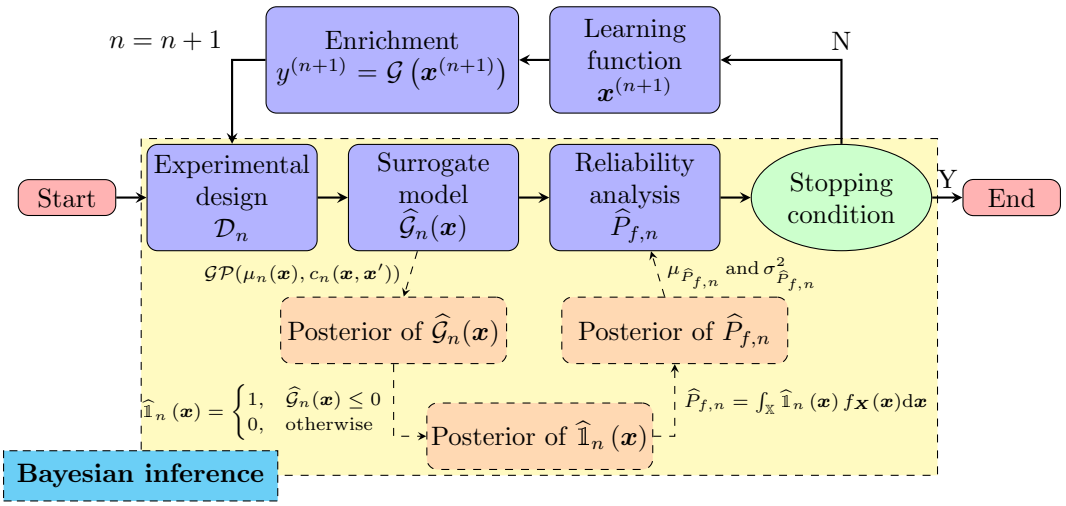


Figure 1: General framework of active learning reliability analysis

The basics of Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ are outlined in Appendix A. Due to the Bayesian formalism of Kriging, Bayesian inference can be conducted for the failure probability estimation, resulting in the posterior mean and variance of $\widehat{P}_{f,n}$. This workflow is briefly illustrated in the panel with yellow background in Fig. 1. Starting by assigning a GP prior over $\mathcal{G}(\boldsymbol{x})$ and specifying an ED $\mathcal{D}_n$ of size n, the probabilistic belief over $\mathcal{G}(\boldsymbol{x})$ is represented by the posterior distribution of Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x}) \sim \mathcal{GP}(\mu_n(\boldsymbol{x}), c_n(\boldsymbol{x}, \boldsymbol{x}'))$, as given by Eqs. (A.6), (A.7), (A.8). Then, according to Eq. (2), the posterior distribution of the estimator $\widehat{\mathbb{1}}_n(\boldsymbol{x})$ of the failure indicator function follows a generalized Bernoulli process [27]

$$\widehat{\mathbb{1}}_n(\boldsymbol{x}) \sim \mathcal{GBP}\left(\mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x}), c_{\widehat{\mathbb{1}}_n}(\boldsymbol{x}, \boldsymbol{x}')\right) \tag{5}$$

with the posterior mean $\mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x})$ and covariance $c_{\widehat{\mathbb{1}}_n}(\boldsymbol{x}, \boldsymbol{x}')$ expressed as [27]

$$\mu_{\widehat{\mathbb{1}}_n}(\boldsymbol{x}) = \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) \tag{6}$$

$$c_{\widehat{\mathbb{1}}_n}(\boldsymbol{x}, \boldsymbol{x}') = F_2\left(\begin{bmatrix}0\\0\end{bmatrix}; \begin{bmatrix}\mu_n(\boldsymbol{x})\\ \mu_n(\boldsymbol{x}')\end{bmatrix}, \begin{bmatrix}\sigma_n^2(\boldsymbol{x}), & c_n(\boldsymbol{x}, \boldsymbol{x}')\\ c_n(\boldsymbol{x}', \boldsymbol{x}), & \sigma_n^2(\boldsymbol{x}')\end{bmatrix}\right) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\Phi\left(-\frac{\mu_n(\boldsymbol{x}')}{\sigma_n(\boldsymbol{x}')}\right) \tag{7}$$

where $\Phi(\cdot)$ denotes the cumulative distribution function (CDF) of a standard Gaussian variable; $F_2(\cdot; \boldsymbol{\mu}, \boldsymbol{C})$ denotes the CDF of a bivariate Gaussian vector with the mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{C}$.

Substituting Eq. (5) into Eq. (1), the $\widehat{P}_{f,n}$ is expressed as

$$\widehat{P}_{f,n} = \int_{\mathbb{X}} \widehat{\mathbb{1}}_n(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \tag{8}$$

which is still a random variable, with the posterior mean and variance derived as [27]

$$\mu_{\widehat{P}_{f,n}} = \int_{\mathbb{X}} \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \tag{9}$$

$$\sigma^2_{\widehat{P}_{f,n}} = \int\int_{\mathbb{X}\times\mathbb{X}} F_2\left(\begin{bmatrix}0\\0\end{bmatrix}; \begin{bmatrix}\mu_n(\boldsymbol{x})\\ \mu_n(\boldsymbol{x}')\end{bmatrix}, \begin{bmatrix}\sigma_n^2(\boldsymbol{x}), & c_n(\boldsymbol{x}, \boldsymbol{x}')\\ c_n(\boldsymbol{x}', \boldsymbol{x}), & \sigma_n^2(\boldsymbol{x}')\end{bmatrix}\right) f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}') \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}' - \mu^2_{\widehat{P}_{f,n}} \tag{10}$$

where $\mu_{\widehat{P}_{f,n}}$ can be used as an estimate of failure probability, and $\sigma^2_{\widehat{P}_{f,n}}$ serves as an uncertainty measure about $\mu_{\widehat{P}_{f,n}}$. The latter reflects the epistemic uncertainty associated with $\widehat{P}_{f,n}$, induced by using the Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$.

Generally, Eqs. (9) and (10) need to be computed via a desired quadrature method. Notably, Eq. (10), containing a bivariate Gaussian CDF $F_2(\cdot;\cdot,\cdot)$ to be numerically computed element-wise, can be computationally expensive.

2.2. Sequential experimental design as a Markov decision process

Another critical aspect of the ALR methods lies in efficiently building the sequential experimental design. This problem can be interpreted from the reinforcement learning perspective, where an agent (or decision-maker) learns the optimal policy for sequential decision-making by interactions with an environment. As depicted in Fig. 2, this task is formulated as a MDP defined by a tuple $< S, A, P, R >$ [34]:

- S: The state space;
- A: The action space;
- $P(s, a, s')$: The probability of transitioning to s' when taking action a at state s;
- $R(s, a, s')$: The reward function when transitioning to state s' after taking action a at state s.

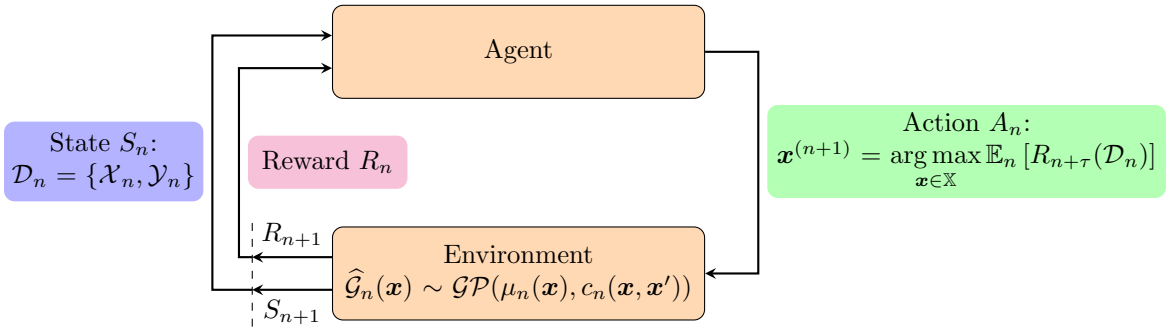


Figure 2: Interpretation of reliability-oriented sequential experimental design as a MDP

A decision rule $\pi_t : S \mapsto A$ maps states to actions at epoch t, and a policy π is a sequence of decision rules $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_N)$. Given a policy $\boldsymbol{\pi}$, an initial state s_n, and a look-ahead horizon τ, the expected cumulative reward can be expressed as [35]

$$V_\tau^{\boldsymbol{\pi}}(s) = \mathbb{E}_{s_n, s_{n+1}, \ldots, s_{n+\tau}|\boldsymbol{\pi}}\left[\sum_{t=0}^{\tau-1} R\left(s_{n+t}, \pi_{n+t}(s_{n+t}), s_{n+t+1}\right) | s = s_n\right] \tag{11}$$

which is also referred to as value function; the expectation is defined with respect to the transition probabilities of states at each epoch. Note that both the discount factor and terminal reward are omitted in Eq. (11) for simplicity.

The objective of solving a MDP is to design an optimal policy $\boldsymbol{\pi}^*$ that maximizes the value function:

$$\boldsymbol{\pi}^* = \left\{\pi_n^*, \ldots, \pi_{n+\tau-1}^*\right\} = \arg\sup_{\boldsymbol{\pi}\in\Pi} V_\tau^{\boldsymbol{\pi}}(s) \tag{12}$$

where Π is the set of all feasible policies.

Following Bellman's principle of optimality, Eqs. (11) and (12) can be formulated using recursive dynamic programming [35]:

$$\begin{aligned} V_{n+t}^*(s) &= \max_{a\in A} \mathbb{E}_{s'}\left[R(s, a, s') + V_{n+t+1}^*(s')\right] \\ V_{n+\pi}^*(s) &= 0 \end{aligned} \tag{13}$$

for $t = \tau-1, \tau-2, \ldots, 0$, where $V_{n+t}^*(s)$ denotes the optimal value function of $\boldsymbol{\pi}^*$. Eq. (13) suffers from the 'curse of dimensionality' raised from the uncountable state and action space when the look-ahead horizon τ is very significant. This challenge can be alleviated by some approximate dynamic programming methods, such as rollout or limited look-ahead strategies [35].

2.3. Problem statement

Having covered the fundamentals of MDP in Section 2.2, we attempt to map them to their analogous counterparts in the ALR methods. Specifically,

- S: The state space $S = \mathbb{X} \times \mathbb{Y}$ is all possible combinations of input and response spaces, and the ED $\mathcal{D}_n$ is the current state s_n.
- A: The action space A is the input space $\mathbb{X}$, and the action corresponds to adding the next sample $\boldsymbol{x}_+^{(n+1)}$ into $\mathcal{D}_n$, i.e., $a_n = \pi_n(\mathcal{D}_n) = \boldsymbol{x}_+^{(n+1)}$.
- $P(s, a, s')$: The transition probability from state $\mathcal{D}_n$ to state $\mathcal{D}_{n+1}$, given an action $\boldsymbol{x}_+^{(n+1)}$, can be readily represented by the Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ such that

$$Y_+^{(n+1)} = \mathcal{N}\left(\mu_n\left(\boldsymbol{x}_+^{(n+1)}\right), \sigma_n^2\left(\boldsymbol{x}_+^{(n+1)}\right)\right) \tag{14}$$

  where the upper case highlights that $Y_+^{(n+1)}$ is a Gaussian random variable. Besides, the subscript '+' intends to distinguish between the existing dataset and newly-added ones.

- $R(s, a, s')$: According to Bayesian decision theory [35], denote $U(\mathcal{D}_n)$ as a real-valued utility function for the current state $\mathcal{D}_n$. Higher utility indicates more favorable outcome for the reliability analysis task. Then, the reward function is expressed as

$$R\left(\mathcal{D}_{n+t}, \boldsymbol{x}_+^{(n+t+1)}, \mathcal{D}_{n+t+1}\right) = U(\mathcal{D}_{n+t+1}) - U(\mathcal{D}_{n+t}), \quad t = 0, \ldots, \tau - 1 \tag{15}$$

  which represents the increase of the utility function, due to adding $\boldsymbol{x}^{(n+t+1)}$ into $\mathcal{D}_{n+t}$.

Analogous to Eq. (11), the value function associated with the reward function $R(\cdot)$ in Eq. (15) is expressed as

$$\begin{aligned} V_\tau^{\boldsymbol{\pi}}\left(\boldsymbol{x}_+^{(n+1)}; \mathcal{D}_n\right) &= \mathbb{E}_{Y_+^{(n+1)}, \ldots, \boldsymbol{x}_+^{(n+\tau)}, Y_+^{(n+\tau)}}\left[\sum_{t=0}^{\tau-1} R\left(\mathcal{D}_{n+t}, \boldsymbol{x}_+^{(n+t+1)}, \mathcal{D}_{n+t+1}\right)\right] \\ &= \mathbb{E}_{Y_+^{(n+1)}, \ldots, \boldsymbol{x}_+^{(n+\tau)}, Y_+^{(n+\tau)}}\left[U\left(\mathcal{D}_{n+\tau}\right)\right] - U(\mathcal{D}_n) \end{aligned} \tag{16}$$

where the expectation is taken with respect to all possible randomness during the whole look-ahead horizon, consisting of both $Y_+^{(n+1)}$ and $\left\{(\boldsymbol{x}_+^{(n+t)}, Y_+^{(n+t)})\right\}_{t=2}^{\tau}$.

Then, in accordance with Eq. (13), there exists

$$\begin{aligned} V_\tau^*\left(\mathcal{D}_n\right) &= \max_{\boldsymbol{x}_+^{(n+1)} \in \mathbb{X}} \left\{ V_1\left(\boldsymbol{x}_+^{(n+1)}; \mathcal{D}_n\right) + \mathbb{E}_{Y_+^{(n+1)}}\left[\max_{\boldsymbol{x}_+^{(n+2)} \in \mathbb{X}} V_{\tau-1}\left(\boldsymbol{x}_+^{(n+2)}; \mathcal{D}_{n+1}\right)\right]\right\} \\ &= \max_{\boldsymbol{x}_+^{(n+1)} \in \mathbb{X}} \left\{ V_1\left(\boldsymbol{x}_+^{(n+1)}; \mathcal{D}_n\right) + \mathbb{E}_{Y_+^{(n+1)}}\left[V_{\tau-1}^*\left(\mathcal{D}_{n+1}\right)\right]\right\} \end{aligned} \tag{17}$$

Hence, the optimal policy is expressed as

$$\boldsymbol{x}^{(n+1)} = \arg\max_{\boldsymbol{x}_+^{(n+1)} \in \mathbb{X}} \left\{ V_1\left(\boldsymbol{x}_+^{(n+1)}; \mathcal{D}_n\right) + \mathbb{E}_{Y_+^{(n+1)}}\left[V_{\tau-1}^*\left(\mathcal{D}_{n+1}\right)\right]\right\} \tag{18}$$

which involves a series of nested maximization and expectation operations. Hence, Eq. (18) is generally difficult to compute when $\tau > 2$.

For analytical tractability, we restrict our attention to the one-step look-ahead case, i.e., $\tau = 1$. Moreover, since only a new point is considered at the current iteration, the notations $\boldsymbol{x}_+^{(1)}$ and $Y_+^{(1)}$ can be simplified as $\boldsymbol{x}_+$ and Y_+, without the risk of confusion. Then, Eq. (18) simplifies to

$$\begin{aligned} \boldsymbol{x}^{(n+1)} &= \arg\max_{\boldsymbol{x}_+ \in \mathbb{X}} V_1\left(\boldsymbol{x}_+; \mathcal{D}_n\right) \\ &= \arg\max_{\boldsymbol{x}_+ \in \mathbb{X}} \left\{\mathbb{E}_{Y_+}\left[U(\mathcal{D}_{n+1})\right] - U(\mathcal{D}_n)\right\} \end{aligned} \tag{19}$$

which is called one-step Bayes optimal policy.

Eq. (19) indicates that the best next point $\boldsymbol{x}^{(n+1)}$ at the iteration n is the one achieving the greatest increase of utility function in expectation when added into $\mathcal{D}_n$. Hence, the general expression $\left\{\mathbb{E}_{Y_+}\left[U(\mathcal{D}_{n+1})\right]-U(\mathcal{D}_n)\right\}$ exactly corresponds to the one-step Bayes optimal learning function.

The remaining concern is how to define a real-value utility function $U(\cdot)$ tailored for the reliability analysis task, so as to gain a theoretically sound and computationally affordable learning function. Eq. (10) indicates that $\sigma^2_{\widehat{P}_{f,n}}$ is a good measure of epistemic uncertainty about $\mu_{\widehat{P}_{f,n}}$, and $-\sigma^2_{\widehat{P}_{f,n}}$ is thus a natural choice for the utility function in Eq. (19). However, $\sigma^2_{\widehat{P}_{f,n}}$ is too computationally demanding. It is more attractive to consider a computationally cheap utility function, which will be detailed in Section 3.

Remark 1. *The rationale behind the one-step look-ahead policy in Eq. (19) could be justified from the perspective of probabilistic model misspecification [40]. Eq. (18) underscores that the optimal policy is defined with respect to both the probabilistic model, say Kriging, of $\mathcal{G}(\boldsymbol{x})$ and the available dataset. Therefore, given the imperfection of our belief about $\mathcal{G}(\boldsymbol{x})$, especially with a limited dataset, a less reliance on the probabilistic model's belief, i.e., limiting the look-ahead horizon, may gain better robustness, along with remarkable computational savings.*

3. The proposed integrated probability of misclassification reduction (IPMR)

A computationally-efficient utility function, referred to as integrated probability of misclassification (IPM), is proposed in the reliability-oriented MDP, resulting in a one-step Bayes optimal learning function named integrated probability of misclassification reduction (IPMR), along with a compatible convergence criterion.

Section 3.1 outlines the definition of IPM. Section 3.2 provides the basic expression for the resulting learning function IPMR, formulated as a double expectation. Then, the inner expectation in IPMR is analytically deduced in Section 3.3, simplifying IPMR to a single expectation. Given the computational challenges faced by IPMR, it is replaced by its upper bound IPMR^{U} in Section 3.4. Then, the pruning of the quadrature set and candidate point in IPMR^{U} are conducted in Sections 3.5 and 3.6, respectively.

3.1. Basic definition of the integrated probability of misclassification (IPM)

Proposition 1. *Denote H_n as*

$$H_n \coloneqq H(\mathcal{D}_n)=\mathbb{E}_{\boldsymbol{X}}\left[P_n(\boldsymbol{x})\right]=\int_{\mathbb{X}} P_n(\boldsymbol{x})f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \tag{20}$$

where $P_n(\boldsymbol{x})=\Phi\left(-\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\right)$ is the so-called probability of misclassification (PM), representing the probability of misclassifying the safe/failure state of $\boldsymbol{x}$ according to the sign of $\mu_n(\boldsymbol{x})$. Hence, H_n can be called integrated probability of misclassification (IPM). Then, $\widehat{P}_{f,n}$ and $\mu_{\widehat{P}_{f,n}}$ satisfy the following expression

$$\mathbb{E}_n\left[\left|\widehat{P}_{f,n}-\mu_{\widehat{P}_{f,n}}\right|\right]\leq 2H_n \tag{21}$$

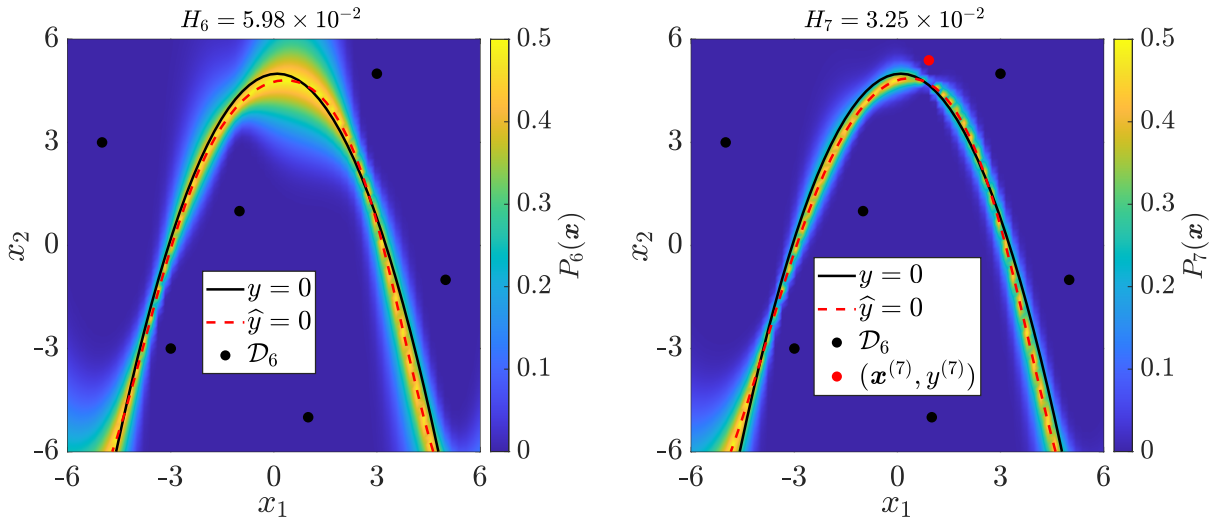
The proof of Proposition 1 is given in Appendix B.

Eq. (21) implies that when $H_n\to 0$, $\mu_{\widehat{P}_{f,n}}$ converges to P_f in expectation. Hence, H_n can be viewed as a measure of epistemic uncertainty about $\mu_{\widehat{P}_{f,n}}$, and it should be reduced as much as possible. For illustration, consider the following bivariate performance function

$$\mathcal{G}(\boldsymbol{X})=5-0.5(X_1-0.1)^2-X_2 \tag{22}$$

where X_1 and X_2 are two uniform variables within $[-6,6]$. The grid of size 80×80 is simply used as a set of quadrature points with equal weights.

In Fig 3, the actual limit state is depicted as a black solid line, and the initial training samples $\mathcal{D}_6=\left\{(\boldsymbol{x}^{(i)},y^{(i)})\right\}_{i=1}^{6}$ are marked as black solid circles. A Kriging $\widehat{\mathcal{G}}_6(\boldsymbol{x})$ is trained, and significant values of $P_6(\boldsymbol{x})$ are observed around the approximated limit state, with the resulting H_6 computed as 5.98×10^{-2} (Fig. 3(a)). Then, a new point $\left(\boldsymbol{x}^{(7)},y^{(7)}\right)$ (red solid circle) is added into $\mathcal{D}_6$, forming $\mathcal{D}_7=\mathcal{D}_6\bigcup\left(\boldsymbol{x}^{(7)},y^{(7)}\right)$. The Kriging $\widehat{\mathcal{G}}_7(\boldsymbol{x})$ is trained and the corresponding $P_7(\boldsymbol{x})$ is shown in Fig. 3(b). Much smaller values of $P_7(\boldsymbol{x})$ are observed in the vicinity of the limit state, and the resulting H_7 is equal to 3.25×10^{-2}, which is far less than H_6.



(a) $P_6(\boldsymbol{x})$ and H_6 (b) $P_7(\boldsymbol{x})$ and H_7

Figure 3: Illustration of IPM in a bivariate toy function

Therefore, $-H_n$ is a favorable utility function $U(\cdot)$ for $\mathcal{D}_n$. Then, following the one-step Bayes optimal policy in Eq. (19), the $\boldsymbol{x}^{(n+1)}$ at iteration n is selected as

$$\begin{aligned} \boldsymbol{x}^{(n+1)} &= \arg\max_{\boldsymbol{x}_+ \in \mathbb{X}} \left\{ \mathbb{E}_{Y_+} \left[-H(\mathcal{D}_{n+1}) \right] - \left(-H(\mathcal{D}_n) \right) \right\} \\ &= \arg\max_{\boldsymbol{x}_+ \in \mathbb{X}} \left\{ H_n - \mathbb{E}_{Y_+} \left[H(\mathcal{D}_{n+1}) \right] \right\} \end{aligned} \tag{23}$$

where $H\left(\mathcal{D}_{n+1}\right)$ represents the one-step look-ahead IPM, when a pair of new point and its response $(\boldsymbol{x}_+, y_+)$ is added into $\mathcal{D}_n$, i.e., $\mathcal{D}_{n+1} = \mathcal{D}_n \bigcup (\boldsymbol{x}_+, y_+)$. Then, the specific expression of the one-step Bayes optimal learning function $\left\{ H_n - \mathbb{E}_{Y_+} \left[H(\mathcal{D}_{n+1}) \right] \right\}$ will be derived in Section 3.2.

3.2. General expression of IPMR

Analogous to Eq. (20), the look-ahead IPM $H\left(\mathcal{D}_{n+1}\right)$ is expressed in terms of the posterior of Kriging $\widehat{\mathcal{G}}_{n+1}(\boldsymbol{x})$ calibrated from $\mathcal{D}_{n+1}$. This can be readily provided by Kriging update formulas, as outlined in Appendix C.

Specifically, when $\mathcal{D}_n$ is enriched with $(\boldsymbol{x}_+, y_+)$, the look-ahead posterior of $\widehat{\mathcal{G}}_{n+1}(\boldsymbol{x})$ can be directly obtained from the posterior of $\widehat{\mathcal{G}}_n(\boldsymbol{x})$, without needing to re-estimate its parameters, as expressed in Eqs. (C.1), (C.2) and (C.3). On this basis, the look-ahead IPM $H\left(\mathcal{D}_{n+1}\right)$ can be expressed as

$$H_{n+1}\left(\boldsymbol{x}_+, y_+\right) \coloneqq H(\mathcal{D}_{n+1}) = \mathbb{E}_{\boldsymbol{X}} \left[P_{n+1}\left(\boldsymbol{x}; \boldsymbol{x}_+, y_+\right) \right] \tag{24}$$

which is a function of $\boldsymbol{x}_+$, y_+, with the current ED $\mathcal{D}_n$ omitted for simplicity; $P_{n+1}(\boldsymbol{x}; \boldsymbol{x}_+, y_+)$ denotes the look-ahead PM and is given as

$$P_{n+1}(\boldsymbol{x}; \boldsymbol{x}_+, y_+) = \begin{cases} \begin{cases} \Phi\left(a(\boldsymbol{x}) + b(\boldsymbol{x}) z_+\right), & z_+ \le z_{\lim} \\ 1 - \Phi\left(a(\boldsymbol{x}) + b(\boldsymbol{x}) z_+\right), & z_+ > z_{\lim} \end{cases}, & c_n(\boldsymbol{x}, \boldsymbol{x}_+) > 0 \\ \begin{cases} 1 - \Phi\left(a(\boldsymbol{x}) + b(\boldsymbol{x}) z_+\right), & z_+ < z_{\lim} \\ \Phi\left(a(\boldsymbol{x}) + b(\boldsymbol{x}) z_+\right), & z_+ \ge z_{\lim} \end{cases}, & \text{otherwise} \end{cases} \tag{25}$$

where $a(\boldsymbol{x}) = \frac{\mu_n(\boldsymbol{x})}{\sigma_{n+1}(\boldsymbol{x})}$, $b(\boldsymbol{x}) = \frac{c_n(\boldsymbol{x}, \boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)\sigma_{n+1}(\boldsymbol{x})}$, $z_+ = \frac{y_+ - \mu_n(\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)}$, $z_{\lim} = -\frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{c_n(\boldsymbol{x}, \boldsymbol{x}_+)}$. The detailed derivation of Eq. (25) is given in Appendix D.

Recall that the actual performance function response y_+ at $\boldsymbol{x}_+$ is unknown; hence, $P_{n+1}\left(\boldsymbol{x}; \boldsymbol{x}_+, y_+\right)$ in Eq. (25) and $H_{n+1}\left(\boldsymbol{x}_+, y_+\right)$ in Eq. (24) are unknown as well. To remedy this bottleneck, the y_+ is replaced by the Kriging prediction at $\boldsymbol{x}_+$, denoted as $Y_+ \sim \mathcal{N}\left(\mu_n(\boldsymbol{x}_+), \sigma_n^2(\boldsymbol{x}_+)\right)$, and Eq. (24) is thus transformed as

$$\mathcal{H}_{n+1}(\boldsymbol{x}_+) = \mathbb{E}_{\boldsymbol{X}} \left[\mathcal{P}_{n+1}(\boldsymbol{x}; \boldsymbol{x}_+) \right] \tag{26}$$

in which

$$\mathcal{P}_{n+1}(\boldsymbol{x};\boldsymbol{x}_+)=\begin{cases}\begin{cases}\Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})Z_+\right), & Z_+\le z_{\lim}\\ 1-\Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})Z_+\right), & Z_+>z_{\lim}\end{cases}, & c_n(\boldsymbol{x},\boldsymbol{x}_+)>0\\ \begin{cases}1-\Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})Z_+\right), & Z_+<z_{\lim}\\ \Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})Z_+\right), & Z_+\ge z_{\lim}\end{cases}, & \text{otherwise}\end{cases} \tag{27}$$

where $Z_+=\frac{Y_+-\mu_n(\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)}$ is a standard normal variable. Obviously, both $\mathcal{P}_{n+1}(\boldsymbol{x};\boldsymbol{x}_+)$ in Eq. (27) and $\mathcal{H}_{n+1}(\boldsymbol{x}_+)$ in Eq. (26) become the functions of $\boldsymbol{x}_+$ solely. Notably, they are random quantities through Y_+, or equivalently Z_+.

In accordance with Eq. (23), the one-step Bayes optimal learning function is expressed as

$$\begin{aligned}\mathrm{IPMR}_n\left(\boldsymbol{x}_+\right)&=H_n-\mathbb{E}_{Y_+}\left[\mathcal{H}_{n+1}\left(\boldsymbol{x}_+\right)\right]\\&=H_n-\mathbb{E}_{Y_+}\left[\mathbb{E}_{\boldsymbol{X}}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right]\right]\\&\overset{*}{=}\mathbb{E}_{\boldsymbol{X}}\left[P_n(\boldsymbol{x})\right]-\mathbb{E}_{\boldsymbol{X}}\left[\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right]\right]\\&=\mathbb{E}_{\boldsymbol{X}}\left[P_n(\boldsymbol{x})-\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right]\right]\\&=\mathbb{E}_{\boldsymbol{X}}\left[I_n\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right]\end{aligned} \tag{28}$$

where the exchange of two expectations in the equality '$\overset{*}{=}$' utilizes the Fubini-Tonelli theorem; $I_n(\boldsymbol{x};\boldsymbol{x}_+)=P_n(\boldsymbol{x})-\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right]$ represents the inner expectation.

Then, the best next point $\boldsymbol{x}^{(n+1)}$ at iteration n is selected as

$$\boldsymbol{x}^{(n+1)}=\underset{\boldsymbol{x}_+\in\mathcal{X}_{\mathrm{C}}}{\arg\max}\,\mathrm{IPMR}_n\left(\boldsymbol{x}_+\right) \tag{29}$$

where $\mathcal{X}_{\mathrm{C}}$ denotes the candidate pool. Obviously, $\boldsymbol{x}^{(n+1)}$ is the point achieving the maximum reduction of IPM in expectation. Hence, this one-step Bayes optimal learning function is called integrated probability of misclassification reduction (IPMR).

Eq. (28) indicates that IPMR involves two expectations, i.e., the inner expectation $I_n(\boldsymbol{x};\boldsymbol{x}_+)$ and the subsequent outer expectation $\mathbb{E}_{\boldsymbol{X}}\left[I_n(\boldsymbol{x};\boldsymbol{x}_+)\right]$. They will be sequentially addressed in the following subsections.

3.3. Inner expectation in IPMR

Proposition 2. *The inner expectation $I_n(\boldsymbol{x};\boldsymbol{x}_+)$ in Eq. (28) is analytically expressed as*

$$I_n(\boldsymbol{x};\boldsymbol{x}_+)=\Phi\left(-\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right)-2\Phi_2\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})},-\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|};-|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|\right) \tag{30}$$

where $\rho_n(\boldsymbol{x},\boldsymbol{x}_+)=\frac{c_n(\boldsymbol{x},\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x})\sigma_n(\boldsymbol{x}_+)}$ denotes the the posterior correlation coefficient of Kriging $\widehat{\mathcal{G}}_n(\cdot)$ between $\boldsymbol{x}$ and $\boldsymbol{x}_+$; $\Phi_2\left(h_1,h_2;r\right)$ is the CDF of a standard bivariate Gaussian vector with a correlation coefficient r.

Proposition 3. *The lower and upper bounds of $I_n(\boldsymbol{x};\boldsymbol{x}_+)$ are expressed as*

$$\begin{aligned}I_n^{\mathrm{L}}(\boldsymbol{x};\boldsymbol{x}_+)&=0\\I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)&=\Phi\left(\frac{-\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right)\end{aligned} \tag{31}$$

The proof of Proposition 2 is provided in Appendix E. Then, the proof of Proposition 3 is provided in Appendix F.

According to Eqs. (28) and (31), the upper bound of $\mathrm{IPMR}_n(\boldsymbol{x}_+)$ is naturally given as

$$\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)=\mathbb{E}_{\boldsymbol{X}}\left[I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)\right] \tag{32}$$

The remaining single expectations in both $\mathrm{IPMR}_n(\boldsymbol{x}_+)$ and $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$ generally have no analytical solution. Therefore, they need to be numerically computed via some favorable quadrature methods, such as variance-amplified importance sampling (VAIS), which will be detailed in Section 3.4.

3.4. Computational challenge encountered by the outer expectation in IPMR

Taking the VAIS-based computation of $\mathrm{IPMR}_n(\boldsymbol{x}_+)$ as an example, it can be rewritten as

$$\mathrm{IPMR}_n(\boldsymbol{x}_+) = \int_{\mathbb{X}} I_n\left(\boldsymbol{x};\boldsymbol{x}_+\right) \frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})} h_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \tag{33}$$

where $h_{\boldsymbol{X}}(\boldsymbol{x})$ is the importance sampling density. The optimal importance sampling density is generally unavailable, due to the lack of exact knowledge about the quantity to be estimated. The VAIS takes a simple but effective approach: $h_{\boldsymbol{X}}(\boldsymbol{x})$ is constructed by amplifying the standard deviation of $\boldsymbol{X}$, while keeping the mean vector unchanged, that is, $h_{\boldsymbol{X}}(\boldsymbol{x}) = \boldsymbol{\phi}\left(\boldsymbol{x};\boldsymbol{0},\alpha^2\boldsymbol{I}_d\right)$, where $\alpha(>1)$ is the amplification coefficient of standard deviation. Although different amplification coefficients can be assigned to distinct dimensions, only a single value of α is set for all dimensions of $\boldsymbol{X}$.

Then, Eq. (33) can be numerically computed as

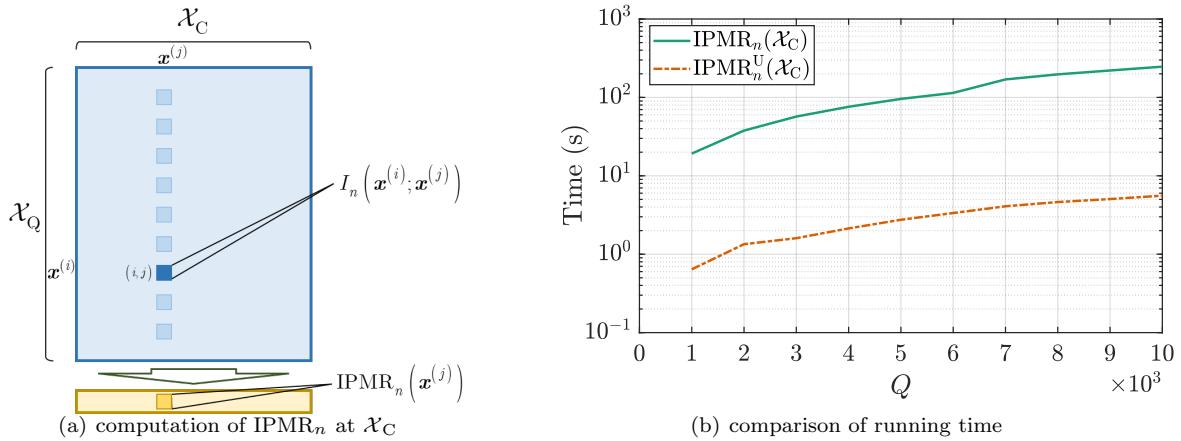
$$\mathrm{IPMR}_n(\boldsymbol{x}_+) \approx \frac{1}{Q}\sum_{i=1}^{Q}\left[I_n(\boldsymbol{x}^{(i)};\boldsymbol{x}_+)\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}\right] \tag{34}$$

where $\mathcal{X}_{\mathrm{Q}} = \{\boldsymbol{x}^{(i)}\}_{i=1}^{Q}$ is a set of Q quadrature points (e.g., Sobol sequence) drawn from $h_{\boldsymbol{X}}(\boldsymbol{x})$. Similarly, $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$ in Eq. (32) can be approximated as

$$\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+) \approx \frac{1}{Q}\sum_{i=1}^{Q}\left[I_n^{\mathrm{U}}(\boldsymbol{x}^{(i)};\boldsymbol{x}_+)\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}\right] \tag{35}$$

When $\mathrm{IPMR}_n(\cdot)$ is used to select the best next point $\boldsymbol{x}^{(n+1)}$ from a candidate pool $\mathcal{X}_{\mathrm{C}}$ of size C, an identical quadrature set $\mathcal{X}_{\mathrm{Q}}$ is usually used to approximate $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+), \forall \boldsymbol{x}_+ \in \mathcal{X}_{\mathrm{C}}$, for convenience. This means that the matrix $\left[I_n(\boldsymbol{x}^{(i)},\boldsymbol{x}^{(j)})\right]_{1\le i\le Q, 1\le j\le C}$ at $\mathcal{X}_{\mathrm{Q}}\times\mathcal{X}_{\mathrm{C}}$ needs to be computed, as illustrated in Fig. 4(a). If Q and C are too significant, this will confront two fatal issues.

- *Computer memory crashing.* The Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ has to provide the correlation coefficient matrix $\left[\rho_n(\boldsymbol{x}^{(i)},\boldsymbol{x}^{(j)})\right]_{1\le i\le Q, 1\le j\le C}$, or equivalently the posterior covariance matrix $\left[c_n(\boldsymbol{x}^{(i)},\boldsymbol{x}^{(j)})\right]_{1\le i\le Q, 1\le j\le C}$, at $\mathcal{X}_{\mathrm{Q}}\times\mathcal{X}_{\mathrm{C}}$, posing high demands on the computer memory if this matrix size is too large.
- *Element-wise computation.* In $I_n(\boldsymbol{x};\boldsymbol{x}_+)$, the $\Phi_2(\cdot,\cdot;\rho)$ with different values of ρ have to be computed element-wise, indicating that a total of $Q\times C$ evaluations of $I_n(\boldsymbol{x}^{(i)};\boldsymbol{x}^{(j)})$ need to be sequentially conducted. Even resorting to parallel computing, this process is still time-consuming.



(a) computation of IPMR_n at $\mathcal{X}_{\mathrm{C}}$ (b) comparison of running time

Figure 4: Comparison between IPMR and $\mathrm{IPMR}^{\mathrm{U}}$

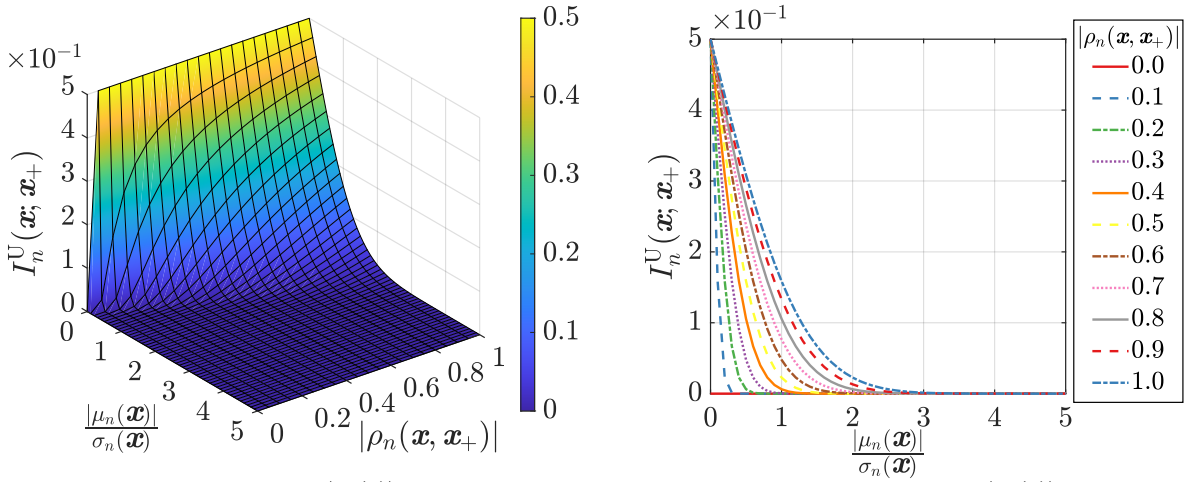
For illustration, assume that the sample size C of $\mathcal{X}_{\mathrm{C}}$ is 5000, and the sample size Q of $\mathcal{X}_{\mathrm{Q}}$ ranges from 1000 to 10000 at intervals of 1000. Parallel computation of $\mathrm{IPMR}_n(\cdot)$ at $\mathcal{X}_{\mathrm{Q}}\times\mathcal{X}_{\mathrm{C}}$ is conducted on an Intel Xeon Gold CPU processor with 20 cores, 3 GHz, and 64G RAM. The corresponding computational time is shown in Fig. 4(b). Notably, when $Q \ge 5000$, $\mathrm{IPMR}_n(\cdot)$ consumes at least 100 s, far greater than that of common learning functions, like U function.

To address the second challenge regarding element-wise computation, it is feasible to substitute $\mathrm{IPMR}_n(\cdot)$ (Eq. (34)) with $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$ (Eq. (35)). Unlike the $I_n(\boldsymbol{x};\boldsymbol{x}_+)$ in $\mathrm{IPMR}_n(\cdot)$, the $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ in $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$ only contains the univariate Gaussian CDF $\Phi(\cdot)$, which can be efficiently computed using vectorization in MATLAB. Fig. 4(b) shows that the computational time of $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$ is comparable to that of common learning functions, significantly less than that of $\mathrm{IPMR}_n(\cdot)$. Therefore, only $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$ is considered hereinafter.

Nevertheless, $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$ still suffers from the first challenge of potential computer memory issue, particularly when $Q \geq \mathcal{O}(10^4)$ and $C \geq \mathcal{O}(10^4)$. To address this challenge, both Q and C are preferred to be pruned, which will be discussed in Sections 3.5 and 3.6, respectively.

3.5. Pruning of quadrature points in IPMR

To prune the quadrature points needed by $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$ in Eq. (35), the locality of the integrand $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ is explored here. Fig. 5(a) depicts $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ as a function of both $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}$ and $|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|$. Notably, $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ achieves its maximum value across the entire limit state $\left\{\boldsymbol{x}\in\mathbb{X}:\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}=0\right\}$. Then, $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ decays rapidly with the increasing of $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}$. Importantly, apart from the limit state, $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ remains significant only when $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}$ is small and $|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|$ is large. This observation is further clarified in Fig. 5(b). When $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\geq 3$, $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ becomes negligible, regardless of the magnitude of $|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|$.



(a) $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ as a function of $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}$ and $|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|$

(b) $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ w.r.t. $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}$

Figure 5: Locality of the integrand $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ in $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$

Therefore, the region with significant $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ in the input space can be defined as

$$\mathbb{X}_{\mathrm{QT}}=\left\{\boldsymbol{x}\in\mathbb{X}:\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\leq\lambda\right\} \tag{36}$$

where the pruning coefficient $\lambda(\leq 3)$ controls the span of $\mathbb{X}_{\mathrm{QT}}$. Then, the quadrature points within $\mathbb{X}_{\mathrm{QT}}$ are denoted as

$$\mathcal{X}_{\mathrm{QT}}=\left\{\boldsymbol{x}^{(i)}\in\mathcal{X}_{\mathrm{Q}}:\frac{|\mu_n(\boldsymbol{x}^{(i)})|}{\sigma_n(\boldsymbol{x}^{(i)})}\leq\lambda\right\} \tag{37}$$

with size Q_{T}.

In this way, it is sufficient to only consider $\mathcal{X}_{\mathrm{QT}}$, rather than $\mathcal{X}_{\mathrm{Q}}$, in the numeric computation of $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$. Consequently, Eq. (35) further reduces to

$$\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)\approx\frac{1}{Q}\sum_{i=1}^{Q_{\mathrm{T}}}\left[I_n^{\mathrm{U}}(\boldsymbol{x}^{(i)};\boldsymbol{x}_+)\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}\right] \tag{38}$$

Three parameters have not yet been specified: the amplification coefficient α, the quadrature size Q, and the pruning coefficient λ. These will be further discussed in Section 4.3.

3.6. Pruning of candidate points in IPMR

To prune the candidate pool $\mathcal{X}_{\mathrm{C}}$ in $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$, let's shed light on what kind of candidate point $\boldsymbol{x}_+$ is more likely to attain the maximum value of $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$.

Eq. (31) indicates that $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ is a function of $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}$ and $|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|$, with only the latter encoding the impact of $\boldsymbol{x}_+$. Fig. 5 illustrates that the quadrature point $\boldsymbol{x}$ with great value of $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$ has the following two characteristics simultaneously: (1) the $\boldsymbol{x}$ is located in $\mathbb{X}_{\mathrm{QT}}$, i.e., $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}$ is as small as possible; (2) the $|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|$ is sufficiently great. Therefore, in order to gain a significant value of $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$, the candidate point $\boldsymbol{x}_+$ should have high correlation with the quadrature points in $\mathcal{X}_{\mathrm{QT}}$, particularly those in the close vicinity of the limit state. Generally, when $\boldsymbol{x}_+$ is close to $\boldsymbol{x}$, a great value of $\rho_n(\boldsymbol{x},\boldsymbol{x}_+)$ will be obtained. Therefore, the $\boldsymbol{x}_+$ is preferred to be close to those quadrature points in $\mathcal{X}_{\mathrm{QT}}$, implying that the $\boldsymbol{x}_+$ with a great value of $P_n(\boldsymbol{x}_+)$ is more likely to achieve a high value of $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$.

For illustration, consider again the toy bivariate analytical function in Eq. (22) with the initial ED $\mathcal{D}_6$. Two different candidate points, $\boldsymbol{x}_+^{(a)}$ and $\boldsymbol{x}_+^{(b)}$, are of interest, marked as red and purple solid circles in Fig. 6(b), respectively. $\boldsymbol{x}_+^{(a)}$ is very close to the limit state, while $\boldsymbol{x}_+^{(b)}$ is the opposite. Therefore, $P_n(\boldsymbol{x}_+^{(a)})$ is greater than $P_n(\boldsymbol{x}_+^{(b)})$.

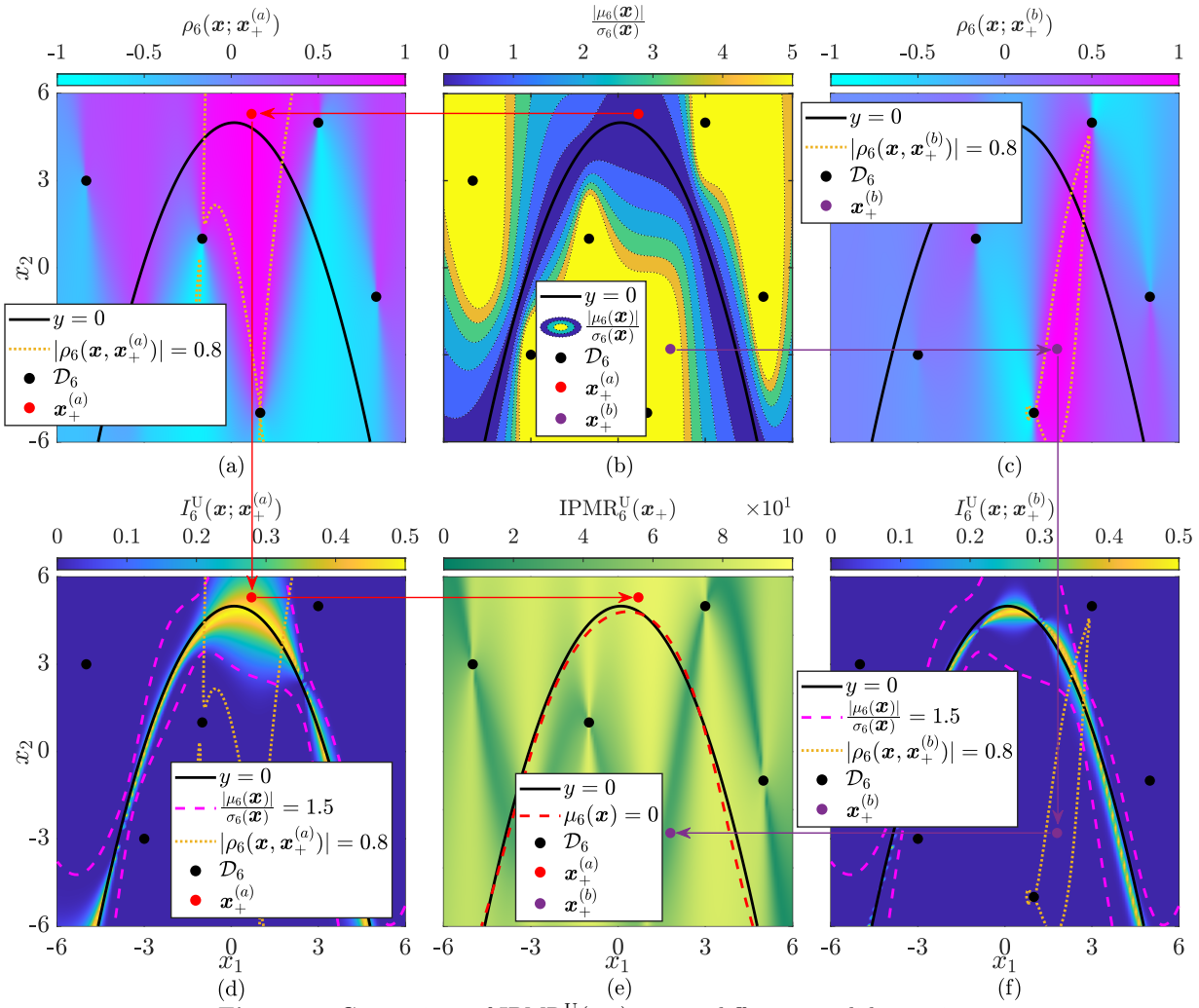


Figure 6: Comparison of $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$ at two different candidate points

Fig. 6(a) illustrates $\rho_6(\boldsymbol{x},\boldsymbol{x}_+^{(a)})$ within the input space, and the regions with $\left|\rho_6(\boldsymbol{x},\boldsymbol{x}_+^{(a)})\right| \geq 0.8$ are enclosed by yellow dotted lines, implying high correlation with $\boldsymbol{x}_+^{(a)}$. Significant values of $\rho_6(\boldsymbol{x},\boldsymbol{x}_+^{(a)})$ are primarily observed in regions around $\boldsymbol{x}_+^{(a)}$. Fig. 6(d) shows the corresponding $I_6^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+^{(a)})$ in the input space, and the $\mathbb{X}_{\mathrm{QT}}$ with $\lambda = 1.5$ is enclosed by magenta dashed lines. There is a substantial overlap between the yellow dotted lines and magenta dashed lines, indicating regions with significant values of $I_6^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+^{(a)})$.

In comparison, Fig. 6(c) depicts $\rho_6(\boldsymbol{x}, \boldsymbol{x}_+^{(b)})$ within the input space, and the regions with great values of $\rho_6(\boldsymbol{x}, \boldsymbol{x}_+^{(b)})$ are generally around $\boldsymbol{x}_+^{(b)}$, surrounded by yellow dotted lines. This region only has a minor overlap with $\mathbb{X}_{\mathrm{QT}}$, as illustrated in Fig. 6(f), and the corresponding $I_6^{\mathrm{U}}(\boldsymbol{x}; \boldsymbol{x}_+^{(b)})$ is very minor across the entire input space. Consequently, $\mathrm{IPMR}_6^{\mathrm{U}}(\boldsymbol{x}_+^{(b)})$ is much smaller than $\mathrm{IPMR}_6^{\mathrm{U}}(\boldsymbol{x}_+^{(a)})$, as depicted in Fig. 6(e). According to $\mathrm{IPMR}_6^{\mathrm{U}}(\cdot)$, $\boldsymbol{x}_+^{(a)}$ is preferable to $\boldsymbol{x}_+^{(b)}$.

On this basis, it is feasible to compute $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$ only on those highly probable candidate points, rather than the entire candidate pool $\mathcal{X}_{\mathrm{C}}$. To this end, the $\mathcal{X}_{\mathrm{C}}$ is pruned to a set of C_{T} points having the greatest values of $P_n(\boldsymbol{x})$, denoted as

$$\mathcal{X}_{\mathrm{CT}} = \{\boldsymbol{x}_+ \in \mathcal{X}_{\mathrm{C}} : P_n(\boldsymbol{x}_+) \geq p_{\mathrm{T}}\} \tag{39}$$

where p_{T} is the C_{T}-th greatest value in $\{P_n(\boldsymbol{x}^{(i)})\}_{i=1}^{C}$. C_{T} is specified as 5000, proven to be reasonable in various numerical examples.

Finally, Eq. (29) further reduces to

$$\boldsymbol{x}^{(n+1)} = \arg\max_{\boldsymbol{x}_+ \in \mathcal{X}_{\mathrm{CT}}} \mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+) \tag{40}$$

Remark 2. *The PM $P_n(\boldsymbol{x})$ can be directly used as a learning function, as given in Eq. (H.1). The advantages of the proposed* $\mathrm{IPMR}^{\mathrm{U}}$ *over PM are clarified as follows. The PM only considers the information of $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}$ in the input space, and any point $\boldsymbol{x}_+$ satisfying $\mu_n(\boldsymbol{x}_+) = 0$ can be selected as the best next point $\boldsymbol{x}^{(n+1)}$, as plotted by the red dashed line in Fig. 3(a). However, $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$ accounts for both $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}$ and $|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|$, with the latter encoding the impact of adding $\boldsymbol{x}_+$. Therefore, $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$ varies on the limit-state surface $\{\boldsymbol{x}_+ \in \mathbb{X} : \mu_n(\boldsymbol{x}_+) = 0\}$, as depicted by the red dashed line in Fig. 6(e). Furthermore, the points belonging to $\{\boldsymbol{x}_+ \in \mathbb{X} : \mu_n(\boldsymbol{x}_+) = 0\}$ do not necessarily produce the biggest reduction of IPM. Compared to PM,* $\mathrm{IPMR}^{\mathrm{U}}$ *quantifies the impact of adding $\boldsymbol{x}_+$ on the reduction of IPM. Besides,* $\mathrm{IPMR}^{\mathrm{U}}$ *just consumes comparable running time to PM, owing to the pruning of both the quadrature set and candidate pool. Actually, the proposed* $\mathrm{IPMR}^{\mathrm{U}}$ *can be viewed as a learning function derived from combining reinforcement learning and PM to some extent. Hence, comparisons between them may highlight the benefit of incorporating reinforcement learning paradigm, which will be shown in Section 5.*

Remark 3. *Two existing learning functions, least improvement function (LIF) [41] and expected integrated error reduction (EIER) [42], were developed with the same objective of maximizing the reduction of IPM. Taking EIER as an example (refer to Appendix H), the advantages of the proposed IPMR over EIER are three-fold. (1) Eq. (H.4) shows that EIER can be viewed as a crude version of IPMR. Different from EIER resorting to the crude MCS to approximate the inner expectation and performing the retraining of Kriging via complex block matrix inversion (Eq. (H.5)), the proposed IPMR obtains the analytical expression of $I_n(\boldsymbol{x}; \boldsymbol{x}_+)$ in Eq. (30), with the aid of Kriging update formulas. Furthermore, $I_n(\boldsymbol{x}; \boldsymbol{x}_+)$ is substituted by its upper bound $I_n^{\mathrm{U}}(\boldsymbol{x}; \boldsymbol{x}_+)$ in Eq. (31), supporting fast computation. (2) In IPMR, the non-negativity of the integrand $I_n(\boldsymbol{x}; \boldsymbol{x}_+)$ has been proved in Appendix F, eliminating the need for the* $\max(\cdot, 0)$ *operation in Eq. (H.5). This leads to a theoretically rigorous and mathematically concise expression. (3) The double summation in EIER makes it computationally unaffordable when a large-size candidate pool is considered. By contrast, owing to the pruning of both the quadrature set and candidate pool, IPMR comes with a comparable computational cost to those common learning functions, e.g., U function.*

4. Bayesian reinforcement learning reliability method

Apart from Bayesian inference of $\widehat{P}_{f,n}$ (Section 2.1) and the learning function $\mathrm{IPMR}^{\mathrm{U}}$ in Section 3, another two components of the proposed BRLR method need to be specified. Section 4.1 presents the initial ED, and Section 4.2 elucidates the convergence criterion. Then, the setting of VAIS is discussed in Section 4.3. Finally, the implementation of the proposed BRLR method is given in Section 4.4.

4.1. Initial experimental design

To obtain a well-behaved Kriging at the initial stage, the initial ED is preferred to be as uniformly as possible. The 'four-sigma' rule is considered, where the upper and lower bounds for each dimension of the sampling domain $\mathbb{X}_{\mathrm{s}}$ are set as

$$x_k^{\pm} = F_{X_k}^{-1}\big(\Phi(\pm 4)\big),\ k = 1, \ldots, d \tag{41}$$

where $F_{X_k}(\cdot)$ is the CDF of the kth component X_k in $\boldsymbol{X}$. Then, the $\mathbb{X}_{\mathrm{s}}$ is assembled by the following tensorization

$$\mathbb{X}_{\mathrm{s}} = \prod_{k=1}^{d} [x_k^-, x_k^+] \tag{42}$$

Finally, the Latin centroidal Voronoi tessellation technique [43] is employed to generate the uniform points $\mathcal{X}_{n_0} = \{\boldsymbol{x}^{(i)}\}_{i=1}^{n_0}$ within $\mathbb{X}_{\mathrm{s}}$, with $n_0 = \max(d+1, 10)$.

4.2. Convergence criterion

A hybrid convergence criterion considering two individual ones simultaneously is developed here. First, recall that the IPM H_n in Eq. (20) measures the epistemic uncertainty about $\mu_{\widehat{P}_{f,n}}$, and the primary goal of IPMR is to reduce the IPM at most per iteration. Hence, the H_n itself can be used to check the convergence of sequential experimental design process. For compatibility, the metric $\frac{H_n}{\mu_{\widehat{P}_{f,n}}}$ is used in the first convergence criterion, defined as

$$\Delta_{H_n} = \frac{\frac{H_n}{\mu_{\widehat{P}_{f,n}}}}{\max\limits_{n_0 \le i \le n} \left(\frac{H_i}{\mu_{\widehat{P}_{f,i}}} \right)} \le \varepsilon_H \tag{43}$$

which stipulates that $\frac{H_n}{\mu_{\widehat{P}_{f,n}}}$ should fall below ε_H times its highest ever value. Eq. (21) indicates that H_n is the upper bound for the absolute relative error of $\mu_{\widehat{P}_{f,n}}$ in expectation, but the specific level of excess may vary with the problems at hand. Here, the tolerance ε_H is specified as 0.4 and 0.5 in static and dynamic reliability problems, respectively.

The second convergence criterion is defined based on the stabilization of $\mu_{\widehat{P}_{f,n}}$, given by

$$\Delta_{P_{f,n}} = \frac{\left| \mu_{\widehat{P}_{f,n}} - \mu_{\widehat{P}_{f,n-1}} \right|}{\mu_{\widehat{P}_{f,n-1}}} \le \varepsilon_{P_f} \tag{44}$$

where the tolerance ε_{P_f} is set as 5×10^{-3}.

Finally, the hybrid convergence criterion requires that

$$\left(\Delta_{H_n} \le \varepsilon_H\right) \bigcap \left(\Delta_{P_{f,n}} \le \varepsilon_{P_f}\right) \tag{45}$$

within two successive iterations.

4.3. Settings of VAIS in the IPMR-based sequential experimental design

During the IPMR-based sequential experimental design process, a total of three single integrals, i.e., $\mu_{\widehat{P}_{f,n}}$ in Eq. (9), H_n in Eq. (20), and $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+), \forall \boldsymbol{x}_+ \in \mathcal{X}_{\mathrm{CT}}$ (Eq. (38)), need to be numerically estimated via the VAIS. Notably, the former two integrals only involve a single computation at each iteration, while the last one entails a total of C_{T} runs of computation at $\mathcal{X}_{\mathrm{CT}}$.

With respect to $\mu_{\widehat{P}_{f,n}}$ and H_n, the corresponding VAIS-based estimates are expressed as

$$\tilde{\mu}_{\widehat{P}_{f,n}} = \frac{1}{Q_1} \sum_{i=1}^{Q_1} \left[\Phi\left(-\frac{\mu_n(\boldsymbol{x}^{(i)})}{\sigma_n(\boldsymbol{x}^{(i)})} \right) \frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})} \right] \tag{46}$$

$$\tilde{H}_n = \frac{1}{Q_2} \sum_{i=1}^{Q_2} \left[\Phi\left(-\frac{\left|\mu_n(\boldsymbol{x}^{(i)})\right|}{\sigma_n(\boldsymbol{x}^{(i)})} \right) \frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})} \right] \tag{47}$$

respectively. Then, the variances of $\tilde{\mu}_{\widehat{P}_{f,n}}$ and $\tilde{H}_n$ are expressed as

$$\mathbb{V}\left[\tilde{\mu}_{\widehat{P}_{f,n}}\right] = \frac{1}{Q_1(Q_1-1)} \sum_{i=1}^{Q_1} \left[\Phi\left(-\frac{\mu_n(\boldsymbol{x}^{(i)})}{\sigma_n(\boldsymbol{x}^{(i)})} \right) \frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})} - \tilde{\mu}_{\widehat{P}_{f,n}} \right]^2 \tag{48}$$

$$\mathbb{V}\left[\tilde{H}_n\right] = \frac{1}{Q_2(Q_2-1)} \sum_{i=1}^{Q_2} \left[\Phi\left(-\frac{\left|\mu_n(\boldsymbol{x}^{(i)})\right|}{\sigma_n(\boldsymbol{x}^{(i)})} \right) \frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})} - \tilde{H}_n \right]^2 \tag{49}$$

respectively. In this way, their coefficient of variation (COV)s are given as $\mathrm{COV}\left[\tilde{\mu}_{\widehat{P}_{f,n}}\right] = \frac{\sqrt{\mathbb{V}\left[\tilde{\mu}_{\widehat{P}_{f,n}}\right]}}{\tilde{\mu}_{\widehat{P}_{f,n}}}$ and $\mathrm{COV}\left[\tilde{H}_n\right] = \frac{\sqrt{\mathbb{V}[\tilde{H}_n]}}{\tilde{H}_n}$, respectively.

For convenience, set $Q_1 = Q_2 = Q$, and the Q is considered to be sufficient when the following expression is satisfied

$$\left(\mathrm{COV}\left[\tilde{\mu}_{\widehat{P}_{f,n}}\right] \le \varepsilon_{\mathrm{Q}}\right) \bigcap \left(\mathrm{COV}\left[\tilde{H}_n\right] \le \varepsilon_{\mathrm{Q}}\right) \tag{50}$$

where the tolerance ε_{Q} is set as 5%.

The VAIS-based estimators in Eqs. (46) and (47) are conducted in an adaptive manner during the sequential experimental design process. Specifically, set the amplification coefficient α as 1.5, and the initial quadrature size Q_{seq} as 2×10^5. At the beginning of the sequential experimental design process, Eqs. (46) and (47) are computed based on $\mathcal{X}_{\mathrm{Q}} = \left\{\boldsymbol{x}^{(i)}\right\}_{i=1}^{Q_{\mathrm{seq}}}$; then, if the current values of $\mathrm{COV}\left[\tilde{\mu}_{\widehat{P}_{f,n}}\right]$ and $\mathrm{COV}\left[\tilde{H}_n\right]$ fail to satisfy Eq. (50), another set of Q_{seq} quadrature points is sequentially added into the current $\mathcal{X}_{\mathrm{Q}}$ until Eq. (50) is fulfilled. Notably, the quadrature size at the current iteration is taken as the initial quadrature size at the next iteration during the sequential experimental design process. Generally, after several iterations, the quadrature size will remain unchanged at subsequent iterations. Moreover, since the integrand $I_n(\boldsymbol{x}; \boldsymbol{x}_+)$ in $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$ is similar to the integrand in H_n, the quadrature size Q for $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$ is directly set to be equal to that of $\mu_{\widehat{P}_{f,n}}$ and H_n at each iteration.

Remark 4. *The transformation of input random vector $\boldsymbol{Z}$ to standard normal vector $\boldsymbol{X}$ in Eq. (3) is mainly attributed to the usage of VAIS in the numeric computation of $\mu_{\widehat{P}_{f,n}}$, H_n and $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$. Therefore, if other favorable integration methods conducted in the original input space are available, such transformation is unnecessary.*

For illustration, the performances of VAIS in two different examples are shown in Appendix G. It is observed that the quadrature set $\mathcal{X}_{\mathrm{Q}}$ whose size Q determined by Eq. (50) provides stable results of $\tilde{\mu}_{\widehat{P}_{f,n}}$ and $\tilde{H}_n$. Then, the pruning coefficient $\lambda = 1.5$ is a very favorable choice for $\mathrm{IPMR}^{\mathrm{U}}$.

To summarize, an identical quadrature set $\mathcal{X}_{\mathrm{Q}}$ is used in the computation of $\mu_{\widehat{P}_{f,n}}$, H_n, and $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+), \forall \boldsymbol{x}_+ \in \mathcal{X}_{\mathrm{CT}}$, with the quadrature size Q determined by Eq. (50). The pruning coefficient λ in $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$ is set as 1.5 for reassurance. Consequently, the size Q_{T} of the pruned quadrature set $\mathcal{X}_{\mathrm{QT}}$ in $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$ is generally $\mathcal{O}(10^3)$, and very minor running time is consumed by $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$ per iteration. Besides, $\mathcal{X}_{\mathrm{Q}}$ serves as the candidate pool $\mathcal{X}_{\mathrm{C}}$ for selecting the best next point $\boldsymbol{x}^{(n+1)}$ via $\mathrm{IPMR}^{\mathrm{U}}$.

Finally, Algorithm 1 presents the workflow of a single iteration during the IPMR-based sequential experimental design. Thanks to the three workarounds, i.e., substituting IPMR by its upper bound $\mathrm{IPMR}^{\mathrm{U}}$ (Steps 7 and 8), the pruning of $\mathcal{X}_{\mathrm{Q}}$ (Step 4), and the pruning of $\mathcal{X}_{\mathrm{C}}$ (Step 5), IPMR-based sequential experimental design comes with a comparable running time to those common learning functions.

Algorithm 1 IPMR-based sequential experimental design

Input: Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ and the quadrature set $\mathcal{X}_{\mathrm{Q}}$ of size Q at iteration n.

1: Set $\mathcal{X}_{\mathrm{C}} = \mathcal{X}_{\mathrm{Q}}$.
2: Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ provides posterior mean $\mu_n(\boldsymbol{x})$ and variance $\sigma_n^2(\boldsymbol{x})$ at $\mathcal{X}_{\mathrm{Q}}$. ▷ Eqs. (A.6) and (A.7)
3: Compute the PM $P_n(\boldsymbol{x})$ at $\mathcal{X}_{\mathrm{Q}}$. ▷ Eq. (H.1)
4: Obtain the pruned quadrature set $\mathcal{X}_{\mathrm{QT}}$ of size Q_{T} from $\mathcal{X}_{\mathrm{Q}}$. ▷ Eq. (37)
5: Obtain the pruned candidate pool $\mathcal{X}_{\mathrm{CT}}$ of size C_{T} from $\mathcal{X}_{\mathrm{C}}$. ▷ Eq. (39)
6: Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ provides posterior covariance matrix $\left[c_n(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})\right]_{1 \le i \le Q_{\mathrm{T}}, 1 \le j \le C_{\mathrm{T}}}$ at $\mathcal{X}_{\mathrm{QT}} \times \mathcal{X}_{\mathrm{CT}}$. ▷ Eq. (A.8)
7: Compute $\left[I_n^{\mathrm{U}}(\boldsymbol{x}^{(i)}; \boldsymbol{x}^{(j)})\right]_{1 \le i \le Q_{\mathrm{T}}, 1 \le j \le C_{\mathrm{T}}}$ at $\mathcal{X}_{\mathrm{QT}} \times \mathcal{X}_{\mathrm{CT}}$. ▷ Eq. (31)
8: Compute $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}^{(j)}), \forall \boldsymbol{x}^{(j)} \in \mathcal{X}_{\mathrm{CT}}$. ▷ Eq. (38)
9: Select the best next point $\boldsymbol{x}^{(n+1)}$ from $\mathcal{X}_{\mathrm{CT}}$. ▷ Eq. (40)

Output: The $\boldsymbol{x}^{(n+1)}$ at iteration n.

Remark 5. *A learning function called stepwise uncertainty reduction (SUR) was derived from the upper bound of $\sigma^2_{\widehat{P}_{f,n}}$ [28] and is outlined in Appendix H. Similar to the integrand $I_n(\boldsymbol{x}; \boldsymbol{x}_+)$ in $\mathrm{IPMR}_n(\boldsymbol{x}_+)$, $\mathrm{SUR}_n(\boldsymbol{x}_+)$ also involves the bivariate Gaussian CDF $F_2(\cdot;\cdot,\cdot)$ that has to be computed element-wise, as*

given in Eq. (H.8). *Besides, the locality of the integrand in* $\mathrm{SUR}_n(\cdot)$ *and the explicit impact of* $\boldsymbol{x}_+$ *on* $\mathrm{SUR}_n(\cdot)$ *are actually difficult to explore. Hence, the pruning of both quadrature set and candidate pool did not be conducted in the existing literature. Obviously, compared to* $\mathrm{IPMR}^{\mathrm{U}}$*, SUR suffers from intensive computational burden.*

4.4. Implementation

The implementation of the proposed BRLR method is outlined in Fig. 7, and the main steps are summarized as follows.

(1) *Initialization.* Generate a set of input samples $\mathcal{X}_{n_0}$ and evaluate the performance function on $\mathcal{X}_{n_0}$ to obtain $\mathcal{Y}_{n_0}$, forming the initial ED $\mathcal{D}_{n_0} = \{\mathcal{X}_{n_0}, \mathcal{Y}_{n_0}\}$ (Section 4.1); then, set $n = n_0$.

(2) *Kriging.* Train a Kriging $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ based on the current ED $\mathcal{D}_n$, as detailed in Appendix A.

(3) *Failure probability.* Estimate the posterior mean $\mu_{\widehat{P}_{f,n}}$ (Eq. (46)) and the IPM H_n (Eq. (47)) using VAIS, with the quadrature size Q determined according to Eq. (50). Meanwhile, the corresponding quadrature set $\mathcal{X}_{\mathrm{Q}}$ serves as the candidate pool $\mathcal{X}_{\mathrm{C}}$ at this iteration.

(4) *Convergence criterion.* If Eq. (45) is satisfied, skip to Step 7; otherwise, continue to Step 5.

(5) *Learning function.* Compute $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$ at $\mathcal{X}_{\mathrm{CT}}$ and select the best next point $\boldsymbol{x}^{(n+1)}$, as detailed in Algorithm 1.

(6) *Enrichment.* Evaluate the performance function on $\boldsymbol{x}^{(n+1)}$, that is, $y^{(n+1)} = \mathcal{G}\left(\boldsymbol{x}^{(n+1)}\right)$. Then, $\mathcal{D}_{n+1} = \mathcal{D}_n \bigcup \left(\boldsymbol{x}^{(n+1)}, y^{(n+1)}\right)$, $n = n+1$, and go back to Step 2.

(7) *End.* The $\mu_{\widehat{P}_{f,n}}$ at Step 3 is considered the final result of this algorithm.

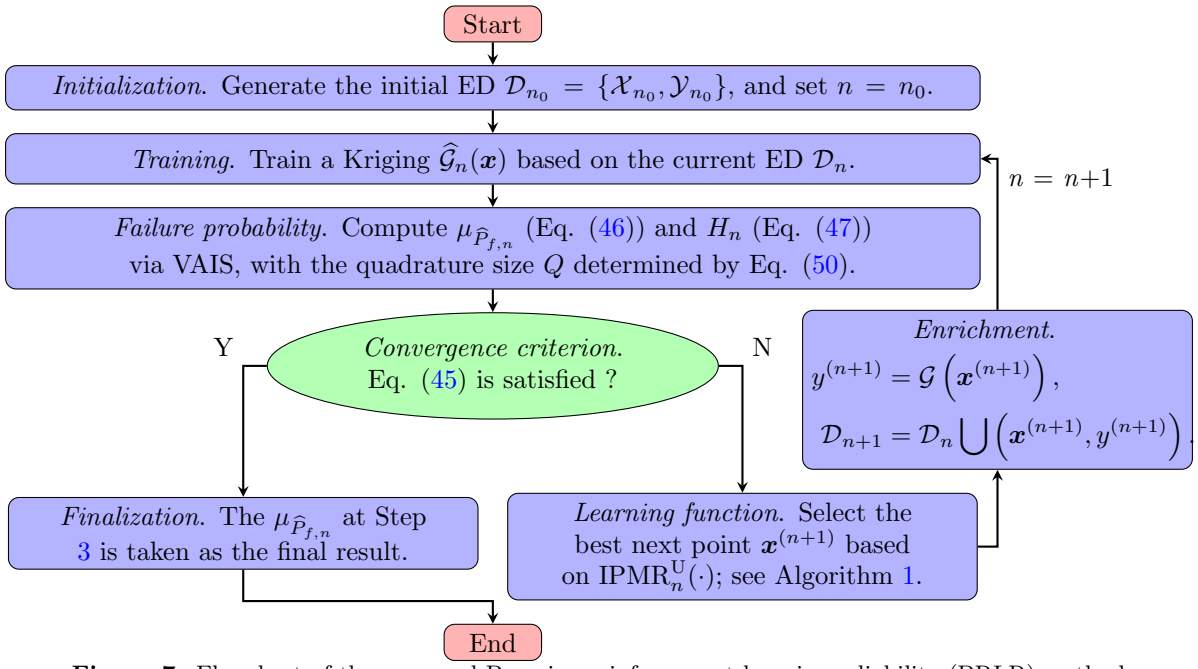


Figure 7: Flowchart of the proposed Bayesian reinforcement learning reliability (BRLR) method

To avoid confusion, 'the proposed method' exclusively refers to the BRLR method equipped with $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$, while 'the proposed method (exact)' specifically denotes the one incorporating $\mathrm{IPMR}_n(\cdot)$.

5. Numerical examples

The performance of the proposed reliability method is demonstrated through four examples of varying complexity. The MCS is conducted to provide the reference failure probability $\widehat{P}_f^{\mathrm{MCS}}$. For comparison, several existing reliability methods are conducted, including adaptive Krigng-subset simulation (AK-SS) and the BALR methods with several existing learning functions. AK-SS is built following the recommendations from [14]: the sample size of each subset and the conditional probability in SS are set as 10^5 and 0.15, respectively; the learning function PM (Eq. (H.1)) is considered; the convergence criterion is defined in terms of the combination of 'β-bound' and 'β-stable' criteria, with β the reliability index; the tolerances for the two criteria are set as 15% and 0.1%, respectively, The initial ED and convergence criterion in the

BALR methods are set consistent with the proposed reliability method (Fig. 7) for comparison purposes. Five existing learning functions, including PM [44], RLCB [31], REIF [32], EIER [42] and SUR [28], are outlined in Appendix H. As elucidated in Remarks 3 and 5, both EIER and SUR suffer from unaffordable computational burden, due to their inherent limitations. For computational cost considerations, they are not conducted in those examples and only $\mathrm{IPMR}_n(\cdot)$ will be compared with $\mathrm{IPMR}_n^{\mathrm{U}}(\cdot)$ in Section 5.4. Besides, results from other reliability methods available in the literature are provided for comparison.

In a reliability method, the total number N_{call} of performance function evaluations and the estimated failure probability $\widehat{P}_f$ are taken as the metrics of computational efficiency and accuracy, respectively. Then, the ALR, BALR, and BRLR methods are repeated 10 times to gain the means and COVs of N_{call} and $\widehat{P}_f$. Further, the relative error of $\widehat{P}_{f,\mathrm{mean}}$ with respect to $\widehat{P}_f^{\mathrm{MCS}}$ is computed as

$$\delta_{\widehat{P}_f} = \frac{\left|\widehat{P}_{f,\mathrm{mean}} - \widehat{P}_f^{\mathrm{MCS}}\right|}{\widehat{P}_f^{\mathrm{MCS}}} \times 100\% \tag{51}$$

Besides, the mean of the computational time T_{c} of each reliability method is provided in the last two numerical examples.

5.1. A four-branch function

The first example considers a four-branch problem [12, 13], which is a prevalent benchmark in structural reliability analysis. The performance function $\mathcal{G}(\boldsymbol{X})$ is expressed as

$$\mathcal{G}(\boldsymbol{X}) = \min \begin{Bmatrix} a + 0.1(X_1 - X_2)^2 - \dfrac{X_1 + X_2}{\sqrt{2}} \\ a + 0.1(X_1 - X_2)^2 + \dfrac{X_1 + X_2}{\sqrt{2}} \\ (X_1 - X_2) + \dfrac{b}{\sqrt{2}} \\ (X_2 - X_1) + \dfrac{b}{\sqrt{2}} \end{Bmatrix} \tag{52}$$

where X_1 and X_2 are two independent, standard Gaussian random variables; the two constants a and b govern the order of magnitude of P_f. Two different cases are considered: $a = 3$ and $b = 6$ in the first one; $a = 5$ and $b = 9$ in the second one.

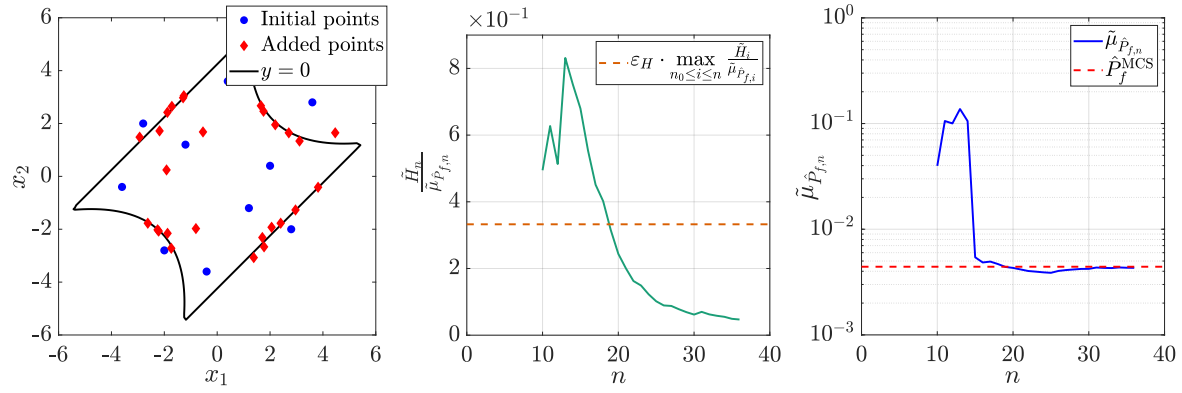
5.1.1. Case 1: $a = 3$ and $b = 6$

The failure probability $\widehat{P}_f^{\mathrm{MCS}} = 4.416 \times 10^{-3}$ provided by MCS is taken as the reference result. Fig. 8 illustrates the performance of one run of the proposed reliability method for the four-branch function (Case 1). The initial samples, as marked as blue circles, are scattered across the entire input space, and most new training samples added by $\mathrm{IPMR}^{\mathrm{U}}$, as plotted as red diamonds, are located in the close vicinity of the limit state, as shown in Fig. 8(a). During the IPMR-informed sequential experimental design process, the IPM H_n gradually shrinks (Fig. 8(b)), implying that the epistemic uncertainty about $\mu_{\widehat{P}_{f,n}}$ is significantly reduced. Consequently, it is evident from Fig. 8(c) that the $\mu_{\widehat{P}_{f,n}}$ produced by the proposed method gradually converges to the reference value.

Table 1 providess a comparison of the results obtained from various reliability methods for the four-branch function (Case 1). It is observed that most reliability methods provide accurate estimates of failure probability, with relative errors below 2%. Compared to other reliability methods, the proposed approach requires much fewer calls to the performance function.

5.1.2. Case 2: $a = 5$ and $b = 9$

The failure probability $\widehat{P}_f^{\mathrm{MCS}} = 7.09 \times 10^{-6}$ produced by MCS is regarded as the reference result. Table 2 provides a comparison of the results of various reliability methods for the four-branch function (Case 2). In the BALR methods, PM achieves favorable performance in terms of $\widehat{P}_f$ and N_{call}. However, RLCB and REIF provide biased estimates of the failure probability, with relative errors above 4%. By contrast, the proposed reliability method provides comparable accuracy of $\widehat{P}_f$ to PM, while requiring a smaller number of calls to the performance function.



(a) sample location of training samples (b) $\frac{\tilde{H}_n}{\tilde{\mu}_{\hat{P}_{f,n}}}$ (c) $\tilde{\mu}_{\hat{P}_{f,n}}$

Figure 8: Illustration of the proposed reliability method in the four-branch function (Case 1)

Table 1: Reliability results in the four-branch function (Case 1)

Method		N_{call}		$\widehat{P}_f$		$\delta_{\widehat{P}_f}(\%)$
		Mean	COV(%)	Mean ($\times 10^{-3}$)	COV(%)	
MCS [12]		10^6	-	4.416	-	-
AK-MCS	U [12]	96	-	4.416	-	-
	LIF [41]	38	-	4.380	-	0.815
	REIF [32]	146.8	-	4.455	4.730	0.883
	LAKSE [45]	65.6	-	4.065	4.950	7.948
	Refined U [46]	76	-	4.432	-	0.362
	KO [47]	90	-	4.182	4.880	5.299
	WKO [47]	71.4	-	4.448	4.730	0.725
PAK-B^n [48]		76.8	-	4.422	-	0.136
AK-KB^n [48]		74.7	-	4.419	-	0.068
BALR	PM	42.4	9.447	4.370	1.457	1.047
	RLCB	51.1	14.095	4.397	2.235	0.424
	REIF	57.4	13.942	4.346	2.514	1.587
AK-SS		52.2	21.290	4.472	2.963	1.274
Proposed		37	18.063	4.371	0.690	1.027

Table 2: Reliability results in the four-branch function (Case 2)

Method	N_{call}		$\widehat{P}_f$		$\delta_{\widehat{P}_f}(\%)$
	Mean	COV(%)	Mean ($\times 10^{-6}$)	COV(%)	
MCS	10^8 [27]	-	7.090	-	-
AK-MCMC [27]	139.5	-	7.100	1.370	0.141
PA-BFPL (k=5) [27]	60	-	7.040	2.170	0.705
BALR (PM)	37.5	10.536	7.147	0.875	0.806
BALR (RLCB)	45.7	12.466	7.421	2.339	4.674
BALR (REIF)	49.4	9.552	7.382	1.143	4.116
AK-SS	40.6	28.279	7.306	17.519	3.049
Proposed	34.5	12.272	7.038	1.650	0.738

5.2. A two-dimensional truss under vertical loads

Consider the static reliability analysis of a two-dimensional truss, which is also a common benchmark in structural reliability analysis [44, 13]. Fig. 9 shows that this truss is composed of 23 bars and 13 nodes. The random input vector is assembled as $\boldsymbol{Z} = \{E_1, E_2, A_1, A_2, P_1, \ldots, P_6\}$, where A_1 and E_1 are the cross section and Young's modulus of horizontal bars, respectively; A_2 and E_2 are the cross section and Young's modulus of diagonal bars, respectively; $P_1, \ldots, P_6$ are the vertical loads applied on the upper nodes of the truss. The statistical information for those parameters is listed in Table 3.

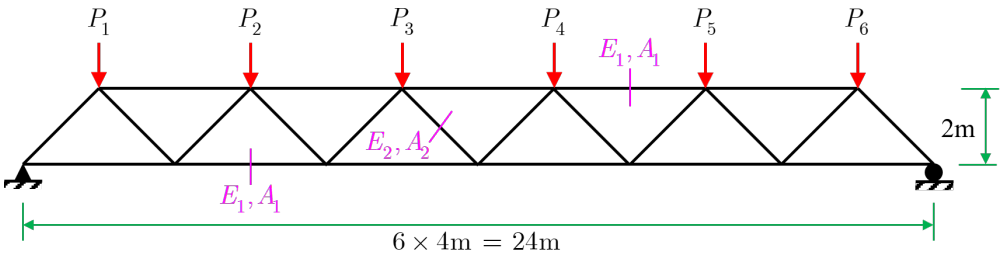


Figure 9: Illustration of a planar truss structure

Table 3: Statistical information of input variables in the planar truss

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}
$P_1, \cdots, P_6$	N	Gumbel	5.0×10^{4}	7.5×10^{3}

Finite-element analysis of this truss is conducted by an in-house MATLAB code. The vertical deflection of the mid-span node, denoted as $U(\boldsymbol{Z})$, is of interest. The maximum allowable mid-span deflection is set to 14 mm. Then, the performance function $\mathcal{G}(\boldsymbol{Z})$ is defined as

$$\mathcal{G}(\boldsymbol{Z}) = 14 - U(\boldsymbol{Z}) \tag{53}$$

and the failure probability $\widehat{P}_f^{\text{MCS}} = 3.45 \times 10^{-5}$ offered by MCS is taken as the reference result.

Fig. 10 illustrates the comparison of three learning functions, i.e., the proposed IPMR^{U}, PM and RLCB, in the planar truss example. Overall, the IPM H_n (Eq. (20)) is substantially reduced by all three learning functions, but their rates of decrease differ significantly. IPMR^{U} enables reducing the IPM H_n to the targeted level with only 35 runs of finite-element analysis. By contrast, PM requires approximately 60 runs of finite element analysis to achieve the targeted reduction of IPM H_n. RLCB entails approximately 130 calls to performance function but with a worse estimate of the failure probability.

The significant differences between the three learning functions can be attributed to their individual objectives. RLCB (Eq. (H.2)) aims to empirically balance the closeness of the Kriging mean to the limit state and the Kriging variance, independently of the IPM. Consequently, RLCB struggles to efficiently reduce the IPM H_n. PM selects the point maximizing the PM value as the best next point per iteration, but it does not explicitly quantify the impact of adding a new point on the reduction of IPM. Hence, the new point added by PM may not necessarily yield the maximum reduction of IPM. In contrast, the new point selected by IPMR^{U} is optimal with respect to reducing the IPM per iteration, leading to the fastest reduction of IPM.

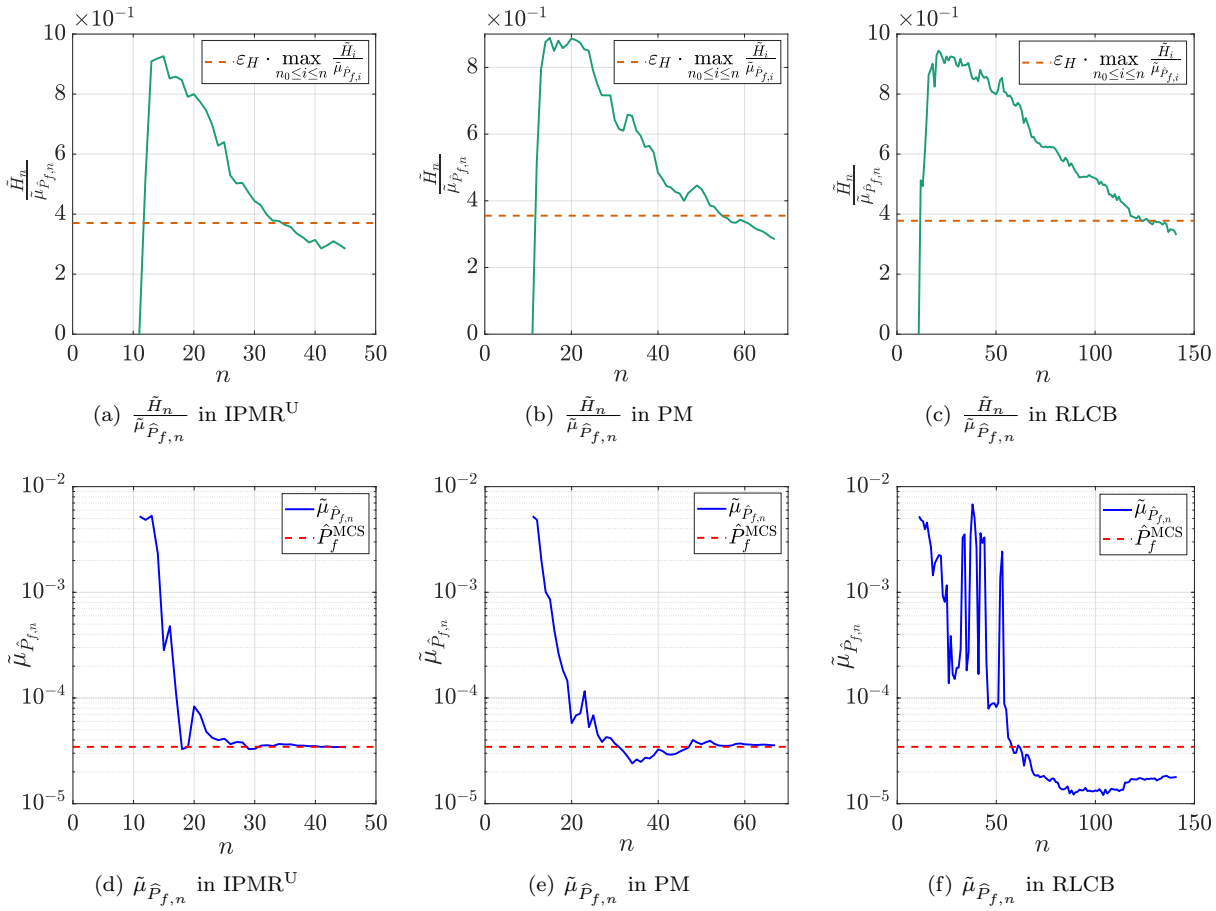
Table 4 presents the results obtained from different reliability methods for this planar truss example. Both RLCB and REIF produce biased estimates of the failure probability but require a significant number of finite element analysis, due to the mismatch between their objectives and IPM. In both BALR and AK-SS, PM provides fair accuracy of $\widehat{P}_f$, with $\delta_{\widehat{P}_f}$ approximately 2%. Compared to PM, the proposed IPMR^{U} achieves a 33.2% reduction in N_{call}.

5.3. A reinforced concrete frame under earthquakes

A three-bay, six-story planar reinforced concrete frame subject to earthquake excitation is considered here. Fig. 11 illustrates the basic geometry of the frame, along with the reinforcement details of columns and beams. The concrete slab at each floor has a thickness of 100 mm. The finite-element model of this frame is built using the OpenSees software [50]. Both columns and beams are modeled using force-based elements with fiber-discretized cross sections. The uniaxial constitutive relationships of concrete and rebar are represented by the Concrete-01 and Steel-01 material models, respectively. The in-plane stiffness of the concrete slab is simply described by the 'RigidLink' command, and the self-weights of these concrete slabs are treated as uniformly-distributed loads applied on the beams beneath them. Rayleigh damping is adopted with the damping ratio of 5%.

A simple unidirectional earthquake ground motion model is considered, given by [18]

$$\ddot{u}_g(t) = A_1 \ddot{u}_{\text{NS}}(t) + A_2 \ddot{u}_{\text{WE}}(t) \tag{54}$$



(a) $\frac{\tilde{H}_n}{\tilde{\mu}_{\hat{P}_{f,n}}}$ in IPMR$^{\mathrm{U}}$ (b) $\frac{\tilde{H}_n}{\tilde{\mu}_{\hat{P}_{f,n}}}$ in PM (c) $\frac{\tilde{H}_n}{\tilde{\mu}_{\hat{P}_{f,n}}}$ in RLCB

(d) $\tilde{\mu}_{\hat{P}_{f,n}}$ in IPMR$^{\mathrm{U}}$ (e) $\tilde{\mu}_{\hat{P}_{f,n}}$ in PM (f) $\tilde{\mu}_{\hat{P}_{f,n}}$ in RLCB

Figure 10: Comparison between IPMR$^{\mathrm{U}}$, PM and RLCB in the planar truss example

Table 4: Reliability results in the planar truss example

Method	N_{call}		$\widehat{P}_f$		$\delta_{\widehat{P}_f}(\%)$
	Mean	COV(%)	Mean ($\times 10^{-5}$)	COV(%)	
MCS	10^6	-	3.450	-	-
AK-MCS(U) [44]	124	-	3.700	-	7.246
AK-MCS(LIF) [41]	121	-	3.31	-	4.058
AKEE-SS [49]	80	-	3.247	-	5.884
BALR (PM)	67.7	12.241	3.377	6.815	2.110
BALR (RLCB)	137.4	7.534	1.969	12.141	42.938
BALR (REIF)	167.8	5.336	1.987	11.319	42.420
AK-SS	69.6	14.544	3.534	6.017	2.425
Proposed	45.2	11.973	3.447	2.883	0.097

where $\ddot{u}_{\mathrm{NS}}(t)$ and $\ddot{u}_{\mathrm{WE}}(t)$ are the amplitude-normalized components of El-Centro accelerogram in N-S and W-E directions, respectively, as show in Figs. 12(a) and 12(b); A_1 and A_2 are the corresponding amplitude coefficients.

The random input vector is assembled by both material parameters and amplitude coefficients, that is, $\boldsymbol{Z} = \{f_{cc}, \varepsilon_{cc}, f_{cu}, \varepsilon_{cu}, f_c, \varepsilon_c, f_u, \varepsilon_u, f_y, E_0, b, A_1, A_2\}$. Then, statistic information for those random variables is given in Table 5. When those random variables take their means, the typical uniaxial stress-strain curves of concrete and rebar at the end section of the leftmost bottom column are shown in Figs. 13(a) and 13(b), respectively. Meanwhile, the typical hysteretic curve of the leftmost bottom column is illustrated in Fig. 13(c). Clearly, both material- and structure-level nonlinearity are observed.

The inter-story drift of this frame is of interest, and the threshold is set as 72 mm. Then, *system* failure

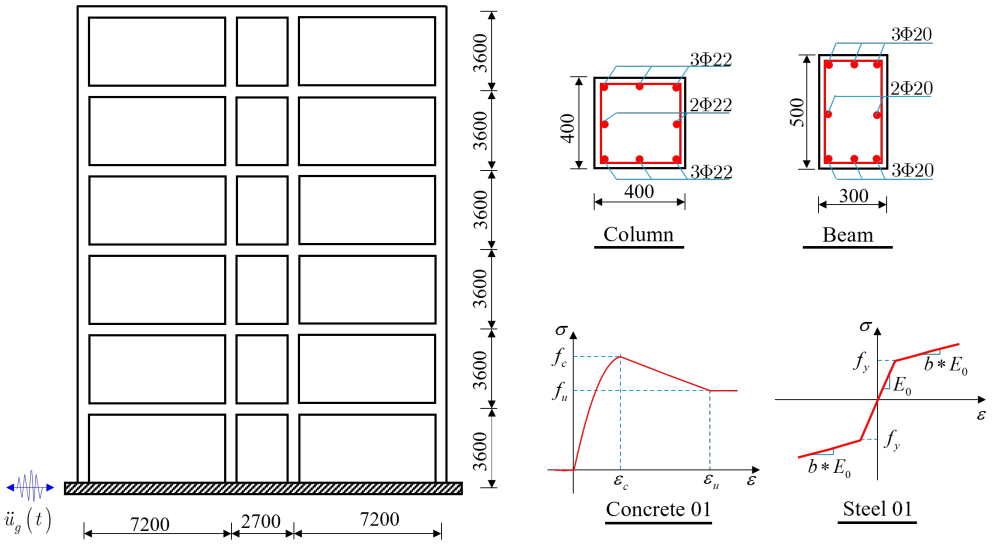
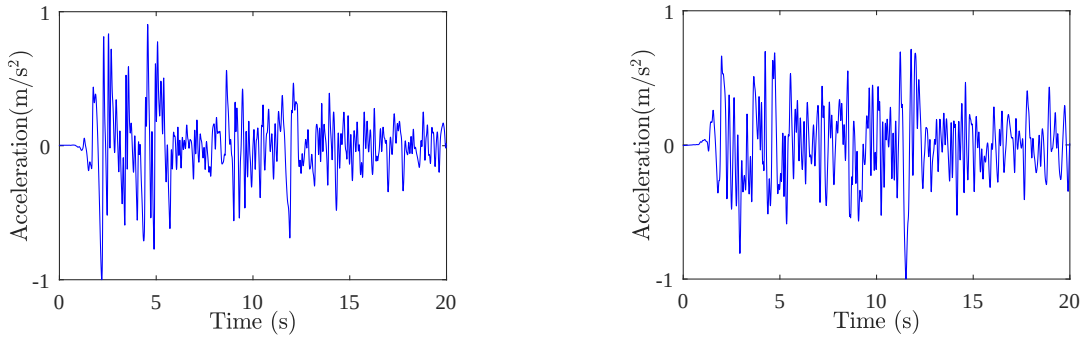


Figure 11: A planar reinforced concrete frame under earthquakes



(a) N-S component (b) W-E component

Figure 12: El-Centro accelerograms in two orthogonal directions

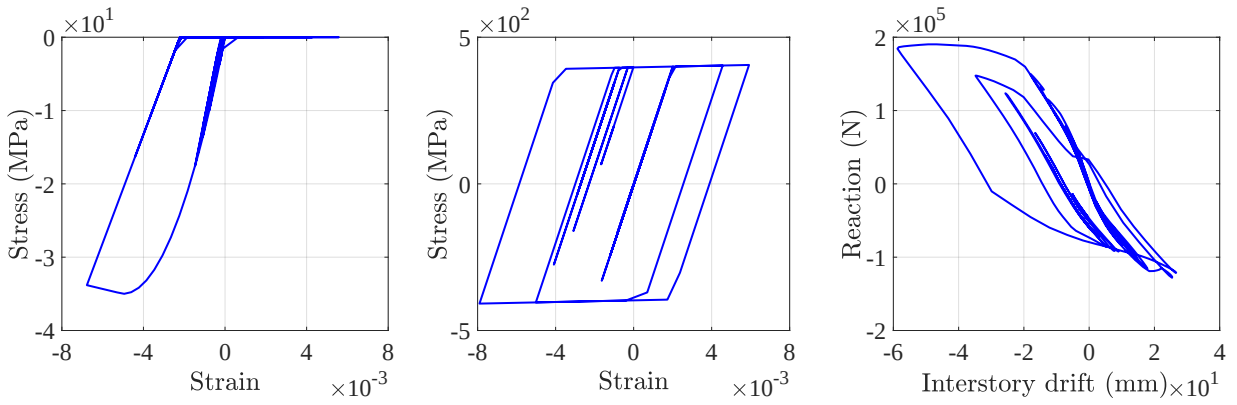
Table 5: Statistical information of random variables in the reinforced concrete frame [20]

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	MPa	Initial Young's modulus of rebar	Lognormal	200	0.1
b	-	Strain-hardening ratio of rebar	Lognormal	0.007	0.05
A_1	-	Amplitude coefficient	Gaussian	2	0.1
A_2	-	Amplitude coefficient	Gaussian	2	0.1

probability is defined as

$$P_f = \mathbb{P}\left(\bigcup_{i=1}^{6} |U_i(\boldsymbol{Z}, t)| \geq 72\right) \tag{55}$$

where $U_i(\boldsymbol{Z}, t), i = 1, \ldots, 6$, denotes the inter-story drift between the $(i-1)$-th and i-th floor. In this way,



(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

the *system* performance function is defined as

$$\mathcal{G}(\boldsymbol{Z}) = 72 - \max_{1\le i\le 6}\left(\max_{t\in[0,20\mathrm{s}]}|U_i(\boldsymbol{Z},t)|\right) \tag{56}$$

The $\widehat{P}_f^{\mathrm{MCS}}$ offered by MCS is equal to 2.967×10^{-5}.

Fig. 14 illustrates the comparison of the three learning functions, $\mathrm{IPMR}^{\mathrm{U}}$, PM and RLCB, for the reinforced concrete frame example. $\mathrm{IPMR}^{\mathrm{U}}$ achieves convergence at the cost of 102 runs of dynamic analysis of the frame. By comparison, PM requires over 200 runs of finite element analysis to achieve the targeted reduction of IPM. RLCB reduces the IPM H_n very slowly, with only a minor reduction gained when n reaches 300. For computational cost considerations, RLCB-based sequential experimental design is terminated at the iteration of $n = 300$, and the resulting P_f deviates significantly from the reference value $\widehat{P}_f^{\mathrm{MCS}}$. It is evident that, unlike PM and RLCB, $\mathrm{IPMR}^{\mathrm{U}}$ effectively addresses the challenge posed by nonlinear dynamic reliability problem.

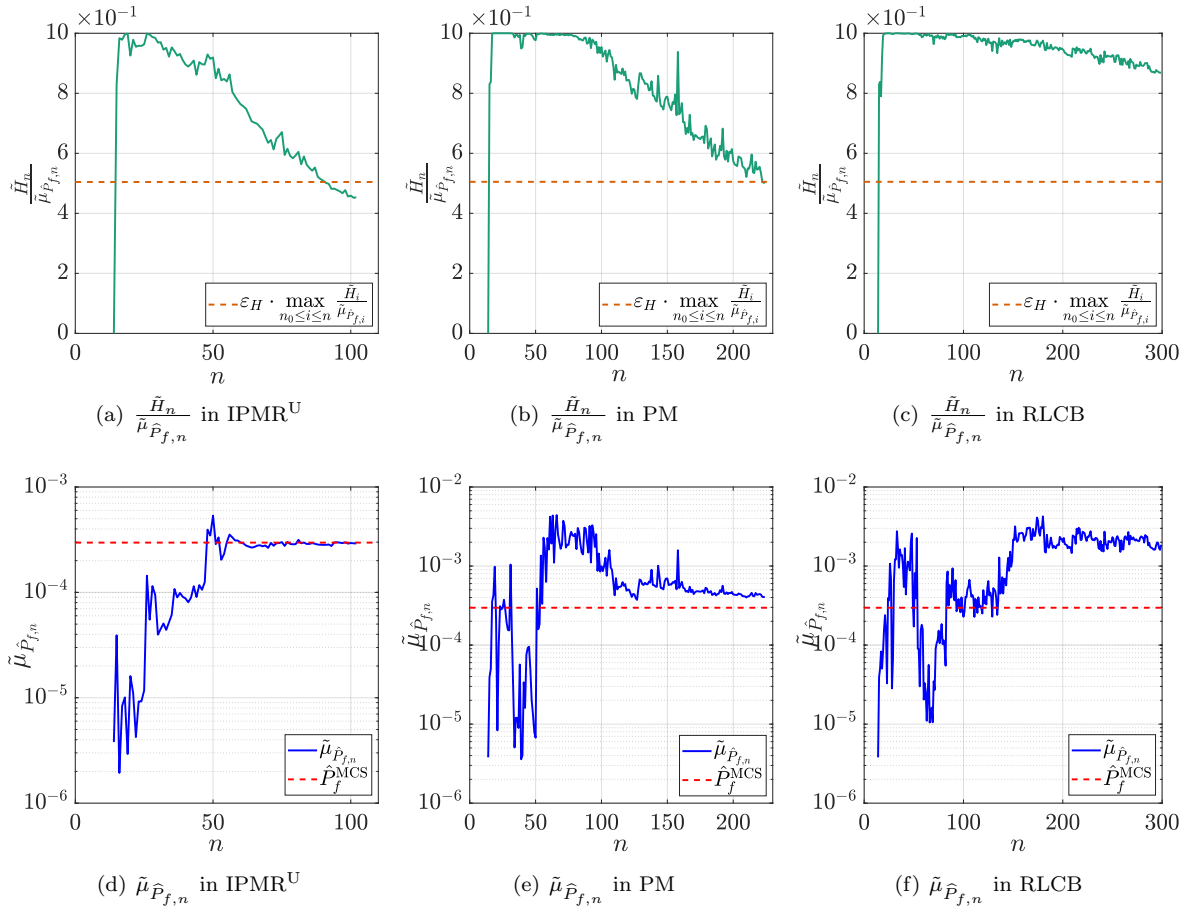
Table 6 lists the results obtained from different reliability methods for the reinforced concrete frame example, along with their computational times for comparison. Since the convergence criterion of AK-SS involves computing the failure probability and its upper and lower bounds, three runs of SS must be performed on Kriging per iteration. In this regard, despite AK-SS requiring a comparable N_{call} to BALR(PM), its total computational time far exceeds that of BALR(PM). In comparison to PM, the proposed $\mathrm{IPMR}^{\mathrm{U}}$ only needs 41.94% of N_{call}, showcasing its high computational efficiency. Evidently, the advantage of the proposed BRLR method over the existing BALR methods becomes more significant when dealing with dynamic reliability problems.

Table 6: Reliability results in the reinforced concrete frame example

Method	N_{call}		$\widehat{P}_f$		$\delta_{\widehat{P}_f}(\%)$	T_{c}(s)
	Mean	COV(%)	Mean ($\times10^{-4}$)	COV(%)		
MCS	5×10^5	-	2.967	-	-	2.926×10^6
BALR (PM)	248	8.038	4.075	8.357	37.359	3.896×10^3
BALR (RLCB)	>300	-	14.067	28.444	374.2	$>4.593\times10^3$
BALR (REIF)	>300	-	53.711	43.061	17104.8	$>4.499\times10^3$
AK-SS	280.6	5.392	2.824	5.195	4.822	1.112×10^4
Proposed	104.9	12.294	2.938	8.064	0.952	1.989×10^3

5.4. A cable-stayed bridge under vehicle loads

The final example considers the Sutong cable-stayed bridge that connects Suzhou and Nantong cities in China. As depicted in Fig. 15, it is a double-pane, twin-pylon, box-girder bridge with a main span of 2088 m. The deck is a streamlined, steel-box girder with a width of 41 m. The heights of the two inverted-Y pylons are 300 m. The stay cables are composed of parallel steel-wire strand and are arranged in double-inclined cable planes, giving rise to 272 cable members.



(a) $\frac{\tilde{H}_n}{\tilde{\mu}_{\hat{P}_{f,n}}}$ in IPMR$^{\mathrm{U}}$ (b) $\frac{\tilde{H}_n}{\tilde{\mu}_{\hat{P}_{f,n}}}$ in PM (c) $\frac{\tilde{H}_n}{\tilde{\mu}_{\hat{P}_{f,n}}}$ in RLCB

(d) $\tilde{\mu}_{\hat{P}_{f,n}}$ in IPMR$^{\mathrm{U}}$ (e) $\tilde{\mu}_{\hat{P}_{f,n}}$ in PM (f) $\tilde{\mu}_{\hat{P}_{f,n}}$ in RLCB

Figure 14: Comparison between IPMR$^{\mathrm{U}}$, PM and RLCB in the reinforced concrete frame example

Finite-element model of this bridge is constructed using the commercial software ANSYS [51], as illustrated in Fig. 16. The steel-box girders, towers, and piers are modeled using the BEAM-4 element, a conventional 3D beam element. The steel-box girder is discretized into many segments based on the suspended points of the stayed cables, and the MASS-21 element is employed to consider the mass of each segment. The stayed cables are represented by the LINK-10 element, a 3D tension-only truss element. Piers are assumed to be fixed to the foundation without considering soil-structure interaction. The vehicle loads (5-wheel heavy-duty vehicles) are considered as a moving concentrated load acting on the bridge deck, with the velocity set as 50 km/h. Finally, this finite-element model consists of 2929 nodes and 3707 elements.

The basic input vector is specified as $\boldsymbol{Z} = \{E_1, D_2, E_3, D_4, E_5, D_6, F_7\}$, where E_1 and D_2 are the Young's modulus and density of the steel-box girder, respectively; E_3 and D_4 are the Young's modulus of the steel-wire strand, respectively; E_5 and D_6 are the Young's modulus and density of concrete in the tower, respectively; F_7 represents the moving vehicle load. Then, statistical information for those random variables is provided in Table 7.

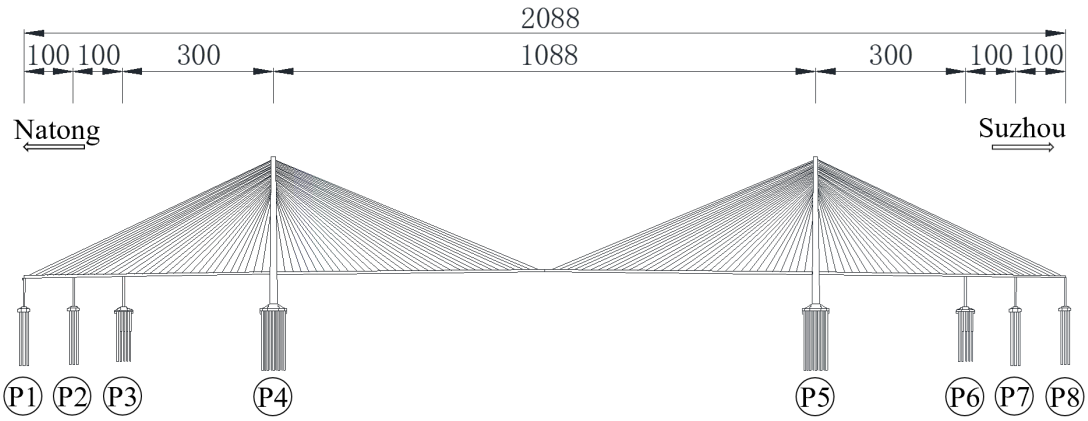
Table 7: Random variables in the cable-stayed bridge

Variables	Units	Distribution	Mean	COV
E_1	Pa	Lognormal	2.0594×10^{11}	0.1
D_2	kg/m^3	Lognormal	9.0810×10^{3}	0.1
E_3	Pa	Lognormal	1.9123×10^{11}	0.1
D_4	kg/m^3	Lognormal	8.606×10^{3}	0.1
E_5	Pa	Lognormal	3.4323×10^{10}	0.15
D_6	kg/m^3	Lognormal	3.7020×10^{3}	0.15
F_7	N	Weibull	5.5×10^{5}	0.1

Of interest is the maximum mid-span deflection $V(\boldsymbol{Z})$ of the main girder, and the threshold is set as 90

(a) front-elevation photo

(b) side-elevation photo



(c) span arrangement

Figure 15: Pictures of the Sutong bridge

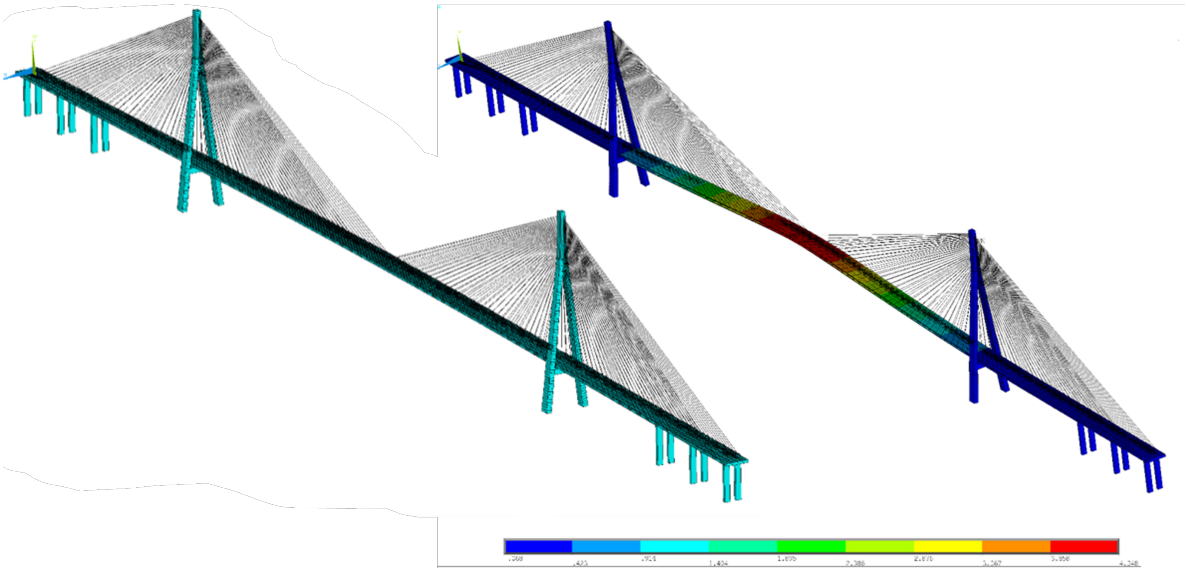


Figure 16: Finite-element model and typical deformation response of the Sutong bridge

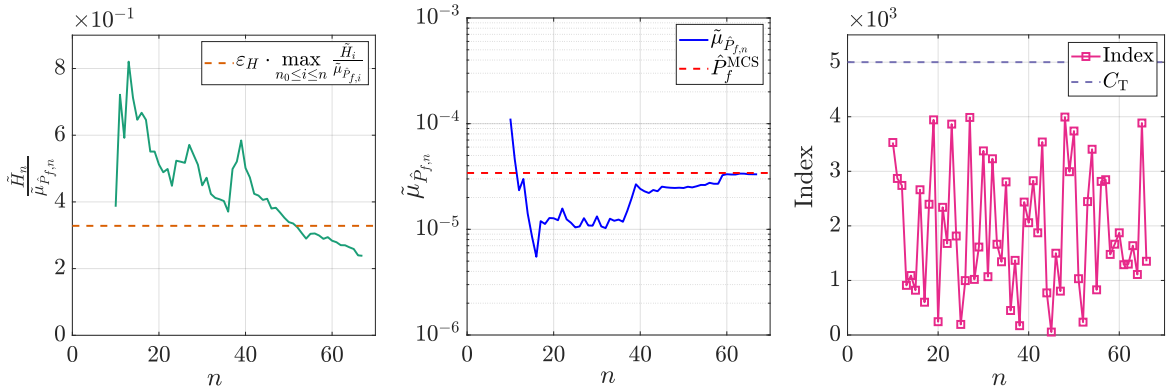
mm. Then, the corresponding performance function $\mathcal{G}(\boldsymbol{Z})$ is defined as

$$\mathcal{G}(\boldsymbol{Z}) = 90 - V(\boldsymbol{Z}) \tag{57}$$

The failure probability $\widehat{P}_f^{\mathrm{MCS}} = 3.414 \times 10^{-5}$ provide by MCS serves as the reference result.

Fig. 17 illustrates a single run of the proposed reliability method for the cable-stayed bridge. As n increases, the H_n shrinks (Fig. 17(a)), and $\mu_{\widehat{P}_{f,n}}$ gradually aligns with the reference value (Fig. 17(b)).

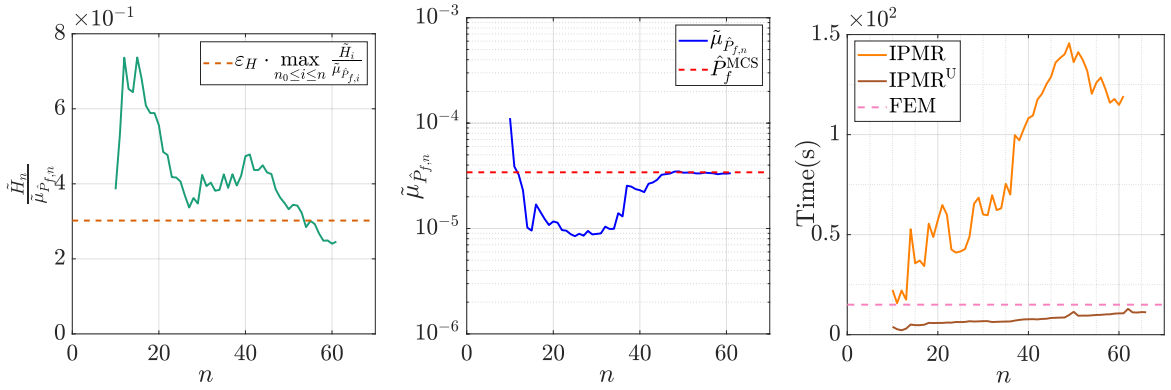
More importantly, Fig. 17 (c) shows the 'index' of the best next point selected at each iteration, and the index 'k' corresponds to the candidate point with the k-th greatest PM value among $\mathcal{X}_{\mathrm{CT}}$. Unlike PM, which selects the point with the greatest PM value (i.e., index = 1) per iteration, the index of the best next point selected by IPMR$^{\mathrm{U}}$ generally ranges from 1 to 4000. This suggests that the point with the greatest PM value does not necessarily achieve the biggest reduction of IPM. Moreover, these indexes are relatively far away from $C_{\mathrm{T}} = 5000$, highlighting the rationale behind the setting of C_{T}.



(a) $\frac{\tilde{H}_n}{\tilde{\mu}_{\widehat{P}_{f,n}}}$ (b) $\tilde{\mu}_{\widehat{P}_{f,n}}$ (c) index of new samples

Figure 17: Illustration of the proposed method in the real-world stayed-cable bridge example

Fig. 18 presents a single run of the proposed reliability method (exact), denoted as the BALR method equipped with IPMR$_n(\cdot)$ instead of IPMR$_n^{\mathrm{U}}(\cdot)$, in this cable-stayed bridge example. Compared to Fig. 17, IPMR$_n(\cdot)$ comes with a slightly smaller number of finite element analysis. Fig. 18(c) presents the running time of IPMR$_n(\cdot)$ per iteration, with the computational time of a single run of finite element analysis of this bridge (14s) included for reference. As highlighted in Section 3.3, IPMR$_n$ involves computing its integrand $\left[I_n(\boldsymbol{x}^{(i)};\boldsymbol{x}^{(j)})\right]_{1\le i\le Q_{\mathrm{T}},1\le j\le C_{\mathrm{T}}}$ at $\mathcal{X}_{\mathrm{QT}}\times\mathcal{X}_{\mathrm{CT}}$ in an element-wise manner (Fig. 4(a)). Therefore, the running time of IPMR$_n$ exceeds 100s in the latter stages. Conversely, the running time of IPMR$_n^{\mathrm{U}}$ consistently remains below 10 s per iteration, underscoring the substantial efficiency advantage of IPMR$^{\mathrm{U}}$.



(a) $\frac{\tilde{H}_n}{\tilde{\mu}_{\widehat{P}_{f,n}}}$ (b) $\tilde{\mu}_{\widehat{P}_{f,n}}$ (c) running time of IPMR vs. IPMR$^{\mathrm{U}}$

Figure 18: Illustration of the proposed method (exact) in the stayed-cable bridge example

Table 8 summarizes the results of various reliability methods in the cable-stayed bridge. Compared to PM, IPMR$^{\mathrm{U}}$ gains comparable accuracy of $\widehat{P}_f$, while requiring only approximately 72% of N_{call}. Although IPMR needs the fewest finite element analysis, its total computational time far exceeds that of those BALR methods. In contrast, IPMR$^{\mathrm{U}}$ requires a comparable number of finite-element analysis to IPMR but with significantly less computational time, just 17.3%. Overall, the proposed BRLR method equipped with IPMR$^{\mathrm{U}}$ exhibits significant advantages in terms of both N_{call} and T_{c}.

Table 8: Reliability results in the real-world cable-stayed bridge example

Method	N_{call}		$\widehat{P}_f$		$\delta_{\widehat{P}_f}(\%)$	T_{c}(s)
	Mean	COV(%)	Mean ($\times 10^{-5}$)	COV(%)		
MCS	5×10^5	-	3.414	-	-	7.112×10^6
BALR (PM)	86.1	13.031	3.398	5.819	0.462	1.181×10^3
BALR (RLCB)	131	9.575	3.924	8.923	22.608	1.793×10^3
BALR (REIF)	144.7	10.537	4.357	15.714	27.631	2.228×10^3
AK-SS	89.9	14.288	3.394	6.048	0.593	2.213×10^3
Proposed	62	10.860	3.389	3.439	0.732	8.923×10^2
Proposed (Exact)	60.7	4.531	3.405	4.132	0.283	5.133×10^3

6. Concluding remarks

A Bayesian reinforcement learning reliability (BRLR) method is proposed, incorporating both Bayesian inference and reinforcement learning. On one hand, Bayesian inference for the failure probability estimation is conducted. On the other hand, unlike the computationally-intensive posterior variance of failure probability, a computationally-cheap measure of epistemic uncertainty about failure probability, as referred to as IPM, is proved to be the upper bound for the absolute relative error of estimated failure probability in expectation and is used as the reward function in the MDP. Then, a one-step Bayes optimal learning function termed IPMR, along with a compatible convergence criterion, is defined. Three effective workarounds are devised to facilitate the IPMR-based sequential experimental design. The efficacy of the proposed BRLR method is demonstrated on four examples of varying complexity. Some concluding remarks are given as follow.

(1) Thanks to the substitution of IPMR by its upper bound IPMR^{U} and the pruning of both the quadrature set and candidate pool, IPMR^{U}-based sequential experimental design avoids both the element-wise computation of bivariate Gaussian CDF and the computer memory constraint, resulting in much less computational time than IPMR itself.

(2) The common learning function PM is unable to explicitly quantify the impact of adding a new point on the reduction of IPM. In contrast, IPMR^{U} allows selecting the optimal new point with the biggest expected reduction of IPM, leading to superior computational efficiency, particularly in dynamic reliability problems.

(3) The advantages of IPMR and IPMR^{U} are context-dependent. In cases where the performance function involves extremely expensive computational model evaluations, IPMR is preferable due to its slightly smaller number of required performance model evaluations. Otherwise, IPMR^{U} is superior to IPMR as it involves much less additional time.

It is admitted that only the single-point sequential experimental design is developed here. With the prevalence of parallel computing in reliability analysis, the batch-point version of IPMR will be investigated in the near future. Besides, exploring the extension of IPMR to the $k(\geq 2)$-step Bayes optimal criterion will be considered.

CRediT authorship contribution statement

Tong Zhou: Conceptualization, Formal analysis, Methodology, Software, Validation, Visualization, Writing - original draft. **Tong Guo**: Writing, review & editing. **Chao Dang**: Writing, review & editing, Conceptualization, Methodology. **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 No. 52125802) is highly appreciated.

Appendix A. Basics of Kriging

Kriging interprets the performance function $\mathcal{G}(\boldsymbol{x})$ as one realization of a Gaussian process, defined by [52]

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

where $\boldsymbol{\beta}^\top \boldsymbol{f}(\boldsymbol{x})$ is the trend function, and universal Kriging assumes that $\boldsymbol{\beta}^\top \boldsymbol{f}(\boldsymbol{x}) = \sum_{i=1}^{P} \beta_i f_i(\boldsymbol{x})$, with $\{f_i(\boldsymbol{x}), i = 1, \ldots, P\}$ a set of P basis functions and $\boldsymbol{\beta} = \{\beta_i, i = 1, \ldots, P\}^\top$ a set of unknown coefficients. A special case of universal Kriging, i.e., linear trend function, is considered: $\boldsymbol{\beta}^\top \boldsymbol{f}(\boldsymbol{x}) = \beta_0 + \sum_{k=1}^{d} \beta_k x_k$. σ^2 is the variance of Gaussian process; $Z(\boldsymbol{x})$ is a stationary Gaussian process with zero mean, unit variance, and a known correlation function. Matern-3/2 correlation function is given as [52]

$$R_{\mathrm{M}}(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}) = \prod_{k=1}^{d} \left(1 + \frac{|x_k - x_k'|}{\theta_k}\right) \exp\left(-\sqrt{3}\frac{|x_k - x_k'|}{\theta_k}\right) \tag{A.2}$$

where $\boldsymbol{\theta} = \{\theta_k > 0\}_{k=1}^{d}$ is a set of unknown parameters.

Assume an ED $\mathcal{D}_n = \{\mathcal{X}_n, \mathcal{Y}_n\}$ is provided, both $\boldsymbol{\beta}$ and σ^2 can be 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 $\boldsymbol{F} \coloneqq (f_j(\boldsymbol{x}^{(i)}))_{1 \le i \le n, 1 \le j \le P}$; $\boldsymbol{R} \coloneqq \left(R_{\mathrm{M}}(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}; \boldsymbol{\theta})\right)_{1 \le i,j \le n}$. Both $\widehat{\boldsymbol{\beta}}$ and $\widehat{\sigma}^2$ depend on $\boldsymbol{\theta}$, and $\boldsymbol{\theta}$ can be estimated as [52]

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

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

Finally, the Kriging predictor conditioned 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 $c_n(\boldsymbol{x}, \boldsymbol{x}')$ defined as [44]

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

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

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

where the subscript n indicates that these quantities condition on $\mathcal{D}_n$; $\boldsymbol{r}(\boldsymbol{x}) = \left[R_{\mathrm{M}}\left(\boldsymbol{x}, \boldsymbol{x}^{(1)}\right), \ldots, R_{\mathrm{M}}\left(\boldsymbol{x}, \boldsymbol{x}^{(n)}\right)\right]^\top$; $\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{F}^\top \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x})$. Note that $\mu_n(\boldsymbol{x})$ is usually taken as the Kriging prediction, and $\sigma_n^2(\boldsymbol{x}) = c_n(\boldsymbol{x}, \boldsymbol{x})$.

Appendix B. Proof of Proposition 1

Proof. First, according to Eq. (8) and (9), there exists

$$\begin{aligned}
\mathbb{E}_n\left[\left|\widehat{P}_{f,n} - \mu_{\widehat{P}_{f,n}}\right|\right] &= \mathbb{E}_n\left[\left|\int_{\mathbb{X}} \widehat{\mathbb{1}}_n(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \int_{\mathbb{X}} \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}\right|\right] \\
&= \mathbb{E}_n\left[\left|\int_{\mathbb{X}} \left[\widehat{\mathbb{1}}_n(\boldsymbol{x}) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right] f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}\right|\right] \\
&\le \int_{\mathbb{X}} \mathbb{E}_n\left[\left|\widehat{\mathbb{1}}_n(\boldsymbol{x}) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right|\right] f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \\
&= \int_{\mathbb{X}} \mathbb{E}_n\left[\left|\widehat{\mathbb{1}}_n(\boldsymbol{x}) - \eta_n(\boldsymbol{x}) + \eta_n(\boldsymbol{x}) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right|\right] f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \\
&\le \int_{\mathbb{X}} \Bigg\{ \underbrace{\mathbb{E}_n\left[\left|\widehat{\mathbb{1}}_n(\boldsymbol{x}) - \eta_n(\boldsymbol{x})\right|\right]}_{①} + \underbrace{\mathbb{E}_n\left[\left|\eta_n(\boldsymbol{x}) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right|\right]}_{②} \Bigg\} f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}
\end{aligned} \tag{B.1}$$

where $\eta_n(\boldsymbol{x}) = \begin{cases} 1, & \mu_n(\boldsymbol{x}) \le 0 \\ 0, & \text{otherwise} \end{cases}$; $|\cdot|$ denotes the absolute-value operator; $\mathbb{E}_n[\cdot]$ denotes the expectation with respect to Kriging predictor $\widehat{\mathcal{G}}_n(\boldsymbol{x})$.

Then, the bracketed term in the expression ① is further given as

$$\left|\widehat{\mathbb{1}}_n(\boldsymbol{x}) - \eta_n(\boldsymbol{x})\right| = \begin{cases} 1, & \widehat{\mathcal{G}}_n(\boldsymbol{x}) \ge 0 \cap \mu_n(\boldsymbol{x}) \le 0 \quad \text{or} \quad \widehat{\mathcal{G}}_n(\boldsymbol{x}) \le 0 \cap \mu_n(\boldsymbol{x}) \ge 0 \\ 0, & \text{otherwise} \end{cases} \tag{B.2}$$

Hence, the expression ① is given as

$$\begin{aligned} ① &= \mathbb{E}_n\left[\left|\widehat{\mathbb{1}}_n(\boldsymbol{x}) - \eta_n(\boldsymbol{x})\right|\right] = \begin{cases} \mathbb{P}\left(\widehat{\mathcal{G}}_n(\boldsymbol{x}) \ge 0\right), & \mu_n(\boldsymbol{x}) \le 0 \\ \mathbb{P}\left(\widehat{\mathcal{G}}_n(\boldsymbol{x}) \le 0\right), & \text{otherwise} \end{cases} = \begin{cases} \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right), & \mu_n(\boldsymbol{x}) \le 0 \\ \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right), & \text{otherwise} \end{cases} \\ &= \Phi\left(-\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\right) = P_n(\boldsymbol{x}) \end{aligned} \tag{B.3}$$

Obviously, $P_n(\boldsymbol{x})$ quantifies the probability of misclassifying the failure/safe state of $\boldsymbol{x}$ according to the sign of $\mu_n(\boldsymbol{x})$. Hence, it is called PM in [44].

Similarly, the bracketed term in the expression ② is further deduced as

$$\begin{aligned} \left|\eta_n(\boldsymbol{x}) - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right| &= \begin{cases} \left|1 - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right|, & \mu_n(\boldsymbol{x}) \le 0 \\ \left|0 - \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right|, & \text{otherwise} \end{cases} = \begin{cases} \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right), & \mu_n(\boldsymbol{x}) \le 0 \\ \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right), & \text{otherwise} \end{cases} \\ &= \Phi\left(-\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\right) = P_n(\boldsymbol{x}) \end{aligned} \tag{B.4}$$

Then, the expression ② is still equal to $P_n(\boldsymbol{x})$.

Finally, substituting Eqs. (B.3) and (B.4) into Eq. (B.1), Eq. (21) can be proved. □

Appendix C. Kriging update formulas

When a new point and its response $(\boldsymbol{x}_+, y_+)$ are added to the current ED $\mathcal{D}_n$, Kriging update formulas provide the look-ahead posteriors of Kriging $\widehat{\mathcal{G}}_{n+1}(\boldsymbol{x})$ as follows [53]

$$\mu_{n+1}(\boldsymbol{x}) = \mu_n(\boldsymbol{x}) + \frac{c_n(\boldsymbol{x}, \boldsymbol{x}_+)}{\sigma_n^2(\boldsymbol{x}_+)}\left(y_+ - \mu_n(\boldsymbol{x}_+)\right) \tag{C.1}$$

$$\sigma_{n+1}^2(\boldsymbol{x}) = \sigma_n^2(\boldsymbol{x}) - \frac{c_n^2(\boldsymbol{x}, \boldsymbol{x}_+)}{\sigma_n^2(\boldsymbol{x}_+)} \tag{C.2}$$

$$c_{n+1}(\boldsymbol{x}, \boldsymbol{x}') = c_n(\boldsymbol{x}, \boldsymbol{x}') - \frac{c_n(\boldsymbol{x}, \boldsymbol{x}_+)c_n(\boldsymbol{x}', \boldsymbol{x}_+)}{\sigma_n^2(\boldsymbol{x}_+)} \tag{C.3}$$

which are directly obtained based on the current posteriors of $\widehat{\mathcal{G}}_n(\boldsymbol{x})$ in Eqs. (A.6), (A.7) and (A.8).

Therefore, Kriging update formulas are computationally cheap and differ from the re-training of the parameters $\{\boldsymbol{\beta}, \sigma^2, \boldsymbol{\theta}\}$ of Kriging $\widehat{\mathcal{G}}_{n+1}(\boldsymbol{x})$ according to the augmented ED $\mathcal{D}_{n+1} = \mathcal{D}_n \bigcup (\boldsymbol{x}_+, y_+)$ [20].

Moreover, it can be observed that $\mu_{n+1}(\boldsymbol{x})$ depends on the future outcome y_+ at $\boldsymbol{x}_+$, while both $\sigma_{n+1}^2(\boldsymbol{x})$ and $c_{n+1}(\boldsymbol{x}, \boldsymbol{x}')$ are independent of y_+. Since y_+ is unknown without evaluating the computational model on $\boldsymbol{x}_+$, these quantities depending on it are actually random quantities.

Appendix D. Derivation of Eq. (25)

Analogous to the PM $P_n(\boldsymbol{x})$, the look-ahead PM $P_{n+1}(\boldsymbol{x}; \boldsymbol{x}_+, y_+)$ is expressed as

$$P_{n+1}(\boldsymbol{x}; \boldsymbol{x}_+, y_+) = \Phi\left(-\frac{|\mu_{n+1}(\boldsymbol{x})|}{\sigma_{n+1}(\boldsymbol{x})}\right) = \begin{cases} \underbrace{\Phi\left(\frac{\mu_{n+1}(\boldsymbol{x})}{\sigma_{n+1}(\boldsymbol{x})}\right)}_{Ⓐ}, & \underbrace{\mu_{n+1}(\boldsymbol{x}) \le 0}_{Ⓑ} \\ 1 - \Phi\left(\frac{\mu_{n+1}(\boldsymbol{x})}{\sigma_{n+1}(\boldsymbol{x})}\right), & \text{otherwise} \end{cases} \tag{D.1}$$

where the associated components are detailed below.

First, substitute Eqs. (C.1) and (C.2) into the expression Ⓐ in Eq. (D.1), yielding

$$\begin{aligned}
\text{Ⓐ} &= \Phi\left(\frac{\mu_{n+1}(\boldsymbol{x})}{\sigma_{n+1}(\boldsymbol{x})}\right) \\
&= \Phi\left(\frac{\mu_n(\boldsymbol{x}) + \frac{c_n(\boldsymbol{x},\boldsymbol{x}_+)}{\sigma_n^2(\boldsymbol{x}_+)}\left(y_+ - \mu_n(\boldsymbol{x}_+)\right)}{\sigma_{n+1}(\boldsymbol{x})}\right) \\
&= \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_{n+1}(\boldsymbol{x})} + \frac{c_n(\boldsymbol{x},\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)\sigma_{n+1}(\boldsymbol{x})} \cdot \frac{y_+ - \mu_n(\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)}\right) \\
&= \Phi\left(a(\boldsymbol{x}) + b(\boldsymbol{x})z_+\right)
\end{aligned} \tag{D.2}$$

where $a(\boldsymbol{x}) = \frac{\mu_n(\boldsymbol{x})}{\sigma_{n+1}(\boldsymbol{x})}$ and $b(\boldsymbol{x}) = \frac{c_n(\boldsymbol{x},\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)\sigma_{n+1}(\boldsymbol{x})}$ are two variables unrelated to the response y_+; $z_+ = \frac{y_+ - \mu_n(\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)}$ is the only variable depending on y_+.

Then, plug Eq. (C.1) into the condition Ⓑ in Eq. (D.1), yielding

$$\begin{aligned}
\text{Ⓑ} &\Rightarrow \mu_{n+1}(\boldsymbol{x}) \le 0 \\
&\Rightarrow \mu_n(\boldsymbol{x}) + \frac{c_n(\boldsymbol{x},\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)} \cdot \frac{y_+ - \mu_n(\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)} \le 0 \\
&\Rightarrow \mu_n(\boldsymbol{x}) + \frac{c_n(\boldsymbol{x},\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)} z_+ \le 0 \\
&\Rightarrow \frac{c_n(\boldsymbol{x},\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)} z_+ \le -\mu_n(\boldsymbol{x}) \\
&\Rightarrow \begin{cases} z_+ \le -\frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{c_n(\boldsymbol{x},\boldsymbol{x}_+)}, & c_n(\boldsymbol{x},\boldsymbol{x}_+) > 0 \\ z_+ \ge -\frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{c_n(\boldsymbol{x},\boldsymbol{x}_+)}, & \text{otherwise} \end{cases} \\
&\Rightarrow \begin{cases} z_+ \le z_{\lim}, & c_n(\boldsymbol{x},\boldsymbol{x}_+) > 0 \\ z_+ \ge z_{\lim}, & \text{otherwise} \end{cases}
\end{aligned} \tag{D.3}$$

where the variable $z_{\lim} = -\frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{c_n(\boldsymbol{x},\boldsymbol{x}_+)}$ is unrelated to y_+.

Substitute Eqs. (D.2) and (D.3) into Eq. (D.1), resulting in

$$\begin{aligned}
P_{n+1}(\boldsymbol{x}; \boldsymbol{x}_+, y_+) &= \begin{cases} \Phi\left(a(\boldsymbol{x}) + b(\boldsymbol{x})z_+\right), & \begin{cases} z_+ \le z_{\lim}, & c_n(\boldsymbol{x},\boldsymbol{x}_+) > 0 \\ z_+ \ge z_{\lim}, & \text{otherwise} \end{cases} \\ 1 - \Phi\left(a(\boldsymbol{x}) + b(\boldsymbol{x})z_+\right), & \begin{cases} z_+ > z_{\lim}, & c_n(\boldsymbol{x},\boldsymbol{x}_+) > 0 \\ z_+ < z_{\lim}, & \text{otherwise} \end{cases} \end{cases} \\
&= \begin{cases} \begin{cases} \Phi\left(a(\boldsymbol{x}) + b(\boldsymbol{x})z_+\right), & z_+ \le z_{\lim} \\ 1 - \Phi\left(a(\boldsymbol{x}) + b(\boldsymbol{x})z_+\right), & z_+ > z_{\lim} \end{cases}, & c_n(\boldsymbol{x},\boldsymbol{x}_+) > 0 \\ \begin{cases} 1 - \Phi\left(a(\boldsymbol{x}) + b(\boldsymbol{x})z_+\right), & z_+ < z_{\lim} \\ \Phi\left(a(\boldsymbol{x}) + b(\boldsymbol{x})z_+\right), & z_+ \ge z_{\lim} \end{cases}, & \text{otherwise} \end{cases}
\end{aligned} \tag{D.4}$$

which comprises a total of four distinct cases.

Appendix E. Proof of Proposition 2

Proof. In the $I_n(\boldsymbol{x};\boldsymbol{x}_+)$ in Eq. (28), the second term $\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right]$ can be expanded as

$$
\begin{aligned}
&\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right] \\
=&\int_{-\infty}^{+\infty} P_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+,y_+\right) f_{Y_+}(y_+)\mathrm{d}y_+ \\
\overset{1}{=}&\int_{-\infty}^{+\infty} P_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+,y_+\right) \frac{1}{\sigma_n(\boldsymbol{x}_+)}\phi\left(\frac{y_+-\mu_n(\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)}\right)\mathrm{d}y_+ \\
\overset{2}{=}&\int_{-\infty}^{+\infty} P_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+,y_+\right) \phi\left(z_+\right)\mathrm{d}z_+ \\
=&\begin{cases}
\underbrace{\int_{-\infty}^{z_{\lim}} \Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\phi\left(z_+\right)\mathrm{d}z_+ + \int_{z_{\lim}}^{+\infty}\left[1-\Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\right]\phi\left(z_+\right)\mathrm{d}z_+}_{①}, & c_n(\boldsymbol{x},\boldsymbol{x}_+)>0 \\
\underbrace{\int_{-\infty}^{z_{\lim}} \left[1-\Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\right]\phi\left(z_+\right)\mathrm{d}z_+ + \int_{z_{\lim}}^{+\infty}\Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\phi\left(z_+\right)\mathrm{d}z_+}_{②}, & \text{otherwise}
\end{cases}
\end{aligned}
\tag{E.1}
$$

where the equality '$\overset{1}{=}$' uses the expression of the PDF $f_{Y_+}(y_+)$ of Y_+, and $\phi(\cdot)$ represents the PDF of a standard Gaussian variable; the equality '$\overset{2}{=}$' utilizes the change of variables: $Z_+ = \frac{Y_+-\mu_n(\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)}$. The two mutually exclusive cases in Eq. (E.1) are discussed below.

First, when $c_n(\boldsymbol{x},\boldsymbol{x}_+)>0$, the expression ① in Eq. (E.1) is derived as

$$
\begin{aligned}
① =& \int_{-\infty}^{z_{\lim}} \Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\phi\left(z_+\right)\mathrm{d}z_+ + \int_{z_{\lim}}^{+\infty}\phi\left(z_+\right)\mathrm{d}z_+ - \int_{z_{\lim}}^{+\infty}\Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\phi\left(z_+\right)\mathrm{d}z_+ \\
\overset{1}{=}& \int_{-\infty}^{z_{\lim}} \Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\phi\left(z_+\right)\mathrm{d}z_+ + \Phi\left(-z_{\lim}\right) - \left[\Phi\left(\frac{a(\boldsymbol{x})}{\sqrt{1+b^2(\boldsymbol{x})}}\right) - \int_{-\infty}^{z_{\lim}}\Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\phi\left(z_+\right)\mathrm{d}z_+\right] \\
=& 2\int_{-\infty}^{z_{\lim}} \Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\phi\left(z_+\right)\mathrm{d}z_+ + \Phi\left(-z_{\lim}\right) - \Phi\left(\frac{a(\boldsymbol{x})}{\sqrt{1+b^2(\boldsymbol{x})}}\right) \\
\overset{2}{=}& 2\Phi_2\left(\frac{a(\boldsymbol{x})}{\sqrt{1+b^2(\boldsymbol{x})}}, z_{\lim}; \frac{-b(\boldsymbol{x})}{\sqrt{1+b^2(\boldsymbol{x})}}\right) + \Phi\left(-z_{\lim}\right) - \Phi\left(\frac{a(\boldsymbol{x})}{\sqrt{1+b^2(\boldsymbol{x})}}\right) \\
=& 2\Phi_2\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}, \frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{-c_n(\boldsymbol{x},\boldsymbol{x}_+)}; \frac{-c_n(\boldsymbol{x},\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)\sigma_n(\boldsymbol{x})}\right) + \Phi\left(\frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{c_n(\boldsymbol{x},\boldsymbol{x}_+)}\right) - \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)
\end{aligned}
\tag{E.2}
$$

where $\Phi_2\left(h_1,h_2;r\right)$ is the CDF of a standard bivariate Gaussian vector with a correlation coefficient r, i.e., $\Phi_2\left(h_1,h_2;r\right) = F_2\left(\begin{bmatrix}h_1\\h_2\end{bmatrix};\begin{bmatrix}0\\0\end{bmatrix},\begin{bmatrix}1 & r\\ r & 1\end{bmatrix}\right)$. The equalities '$\overset{1}{=}$' and '$\overset{2}{=}$' adopt the formulas with indexes 10010.8 and 10010.1 in [54], respectively.

Regarding $\Phi_2\left(h_1,h_2;r\right)$, the following three relationships always hold [55]

$$
\begin{cases}
\Phi_2\left(h_1,-h_2;-r\right) &= \Phi\left(h_1\right) - \Phi_2\left(h_1,h_2;r\right) \\
\Phi_2\left(-h_1,h_2;-r\right) &= \Phi\left(h_2\right) - \Phi_2\left(h_1,h_2;r\right) \\
\Phi_2\left(-h_1,-h_2;r\right) &= \Phi_2\left(h_1,h_2;r\right) - \Phi\left(h_1\right) - \Phi\left(h_2\right) + 1
\end{cases}
\tag{E.3}
$$

Then, according to the first expression in Eq. (E.3), Eq. (E.2) is equivalent to

$$
\begin{aligned}
① =& 2\left[\Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) - \Phi_2\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}, \frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{c_n(\boldsymbol{x},\boldsymbol{x}_+)}; \frac{c_n(\boldsymbol{x},\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)\sigma_n(\boldsymbol{x})}\right)\right] + \Phi\left(\frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{c_n(\boldsymbol{x},\boldsymbol{x}_+)}\right) - \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) \\
=& \Phi\left(\frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{c_n(\boldsymbol{x},\boldsymbol{x}_+)}\right) + \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) - 2\Phi_2\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}, \frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{c_n(\boldsymbol{x},\boldsymbol{x}_+)}; \frac{c_n(\boldsymbol{x},\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)\sigma_n(\boldsymbol{x})}\right)
\end{aligned}
\tag{E.4}
$$

Second, when $c_n(\boldsymbol{x},\boldsymbol{x}_+)<0$, the expression ② in Eq. (E.1) is derived as

$$\begin{aligned}
② &= \int_{-\infty}^{z_{\lim}} \phi\left(z_+\right) \mathrm{d}z_+ - \int_{-\infty}^{z_{\lim}} \Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\phi\left(z_+\right)\mathrm{d}z_+ + \left[\Phi\left(\frac{a(\boldsymbol{x})}{\sqrt{1+b^2(\boldsymbol{x})}}\right) - \int_{-\infty}^{z_{\lim}} \Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\phi\left(z_+\right)\mathrm{d}z_+\right] \\
&= \Phi\left(z_{\lim}\right) + \Phi\left(\frac{a(\boldsymbol{x})}{\sqrt{1+b^2(\boldsymbol{x})}}\right) - 2\int_{-\infty}^{z_{\lim}} \Phi\left(a(\boldsymbol{x})+b(\boldsymbol{x})z_+\right)\phi\left(z_+\right)\mathrm{d}z_+ \\
&= \Phi\left(z_{\lim}\right) + \Phi\left(\frac{a(\boldsymbol{x})}{\sqrt{1+b^2(\boldsymbol{x})}}\right) - 2\Phi_2\left(\frac{a(\boldsymbol{x})}{\sqrt{1+b^2(\boldsymbol{x})}}, z_{\lim}; \frac{-b(\boldsymbol{x})}{\sqrt{1+b^2(\boldsymbol{x})}}\right) \\
&= \Phi\left(\frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{-c_n(\boldsymbol{x},\boldsymbol{x}_+)}\right) + \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) - 2\Phi_2\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}, \frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{-c_n(\boldsymbol{x},\boldsymbol{x}_+)}; \frac{-c_n(\boldsymbol{x},\boldsymbol{x}_+)}{\sigma_n(\boldsymbol{x}_+)\sigma_n(\boldsymbol{x})}\right)
\end{aligned} \tag{E.5}$$

Therefore, the unified expression of $\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right]$ can be obtained by assembling Eqs. (E.4) and (E.5), that is,

$$\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right] = \Phi\left(\frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{|c_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right) + \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) - 2\Phi_2\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}, \frac{\sigma_n(\boldsymbol{x}_+)\mu_n(\boldsymbol{x})}{|c_n(\boldsymbol{x},\boldsymbol{x}_+)|}; \frac{|c_n(\boldsymbol{x},\boldsymbol{x}_+)|}{\sigma_n(\boldsymbol{x}_+)\sigma_n(\boldsymbol{x})}\right) \tag{E.6}$$

Then, two additional simplifications are performed on Eq. (E.6).

First, denote

$$\rho_n(\boldsymbol{x},\boldsymbol{x}_+) = \frac{c_n\left(\boldsymbol{x},\boldsymbol{x}_+\right)}{\sigma_n(\boldsymbol{x})\sigma_n(\boldsymbol{x}_+)} \in (-1,1) \tag{E.7}$$

as the posterior correlation coefficient of Kriging $\widehat{\mathcal{G}}_n(\cdot)$ between $\boldsymbol{x}$ and $\boldsymbol{x}_+$; then, substitute Eq. (E.7) into Eq. (E.6), giving rise to

$$\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right] = \Phi\left(\frac{\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right) + \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) - 2\Phi_2\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}, \frac{\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}; |\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|\right) \tag{E.8}$$

Second, if $\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}$ in Eq. (E.8) is replaced by its negative counterpart, i.e., $-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}$, there exists

$$\begin{aligned}
&\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right]\Big|_{-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}} \\
&= \Phi\left(\frac{-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right) + \Phi\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) - 2\Phi_2\left(-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}, \frac{-\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}; |\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|\right) \\
&\overset{*}{=} 1 - \Phi\left(\frac{\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right) + 1 - \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) \\
&\quad - 2\left[\Phi_2\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}, \frac{\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}; |\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|\right) - \Phi\left(\frac{\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right) - \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) + 1\right] \\
&= \Phi\left(\frac{\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right) + \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) - 2\Phi_2\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}, \frac{\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}; |\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|\right) \\
&= \mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right]\Big|_{\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}}
\end{aligned} \tag{E.9}$$

where the equality '$\overset{*}{=}$' utilizes the third expression in Eq. (E.3). Eq. (E.9) indicates that $\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}(\boldsymbol{x};\boldsymbol{x}_+)\right]$ is an even function with respect to $\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}$.

Therefore, $\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}(\boldsymbol{x};\boldsymbol{x}_+)\right]$ is finally expressed as

$$\mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}\left(\boldsymbol{x};\boldsymbol{x}_+\right)\right] = \Phi\left(\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right) + \Phi\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\right) - 2\Phi_2\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}, \frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}; |\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|\right) \tag{E.10}$$

which is a function of $|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|$ and $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}$.

Finally, according to Eq. (28), the $I_n(\boldsymbol{x}; \boldsymbol{x}_+)$ is expressed as

$$
\begin{aligned}
I_n(\boldsymbol{x}; \boldsymbol{x}_+) &= P_n(\boldsymbol{x}) - \mathbb{E}_{Y_+}\left[\mathcal{P}_{n+1}(\boldsymbol{x}; \boldsymbol{x}_+)\right] \\
&= 1 - \Phi\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\right) - \Phi\left(\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|}\right) - \Phi\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\right) + 2\Phi_2\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}, \frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|}; |\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|\right) \\
&= 1 - \Phi\left(\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|}\right) - 2\Phi\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\right) + 2\Phi_2\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}, \frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|}; |\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|\right) \\
&= \Phi\left(-\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|}\right) - 2\left[\Phi\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\right) - \Phi_2\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}, \frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|}; |\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|\right)\right] \\
&\overset{\oplus}{=} \Phi\left(-\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|}\right) - 2\Phi_2\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}, -\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|}; -|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|\right)
\end{aligned}
\tag{E.11}
$$

where the equality '$\overset{\oplus}{=}$' adopts the first expression in Eq. (E.3). In this way, the analytical expression of $I_n(\boldsymbol{x}, \boldsymbol{x}_+)$ in Eq. (30) can be proved. □

Appendix F. Proof of Proposition 3

Proof. First, the upper bound of $I_n(\boldsymbol{x}; \boldsymbol{x}_+)$, denoted as $I_n^{\mathrm{U}}(\boldsymbol{x}; \boldsymbol{x}_+)$, in Eq. (31) is naturally obtained based on the fact that the second term in the right-hand side of Eq. (30) is non-negative.

Second, the lower bound of $I_n(\boldsymbol{x}; \boldsymbol{x}_+)$, denoted as $I_n^{\mathrm{L}}(\boldsymbol{x}; \boldsymbol{x}_+)$, in Eq. (31) is derived as follows. For brevity, the two terms, $\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}$ and $|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|$, in $I_n(\boldsymbol{x}; \boldsymbol{x}_+)$ are simplified as two non-negative variables, $A \in [0, +\infty)$ and $B \in [0, 1)$, respectively. Then, $I_n(\boldsymbol{x}; \boldsymbol{x}_+)$ in Eq. (30) is reformulated as

$$
I_n(\boldsymbol{x}; \boldsymbol{x}_+) = \Phi\left(-\frac{A}{B}\right) - 2\Phi_2\left(A, -\frac{A}{B}; -B\right) \tag{F.1}
$$

With respect to $\Phi_2(h_1, h_2; r)$, its partial derivatives with respect to the three components are expressed as [56]

$$
\begin{cases}
\frac{\partial \Phi_2(h_1, h_2; r)}{\partial h_1} &= \phi(h_1)\,\Phi\left(\frac{h_2 - r h_1}{\sqrt{1-r^2}}\right) \\
\frac{\partial \Phi_2(h_1, h_2; r)}{\partial h_2} &= \phi(h_2)\,\Phi\left(\frac{h_1 - r h_2}{\sqrt{1-r^2}}\right) \\
\frac{\partial \Phi_2(h_1, h_2; r)}{\partial r} &= \frac{1}{2\pi\sqrt{1-r^2}} \exp\left(-\frac{h_1^2 - 2r h_1 h_2 + h_2^2}{2(1-r^2)}\right)
\end{cases}
\tag{F.2}
$$

Then, according to Eq. (F.2), $\frac{\partial I_n(\boldsymbol{x}; \boldsymbol{x}_+)}{\partial B}$ can be obtained from chain rule such that

$$
\begin{aligned}
\frac{\partial I_n(\boldsymbol{x}; \boldsymbol{x}_+)}{\partial B} &= \phi\left(-\frac{A}{B}\right)\frac{A}{B^2} - 2\left[\phi\left(-\frac{A}{B}\right)\Phi\left(\frac{A - (-B)\left(-\frac{A}{B}\right)}{\sqrt{1-B^2}}\right)\frac{A}{B^2}\right. \\
&\quad \left. + \frac{1}{2\pi\sqrt{1-B^2}} \exp\left(-\frac{A^2 - 2(-B)A\left(-\frac{A}{B}\right) + \frac{A^2}{B^2}}{2(1-B^2)}\right) \times (-1)\right] \\
&= \frac{A}{B^2}\phi\left(\frac{A}{B}\right) - 2\left[\frac{A}{2B^2}\phi\left(\frac{A}{B}\right) - \frac{1}{2\pi\sqrt{1-B^2}} \exp\left(-\frac{A^2(\frac{1-B^2}{B^2})}{2(1-B^2)}\right)\right] \\
&= \frac{A}{B^2}\phi\left(\frac{A}{B}\right) - \frac{A}{B^2}\phi\left(\frac{A}{B}\right) + \frac{2}{\sqrt{2\pi}\sqrt{1-B^2}}\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(\frac{A}{B})^2}{2}\right) \\
&= \frac{2}{\sqrt{2\pi}\sqrt{1-B^2}}\phi\left(\frac{A}{B}\right) \\
&\geq 0
\end{aligned}
\tag{F.3}
$$

which equals 0 only when $B = 0$. Therefore, $I_n(\boldsymbol{x}; \boldsymbol{x}_+)$ is a monotonically increasing function with respect to $|\rho_n(\boldsymbol{x}, \boldsymbol{x}_+)|$.

When $r=0$, $\Phi_2(h_1,h_2;r)=\Phi(h_1)\Phi(h_2)$. Therefore, if $|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|\to 0$, $I_n(\boldsymbol{x};\boldsymbol{x}_+)$ reduces to

$$I_n(\boldsymbol{x};\boldsymbol{x}_+)=\Phi\left(-\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right)-2\Phi\left(-\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right)\Phi\left(\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\right) \tag{F.4}$$

Further, there exists

$$\begin{aligned}|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|\to 0 &\Rightarrow -\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\to-\infty\\ &\Rightarrow \Phi\left(-\frac{\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}}{|\rho_n(\boldsymbol{x},\boldsymbol{x}_+)|}\right)\to 0\\ &\Rightarrow I_n(\boldsymbol{x};\boldsymbol{x}_+)\to 0\end{aligned} \tag{F.5}$$

Consequently, the lower bound of $I_n(\boldsymbol{x};\boldsymbol{x}_+)$, denoted as $I_n^{\mathrm{U}}(\boldsymbol{x};\boldsymbol{x}_+)$, is 0. □

Appendix G. Performances of VAIS in two examples

Fig. G.19 illustrates the performances of VAIS during the IPMR-based sequential experimental design in two examples: the planar truss example in Section 5.2 and the cable-stayed bridge example in Section 5.4.

In Fig. G.19(a), the quadrature size Q increases to 6×10^5 according to Eq. (50), as enclosed by red dashed lines. The corresponding quadrature set $\mathcal{X}_{\mathrm{Q}}$ provides stable estimates of $\mu_{\widehat{P}_{f,n}}$, H_n, and $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$. When the pruning coefficient $\lambda>1.0$, the $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$ value computed based on the pruned quadrature set $\mathcal{X}_{\mathrm{QT}}$ is highly consistent with that based on the full $\mathcal{X}_{\mathrm{Q}}$. Therefore, λ can be reassuringly set as 1.5. Moreover, when $\lambda=1.5$, the ratio $\frac{Q_{\mathrm{T}}}{Q}$ is approximately 6×10^{-3}. Hence, Q_{T} is roughly 3.6×10^3, and the total running time of $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+),\forall\boldsymbol{x}_+\in\mathcal{X}_{\mathrm{CT}}$, is just 2s, as shown in Fig. 4(b).

Similarly, Fig. G.19(b) shows that the quadrature size Q increases to 4×10^5 according to Eq. (50), and such quadrature size yields favorable results for the three integrals. Then, when $\lambda=1.5$, the corresponding value of $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+)$ aligns well with that based on the full $\mathcal{X}_{\mathrm{Q}}$. Additionally, the sample size Q_{T} of the pruned quadrature set $\mathcal{X}_{\mathrm{QT}}$ is only 1.4×10^3, and $\mathrm{IPMR}_n^{\mathrm{U}}(\boldsymbol{x}_+),\forall\boldsymbol{x}_+\in\mathcal{X}_{\mathrm{CT}}$, just consumes almost 1s, as illustrated in Fig. 4(b).

Appendix H. Five existing learning functions

An overview of five existing learning functions, namely PM [44], RLCB [31], REIF [32], EIER [42] and SUR [29], is provided for comparison.

(1) PM

The PM is expressed as [44]

$$\mathrm{PM}_n(\boldsymbol{x})=\Phi\left(-\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\right) \tag{H.1}$$

then, the best next point is selected as $\boldsymbol{x}^{(n+1)}=\arg\max_{\boldsymbol{x}\in\mathcal{X}_{\mathrm{C}}}\mathrm{PM}_n(\boldsymbol{x})$.

(2) RLCB

The RLCB is expressed as [31]

$$\mathrm{RLCB}_n(\boldsymbol{x})=|\mu_n(\boldsymbol{x})|-\phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\sigma_n(\boldsymbol{x}) \tag{H.2}$$

then, $\boldsymbol{x}^{(n+1)}=\arg\min_{\boldsymbol{x}\in\mathcal{X}_{\mathrm{c}}}\mathrm{RLCB}_n(\boldsymbol{x})$.

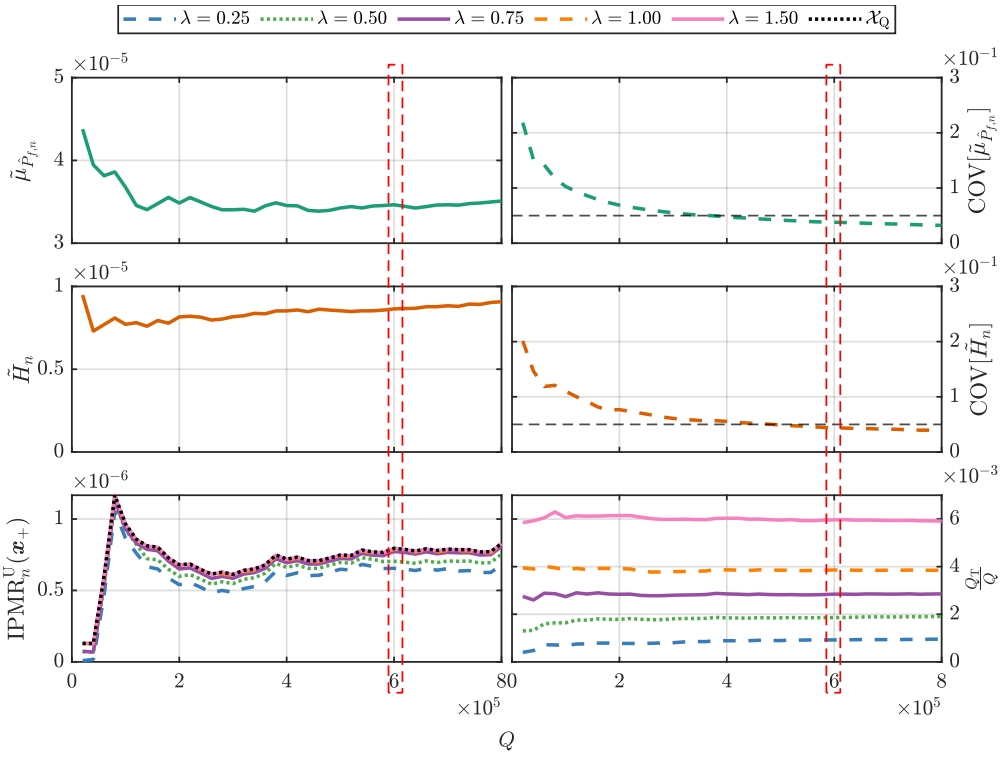
(3) REIF

The REIF is defined as [32]

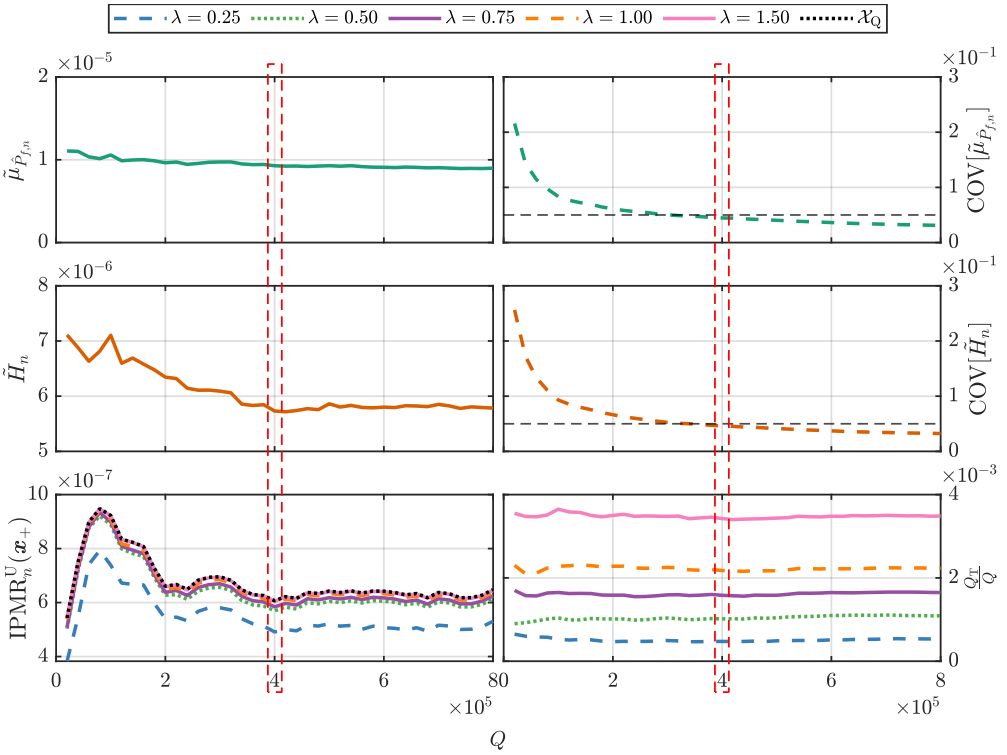
$$\mathrm{REIF}_n(\boldsymbol{x})=\mu_n(\boldsymbol{x})\left[1-2\Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right]+\sigma_n(\boldsymbol{x})\left[2-\sqrt{\frac{2}{\pi}}\exp\left(-\frac{1}{2}\frac{\mu_n^2(\boldsymbol{x})}{\sigma_n^2(\boldsymbol{x})}\right)\right] \tag{H.3}$$

then, $\boldsymbol{x}^{(n+1)}=\arg\max_{\boldsymbol{x}\in\mathcal{X}_{\mathrm{C}}}\mathrm{REIF}_n(\boldsymbol{x})$.

(4) EIER



(a) the 38-th iteration of IPMR in the planar truss example



(b) the 9-th iteration of IPMR in the cable-stayed bridge example

Figure G.19: Illustration of sequential VAIS on two examples

EIER is defined with the aim similar to the proposed IPMR, given as [42]:

$$\begin{aligned}\mathrm{EIER}_n(\boldsymbol{x}_+) &= \mathbb{E}_{\boldsymbol{X}}\left[\mathbb{E}_{Y_+}\left[\max\left(P_n(\boldsymbol{x}) - P_{n+1}(\boldsymbol{x};\boldsymbol{x}_+,Y_+),0\right)\right]\right] \\ &= \mathbb{E}_{\boldsymbol{X}}\left[\mathbb{E}_{Y_+}\left[\max\left(\Phi\left(-\frac{|\mu_n(\boldsymbol{x})|}{\sigma_n(\boldsymbol{x})}\right) - \Phi\left(-\frac{|\mu_{n+1}(\boldsymbol{x},\boldsymbol{x}_+,Y_+)|}{\sigma_{n+1}(\boldsymbol{x};\boldsymbol{x}_+,Y_+)}\right),0\right)\right]\right]\end{aligned} \tag{H.4}$$

then, $\boldsymbol{x}^{(n+1)} = \arg\max_{\boldsymbol{x}_+ \in \mathcal{X}_C} \mathrm{EIER}_n(\boldsymbol{x}_+)$.

Due to the lack of analytical solutions for both two expectations in Eq. (H.4), $\mathrm{EIER}_n(\boldsymbol{x}_+)$ is approximated using a double summation [42]:

$$\mathrm{EIER}_n(\boldsymbol{x}_+) \approx \frac{1}{N \times Q} \sum_{i=1}^{N} \sum_{j=1}^{Q} \max \left[\Phi\left(-\frac{\left|\mu_n(\boldsymbol{x}^{(j)})\right|}{\sigma_n(\boldsymbol{x}^{(j)})} \right) - \Phi\left(-\frac{\left|\mu_{n+1}\left(\boldsymbol{x}^{(j)}; \boldsymbol{x}_+, y_+^{(i)}\right)\right|}{\sigma_{n+1}\left(\boldsymbol{x}^{(j)}; \boldsymbol{x}_+, y_+^{(i)}\right)} \right), 0 \right] \tag{H.5}$$

where $\left\{\boldsymbol{x}^{(j)}\right\}_{j=1}^{Q}$ denotes a set of Q quadrature points drawn from $f_{\boldsymbol{X}}(\boldsymbol{x})$; $\left\{y_+^{(i)}\right\}_{i=1}^{N}$ denotes a set of N Kriging realizations evaluated at $\boldsymbol{x}_+$. For each $y_+^{(i)}$, the retraining of Kriging is performed via complex block matrix inversion to provide both $\mu_{n+1}(\cdot)$ and $\sigma_{n+1}^2(\cdot)$. When Q and N are significant, this double summation can lead to substantial computational burden.

(5) SUR

In SUR, the uncertainty measure H_n^{SUR} is defined in terms of the upper bound of $\sigma^2_{\widehat{P}_{f,n}}$ in Eq. (10), given by

$$H_n^{\mathrm{SUR}} = \int_{\mathbb{X}} \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\left(1 - \Phi\left(\frac{\mu_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right)\right) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \tag{H.6}$$

Then, the basic expression of SUR is defined as [28]

$$\mathrm{SUR}_n(\boldsymbol{x}_+) = \mathbb{E}_{Y_+}\left[\mathcal{H}_{n+1}^{\mathrm{SUR}}(\boldsymbol{x}_+)\right] \tag{H.7}$$

and the best next point is selected as $\boldsymbol{x}^{(n+1)} = \arg\min_{\boldsymbol{x}_+ \in \mathcal{X}_C} \mathrm{SUR}_n(\boldsymbol{x}_+)$.

Utilizing Kriging update formulas, the double integral in SUR can be reduced to a single one [29]:

$$\mathrm{SUR}_n(\boldsymbol{x}_+) = \int_{\mathbb{X}} F_2\left(\begin{bmatrix} a(\boldsymbol{x}) \\ -a(\boldsymbol{x}) \end{bmatrix}; \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} c(\boldsymbol{x}), & 1 - c(\boldsymbol{x}) \\ 1 - c(\boldsymbol{x}), & c(\boldsymbol{x}) \end{bmatrix}\right) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \tag{H.8}$$

where $a(\boldsymbol{x}) = \frac{\mu_n(\boldsymbol{x})}{\sigma_{n+1}(\boldsymbol{x})}$ and $c(\boldsymbol{x}) = \frac{\sigma_n^2(\boldsymbol{x})}{\sigma_{n+1}^2(\boldsymbol{x})}$. Further, it can be approximated based on VAIS as

$$\mathrm{SUR}_n(\boldsymbol{x}_+) \approx \frac{1}{Q} \sum_{i=1}^{Q} F_2\left(\begin{bmatrix} a(\boldsymbol{x}^{(i)}) \\ -a(\boldsymbol{x}^{(i)}) \end{bmatrix}; \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} c(\boldsymbol{x}^{(i)}), & 1 - c(\boldsymbol{x}^{(i)}) \\ 1 - c(\boldsymbol{x}^{(i)}), & c(\boldsymbol{x}^{(i)}) \end{bmatrix}\right) \frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})} \tag{H.9}$$

Similar to $\mathrm{IPMR}_n(\boldsymbol{x}_+)$ in Eq. (E.11), $\mathrm{SUR}_n(\boldsymbol{x}_+)$ involves computing the bi-variate Gaussian CDF $F_2(\cdot;\cdot,\cdot)$ at $\mathcal{X}_Q \times \mathcal{X}_C$, which has to be conducted element-wise. Moreover, it is more challenging to explore the locality of the integrand in Eq. (H.8). Therefore, the pruning of $\mathcal{X}_Q$ and $\mathcal{X}_C$ has not yet conducted in the existing literature. Obviously, SUR suffers from intensive computational time.

References

[1] R. E. Melchers, A. T. Beck, Structural reliability analysis and prediction, John wiley & sons, 2018.

[2] 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).

[3] R. Y. Rubinstein, D. P. Kroese, Simulation and the Monte Carlo method, John Wiley & Sons, 2016.

[4] A. Tabandeh, G. Jia, P. Gardoni, A review and assessment of importance sampling methods for reliability analysis, Structural Safety 97 (2022). doi:10.1016/j.strusafe.2022.102216.

[5] R. Melchers, Structural system reliability assessment using directional simulation, Structural Safety 16 (1-2) (1994) 23 – 37. doi:10.1016/0167-4730(94)00026-M.

[6] 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.

[7] P. Koutsourelakis, H. Pradlwarter, G. Schuëller, Reliability of structures in high dimensions, part I: Algorithms and applications, Probabilistic Engineering Mechanics 19 (4) (2004) 409 – 417. doi:10.1016/j.probengmech.2004.05.001.

[8] Y.-G. Zhao, Z.-H. Lu, Structural reliability: approaches from perspectives of statistical moments, John Wiley & Sons, 2021.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.
[14] 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.
[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] R. G. Ghanem, P. D. Spanos, Stochastic finite elements: a spectral approach, Springer-Verlag Inc, 1991.
[17] X. Zeng, R. Ghanem, Projection pursuit adaptation on polynomial chaos expansions, Computer Methods in Applied Mechanics and Engineering 405 (2023). doi:10.1016/j.cma.2022.115845.
[18] 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.
[19] 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.
[20] 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.
[21] D. Wang, D. Zhang, Y. Meng, M. Yang, C. Meng, X. Han, Q. Li, Ak-hrn: An efficient adaptive kriging-based n-hypersphere rings method for structural reliability analysis, Computer Methods in Applied Mechanics and Engineering 414 (2023). doi:10.1016/j.cma.2023.116146.
[22] T. Zhou, Y. Peng, Gaussian process regression based on deep neural network for reliability analysis in high dimensions, Structural and Multidisciplinary Optimization 66 (6) (2023). doi:10.1007/s00158-023-03582-0.
[23] T. Zhou, Y. Peng, Reliability analysis using adaptive polynomial-chaos kriging and probability density evolution method, Reliability Engineering and System Safety 220 (2022). doi:10.1016/j.ress.2021.108283.
[24] 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.
[25] C. Dang, P. Wei, J. Song, M. Beer, Estimation of failure probability function under imprecise probabilities by active learning-augmented probabilistic integration, ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering 7 (4) (2021). doi:10.1061/AJRUA6.0001179.
[26] 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.
[27] 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.
[28] 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.
[29] 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.
[30] 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.
[31] J. Yi, Q. Zhou, Y. Cheng, J. Liu, Efficient adaptive kriging-based reliability analysis combining new learning function and error-based stopping criterion, Structural and Multidisciplinary Optimization 62 (5) (2020) 2517 – 2536. doi:10.1007/s00158-020-02622-3.
[32] X. Zhang, L. Wang, J. D. Sørensen, Reif: A novel active-learning function toward adaptive kriging surrogate models for structural reliability analysis, Reliability Engineering and System Safety 185 (2019) 440 – 454. doi:10.1016/j.ress.2019.01.014.
[33] H. Wang, G. Lin, J. Li, Gaussian process surrogates for failure detection: A bayesian experimental design approach, Journal of Computational Physics 313 (2016) 247 – 259. doi:10.1016/j.jcp.2016.02.053.
[34] R. S. Sutton, A. G. Barto, Reinforcement learning: An introduction, MIT press, 2018.
[35] C. Dimitrakakis, R. Ortner, Decision Making Under Uncertainty and Reinforcement Learning: Theory and Algorithms, Springer International Publishing AG, 2022.
[36] T. Blau, E. V. Bonilla, I. Chades, A. Dezfouli, Optimizing sequential experimental design with deep reinforcement learning, Vol. 162, ML Research Press, 2022, p. 2107 – 2128.
[37] H.-E. Byun, B. Kim, J. H. Lee, Multi-step lookahead bayesian optimization with active learning using reinforcement learning and its application to data-driven batch-to-batch optimization, Computers and Chemical Engineering 167 (2022). doi:10.1016/j.compchemeng.2022.107987.
[38] C. Andriotis, K. Papakonstantinou, Deep reinforcement learning driven inspection and maintenance planning under incomplete information and constraints, Reliability Engineering and System Safety 212 (2021). doi:10.1016/j.ress.2021.107551.
[39] Z. Xiang, Y. Bao, Z. Tang, H. Li, Deep reinforcement learning-based sampling method for structural reliability assessment, Reliability Engineering and System Safety 199 (2020). doi:10.1016/j.ress.2020.106901.
[40] R. Garnett, Bayesian optimization, Cambridge University Press, 2023.
[41] Z. Sun, J. Wang, R. Li, C. Tong, Lif: A new kriging based learning function and its application to structural reliability analysis, Reliability Engineering and System Safety 157 (2017) 152 – 165. doi:10.1016/j.ress.2016.09.003.
[42] 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.
[43] 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.
[44] 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.
[45] X. You, M. Zhang, D. Tang, Z. Niu, An active learning method combining adaptive kriging and weighted penalty for structural reliability analysis, Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability 236 (1) (2022) 160 – 172. doi:10.1177/1748006X211016148.
[46] D. Gu, W. Han, J. Guo, H. Guo, S. Gao, X. Liu, A kriging-based adaptive adding point strategy for structural reliability analysis, Probabilistic Engineering Mechanics 74 (2023). doi:10.1016/j.probengmech.2023.103514.

[47] K. Khorramian, F. Oudah, New learning functions for active learning kriging reliability analysis using a probabilistic approach: Ko and wko functions, Structural and Multidisciplinary Optimization 66 (8) (2023). doi:10.1007/s00158-023-03627-4.
[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] J. Zhang, M. Xiao, L. Gao, An active learning reliability method combining kriging constructed with exploration and exploitation of failure region and subset simulation, Reliability Engineering and System Safety 188 (2019) 90 – 102. doi:10.1016/j.ress.2019.03.002.
[50] 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.
[51] T. Stolarski, Y. Nakasone, S. Yoshimoto, Engineering analysis with ANSYS software, Butterworth-Heinemann, 2018.
[52] 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.
[53] 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.
[54] D. B. Owen, A table of normal integrals, Communications in Statistics - Simulation and Computation 9 (4) (1980) 389–419. doi:10.1080/03610918008812164.
[55] Z. Drezner, Computation of the bivariate normal integral, Mathematics of Computation 32 (141) (1978) 277 – 279. doi:10.1090/S0025-5718-1978-0461849-9.
[56] Z. Drenzer, G. Wesolowsky, On the computation of the bivariate normal integral, Journal of Statistical Computation and Simulation 35 (1-2) (1990) 101 – 107. doi:10.1080/00949659008811236.