Metamodel-based subset simulation adaptable to target computational capacities: The case for high-dimensional or rare event reliability analysis

Zeyu Wang[1,2], Abdollah Shafieezadeh[2]
[1] Department of Civil Engineering, Tsinghua University, Beijing 100084, China
[2] The Ohio State University, Columbus, OH, 43210, United States
**Corresponding Author: Shafieezadeh.1@osu.edu

Abstract

Metamodel-based approaches to reliability analysis, e.g. adaptive Kriging, are computationally challenged by the complexity of reliability problems, thus limiting the application of these methods to problems that are low-dimensional or not rare. Here, we propose a reliability analysis approach via integration of subset simulation and adaptive kriging (RASA) for unbiased estimation of failure probabilities of high-dimensional or rare event problems. Concepts of Conditional Failure Probability Curves and Dynamic Learning Function are introduced to decompose the original problem to sub-reliability problems and adaptively identify intermediate failure thresholds of limit state functions corresponding to the sub-reliability problems. The reliability decomposition and the establishment of target intermediate failure thresholds are guided by the available computational capacity; thus, enabling RASA to control the computational cost associated with the estimation of the intermediate failure thresholds in each subset and consequently to analyze the reliability of medium to high-dimensional problems or rare events. Three numerical examples are investigated as benchmark to explore the performance of the proposed method. Results indicate that the proposed method has high accuracy and has the ability to adjust to available computational resources.

Key words: *Reliability analysis; Surrogate model; Subset Simulation; Adaptive Kriging; Condition Failure Probability Curve; Dynamic Learning Function;*

Nomenclature

$\mathbb{c}$	Function of the computational capacity	N_{ss}^{max}	Maximum number of candidate design samples in Kriging-based SS	$\boldsymbol{x}_{tr}$	Training samples
C_r	Required Computational storage	$\mathbb{p}$	Parameter sets for Kriging surrogate model	$\boldsymbol{x}_{tr}^{*}$	Next best training samples
$COV_{\hat{P}_f^{mcs}}$	Coefficient of variation of $\hat{P}_f^{mcs}$	p_0	Intermediate probability of failure	$\boldsymbol{Y}$	Vector of the responses from Kriging
$COV_{\hat{P}_f^{ss}}$	Coefficient of variation of $\hat{P}_f^{ss}$	p_0^{*}	Optimal intermediate probability of failure	Z	The Gaussian process
COV_{thr}	Coefficient of variation of $\hat{P}_f^{mcs}$	P_f	Probability of failure (ground truth)	$\boldsymbol{\beta}$	The vector of coefficients for Kriging basis
$\boldsymbol{f}$	The Kriging basis function	$\hat{P}_f^{di}$	Probability of failure estimated with deterministic indicator	Γ_{thr}	Stopping criterion threshold
$\boldsymbol{F}$	The vector of $\boldsymbol{f}$	P_f^{mcs}	Probability of failure estimated through MCS	$\boldsymbol{\theta}$	The vector of hyperparameters for Kriging
g	True limit state function	$\hat{P}_f^{mcs}$	Probability of failure estimated through Kriging-based MCS	Θ_t	pdf of identified intermediate failure thresholds

$\hat{g}$	Estimated limit state function	$\hat{P}_f^{si}$	Probability of failure estimated with stochastic indicator	$\mu_{\hat{g}}$	The mean value of Kriging responses
g_i	True sub-limit state function in SS	P_f^{ss}	Probability of failure estimated with SS	ρ	Probability density function (pdf)
$\hat{g}_i$	Estimated sub-limit state function in SS	$\hat{P}_f^{ss}$	Probability of failure estimated through Kriging-based SS	σ^2	Variance of Gaussian Process
I_g	Indicator based on g	p_k	Intermediate failure probability for SS	$\sigma_{\hat{g}}^2$	The variance of Kriging Responses
$I_{\hat{g}}^{si}$	Indicator based on $\hat{g}$	$\hat{p}_k$	Intermediate failure probability for Kriging-based SS	φ	pdf of Gaussian distribution
m	Number of samples for Kriging training	$\boldsymbol{r}$	Correlation vector	Φ	cdf of Gaussian distribution
n_{ss}	Number of subsets	R	The Kriging correlation function between two points	Ω	Probabilistic domain of $\boldsymbol{x}$
N_{cds}	Number of candidate design samples for any algorithm	S	Pool of generated samples in Ω_f	Ω_f	Failure domain of Ω
N_d	Number of the dimension of random variables	t_i	True intermediate failure thresholds	Ω_i	The *i*th subsets in Subset Simulation
N_{MCS}	Number of samples for MCS	$\hat{t}_i$	Estimated intermediate failure thresholds	Ω_s	Failure domain of Ω
N_{ss}	Number of candidate design samples in SS	$\boldsymbol{x}$	The vector of random variables		

1. Introduction

Reliability analysis has become increasingly essential in various fields of engineering and science. These analyses are often concerned with the quantitative assessment of the safety of systems. In the processes of design and manufacturing complex systems such as rockets and their propulsion systems, satellites and Unmanned Aerial Vehicles, reliability analysis is crucial for considering extreme requirements of mission success under unpredictable conditions [1]. Reliability analysis is also embedded in the procedure of 'design by reliability', which ensures that estimated reliabilities of designed mechanical components and systems are acceptable [2]. Moreover, safety is a critically important consideration in nuclear engineering, since the failure of a nuclear power plant may lead to devastating consequences for the society [3]. Another area of interest is the hazard performance of structures and infrastructure systems. These systems are vulnerable against various hazards such as earthquake, tsunami, flood, and tornado [4]. Reliability analysis enables analyzing the performance of components and systems in terms of the probability of failing to meet a prescribed objective considering aleatoric and epistemic uncertainties. Therefore, estimation of failure probability, here denoted as P_f, is indispensable for quantification of risks and design and risk management of various systems. In this article, P_f is commonly defined as:

$$P_f = \int_{\Omega_{g(x)\leq 0}} \rho(\boldsymbol{x})\, dx = \int_{\Omega} I_g(\boldsymbol{x})\rho(\boldsymbol{x})d\boldsymbol{x} \tag{1}$$

where $\rho(\boldsymbol{x})$ is the Probability Density Function (PDF) of the random variables, $\boldsymbol{x}$, Ω is the domain of $\boldsymbol{x}$, $\Omega_{g\leq 0}$ is the integration domain where the performance function satisfies $g(\boldsymbol{x}) \leq 0$, and I_g is the indicator

function, with I_g=1, when $g(\boldsymbol{x}) \leq 0$ and I_g=0 when $g(\boldsymbol{x}) > 0$. A number of techniques have been developed and implemented for estimating reliability or other tasks of uncertainty quantification. These include, among others, crude Monte Carlo simulation (MCS) [5], [6], importance sampling [7] , Subset Simulation (SS) [8], first or second order reliability method (FORM & SORM) [9], [10] and surrogate model-based approaches such as those based on response surface [11], [12], [13], polynomial chaos expansion [14], support vector regression [15], [16], or Kriging [17]–[20].

Kriging-based reliability analysis methods have gained significant attention due to their computational efficiency and accuracy. Different from other surrogate models that provide only the best estimate of responses, Kriging model for outputs provide a normal distribution with Kriging mean and variance. By taking advantage of the stochastic output model, the limit state in reliability problems can be adaptively refined by strategically enriching the set of training points in the vicinity of the limit state $g(\boldsymbol{x}) = 0$. Advancements in Kriging-based reliability analysis are elaborated in [17], [21], [22]. Among these methods, Efficient Global Reliability Analysis (EGRA) by Bichon et al. [23] and Adaptive Kriging with Monte Carlo Simulation(AK-MCS) by Echard et al. [17] are widely accepted and used as benchmark to analyze the performance of other techniques. To enhance sampling strategies for rare events, Echard et al. [7], Balesdent et al. [24] and Dubourg et al. [25] used importance sampling techniques in association with adaptive Kriging models. Moreover, studies in [26],[27], and [28] have proposed combining Subset Simulation and Kriging. Truncating insignificant candidate samples that have small values of probability density is also shown to be an effective strategy in improving the computational efficiency of Kriging-based reliability analysis [12], [13], [29]. While past developments enhanced the performance of Kriging-based reliability analysis in different aspects, these methods still face key challenges. For example, training a Kriging model becomes extremely computationally demanding or even intractable when the reliability problems involve a large number of random variables. To obtain reliable estimates for problems with small failure probability, a very large number of candidate design samples are required. Significant computational demands are faced in these cases due to complex matrix operations needed for constructing correlation functions for training and candidate design samples as well as optimization procedures for estimating hyper-parameters. These limitations for Kriging-based methods have significant implications for their applicability, as real-world reliability problems are often high-dimensional or have small failure probability.

To address this gap, a new method called Reliability Analysis using Subset simulation and Adaptive Kriging (RASA) is proposed in this paper. This approach integrates Kriging surrogate modeling with Subset Simulation. The main idea behind RASA is to decompose highly computationally demanding or intractable problems such as high-dimensional reliability problems to a number of sub-reliability problems each with a controlled number of candidate design samples for adaptive Kriging. This is accomplished by training Kriging surrogate models based on the candidate design samples in the subsets via Subset Simulation and not the entire candidate design samples, which is used in regular Kriging-based reliability analysis. The intermediate failure probability, denoted as p_o, in Subset Simulation is considerably larger than the target failure probability, P_f. The implication is that the required number of candidate design samples in the subsets is significantly smaller than the total number of candidate samples in regular Kriging-based MCS. The relatively very small number of candidate design samples in the subsets offers the Kriging model the capability to tackle computationally demanding reliability problems such as high dimensional or rare event problems. Moreover, RASA can strategically adjust p_o and the number of candidate design samples in each subset, denoted as N_{ss}, to satisfy the requirement of the coefficient of variation of estimated failure probability, denoted as $COV_{\hat{p}_f^{ss}}$. If the required threshold of $COV_{\hat{p}_f^{ss}}$ is not prescribed, RASA can also find the optimal value of p_o and N_{ss} to minimize the coefficient of variation. However, the most challenging task here is to identify the intermediate failure thresholds in the process of implementing Subset Simulation based on the information provided by the Kriging model. Toward this goal, two novel concepts called Conditional Failure Probability Curve (CFPC) and Dynamic Learning Function (DLF) are proposed in this article. CFPC is a curve that represents the relation between

the value of intermediate failure threshold and intermediate failure probability with the corresponding confidence interval. On the other hand, DLF is a learning function to strategically add training points that can reduce the uncertainty near the identified intermediate thresholds. After the final intermediate failure threshold is identified to be smaller than zero, conditional failure probability for the last subset is estimated to complete the RASA algorithm. The proposed reliability analysis method offers the capability to control the number of candidate design samples throughout the process to completely avoid the otherwise computational intractability of complex reliability problems. Three numerical examples are presented in this article to showcase the performance of the proposed method in solving challenging problems.

The paper is organized in seven sections. In Section 2, the elements of Kriging along with stochastic indicator-based MCS are briefly reviewed. Section 3 presents an overview of Subset Simulation. In Section 4, the proposed method RASA is presented along with several theorems and propositions. In Section 5, Conditional Failure Probability Curves and Dynamic Learning Function with corresponding computational details are introduced. In Section 6, two examples with different dimensions and failure probabilities are investigated to examine the applicability and performance of RASA for challenging problems. Section 7 presents the conclusions of this research.

2. Adaptive Kriging-based Reliability Analysis

2.1 The Kriging elements

The Kriging or the Gaussian Process Regression (GPR), $\hat{g}(\boldsymbol{x})$, has been widely used in reliability analysis for its high accuracy and efficiency [30]. The mathematical form of this model is shown below,

$$\hat{g}(\boldsymbol{x}) = F(\boldsymbol{\beta}, \boldsymbol{x}) + Z(\boldsymbol{x}) = \boldsymbol{\beta}^T \boldsymbol{f}(\boldsymbol{x}) + Z(\boldsymbol{x}) \tag{2}$$

where $F(\boldsymbol{\beta}, \boldsymbol{x})$ is the regression component representing the general trend of the response and $Z(\boldsymbol{x})$ is the stochastic interpolation based on the Gaussian assumption. $F(\boldsymbol{\beta}, \boldsymbol{x})$ can be expanded to $\boldsymbol{f}(\boldsymbol{x})$ as the Kriging basis and $\boldsymbol{\beta}$ as the regression coefficients. $\boldsymbol{\beta}^T f(\boldsymbol{x})$ often takes ordinary (β_0), linear ($\beta_0 + \sum_{i=1}^{N_d} \beta_i x_i$) or quadratic ($\beta_0 + \sum_{i=1}^{N_d} \beta_i x_i + \sum_{i=1}^{N_d} \sum_{j=i}^{N_d} \beta_{ij} x_i x_j$) forms, where N_d is the dimension of the random variable $\boldsymbol{x}$. The ordinary Kriging model is implemented throughout this paper. Moreover, $Z(\boldsymbol{x})$ is a Gaussian process with zero mean and covariance matrix as shown below,

$$\mathrm{Cov}\left(Z(\boldsymbol{x}_i), Z(\boldsymbol{x}_j)\right) = \sigma^2 R(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\theta}) \tag{3}$$

where $\mathrm{Cov}(\cdot)$ denotes the operation of covariance, σ^2 is the process variance or the generalized mean square error of the regression component, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are two observations, and $R(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\theta})$ is the correlation function representing the correlation of the process with hyper-parameter $\boldsymbol{\theta}$. Candidate forms of these correlation functions include linear, exponential, Gaussian, and Matérn functions, among others. In this paper, the Gaussian kernel function is implemented, which has the following formulation,

$$R(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\theta}) = \prod_{k=1}^{N_d} \exp\left(-\theta^k \left(x_i^k - x_j^k\right)^2\right) \tag{4}$$

The hyper-parameter $\boldsymbol{\theta}$ can be estimated via maximum likelihood estimation (MLE) or cross validation [30]. Moreover, the anisotropic Kriging model is adopted in this study, which means that the hyper-parameter $\boldsymbol{\theta}$ is optimized in each dimension. To keep the consistency with previous studies, the range of θ^k in the optimization process is considered as $(0,10)$ [4]. The maximum likelihood estimation of $\boldsymbol{\theta}$ can be presented as,

$$\boldsymbol{\theta} = \underset{\boldsymbol{\theta}'}{\operatorname{argmin}} \left(\left| \boldsymbol{R}(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\theta}') \right|^{\frac{1}{m}} \sigma^2 \right) \tag{5}$$

where m denotes the number of training samples for Kriging. Accordingly, the regression coefficient $\boldsymbol{\beta}$, and the Kriging estimated mean and variance can be determined as follows [30],

$$\begin{aligned} \boldsymbol{\beta} &= (\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{F})^{-1} \boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{Y} \\ \mu_{\hat{g}}(\boldsymbol{x}) &= \boldsymbol{f}^T(\boldsymbol{x}) \boldsymbol{\beta} + \boldsymbol{r}^T(\boldsymbol{x}) \boldsymbol{R}^{-1} (\boldsymbol{y} - \boldsymbol{F}\boldsymbol{\beta}) \\ \sigma_{\hat{g}}^2(\boldsymbol{x}) &= \sigma^2 (1 - \boldsymbol{r}^T(\boldsymbol{x}) \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) + (\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x}))^T (\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{F})^{-1} (\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x}))) \end{aligned} \tag{6}$$

where σ^2 denotes the process variance calculated as $\frac{1}{m}(\boldsymbol{y} - \boldsymbol{\beta F})^t \boldsymbol{R}^{-1} (\boldsymbol{y} - \boldsymbol{\beta F})$ and $\boldsymbol{F}$ is the matrix of the basis function, $\boldsymbol{f}(\boldsymbol{x})$, evaluated at known training points, i.e. $F_{ij} = f_j(\boldsymbol{x}_i)$, $i = 1, 2, \ldots, m$; $j = 1,2, \ldots, p$, where p is the pth element of the regression term, $\boldsymbol{r}(\boldsymbol{x})$ is the vector of correlation between known training points $\boldsymbol{x}_i$ and an unknown point $\boldsymbol{x}$: $r_i = R(\boldsymbol{x}, \boldsymbol{x}_i, \boldsymbol{\theta})$, $i = 1,2 \ldots m$, and R is the autocorrelation matrix for known training points: $R_{ij} = R(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{\theta})$, $i = 1,2, \ldots, m; j = 1,2, \ldots, m$. The responses from Kriging follow a normal distribution with Kriging mean $\mu_{\hat{g}}(\boldsymbol{x})$ and Kriging variance $\sigma_{\hat{g}}^2(\boldsymbol{x})$,

$$\hat{g}(\boldsymbol{x}) \sim N\left(\mu_{\hat{g}}(\boldsymbol{x}), \sigma_{\hat{g}}^2(\boldsymbol{x})\right) \tag{7}$$

Compared with points that are farther away from the training points, responses of points close to the training points are expected to have less uncertainty.

2.2 Adaptive Kriging-based reliability analysis

Reliability analysis methods based on adaptive Kriging aim to replace the true limit state function (LSF) with a Kriging model. The surrogate model is adaptively trained to reach the desired accuracy. The general procedure of these methods is summarized in Algorithm 1.

Algorithm 1. Adaptive Kriging-based Reliability Analysis

1. Define initial parameters such as Kriging trends, optimization algorithm types, Generating initial candidate design samples *S* with sampling techniques (e.g., Latin Hypercube Sampling (LHS)).
2. Randomly select initial training samples $\boldsymbol{x}_{tr}$ from *S* and evaluate their responses, $g(\boldsymbol{x}_{tr})$.
3. Construct the Kriging model $\hat{g}(\boldsymbol{x})$ based on $\boldsymbol{x}_{tr}$ and $g(\boldsymbol{x}_{tr})$.
4. Estimate the mean $\mu_{\hat{g}}(\boldsymbol{x})$, standard deviation $\sigma_{\hat{g}}(\boldsymbol{x})$ and $\hat{P}_f^{mcs}$ for *S* with $\hat{g}(\boldsymbol{x})$ using the *deterministic* or *stochastic indicator*.
5. Search for the next best training points $\boldsymbol{x}_{tr}^*$ using a *learning function* and update the set of training samples $\boldsymbol{x}_{tr}$.
6. Check if the *stopping criterion* is satisfied:
   (a) If satisfied, go to step 7.
   (b) If not satisfied, estimate the response $g(\boldsymbol{x}_{tr}^*)$ for $\boldsymbol{x}_{tr}^*$ and go back to step 3.
7. Output $\hat{P}_f^{mcs}$.

Two indicator functions are available for Kriging-based MCS: the deterministic indicator [17], [31] and the stochastic indicator [25], [32] (denoted as DI and SI, respectively, hereafter). For DI, the probability of failure can be estimated as,

$$\hat{P}_f^{mcs} = \hat{P}_f^{di} = \frac{1}{N_{MCS}} \sum_{i=1}^{N_{MCS}} I_g(\boldsymbol{x}_i), \qquad \boldsymbol{x} \in S \tag{8}$$

where $\hat{P}_f^{di}$ is the estimated failure probability with DI based on the Kriging model. N_{MCS} denotes the number of samples, $\boldsymbol{x}_i$, generated from the probability distribution of random variables and $I_{\hat{g}}^{dc}$ is the indicator function for DI,

$$I_{\hat{g}}^{di}(\boldsymbol{x}_i) = \begin{cases} 1, & \mu_{\hat{g}}(\boldsymbol{x}_i) \le 0 \\ 0, & \mu_{\hat{g}}(\boldsymbol{x}_i) > 0 \end{cases}, \qquad \boldsymbol{x}_i \in S \tag{9}$$

The estimate of failure probability based on stochastic indicator can be derived as,

$$\hat{P}_f^{mcs} = \hat{P}_f^{si} = \frac{1}{N_{MCS}} \mathrm{E}[\mathbb{U}] = \frac{1}{N_{MCS}} \sum_{i=1}^{N_{MCS}} \Phi\left(\frac{-\mu_{\hat{g}}(\boldsymbol{x}_i)}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right), \qquad \boldsymbol{x}_i \in S \tag{10}$$

$$\mathbb{U} = \sum_{i=1}^{N_{MCS}} I_{\hat{g}}^{si}(\boldsymbol{x}_i), \boldsymbol{x}_i \in S \tag{11}$$

where $\hat{P}_f^{si}$ is the estimated failure probability with SI based on the Kriging model, E[·] is the expectation operator, and $I_{\hat{g}}^{si}(\boldsymbol{x}_i)$ denotes the indicator function for stochastic indicator,

$$I_{\hat{g}}^{si}(\boldsymbol{x}_i) = \begin{cases} 1, & \text{with probability } \Phi\left(\dfrac{-\mu_{\hat{g}}(\boldsymbol{x}_i)}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right) \\ 0, & \text{with probability } 1 - \Phi\left(\dfrac{-\mu_{\hat{g}}(\boldsymbol{x}_i)}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right) \end{cases}, \qquad \boldsymbol{x}_i \in S \tag{12}$$

where $\Phi(x)$ is the cumulative distribution function (CDF) of the univariate standard normal distribution. The performance of the SI-based MCS is equivalent to the deterministic classification-based approach [32]. Note that $\mathbb{U}$ is a random variable, thus, $\hat{P}_f^{si}$ is the mean value of $\frac{\mathbb{U}}{N_{MCS}}$ according to Eq. (11). In this paper, the SI-based MCS is adopted because it has the capability to provide stochastic information for the estimated failure probabilities.

3. Subset Simulation

By adaptively decomposing the original limit state function into a series of computationally less demanding LSFs with intermediate failure thresholds, Au and Beck [8] proposed Subset Simulation to efficiently estimate probabilities of failure. Let's denote the subsets in Subset Simulation as $\Omega_1 \supset \Omega_2 \supset \cdots \supset \Omega_{n_{ss}} = \Omega_f$ and $\Omega_f = \bigcap_{i=1}^{n_{ss}} \Omega_i$, where n_{ss} denotes the number of subsets and Ω_f is the failure domain equivalent to the original LSF. The subset Ω_i is the failure domain corresponding to the LSF,

$$\Omega_i = \{\boldsymbol{x}: g(\boldsymbol{x}) \le t_i\} \tag{13}$$

where t_i is the so-called intermediate failure threshold: $t_1 > t_2 > \cdots > t_m = 0$. An illustration of Subset Simulation is shown in Fig. 1. The probability of failure can then be estimated as,

$$P_f \approx P_f^{ss} = \mathrm{P}(\Omega_{n_{ss}}) = \mathrm{P}\left(\bigcap_{i=1}^{n_{ss}} \Omega_i\right) = \mathrm{P}(\Omega_1) \cdot \prod_{i=1}^{n_{ss}-1} \mathrm{P}(\Omega_{i+1}|\Omega_i) \tag{14}$$

where $\mathrm{P}(\Omega_1)$ and $\mathrm{P}(\Omega_{i+1}|\Omega_i)$ can be estimated using MCS and MCMC with the corresponding limit state functions $g(\boldsymbol{x}) \leq t_i$, respectively. The probability of failure can be estimated as,

$$P_f^{ss} = \mathrm{P}(\Omega_1) \cdot \prod_{i=1}^{n_{ss}-1} \mathrm{P}(\Omega_{i+1}|\Omega_i) = \prod_{i=1}^{n_{ss}} p_i \tag{15}$$

In the implementation of Subset Simulation, the intermediate conditional probability is typically set as: $\mathrm{P}(\Omega_{i+1}|\Omega_i) \approx p_0 = 0.1$. It is shown that the speed of convergence and accuracy are related to the definition of p_0. The steps for implementing Subset Simulation are presented in Algorithm 2.

Algorithm 2. Subset Simulation

1. Generate N_{SS} samples $\boldsymbol{x}_k \ = 1, \ldots N_{SS}$ through the crude MCS and evaluate their responses $g(\boldsymbol{x}_k), k = 1, \ldots N_{SS}$.
2. $i = 1$
3. (a) If $i = 1$, determine t_1 such that $\mathrm{P}(\Omega_1) \approx p_0$.
   (b) If $i > 1$, determine the intermediate failure thresholds t_i such that the conditional probabilities satisfy $\mathrm{P}(\Omega_{i+1}|\Omega_i) \approx p_0$.
4. Generate samples in Ω_{i+1} using the Markov Chain Monte Carlo Simulation technique (MCMC), where $i \geq 1$.
5. $i = i + 1$. Return to step 3, if $t_i > 0$; otherwise, continue to step 6.
6. Estimate the last failure probability $p_{n_{ss}} = \mathrm{P}\left(\Omega_{n_{ss}}|\Omega_{n_{ss}-1}\right)$ for the final subset $\Omega_{n_{ss}}$ with $t_{n_{ss}} = 0$.
7. Estimate the failure probability P_f^{ss}.

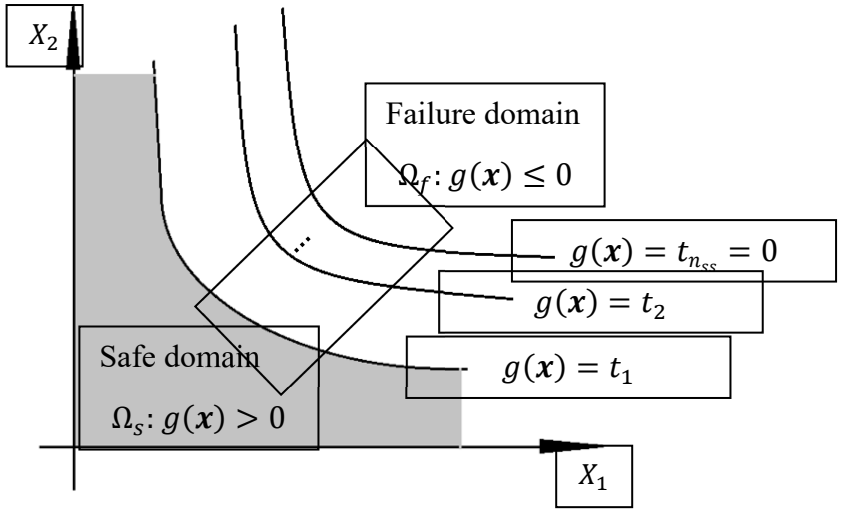

Fig. 1 Illustration of safe and failure domains and the limit state $g(\boldsymbol{x}) = 0$ in Subset Simulation

4. RASA: The Conceptual Framework

The core idea of RASA is to control the size of the pool of candidate design samples to enable analysis of computationally very demanding reliability problems such as high-dimensional or rare event problems. In this paper, high-dimensional problems refer to reliability analysis for limit states that involve 20 or more random variables ($N_d \geq 20$). Toward this goal, Subset Simulation is integrated with the Kriging surrogate model to enhance the computational performance of reliability analysis. Generally, a reliable

estimate of the probability of failure can be obtained through Algorithm 1. The coefficient of variation of failure probability, $COV_{\hat{P}_f^{mcs}}$, can be determined as,

$$COV_{\hat{P}_f^{mcs}} = \sqrt{\frac{1-\hat{P}_f^{mcs}}{N_{cds}\hat{P}_f^{mcs}}} \tag{16}$$

where N_{cds} denotes the number of candidate design samples. If the target $COV_{\hat{P}_f^{mcs}}$ is prescribed, the required computational capacity C_r to solve the problem can be expressed as,

$$C_r = \mathbb{c}(N_d, N_{cds}, \mathbb{P}) \tag{17}$$

where $N_{cds} = N_{mcs}$ for Algorithm 1 and $N_{cds} = N_{ss}$ for Algorithm 2, $\mathbb{P}$ is the set of parameters defining the Kriging surrogate model, and $\mathbb{c}(\cdot)$ denotes the required computational capacity for an algorithm parameterized by N_d, N_{cds} and $\mathbb{P}$. Basically, C_r increases as N_d and N_{cds} increase,

$$C_r \propto N_d, N_{cds} \tag{18}$$

As N_d is typically prescribed, reducing N_{cds} can improve the computational efficiency. However, reducing N_{cds} can increase $COV_{\hat{P}_f^{mcs}}$ according to Eq. (16). Therefore, a key task is to design an algorithm that satisfies the condition $COV_{P_f} \leq COV_{thr}$ with respect to the prescribed N_{cds}, where COV_{P_f} generally refers to the COV of the probability of failure estimated through any appliable algorithm. Toward this goal, the next section presents a novel approach to improve the computational efficiency of reliability analysis with limited computational capacity.

4.1 Proposed RASA method

RASA is aimed at adaptively adjusting N_{ss} and p_0 so that the probability of failure can be estimated with limited available computationally capacity. First, $\hat{P}_f^{ss}$ is roughly estimated based on the initial parameter sets. Then N_{ss} is adaptively updated and adjusted to satisfy the requirement of COV_{thr}. However, it may happen that the updated $N_{ss} = N_{ss}^{max}$ can still not satisfy the condition that $COV_{\hat{P}_f^{ss}} < COV_{\text{thr}}$, where N_{ss}^{max} denotes the max number of candidate design samples that can be carried out and $COV_{\hat{P}_f^{ss}}$ denotes the COV of the probability of failure estimated through Kriging-based Subset Simulation. Therefore, the value of p_0 also needs to be updated in this setting. Hence, the main steps of RASA can be summarized as adjustment of N_{ss} and p_0 so that C_r can be controlled subject to the constraint of COV_{thr}. Details of the proposed RASA algorithm are elaborated in Algorithm 3. Due to the fact that not all the training samples generated in the last subsets can render useful information for the construction of the surrogate model in the next subsets, only a portion of training samples in previous subsets are selected as the initial training samples for the next subsets to reduce the computational burden. According to an experimental study, the portion can be selected as 20%. After $\hat{P}_f^{ss}$ is roughly estimated, N_{ss} can be updated according to the following equation (the derivation is presented in section 4.2),

$$N_{ss} = \sum_{i=1}^{n_{ss}} \frac{1-\hat{p}_k}{COV_{thr}^2 \hat{p}_k} \tag{19}$$

where $\hat{p}_k$ denotes the intermediate probability of failure in the kth subset estimated through the Kriging surrogate model. However, N_{ss} cannot exceed the maximum computational capacity, N_{ss}^{max}. For Algorithm 3, the minima of $COV_{\hat{P}_f^{ss}}$ can be estimated using the following equation,

$$p_0^* = \underset{p_0 \in (0,1)}{\arg\min}\, COV_{\hat{P}_f^{ss}} \quad \text{s.t. } C_r < C_c \tag{20}$$

where C_c denotes the algorithmic computational capacity and $\arg\min(\cdot)$ stands for argument of the minimum. If N_{ss} and the intermediate failure probabilities $\hat{p}_k, k = 1, 2, \dots, n_{ss}$ in every subset except the last one are kept the same, the optimal intermediate failure probability can be determined by the following equation,

$$\begin{aligned} p_0^* &= \underset{p_0 \in (0,1)}{\arg\min}\, COV_{\hat{P}_f^{ss}} = \underset{p_0 \in (0,1)}{\arg\min} \sqrt{\sum_{i=1}^{n_{ss}} \frac{1-\hat{p}_k}{N_{ss}^{max}\hat{p}_k}} \\ &= \underset{p_0 \in (0,1)}{\arg\min} \left[\sum_{i=1}^{n_{ss}-1} \frac{1-p_0}{N_{ss}^{max} p_0} + \left(1 - \frac{\hat{P}_f^{ss}}{p_0^{n_{ss}-1}}\right) \Big/ \left(N_{ss}^{max} \frac{\hat{P}_f^{ss}}{p_0^{n_{ss}-1}}\right) \right] \\ &= \underset{p_0 \in (0,1)}{\arg\min} \left[\sum_{i=1}^{n_{ss}-1} \frac{1-p_0}{N_{ss}^{max} p_0} + \frac{p_0^{n_{ss}-1} - \hat{P}_f^{ss}}{N_{ss}^{max} \hat{P}_f^{ss}} \right] \end{aligned} \tag{21}$$

Algorithm 3. Reliability Analysis with Subset simulation using Adaptive Kriging (RASA)

1. (a) Define COV_{thr}, and the initial N_{SS} and p_0.
   (b) Generate N_{SS} samples $\boldsymbol{x}_k = 1, \dots, N_{SS}$ through crude MCS and estimate their responses $g(\boldsymbol{x}_k), k = 1, \dots, N_{SS}$.
2. $i = 1$
3. (a) If $i = 1$, identify t_1 using Algorithm 2 such that $\mathrm{P}(\Omega_1) \approx p_0$.
   (b) If $i > 1$, determine the intermediate failure thresholds t_i using Algorithm 4 such that the conditional probabilities satisfy $\mathrm{P}(\Omega_{i+1}|\Omega_i) \approx p_0$.
4. Generate samples in Ω_i through crude MCS (if the probability of failure is not rare) or MCMC based on the remaining points (i.e., seeds).
5. $i = i + 1$. Return to step 3 if $\hat{t} > 0$; Otherwise, continue to step 6.
6. Estimate the intermediate failure probability $\hat{P}_0^m = \mathrm{P}(\Omega_{m-1}|\Omega_m)$ in the last subset $\Omega_m$with $t_m = 0$ using Algorithm 5.
7. Estimate the failure probability $\hat{P}_f^{ss}$ and $COV_{\hat{P}_f^{ss}}$ using Eq. (16) and Eq. (27).
8. Check
   (a) if $COV_{\hat{P}_f^{ss}} > COV_{thr}$ and $N_{ss} \le N_{ss}^{max}$, increase N_{SS} using Eq. (19) and go back to step 2. This step needs to utilize the previous training samples.
   (b) if $COV_{\hat{P}_f^{ss}} > COV_{thr}$ and $N_{ss} \ge N_{ss}^{max}$, go to step 9.
   (c) if $COV_{\hat{P}_f^{ss}} \le COV_{thr}$, output $\hat{P}_f^{ss}$.
9. Estimate p_0^* with corresponding $COV_{\hat{P}_f^{ss}}^{min}$. Check
   (b) if $COV_{\hat{P}_f^{ss}}^{min} \le COV_{thr}$, update p_0 using Eq. (25) and go back to step 2. This step also needs to utilize the previous training samples.
   (b) else, update $p_0 = p_0^*$ and repeat step 2-7, and then output $\hat{P}_f^{ss}$.

where the intermediate probability of failure in the last subset can be estimated as $p_{n_{ss}} = \frac{\hat{P}_f^{ss}}{p_0^{n_{ss}-1}}$ based on the following equation,

$$p_0{}^{n_{ss}} = \hat{P}_f^{ss} \tag{22}$$

Thus,

$$n_{ss} = \frac{log\hat{P}_f^{ss}}{logp_0} \tag{23}$$

Since n_{ss} is an integer, the total number of subsets can be calculated as $n_{ss} = \text{INT}\left[\frac{log\hat{P}_f^{ss}}{logp_0}\right]$, where INT[·] denotes the operator that rounds to the nearest larger integer. Hence, Eq. (23) can be further expanded to,

$$\begin{aligned} p_0^* &= \underset{p_0\in(0,1)}{\arg\min}\left[\sum_{i=1}^{\text{INT}\left[\frac{log\hat{P}_f^{ss}}{logp_0}\right]-1}\frac{1-p_0}{N_{ss}^{max}p_0} + \left(1-\frac{\hat{P}_f^{ss}}{p_0{}^{\text{INT}\left[\frac{log\hat{P}_f^{ss}}{logp_0}\right]-1}}\right)\Big/\left(N_{ss}^{max}\frac{\hat{P}_f^{ss}}{p_0{}^{\text{INT}\left[\frac{log\hat{P}_f^{ss}}{logp_0}\right]-1}}\right)\right] \\ &= \underset{p_0\in(0,1)}{\arg\min}\left[\left(\text{INT}\left[\frac{log\hat{P}_f^{ss}}{logp_0}\right]-1\right)\frac{1-p_0}{N_{ss}^{max}p_0} + \frac{p_0{}^{\text{INT}\left[\frac{log\hat{P}_f^{ss}}{logp_0}\right]-1}-\hat{P}_f^{ss}}{N_{ss}^{max}\hat{P}_f^{ss}}\right] \end{aligned} \tag{24}$$

The corresponding $COV_{\hat{P}_f^{ss}}^{min}$ can be estimated as $\sqrt{\sum_{i=1}^{n_{ss}}\frac{1-p_0^*}{N_{ss}p_0^*}}$. Moreover, p_0 can be determined according to the established COV_{thr} as,

$$p_0 = \mathbb{w}^{-1}(COV_{thr}) \tag{25}$$

where $\mathbb{w}^{-1}$ denotes the inverse of function $\mathbb{w}$, which is defined as,

$$\mathbb{w}\left(p_0, \hat{P}_f^{ss}, N_{ss}^{max}\right) = \sqrt{\left(\text{INT}\left[\frac{log\hat{P}_f^{ss}}{logp_0}\right]-1\right)\frac{1-p_0}{N_{ss}^{max}p_0} + \frac{p_0{}^{\text{INT}\left[\frac{log\hat{P}_f^{ss}}{logp_0}\right]-1}-\hat{P}_f^{ss}}{N_{ss}^{max}\hat{P}_f^{ss}}} \tag{26}$$

Note that $\mathbb{w}(\cdot)$ is typically not a monotonic function, thus, there is a potential that more than one p_0 can satisfy Eq. (26). Therefore, one can select a p_0 such that n_{ss} can be minimized. The advantages of the proposed RASA algorithm are discussed in the next subsections. Moreover, the approach to precisely identify the intermediate failure thresholds are elaborated in Section 5 by introducing two innovative concepts called Conditional Failure Probability Curve and Dynamic Learning Function.

4.2 Unbiased property of the stochastic estimator

Classical Subset Simulation tends to obtain samples in the failure region associated with $P(\Omega_{i+1}|\Omega_i)$ using Markov Chain Monte Carlo simulation (MCMC). This approach is motivated by the fact that generating samples located in the posterior failure region through brute MCS can be computationally very

demanding. The MCMC technique facilitates fast generation of samples that follow the posterior distribution by introducing a proposal density also called 'jumping distribution'. However, the obtained samples are often biased due to the mutual correlation of Markov Chains. Mitigating this effect would require proper definition of the parameters of the proposal density, which becomes challenging as the dimension of the problem increases, thus leaving the potential for the final estimate of failure probability by this method to be inaccurate. This limitation can be completely avoided if one takes advantage of the fast analysis speed of surrogate models. Such models enable generation of all candidate design samples in subsets through brute MCS. Properties of the stochastic estimator in Algorithm 3 for the analysis of failure probability are explored through the following theorems.

Theorem 1. The estimate of the probability of failure using Subset Simulation through brute MCS is unbiased with the coefficient of variation of,

$$COV_{\hat{P}_f^{ss}} \cong \sqrt{\sum_{i=1}^{n_{ss}} \frac{1-\hat{p}_k}{N_{ss}\hat{p}_k}} \tag{27}$$

where n_{ss} denotes the total number of subsets, N_{ss} is the number of generated candidate design samples in each subset and p_k is the intermediate probability of failure.

Proof: According to [8] , we have,

$$E\left[\frac{\hat{P}_f^{ss}-P_f}{P_f}\right] = \sum_{i>j} \delta_i\delta_j E[\mathbb{z}_i\mathbb{z}_j] + \sum_{i>j>k} \delta_i\delta_j\delta_k E[\mathbb{z}_i\mathbb{z}_j\mathbb{z}_k] + \cdots \left(\prod_{i=1}^{n_{ss}} \delta_i\right) E\left[\prod_{i=1}^{n_{ss}} \mathbb{z}_i\right] \tag{28}$$

where δ_i denotes the COV of $\hat{p}_k$ and $\mathbb{z}_i$ is calculated as $\mathbb{z}_i = (\hat{p}_k - p_k)/\delta_i$. Since all the samples are generated through brute MCS and not MCMC, $\mathbb{z}_i$s are mutually uncorrelated; therefore $E[\mathbb{z}_i\mathbb{z}_j] = 0$ for i > j, $E[\mathbb{z}_i\mathbb{z}_j\mathbb{z}_k] = 0$ for $i > j > k$, and $E[\prod_{i=1}^{n_{ss}} \mathbb{z}_i] = 0$. As a result, the stochastic estimator is unbiased. Moreover, considering the following equation [8],

$$E\left[\frac{\hat{P}_f^{ss}-P_f}{P_f}\right]^2 = E\left[\sum_{i=1}^{n_{ss}} \delta_i\mathbb{z}_i + \sum_{i>j} \delta_i\delta_j\mathbb{z}_i\mathbb{z}_j + \prod_{i=1}^{n_{ss}} \delta_i\mathbb{z}_i\right]^2 = \sum_{i,j=1}^{n_{ss}} \delta_i\delta_j\, E[\mathbb{z}_i\mathbb{z}_j] + o\left(\frac{1}{N_{ss}}\right) \tag{29}$$

it can be shown that

$$E\left[\frac{\hat{P}_f^{ss}-P_f}{P_f}\right]^2 \cong \sum_{i=j=1}^{n_{ss}} \delta_i\delta_j\, E[\mathbb{z}_i\mathbb{z}_j] \tag{30}$$

The following equation can be subsequently derived,

$$E\left[\frac{\hat{P}_f^{ss}-P_f}{P_f}\right]^2 \cong \sum\nolimits_{i=1}^{n_{ss}} \sigma_i^2 \tag{31}$$

which means that

$$COV_{\hat{P}_f^{ss}} \cong \sqrt{\sum_{i=1}^{n_{ss}} \frac{1-\hat{p}_k}{N_{ss}\hat{p}_k}} \tag{32}$$

Thus, the theorem is proved.

4.3 The computational capability of RASA

This section demonstrates the computational capability of RASA in estimating the probability of failure relative to regular reliability analysis methods. Moreover, it is shown that for a given constraint on computational capacity, RASA can analyze the probability of rarer events. These features are demonstrated via the following theorems.

Theorem 2. Let $\hat{P}_f^{mcs}$ and $\hat{P}_f^{ss}$ denote the probability of failure by Algorithm 1 and Algorithm 3, respectively. The superior computational performance of Algorithm 3 relative to Algorithm 1 can be demonstrated by the following two cases:
(i) If the numbers of candidate design samples for Algorithm 1 and Algorithm 3 are kept the same (i.e., $N_{mcs} = N_{ss}$ for both algorithms), Algorithm 3 can analyze events with lower probability compared to Algorithm 1. In fact, the logarithmic ratio of the smallest failure probability that Algorithm 3 can analyze relative to Algorithm 1 can be determined as follows,

$$Ł\left(\hat{P}_f^{ss}, \hat{P}_f^{mcs}\right) = \left| log\left(\frac{\hat{P}_f^{ss}}{\hat{P}_f^{mcs}}\right)\right| = \left| log\left(N_{mcs}COV_{thr}^2 + 1\right) + \frac{COV_{thr}^2 N_{ss} p_0}{1-p_0} log p_0 \right| \tag{33}$$

(ii) If the target probability of failure for Algorithm 1 and Algorithm 3 are kept the same, Algorithm 3 has lower computational demand compared to Algorithm 1 in analyzing the same problem. The ratio of the required number of candidate design samples via Subset Simulation over crude MCS can be estimated as,

$$\mathcal{T}(N_{ss}^r, N_{mcs}^r) = \frac{N_{ss}^r}{N_{mcs}^r} = \frac{\hat{P}_f^{mcs}(1-p_0) log \hat{P}_f^{ss}}{p_0\left(1-\hat{P}_f^{mcs}\right) log p_0} \tag{34}$$

where COV_{thr} denotes the threshold of COV for the probability of failure, N_{ss}^r and N_{mcs}^r denote the required number of candidate design samples in Algorithm 1 and Algorithm 3, respectively.

Proof: For case (i) considering Eq. (16), the rarest event that can be analyzed using MCS technique with N_{mcs} samples for a COV less than COV_{thr} is,

$$\hat{P}_f^{mcs} = \frac{1}{N_{mcs}COV_{thr}^2 + 1} \tag{35}$$

Set the intermediate failure probabilities in every subset to be the same $\hat{p}_k = p_0$, thus, for the Subset Simulation-based approach, the number of subsets for intermediate failure thresholds can be determined according to the Eq. (27). Consequently,

$$n_{ss} = \frac{COV_{thr}^2 N_{ss} p_0}{1-p_0} \tag{36}$$

Hence, the rarest event that can be analyzed by the Subset Simulation-based approach has the probability of,

$$\hat{P}_f^{ss} = p_0^{n_{ss}} = p_0^{\frac{COV_{thr}^2 N_{ss} p_0}{1-p_0}} \tag{37}$$

Therefore,

$$Ł\left(\hat{P}_f^{ss}, \hat{P}_f^{mcs}\right) = \left| log\left(\left(p_0^{\frac{COV_{thr}^2 N_{ss} p_0}{1-p_0}}\right) / \left(\frac{1}{N_{mcs} COV_{thr}^2 + 1}\right)\right)\right| = \left| log\left(\left(N_{mcs} COV_{thr}^2 + 1\right) p_0^{\frac{COV_{thr}^2 N_{ss} p_0}{1-p_0}}\right)\right|$$
$$= \left| log\left(N_{mcs} COV_{thr}^2 + 1\right) + \frac{COV_{thr}^2 N_{ss} p_0}{1-p_0} log p_0 \right| \tag{38}$$

For case (ii), the required number of candidate design samples for Algorithm 1 can be estimated according to Eq. (16),

$$N_{mcs}^r = \frac{1 - \hat{P}_f^{mcs}}{\hat{P}_f^{mcs} COV_{thr}^2} \tag{39}$$

From Eq. (22), (23) and (27), $\hat{P}_f^{ss}$ is treated as a constant equal to $\hat{P}_f^{mcs}$, thus N_{ss}^r can be estimated as,

$$N_{ss}^r = \frac{(1-p_0) log \hat{P}_f^{ss}}{COV_{thr}^2 p_0 log p_0} \tag{40}$$

Therefore,

$$\mathcal{T}(N_{ss}^r, N_{mcs}^r) = \frac{N_{ss}^r}{N_{mcs}^r} = \left(\frac{(1-p_0) log \hat{P}_f^{ss}}{COV_{thr}^2 p_0 log p_0}\right) / \left(\frac{1 - \hat{P}_f^{mcs}}{\hat{P}_f^{mcs} COV_{thr}^2}\right) = \frac{(1-p_0) log \hat{P}_f^{ss}}{COV_{thr}^2 p_0 log p_0} \cdot \frac{\hat{P}_f^{mcs} COV_{thr}^2}{1 - \hat{P}_f^{mcs}}$$
$$= \frac{\hat{P}_f^{mcs} (1-p_0) log \hat{P}_f^{mcs}}{p_0 \left(1 - \hat{P}_f^{mcs}\right) log p_0} \tag{41}$$

which proves the theorem.

5. RASA: The Implementation Process

In this section, we present an approach to adaptively identify the intermediate failure thresholds t_i in Eq. (13) through two novel techniques called Conditional Failure Probability Curve and Dynamic Learning Function. First, CFPC is developed to construct a relation between the defined intermediate failure thresholds and failure probability in the selected subsets and characterize the associated confidence intervals. DLF, on the other hand, facilitates effective enrichment of the training set with optimal candidate design samples.

5.1 Conditional Failure Probability Curve

In the procedure of Subset Simulation, the failure thresholds t_i are searched to satisfy the requirement below,

$$P(\Omega_{i+1}|\Omega_i) = P(g(\boldsymbol{x}) \le t_i) = p_0\,, \qquad \boldsymbol{x} \in \Omega_i \quad \text{s.t.} \quad t_i \ge 0 \tag{42}$$

where p_0 is the intermediate failure probability. In RASA, we substitute the true performance function $g(\boldsymbol{x})$ with the Kriging surrogate model $\hat{g}(\boldsymbol{x})$; however, this surrogate model may not ensure precise identification of the true t_i in Eq. (13). Thus, an important task of RASA is to search for an estimate of t_i here denoted as $\hat{t}_i$ such that $\hat{t}_i \cong t_i$. Hereafter, $\hat{t}_i$ and t_i are noted by $\hat{t}$ and t, respectively, for the purpose of notational simplicity. The goal can be achieved by targeted training of the Kriging surrogate model. In this context, the Conditional Failure Probability Curve can be defined as a function of the variable t^*,

$$p_k(t^*) = P(\hat{g}(\boldsymbol{x}) \le t^*), \qquad t^* \ge 0 \tag{43}$$

where $p_k(t^*)$ denotes the intermediate failure probability in the kth subset parametrized by the variable t^*, and $\hat{g}(\boldsymbol{x})$ is the constructed Kriging surrogate model. The corresponding stochastic indicator can be defined as,

$$p_k(t^*) = \frac{1}{N_{MCS}} \sum_{i=1}^{N_{MCS}} \Phi\left(\frac{-\left(\mu_{\hat{g}}(\boldsymbol{x}_i) - t^*\right)}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right), \tag{44}$$

A conceptual sketch of CPFC is shown in Fig. 2. (a) with impacts of training on the estimation of $\hat{t}$ shown in subplots (a) and (b). Note that if $\hat{t}$ changes, the estimated failure probability changes accordingly.

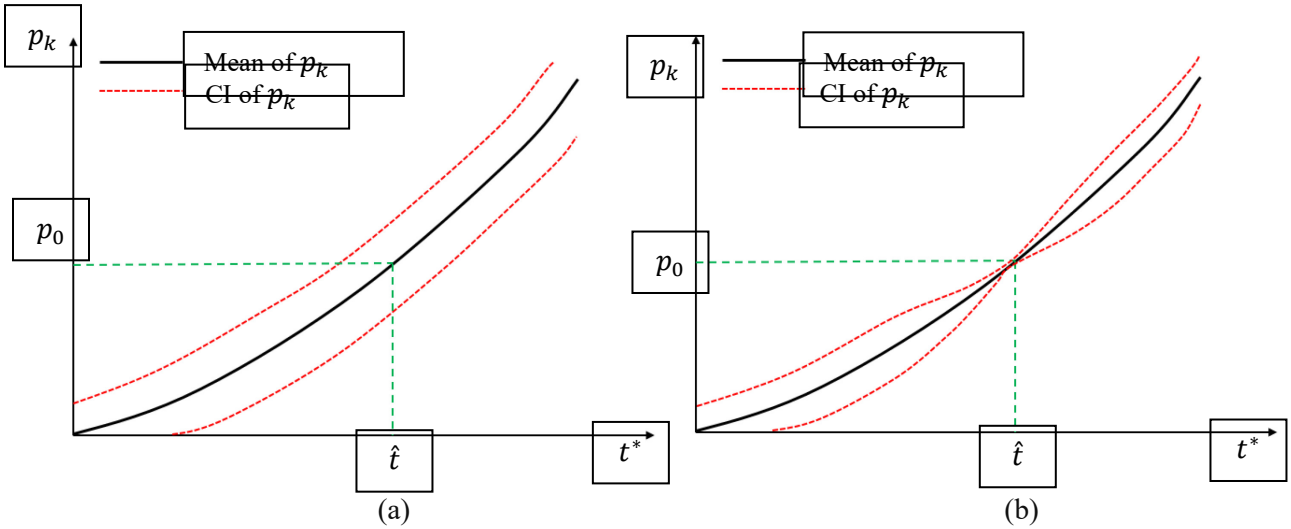

Fig. 2 Conditional Failure Probability Curves with (a) only initial training samples and (b) sufficient training samples in the vicinity of the limit state $\hat{g}(\boldsymbol{x}) - \hat{t} = 0$.

In Fig. 2, the black solid line and red dashed lines denote the mean and confidence interval (CI) of p_k, respectively. Here, $\hat{t}$ is the estimated intermediate failure threshold satisfying the following condition,

$$P(\hat{g}(\boldsymbol{x}) \le \hat{t}) = p_0\,, \qquad \boldsymbol{x} \in \Omega_i \tag{45}$$

However, the identified intermediate failure thresholds $\hat{t}$ are not equal to the true value t. Therefore, the corresponding confidence intervals that reflect the uncertainty of the intermediate failure threshold $\hat{t}$

should be characterized. For an intermediate failure threshold t^*, the probability of failure can be estimated as,

$$\mu_{cfp}\left(p_k(t^*)\right) = E[p_k(t^*)] = \frac{1}{N_{MCS}} \sum_{i=1}^{N_{MCS}} \Phi\left(\frac{-\left(\mu_{\hat{g}}(\boldsymbol{x}_i) - t^*\right)}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right) \tag{46}$$

where $\mu_{cfp}\left(p_k(t^*)\right)$ denotes the mean of the conditional failure probability that is parameterized by t^*. According to the Central Limit Theorem (CLT), t^* follows a normal distribution with the following variance,

$$\sigma^2_{cfp}\left(p_k(t^*)\right) = Var\left(p_k(t^*)\right) = \frac{1}{N^2_{MCS}} \sum_{i=1}^{N_{MCS}} \Phi\left(\frac{\mu_{\hat{g}}(\boldsymbol{x}_i) - t^*}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right) \Phi\left(\frac{-\left(\mu_{\hat{g}}(\boldsymbol{x}_i) - t^*\right)}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right) \tag{47}$$

The probability model of t^* can be represented as,

$$p_k(t^*) \sim N\left(\mu_{cfp}\left(p_k(t^*)\right), \sigma^2_{cfp}\left(p_k(t^*)\right)\right) \tag{48}$$

where $\sigma^2_{cfp}\left(p_k(t^*)\right)$ denotes the variance of the conditional failure probability. Thus, the probability that $p_k(t^*)$ is equal to p_0 can be estimated as,

$$\varphi_{p_k=p_0}\left(p_k(t^*) | \mu_{cfp}\left(p_k(t^*)\right), \sigma^2_{cfp}\left(p_k(t^*)\right)\right) = \frac{1}{\sqrt{2\pi}\sigma_{cfp}\left(p_k(t^*)\right)} exp\left(-\frac{\left(p_0 - \mu_{cfp}\left(p_k(t^*)\right)\right)^2}{2\sigma^2_{cfp}\left(p_k(t^*)\right)}\right) \tag{49}$$

Considering the entire probabilistic space, the PDF of t can be determined as,

$$\Theta_t(t^*) = \frac{\varphi_{p_k=p_0}\left(p_0 | \mu_{cfp}\left(p_k(t^*)\right), \sigma^2_{cfp}\left(p_k(t^*)\right)\right)}{\int_{-\infty}^{\infty} \varphi_{p_k=p_0}\left(\mu_{cfp}\left(p_k(t^*)\right), \sigma^2_{cfp}\left(p_k(t^*)\right)\right) dt^*} \tag{50}$$

where Θ_t denotes the pdf of t. In this article, the intermediate failure threshold $\hat{t}$ is determined using MLE as follows,

$$\hat{t} = \underset{t^* \in \mathbb{R}}{\arg\max}\ \Theta_{\mathrm{t}}(t^*) = \mathrm{E[t]} \tag{51}$$

In Fig. 2, $\hat{t}$ can be identified as the point on the X-axis that corresponds to p_0 on the Y-axis. As the number of training samples in Kriging surrogate model increases, the variance of the random variable t decreases, thus $\hat{t}$ converges to t. The confidence interval of $p_k(t^*)$ can be derived using a method developed by the authors in [33]. Moreover, if the current subset of candidate design samples is denoted as Ω_k, the probability of failure using the Kriging model based on stochastic indicator can be estimated as,

$$p_k(t^*) = \frac{\mathrm{E}[\mathbb{U}(t^*)]}{N_{MCS}} \tag{52}$$

In this approach, for each candidate design sample, $\boldsymbol{x}_i$, $I^{si}_{\hat{g}}(\boldsymbol{x}_i, t^*)$ follows a Bernoulli distribution,

$$I_{\hat{g}}^{si}(\boldsymbol{x}_i, t^*) \sim B\left(\mu_b(\boldsymbol{x}_i, t^*), \sigma_b^2(\boldsymbol{x}_i, t^*)\right), \boldsymbol{x}_i \in \Omega_i, \tag{53}$$

where B is the Bernoulli distribution, $\mu_b(\boldsymbol{x}_i)$ is the Bernoulli mean with $\mu_{\mathrm{b}}(\boldsymbol{x}_i, t^*) = \Phi\left(\frac{-(\mu_{\hat{g}}(\boldsymbol{x}_i)-t^*)}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right)$ and σ_b^2 is the variance of the Bernoulli distribution, where $\sigma_b^2(\boldsymbol{x}_i\, t^*) = \mu_{\mathrm{b}}(\boldsymbol{x}_i, t^*)\left(1-\mu_{\mathrm{b}}(\boldsymbol{x}_i, t^*)\right)$. As $\mathbb{U}(t^*)$ can be derived as the sum of $I_g(\boldsymbol{x}_i, t^*)$, $\boldsymbol{x}_i \in \Omega_i$, it follows that $\mathbb{U}(t^*)$ follows a Poisson Binomial distribution (PBD). As shown in [34], the distribution of $\mathbb{U}(t^*)$ can be denoted as,

$$\mathbb{U}(t^*) \sim PB\left(\mu_{\mathbb{U}}(t^*), \sigma_{\mathbb{U}}^2(t^*), B(t^*)\right) \tag{54}$$

where $\mu_{\mathbb{U}}(t^*)$ and $\sigma_{\mathbb{U}}^2(t^*)$ are the mean value and variance of $\mathbb{U}(t^*)$, respectively. $B(t^*)$ denotes the corresponding Bernoulli distribution of each candidate design samples. According to the probabilistic properties of Poisson Binomial distribution, $\mu_{\mathbb{U}}(t^*) = \sum_{i=1}^{N_{ss}} \mu_{\mathrm{b}}(\boldsymbol{x}_i, t^*)$ and $\sigma_{\mathbb{U}}^2(t^*) = \sum_{i=1}^{N_{ss}} \mu_{\mathrm{b}}(\boldsymbol{x}_i, t^*)\left(1-\mu_{\mathrm{b}}(\boldsymbol{x}_i, t^*)\right)$. Therefore, the CI of $\mathbb{U}(t^*)$ with confidence level α can be derived as,

$$\mathbb{U}(t^*) \in \left(\boldsymbol{\Theta}_{\mathbb{U}}^{-1}\left(\frac{\alpha}{2}, t^*\right), \boldsymbol{\Theta}_{\mathbb{U}}^{-1}\left(1-\frac{\alpha}{2}, t^*\right)\right), \tag{55}$$

where $\boldsymbol{\Theta}_{\mathbb{U}}^{-1}(\cdot)$ is the inverse CDF of PBD with mean $\mu_{\mathbb{U}}(t^*)$ and variance $\sigma_{\mathbb{U}}^2(t^*)$ and α is the confidence level (e.g. $\alpha = 0.05$). Subsequently, according to the Central Limit Theorem, it can be shown that $\mathbb{U}(t^*)$ in distribution converges to a normal distribution,

$$\mathbb{U}(t^*) \sim N\left(\mu_{\mathbb{U}}(t^*), \sigma_{\mathbb{U}}^2(t^*)\right) \tag{56}$$

The CI of $\mathbb{U}(t^*)$ can then be obtained as,

$$\mathbb{U}(t^*) \in [\mu_{\mathbb{U}}(t^*) - \gamma_{ci}\sigma_{\mathbb{U}}(t^*),\ \mu_{\mathbb{U}}(t^*) + \gamma_{ci}\sigma_{\mathbb{U}}(t^*)], \quad \boldsymbol{x}_i \in S \tag{57}$$

where $\gamma_{ci} = 1.96$ for the confidence level $\alpha = 0.05$. The large N_{ss} in Kriging-based reliability analysis problems satisfies the requirement of CLT yielding accurate confidence intervals for $\mathbb{U}(t^*)$. Accordingly, the CI of $p_k(t^*)$ can be derived by integrating Eq. (52) and (57),

$$p_k(t^*) \in \frac{1}{N_{ss}}[\mu_{\mathbb{U}}(t^*) - \gamma_{ci}\sigma_{\mathbb{U}}(t^*),\ \mu_{\mathbb{U}}(t^*) + \gamma_{ci}\sigma_{\mathbb{U}}(t^*)], \quad \boldsymbol{x}_i \in S \tag{58}$$

One should note that the confidence bounds tighten as the number of training samples in the Kriging model increases. However, to accurately estimate the true intermediate failure probability thresholds t_i, the variance near $\hat{t}$ should be significantly reduced. Thus, the training samples are strategically selected in the vicinity of the limit state $\hat{g}(\boldsymbol{x}) - \hat{t} = 0$ using a new learning function in RASA. In the next section, this learning function and an implementation algorithm are introduced to adaptively reduce the uncertainty of $\hat{t}$.

5.2 Dynamic Learning Function

In this section, we introduce Dynamic Learning Function to facilitate strategic selection of best training samples from candidate design samples in the next subset. As the number of training points increases, the

width of the CI of $p_k(\hat{t})$ in Eq. (52) (e.g. the distance between the two dashed red lines in Fig. 2) decreases. Thus, identifying t is equivalent to reducing the uncertainty in the vicinity of the limit state $\hat{g}(\boldsymbol{x}) - \hat{t} = 0$.

Theorem 3. Considering $\rho\left(\hat{g}(\boldsymbol{x}_i), \hat{g}(\boldsymbol{x}_j)\right) = 0$, the optimal active learning strategy for identifying the true intermediate failure threshold t can be expressed as follows,

$$\boldsymbol{x}_{tr}^{*} = \arg\max_{\boldsymbol{x} \in \Omega_i} \left[\Phi\left(\frac{-\left(\mu_{\hat{g}}(\boldsymbol{x}_i) - \hat{t}\right)}{\sigma_{\hat{g}}(\boldsymbol{x}_i)} \right) \Phi\left(\frac{\mu_{\hat{g}}(\boldsymbol{x}_i) - \hat{t}}{\sigma_{\hat{g}}(\boldsymbol{x}_i)} \right) \right] \tag{59}$$

where $\rho\left(\hat{g}(\boldsymbol{x}_i), \hat{g}(\boldsymbol{x}_j)\right)$ denotes the correlation between the response of $\hat{g}(\boldsymbol{x}_i)$ and $\hat{g}(\boldsymbol{x}_j)$, $\boldsymbol{x}_{tr}^{*}$ denotes the new training samples, and $\hat{t}$ is the identified intermediate failure threshold in the last iteration.

Proof: After the estimated intermediate failure threshold $\hat{t}$ is identified in the current iteration, the variance of the intermediate failure threshold can be estimated according to the Central Limit Theorem as,

$$\begin{aligned} \mathrm{Var}\left[\tilde{P}_f^{si}\right] &= \mathrm{Var}\left[\frac{1}{N_{ss}} I_{\hat{g}}^{si}\right] = \frac{1}{N_{ss}^2}\mathrm{Var}\left[I_{\hat{g}}^{si}\right] = \frac{1}{N_{ss}^2}\mathrm{Var}\left[\sum_{i=1}^{N_{ss}} I_{\hat{g}}^{si}(\boldsymbol{x}_i)\right] = \frac{1}{N_{ss}^2}\sum_{i=1}^{N_{ss}}\mathrm{Var}\left[I_{\hat{g}}^{si}(\boldsymbol{x}_i)\right] \\ &= \frac{\sum_{i=1}^{N_{ss}} \Phi\left(\frac{-\mu_{\hat{g}}(\boldsymbol{x}_i - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right)\left(1 - \Phi\left(\frac{-\mu_{\hat{g}}(\boldsymbol{x}_i - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right)\right)}{N_{ss}^2} = \frac{\sum_{i=1}^{N_{ss}} \Phi\left(\frac{-\mu_{\hat{g}}(\boldsymbol{x}_i - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right)\Phi\left(\frac{\mu_{\hat{g}}(\boldsymbol{x}_i) - \hat{t}}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right)}{N_{ss}^2} \end{aligned} \tag{60}$$

Note the fact that the following equation always holds true as $\boldsymbol{x}_i$ is selected as the next training point,

$$\lim_{\sigma_{\hat{g}}(\boldsymbol{x}_i)\to 0}\left(\Phi\left(\frac{-\mu_{\hat{g}}(\boldsymbol{x}_i - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right)\left(1 - \Phi\left(\frac{-\mu_{\hat{g}}(\boldsymbol{x}_i - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right)\right)\right) = 0 \tag{61}$$

Let $\tilde{P}_f^{si\prime}$ denote the stochastic estimator of failure probability after new training samples are added. If $\rho\left(\hat{g}(\boldsymbol{x}_i), \hat{g}(\boldsymbol{x}_j)\right) = 0$, the optimal learning strategy can be represented as,

$$\begin{aligned} \boldsymbol{x}_{tr}^{*} &= \arg\max_{\boldsymbol{x}_i \in \mathrm{S}} \left[\mathrm{Var}\left[\tilde{P}_f^{si}\right] - \mathrm{Var}\left[\tilde{P}_f^{si\prime}\right]\right] \\ &= \arg\max_{\boldsymbol{x}_i \in \mathrm{S}} \frac{1}{N_{ss}^2}\left[\sum_{k=1}^{N_{ss}} \Phi\left(\frac{-\mu_{\hat{g}}(\boldsymbol{x}_k - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_k)}\right)\Phi\left(\frac{\mu_{\hat{g}}(\boldsymbol{x}_k - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_k)}\right)\right. \\ &\quad \left. - \left(\sum_{k=1}^{N_{ss}} \Phi\left(\frac{-\mu_{\hat{g}}(\boldsymbol{x}_k - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_k)}\right)\Phi\left(\frac{\mu_{\hat{g}}(\boldsymbol{x}_k - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_k)}\right) - \Phi\left(\frac{-\mu_{\hat{g}}(\boldsymbol{x}_i - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right)\Phi\left(\frac{\mu_{\hat{g}}(\boldsymbol{x}_i - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right)\right)\right] \\ &= \arg\max_{\boldsymbol{x}_i \in \mathrm{S}} \Phi\left(\frac{-\mu_{\hat{g}}(\boldsymbol{x}_i - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right)\Phi\left(\frac{\mu_{\hat{g}}(\boldsymbol{x}_i - \hat{t})}{\sigma_{\hat{g}}(\boldsymbol{x}_i)}\right), i = 1,2,\ldots N_{ss} \end{aligned} \tag{62}$$

This result shows that the learning strategy without considering Kriging correlation tends to select points with the highest contribution to the variance of $\tilde{P}_f^{si}$. The method to adaptively estimate t is summarized in Algorithm 4. Note that $\hat{t}$ changes in each iteration. The stopping criterion for dynamic learning can be set based on the variance of $p_k(\hat{t})$ as follows,

$$\lim_{\Gamma=\frac{\sigma_{\mathbb{U}}}{\mu_{\mathbb{U}}}\leq\Gamma_{thr}\to 0} (\hat{t} \cong t\,) \tag{63}$$

where Γ is the stopping index and Γ_{thr} is the corresponding threshold. Initially, when the number of training points is insufficient, $\hat{t}$ will not be accurate to satisfy $\hat{t} \cong t$. However, $\hat{t}$ will asymptotically converge to t as new training points accumulate near the limit state $\hat{g}(\boldsymbol{x}) - \hat{t} = 0$. Therefore, two adaptive processes exist in the proposed method. First, the uncertainty or the variance at $\hat{t}$ (i.e., $\sigma_b^2(\boldsymbol{x}_i, \hat{t})$) is reduced adaptively by adding training samples using the proposed Dynamic Learning Function. Second, the estimated intermediate failure threshold $\hat{t}$ adaptively converges to the true one (i.e. t_i in Eq. (13)) via the addition of the new training points.

Algorithm 4. Searching for t_i using DLF and CFPC

1. Prepare the initial training points $\boldsymbol{x}_{in}$. Keep $\boldsymbol{x}_{in}$ and $g(\boldsymbol{x}_{in})$ the same for all simulations. Also, generate candidate design samples S_k from the subset Ω_k, if $k \geq 2$.
2. Construct the Kriging model $\hat{g}_k(\cdot)$ based on the current set of training points.
3. Build the CFPC according to Eq. (44).
4. Search for $\hat{t}$ according to Eq. (51).
5. Search for the next training point $\boldsymbol{x}_{tr}^*$ using Dynamic Learning Function according to Eq. (59) and update the set of training points.
6. Check if the stopping criterion is satisfied according to Eq. (63):
   (a) If satisfied, go to step 7.
   (b) If not satisfied, estimate the response for $\boldsymbol{x}_{tr}^*$ and return to step 2.
7. Output $\hat{t}$.

In the estimation of failure probability using RASA, two stages are distinguished: (a) estimation of the next failure threshold when $\hat{t} > 0$ and (b) estimation of the probability of failure in the last subset when $\hat{t} < 0$. The analyses for the first case can be conducted according to Algorithm 4. However, the failure probability in the last subset, $P\left(\Omega_{n_{ss}} \middle| \Omega_{n_{ss}-1}\right)$, can be estimated as,

$$P\left(\Omega_{n_{ss}} \middle| \Omega_{n_{ss}-1}\right) = P(g(\boldsymbol{x}) \leq 0)\,, \qquad \boldsymbol{x} \in \Omega_{n_{ss}-1} \tag{64}$$

where $\Omega_{n_{ss}}$ is the final subset. In this case, the failure probability can be estimated following Algorithm 3. It should be noted that training samples generated for the previous subsets are not close to the new limit state for the current subset, except for the case in the last subset. Therefore, their contribution to enhancing the accuracy of the Kriging model for the next subset is insignificant. Moreover, considering these points in the construction of the Kriging surrogate model for the next subsets is computationally inefficient and can often lead to high computational demand. Due to these reasons, training samples generated for the previous subsets are not used in searching for the next intermediate threshold and only the initial training samples, $\boldsymbol{x}_{in}$, are used as the starting set for every subset. The method for estimating the conditional failure probability in the last subset follows the same principle of adaptive Kriging-based reliability analysis methods such as those in [4], [17]. This process is summarized in Algorithm 5.

Algorithm 5. Estimating the conditional failure probability for the last subset

1. Use training points generated for the last subset in Algorithm 4 and generate candidate design samples $S_{n_{ss}-1}$ from the subset $\Omega_{n_{ss}-1}$.
2. Construct the Kriging model $\hat{g}_{n_{ss}}(\cdot)$ based on the current set of training points $\boldsymbol{x}_{tr}$.
3. Estimate the mean $\mu_{\hat{g}_{n_{ss}}}(\boldsymbol{x})$, standard deviation $\sigma_{\hat{g}_{n_{ss}}}(\boldsymbol{x})$, and $P\left(\Omega_{n_{ss}}\middle|\Omega_{n_{ss}-1}\right)$ for $S_{n_{ss}-1}$ with $\hat{g}_{n_{ss}}(\cdot)$.
4. Search for the next best training point $\boldsymbol{x}_{tr}^{*}$ using the learning function in Eq. (59), where $\hat{t}=0$. Update the set of training samples.
5. Check if the stopping criterion is satisfied:
   (a) If satisfied, go to step 7.
   (b) If not satisfied, estimate the response $g(\boldsymbol{x}_{tr}^{*})$ for $\boldsymbol{x}_{tr}^{*}$ and go back to step 3.
6. Output $P\left(\Omega_{n_{ss}}\middle|\Omega_{n_{ss}-1}\right)$.

6. Numerical Investigations

In this section, three examples are investigated to evaluate the computational capabilities of RASA in solving computationally demanding reliability problems. The number of random variables for these three examples increases from 9 to 30 and to 110 with different levels of nonlinearity in the limit state. By controlling N_{ss} and p_0, RASA is shown to effectively leverage the available computational capacity to arrive at reliable and unbiased estimates of failure probabilities.

6.1 Turbine example with small probability of failure

The first example is a non-linear problem with the small probability of failure of 1.5×10^{-4} [35], [36]. As shown in Fig. 3, the cantilever tube is subject to three types of external forces including shear forces F_1 and F_2, axial force F_x and torsional moment $\mathrm{F_t}$. The probability distribution properties of the involved nine random inputs are summarized in Table 1. The performance function is defined as,

$$g(x)=\sigma_{cap}-\sigma_{max} \tag{65}$$

where σ_{cap} is the strength capacity and σ_{max} is the maximum von Mises stress calculated as,

$$\sigma_{max}=\sqrt{\sigma_x^2+3\tau_{zx}^2} \tag{66}$$

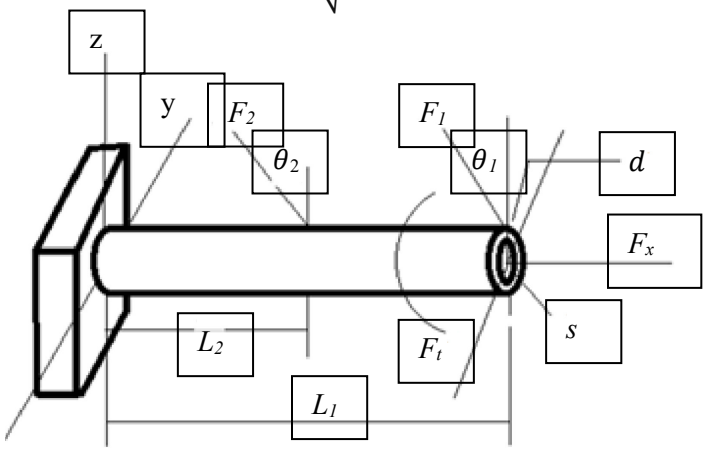

Fig. **3** Cantilever tube and involved variables in Example 1.

$$\sigma_x = \frac{F_x + F_1 sin\theta_1 + F_2 sin\theta_2}{A} + \frac{Mc}{I} \tag{67}$$

where $\theta_1 = 5°$, $\theta_2 = 10°$, and A is the tube area. Moreover, τ_{zx} is the torsional stress, M is the bending moment, c is the radius and I is the moment of inertia. These variables are calculated as,

$$M = F_1 L_1 cos\theta_1 + F_2 L_2 cos\theta_2 \tag{68}$$
$$A = \frac{\pi}{4}[d^2 - (d - 2s)^2] \tag{69}$$
$$c = \frac{d}{2} \tag{70}$$
$$I = \frac{\pi}{64}[d^4 - (d - 2s)^4] \tag{71}$$
$$\tau_{zx} = \frac{F_t d}{2J} \tag{72}$$
$$J = 2I \tag{73}$$

Table 1. Random variables used in Example 1.

Random variable	Distribution type	Parameter 1	Parameter 2
s	Normal	5 mm (μ)	0.1 mm (σ)
d	Normal	42 mm (μ)	0.5 mm (σ)
F_1	Normal	3.0 kN (μ)	0.3 kN (σ)
F_2	Normal	3.0 kN (μ)	0.3 kN (σ)
F_t	Normal	90.0 Nm (μ)	9.0 Nm (σ)
σ_{cap}	Normal	220.0 MPa (μ)	22.0 MPa (σ)
L_1	Uniform	119.75 mm (lb)	120.25 mm (ub)
L_2	Uniform	59.75 mm (lb)	60.25 mm (ub)
F_x	Gumbel	12.0 kN (μ)	1.2 kN (σ)

*Note: μ and σ represent the mean and standard deviation, and lb and ub the lower and upper bounds of random variables, respectively.

In the implementation of Algorithm 3, $N_{ss} = 5000$, $p_0 = 0.1$, $COV_{thr} = 0.05$ and the stopping threshold $\Gamma_{thr} = 0.005$ in Eq. (63) are initially set. Computational details regarding the convergence history of $\hat{t}_i$, effects of N_{ss} and p_0 as well as the computational performance of this method compared to other approaches are elaborated in subsections 6.1.1 to 6.1.4.

6.1.1 Convergence history for intermediate failure thresholds

To check whether the intermediate thresholds t_is are accurately identified, the true intermediate failure thresholds are estimated using the true performance function. The convergence histories for the identification of $\hat{t}_i$ are presented in Fig. 4. Moreover, the identified intermediate thresholds t_i and $\hat{t}_i$, and the number of new calls to the performance function, N_{call}, are summarized in Table 2. To investigate the evolution of the CFPC, Fig. 5 shows the shrinkage of the CIs of $\hat{t}_1$ as the number of calls to the performance function increases.

Table 2. Identified intermediate failure thresholds using g(x)/ĝ(x)for Example 1.

Performance function	t_1 or $\hat{t}_1(N_{call})$	t_2 or $\hat{t}_2(N_{call})$	t_3 or $\hat{t}_3(N_{call})$	t_3 or $\hat{t}_3(N_{call})$
$g(x)$	54.900	29.87	11.33	−3.80
$\hat{g}(x)$	55.00(50+ 63)	29.80(58)	11.40(67)	−3.77(80)

As shown in Table 2, the intermediate thresholds estimated using the proposed method are very close to those estimated via Subset Simulation. Specifically, the intermediate thresholds based on 5000 samples are identified as $\hat{t}_1 = 54.90$, $\hat{t}_2 = 29.80$, $\hat{t}_3 = 11.40$ and $\hat{t}_4 = -3.77$, while the true intermediate thresholds are $t_1 = 55.00$, $t_2 = 29.87$, $t_3 = 11.33$ and $t_4 = -3.80$, respectively. It should be noted that the accuracy of $\hat{t}_i$ can be improved by reducing the stopping threshold Γ_{thr} in Eq. (63). Based on results presented in Fig. 4 and 5, the following observations can be drawn:

- RASA can accurately identify the intermediate failure thresholds t_i. The estimated limit states $\hat{g}(\boldsymbol{x}) = \hat{t}_i, i = 1, \ldots, 4$ deviate from the corresponding true limit states $g(\boldsymbol{x}) = t_i, i = 1, \ldots 4$ for the case where only the initial training set is used to construct the Kriging model. However, as the number of training samples increases adaptively, the accuracy of the estimated limit states increases. This feature can be observed also from the convergence history of $\hat{t}_i$ in Fig. 4, where the estimated thresholds converge rapidly to the true thresholds. Deviations of $\hat{t}_i$ from t_i at the early stages (i.e., when $\mathrm{N_{call}}$ is small) are primarily due to the yet inadequate accuracy of the Conditional Failure Probability Curve that generates large variance for the intermediate failure probability p_o, as shown in Fig. 5 (a) and (b).

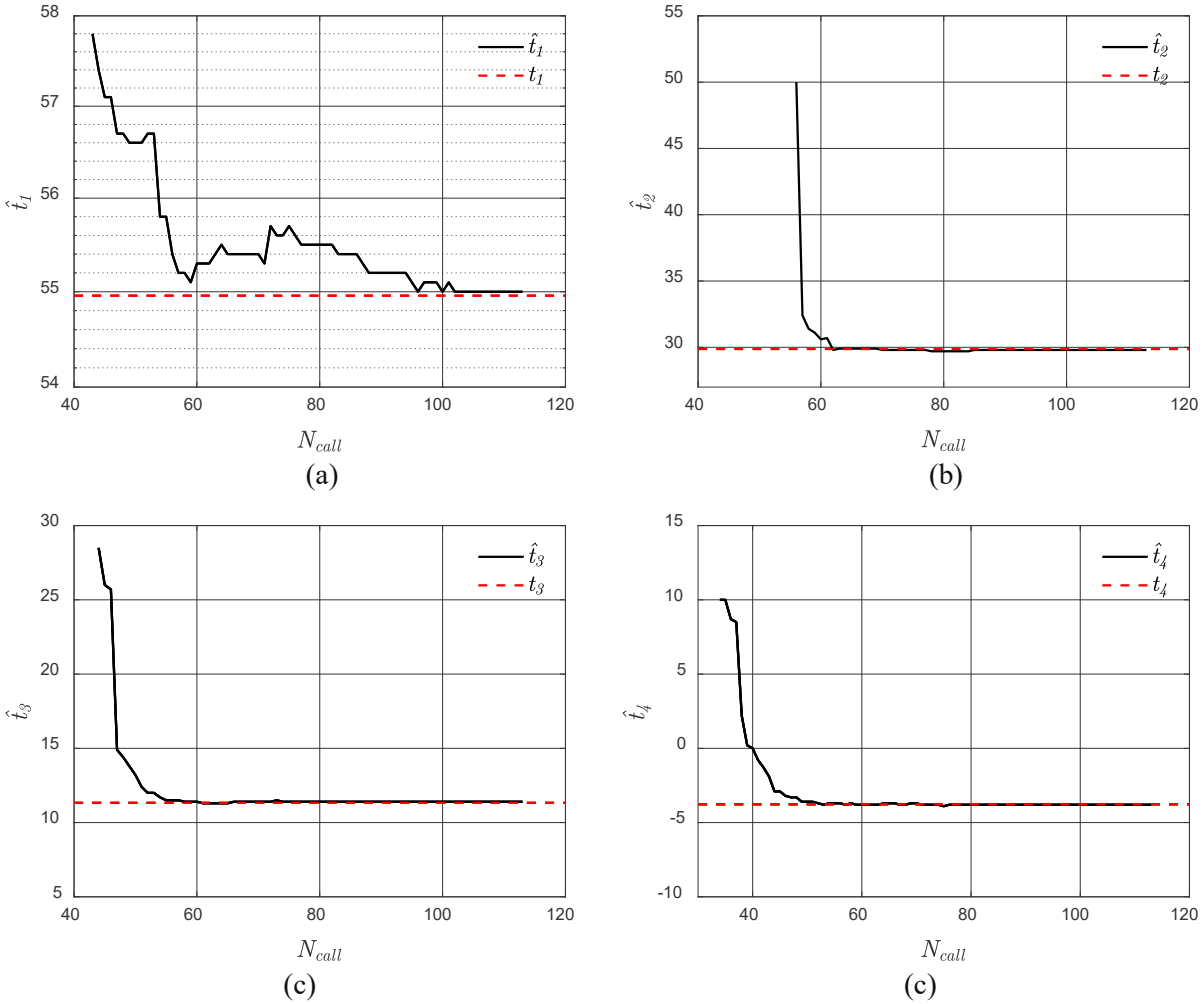

Fig. 4. Convergence history for the estimated intermediate failure threshold for (a) $\hat{t}_1$, (b) $\hat{t}_2$, (c) $\hat{t}_3$, and (d) $\hat{t}_4$.

- Uncertainty of p_o is significantly reduced as the number of strategically selected training points via the proposed Dynamic Learning Function increases. As shown in Fig. 5, the confidence interval for the estimated intermediate failure probability $\hat{p}_o$ tightens as N_{call} increases. In this example, $\hat{t}_1$ is

identified as 56.65, 55.64, 55.25 and 55.00 for $N_{call} = 50$, 70, 90 and 110, respectively. This quantity finally convergences to 55.00, while $t_1 = 54.90$. After the forth intermediate failure threshold $\hat{t}_4$ is found to be smaller than zero, the conditional failure probability $P(\Omega_m|\Omega_{m-1})$ in the last subset is accurately estimated using Algorithm 3.

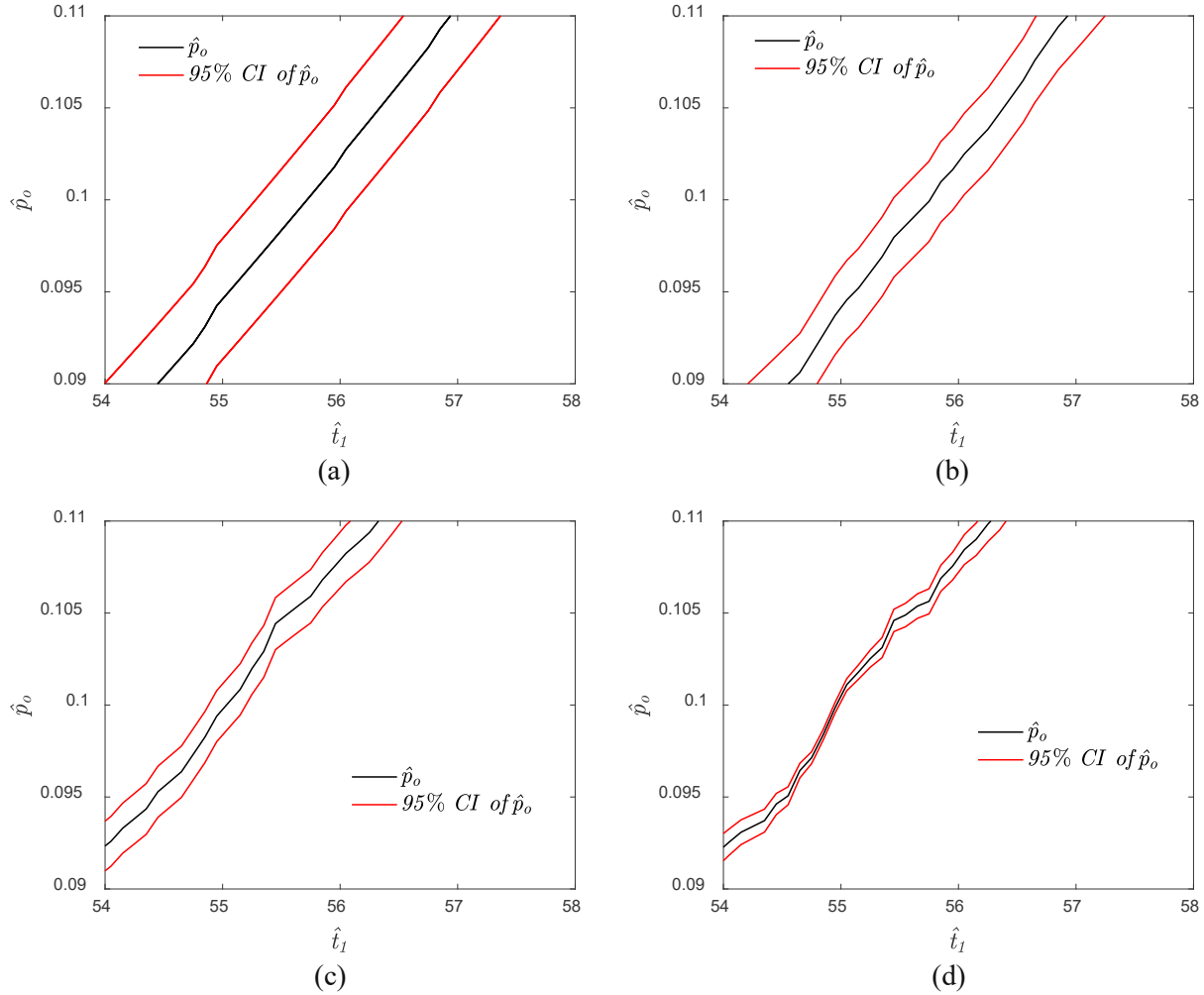

Fig. 5. Illustration of Conditional Failure Probability Curve for identifying t_1 with (a) $N_{call} = 30$, (b) $N_{call} = 50$, (c) $N_{call} = 70$, and (d) $N_{call} = 110$.

6.1.2 Adjustment of N_{ss} and p_0 in RASA

Following the steps presented in Algorithm 3, the number of candidate design samples in the subsets needs to be increased. According to Eq. (16) and (19), the minimum required number of candidate design samples for N_{mcs} and N_{ss} are estimated as 2.30×10^6 and 1.27×10^4, respectively. Matrix operations among other computations that are involved in Kriging-based MCS techniques such as AK-MCS [37], EGRA [38], REAK [39] limit the maximum number of candidate samples that can be considered using regular PCs (e.g., Intel(R) Core(TM) i5-6300HQ CPU, RAM 16.00GB). To effectively utilize the available computational capacity, we have set $N_{ss}^{max} = 6 \times 10^4$. For Subset Simulation, the new number of candidate design samples is set as 1.3×10^4, which means that 8000 extra candidate design samples need to be added for the previous subsets. Due to this update in the candidate design samples, the intermediate failure thresholds also need to be updated. Results for the updated intermediate failure thresholds and the final estimates of the failure probability are summarized in Table 3.

Table 3. Intermediate failure thresholds using updated candidate design samples for Example 1.

N_{ss}	$\hat{t}_1$	$\hat{t}_2$	$\hat{t}_3$	$\hat{t}_4$	$\hat{P}_f^{ss}$
0.5×10^4	55.00	29.80	11.40	-3.77	1.744×10^{-4}
1.3×10^4	55.34	28.26	-11.15	-3.66	1.792×10^{-4}

Results in Table 3 are based on the requirement that $COV_{thr} = 0.05$, which may not be sufficiently small to yield a reliable estimate of failure probability. For this purpose, in some cases, COV_{thr} is set as small as 0.015 to obtain a more stable estimate of failure probabilities. For this case according to Eq. (16) and (19), the minimum required number of candidate design samples for N_{mcs} and N_{ss} are estimated as 2.55×10^7 and 1.41×10^5, respectively. However, $N_{ss}^{max} = 6 \times 10^4$ is applied to this problem, which is smaller than the required value of 1.41×10^5. Therefore, the value of p_0 is adjusted to satisfy the required $COV_{\hat{P}_f^{ss}}$. According to Eq. (24), the minimum value of $COV_{\hat{P}_f^{ss}}$ is estimated as 0.0136, which means that a p_0 exists such that the estimated $COV_{\hat{P}_f^{ss}}$ is equal to 0.015. Moreover, Fig. 6 illustrates the relation between $COV_{\hat{P}_f^{ss}}$ and p_0 based on Eq. (25). According to Fig. 6, $p_0 = 0.43$ can satisfy the requirement that $COV_{\hat{P}_f^{ss}} \leq COV_{thr} = 0.015$ with $N_{ss} = N_{ss}^{max}$.

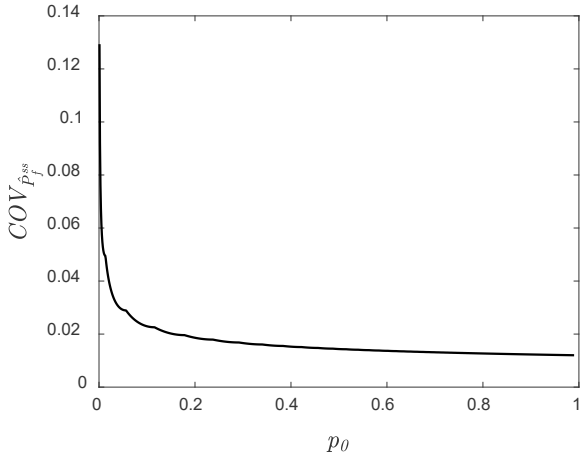

Fig. 6 $COV_{\hat{P}_f^{ss}}$ vs p_0 in Example 1.

Table 4. Intermediate failure thresholds using the updated p_0 for Example 1.

p_0	$\hat{t}_1$	$\hat{t}_2$	$\hat{t}_3$	$\hat{t}_4$	$\hat{P}_f^{ss}$
0.1	55.34	28.26	-11.15	0.452	1.792×10^{-4}
0.43	81.62	64.31	52.06	42.12	1.828×10^{-4}

Hence, the number of subsets is increased from 4 to 11, which means that the corresponding intermediate failure thresholds are also updated. The comparison of the first four intermediate failure thresholds for $p_0 = 0.1$ and 0.43 are summarized in Table 4. According to Theorem 2, $\mathcal{T}(N_{ss}^r, N_{mcs}^r)$ is estimated as 0.0236 for this set, which means that Algorithm 3 needs a very small portion of the number of candidate design samples needed for Algorithm 1. This capability facilitates analysis of problems that would otherwise require high computational resources.

6.1.3 Computational performance of the proposed method

The computational performance of some of the state-of-the-art methods for Example 1 is summarized in Table 5. For this problem with 9-random inputs and the level of failure probability at 1.801×10^{-4} based

on the pure MCS simulation, algorithms such as EGRA, AK-MCS, ISKRA, and REAK would require 10^7 samples to achieve $COV_{thr} = 0.015$. This very large number is not computationally feasible to analyze using most regular PCs. The results here are compared with two other Subset Simulation-based method including AK-SS [26] and the method proposed by Ling et al. [40], which hereafter is referred to as AK-SS (Ling et al.).

Table 5. The comparison of computational performances among EGRA, AK-MCS, ISKRA, REAK, AK-SS, AK-SS (Ling et al.) and RASA.

Methodologies	$\hat{P}_f$	$\overline{N}_{call}$	ECC	Accounts for COV	Is the estimate reliable	Unbiased estimator
MCS	1.801×10^{-4}	1×10^7	-	-	-	-
EGRA	-	-	Yes	No	Depends on N_{mcs}	Yes
AK-MCS	-	-	Yes	Yes	Depends on N_{mcs}	Yes
ISKRA	-	-	Yes	No	Depends on N_{mcs}	No
REAK	-	-	Yes	Yes	Depends on N_{mcs}	No
AK-SS ($p_o = 0.1$& $N_{ss} = 10000$)	1.477×10^{-4}	>500	Depends on p_o and N_{ss}	Yes	Depends on the definition of proposal sampling for MCMC	No
AK-SS(Ling) ($p_o = 0.1$& $N_{ss} = 13000$)	1.639×10^{-4}	412	Depends on p_o and N_{ss}	No	Depends on the definition of proposal sampling for MCMC	No
RASA	1.828×10^{-4}	318	No	Yes	Depends on N_{ss}	Yes

*ECC means the specified algorithm will exceed the computational capacity of the computer built with Intel(R) Core(TM) i5-6300HQ CPU, RAM 16.00GB.

Compared to other Kriging-based Subset Simulation techniques e.g., [26], [40], three major advantages of RASA stand out. First, failure probability analysis using RASA is not hampered by the number of random inputs and the probability of the rare event. By adaptively adjusting the value of N_{ss} and p_0, RASA takes advantage of its algorithmic capability to leverage available computational resources to solve complex problems. This feature is unique to RASA and is not present in other methods in the literature. For example, the method proposed by Ling et al. [40] neglects the check needed for $COV_{\hat{P}_f^{ss}}$. Hence, it can easily lead to high computational demand or a large deviation of the estimated failure probability from the true value if N_{ss} is set inappropriately. Second, due to the quick and effortless analysis of performance functions using Kriging surrogate models, the conditional failure probabilities can be quickly estimated through the surrogate-based MCS and not the MCMC technique. This approach can be used to derive unbiased estimates for conditional failure probabilities and improve the accuracy of the results. For example, the probability of failure estimated through the AK-SS and the method by Ling et al. [40] are 1.4×10^{-4} and 1.639×10^{-4}, which are not close to the one estimated through the pure MCS. This is in part due to the effect of using MCMC to generate samples in the subsets, which requires sophisticated

definitions of the proposal (jumping) sampling function. Finally, RASA can not only provide the convergence history of the estimated intermediate failure thresholds, but also their corresponding confidence intervals.

6.2 A truss structure with 30 input variables

In the second example, the performance of RASA is investigated for structural reliability analysis of a 23-bar truss with 30 input random variables. These variables represent uncertainties in member sizes and properties as well as external loads. The configuration of the structure is shown in Fig. 7. In this figure, 1 to 12 denote the 12 types of bars in the truss. Moreover, the horizontal and diagonal bars are 4 m and $2\sqrt{2}$ m long, respectively. The performance function of this example is defined as,

$$g(x) = 0.15 - |\Delta|, \tag{35}$$

where Δ is the vertical displacement at the midpoint of the truss. The structure is subject to six vertical point loads, V_1 to V_6. These loads follow Gumbel distributions. $A_1 - A_{12}$ and $E_1 - E_{12}$ are random variables representing the cross-section area and Young's modulus of bars, respectively, as shown in Fig. 7. Probability distributions of these 30 random variables are presented in Table 6. For this example, 50 initial training samples are generated, the number of candidate design samples for each subset is set as 5000, $COV_{thr} = 0.015$ and Γ_{thr} is set as 5×10^{-3}. Analysis results for this example are summarized in Table 7. Moreover, the convergence history of estimated intermediate thresholds are shown in Fig. 8.

6.2.1 Convergence history of intermediate failure thresholds

According to the results summarized in Table 7, RASA is very efficient in identifying the intermediate thresholds t_i. In this example, $\hat{t}_1$, $\hat{t}_2$ and $\hat{t}_3$ are identified as 0.01681, 0.0052 and -0.0044, respectively, which are very precise considering the true values $t_1 = 0.01679$, $t_2 = 0.00528$ and $t_3 = -0.00446$. Initially, $\hat{P}_f^{ss}$ is roughly estimated as 2.81×10^{-3} as shown in Table 7. According to the convergence history, the number of calls to the performance function increases significantly for this example compared to the problem with small number of random variables. Specifically, N_{call} is equal to 238, 349 and 327 for identifying the three intermediate failure thresholds.

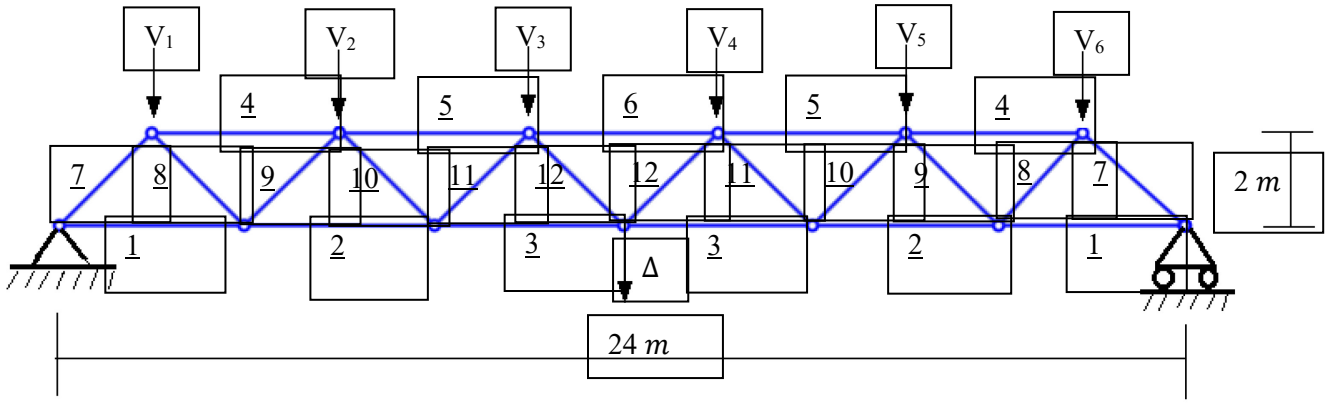

Fig. 7 The truss with 30 random variables in Example 2.

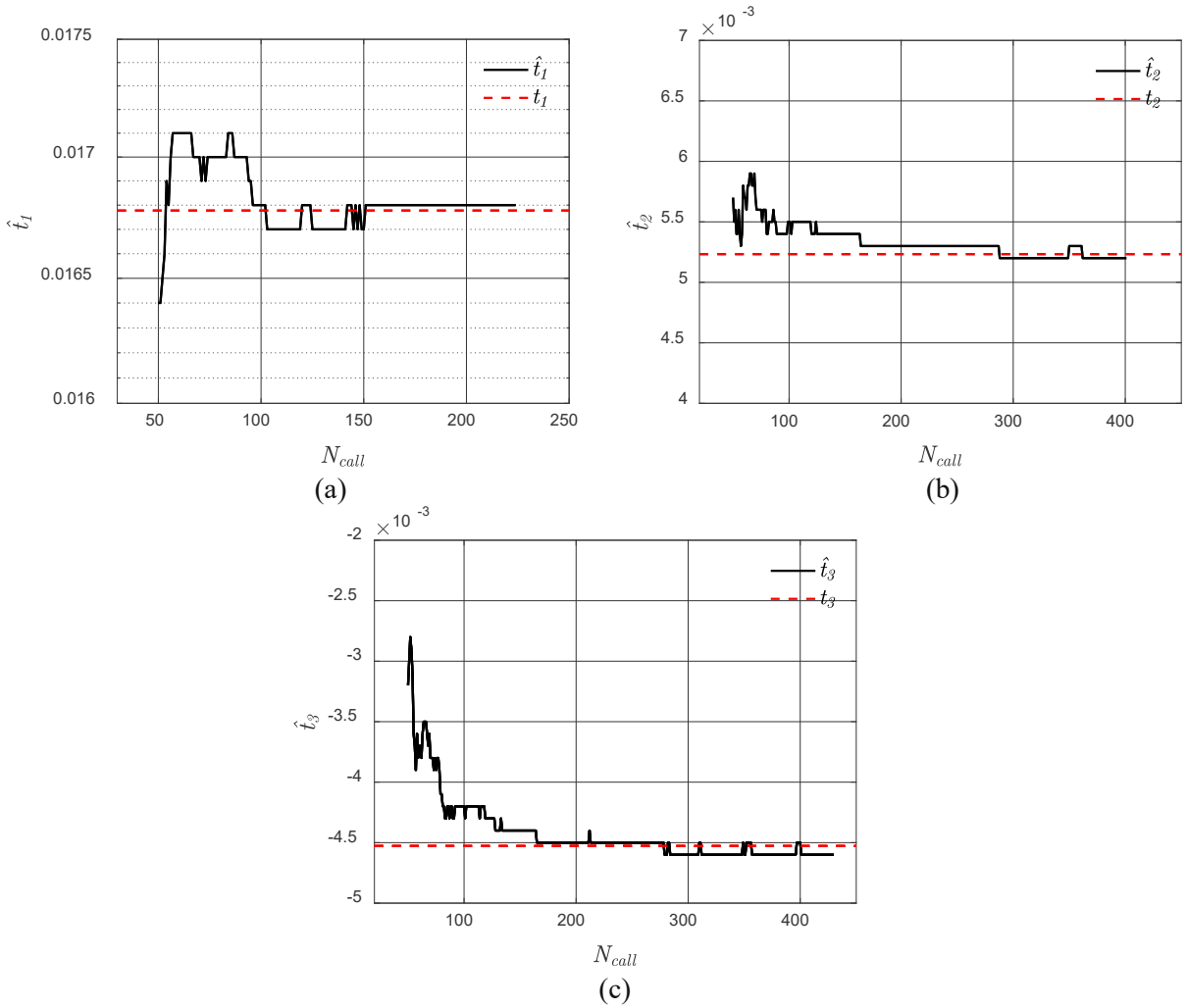

Fig. 8. Convergence history of estimated intermediate failure threshold for (a) $\hat{t}_1$, (b) $\hat{t}_2$, and (c) $\hat{t}_3$ for Example 2.

Table 6. Random variables in Example 2.

Random variable	Distribution	Mean	COV
$V_1 - V_6$	Gumbel	6.5×10^4(N)	0.1
$A_1 - A_3$	Lognormal	2×10^{-3} (m^2)	0.1
$A_4 - A_6$	Lognormal	1.5×10^{-3} (m^2)	0.1
$A_7 - A_9$	Lognormal	1.2×10^{-3} (m^2)	0.1
$A_{10} - A_{12}$	Lognormal	1.0×10^{-3} (m^2)	0.1
$E_1 - E_3$	Lognormal	2.1×10^{11} (N/m)	0.1
$E_4 - E_6$	Lognormal	2.0×10^{11} (N/m)	0.1
$E_7 - E_9$	Lognormal	1.8×10^{11} (N/m)	0.1
$E_{10} - E_{12}$	Lognormal	1.6×10^{11} (N/m)	0.1

*COV denotes the coefficient of variation.

Table 7. Identified intermediate failure thresholds using $g(x)/\hat{g}(x)$for Example 2.

Performance function	t_1 or $\hat{t}_1$ (N_{call})	t_2 or $\hat{t}_2$ (N_{call})	t_3 or $\hat{t}_3$ (N_{call})	$\hat{P}_f^{ss}$
$g(x)$	0.01695	0.0058	-0.0037	2.81×10^{-3}
$\hat{g}(x)$	0.0169 $(50 + 188)$	0.0058 $(48 + 351)$	-0.0037 $(80 + 247)$	2.81×10^{-3}

6.2.2 Adjustment of N_{ss} and p_0 in RASA

According to Eq. (16) and (19), the minimum required number of candidate design samples for N_{mcs} and N_{ss} are estimated as 2.55×10^7 and 1.58×10^6 for this case, respectively. Following the same steps for the previous example, the number of candidate design samples in the subsets needs to be increased. According to Eq. (16) and (19), the minimum required number of candidate design samples for N_{mcs} and N_{ss} are estimated as 2.30×10^6 and 1.27×10^4, respectively. Based on a few experiments, it is realized that the maximum number of candidate design samples that can be handled by the considered PC is about $N_{ss}^{max} = 5 \times 10^4$ for this problem. The new N_{ss} cannot guarantee $COV_{\hat{P}_f^{ss}} \leq COV_{thr}$, therefore, the value of p_0 should be adjusted to leverage the available computational capacity such that this condition can be satisfied. According to Eq. (24), the minimum value of $COV_{\hat{P}_f^{ss}}$ that can be reached is 0.0109. This result subsequently indicates that a p_0 exists that yields the estimated $COV_{\hat{P}_f^{ss}}$ of 0.015.

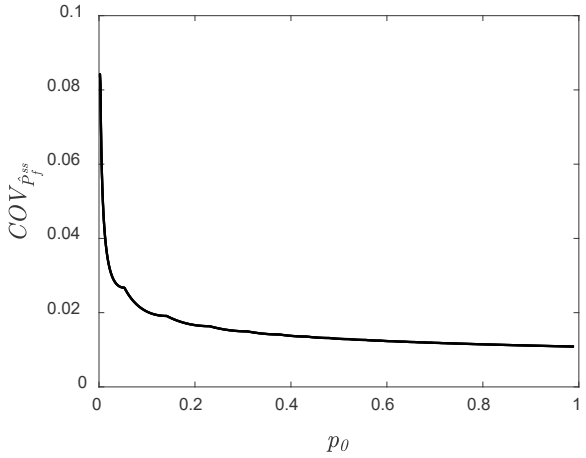

Fig. 9 $COV_{\hat{P}_f^{ss}}$ vs p_0 in Example 2.

Fig. 9 illustrates the relation between $COV_{\hat{P}_f^{ss}}$ and p_0 based on Eq. (25). According to Fig. 9, $p_0 = 0.3$ can satisfy the requirement that $COV_{\hat{P}_f^{ss}} \leq COV_{thr} = 0.015$ with $N_{ss} = N_{ss}^{max}$. Hence, the number of subsets is increased from 3 to 5, which means the corresponding intermediate failure thresholds are also updated. The comparison of the first three intermediate failure thresholds for $p_0 = 0.1$ and 0.3 are summarized in Table 8.

Table 8. Intermediate failure thresholds using updated candidate design samples for Example 2.

N_{ss}	p_0	$\hat{t}_1$	$\hat{t}_2$	$\hat{t}_3$	$\hat{P}_f^{ss}$
0.5×10^4	0.1	0.0169	0.0058	-0.0037	2.81×10^{-3}
5×10^4	0.3	0.0243	0.0162	0.0089	2.77×10^{-3}

6.2.3 Computational performance compared to other methods

Due to the strict requirement of $COV_{thr} = 0.015$, methods such as AK-MCS [17], EGRA [38] and REAK [4] that rely on large pools of candidate design samples ($N_{mcs} = 2.55 \times 10^7$) fail to perform well for this example with 30 random variables. Two methods including AK-SS [26] and the method proposed by Ling et al. [40] are applied to this problem and compared with the proposed method. Simulation results for Example 2 are summarized in the Table 9. The probability of failure estimated through RASA is found to be very close to the estimate by MCS. This accuracy stems in part from the unbiased property of the proposed estimator as proved in Theorem 1. On the other hand, AK-SS and the method by AK-SS (Ling et al.) fail to guarantee an accurate estimate of the failure probability. For example, the estimated failure probability can be unreliable if the initial number of candidate design samples is smaller than 5×10^4, and fails to satisfy the requirement that the COV of estimated failure probability should be smaller than COV_{thr}. The relative errors in failure probability estimates by AK-SS, AK-SS (Ling et al.) and RASA are 15.3%, 10.2% and 1.1%, respectively. However, RASA requires a higher number of calls to the performance function compared with the two other approaches. Generally, RASA is more accurate in terms of the final estimated failure probability. It should be noted that in reliability problems, meeting a target accuracy very often supersedes lowering the computational demand, as the implications of incorrect decisions based on inaccurate reliability estimates can be significant.

Table 9. The comparison of computational performances of AK-SS, AK-SS (Ling et al.) and RASA for Example 2.

Methodologies	$\bar{N}_{call}$	$\hat{P}_f^{ss}$	ECC	Account for COV	Is the estimate reliable	Unbiased estimator
MCS	1×10^8	2.74×10^{-3}	-	-	-	-
AK-SS ($p_o = 0.1$& $N_{ss} = 10000$)	1322	2.32×10^{-3}	Depends on p_o and N_{ss}	Yes	Depends on definition of MCMC	No
AK-SS(Ling) ($p_o = 0.1$& $N_{ss} = 10000$)	1124	2.46×10^{-3}	Depends on p_o and N_{ss}	No	Depends on definition of MCMC	No
RASA	1536	2.77×10^{-3}	No	Yes	Depends on N_{ss}	Yes

*ECC means the specified algorithm will exceed the computational capacity of the computer built with Intel(R) Core(TM) i5-6300HQ CPU, RAM 16.00GB.

6.3 A building example with 110 random variables

The third example investigates a large frame structure with 110 input random variables. This structure consists of 35 members denoted by the circled numbers from 1 to 35 as shown in Fig. 10, and is subjected to five external loads, $P_1 - P_5$. The performance function of this example is defined as,

$$g(x) = 0.06 - |\Delta|, \tag{74}$$

where Δ is the horizontal displacement at the right-top point of this frame (Fig. 10). $\mathrm{E}_1 - \mathrm{E}_{35}$, $\mathrm{I}_1 - \mathrm{I}_{35}$ and $\mathrm{A}_1 - \mathrm{A}_{35}$ are the Young's modulus, moments of inertia and cross-section areas of the 35 beams and columns, respectively. Probability distributions of these 110 random variables are presented in Table 10. For training Kriging models, 50 initial training samples are generated. The initial number of candidate design samples for each subset is set as 5000, and Γ_{thr} is set as 5×10^{-3}. The Initial estimate of the failure probability for this example is 2.89×10^{-3}. Results of this analysis are summarized in Table 11.

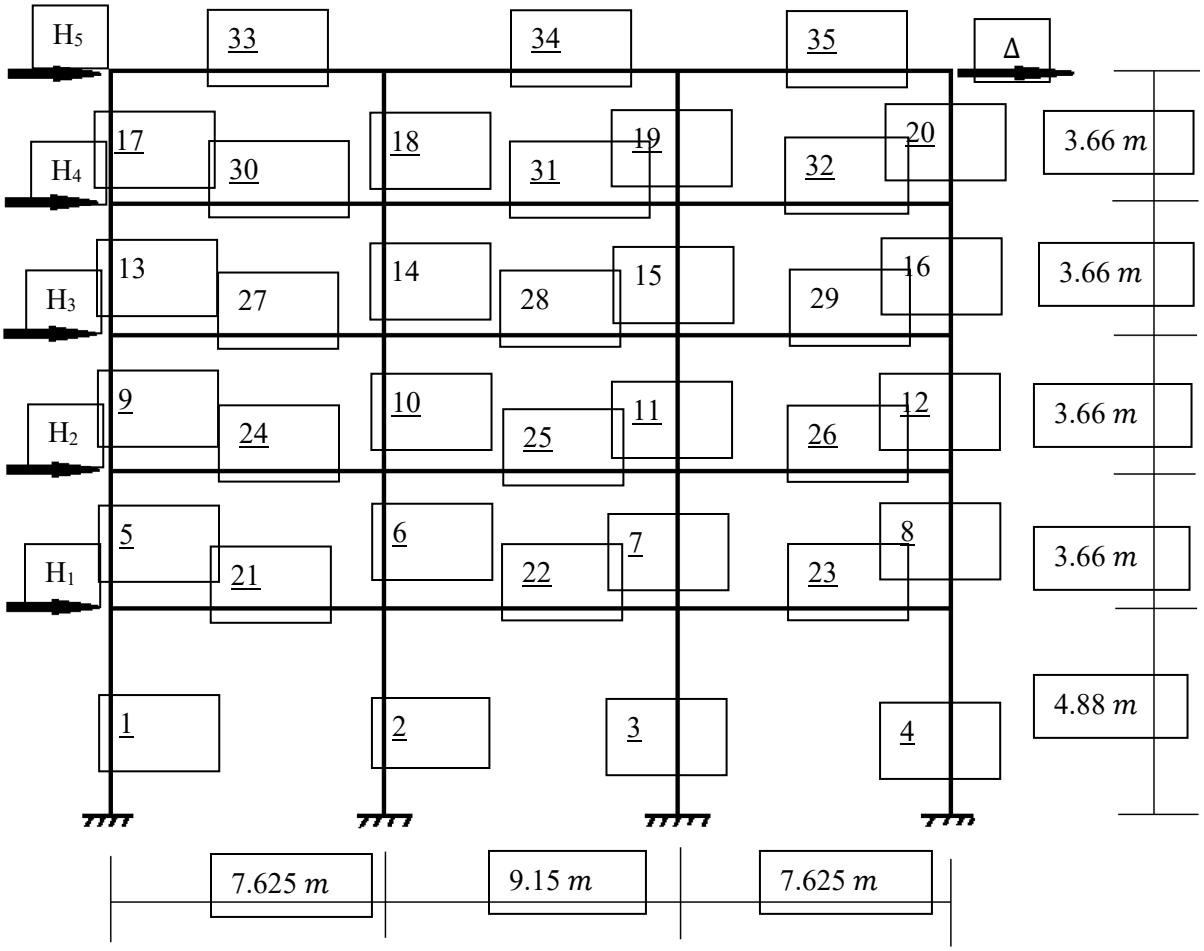

Fig. 10 The frame structure with 110 random variables in Example 3.

Table 10. Random variables in Example 3.

Random variable	Distribution	Mean	COV
$\mathrm{P}_1 - \mathrm{P}_5$	Gumbel	6.5×10^4 (N)	0.1
$\mathrm{E}_1 - \mathrm{E}_4$	Lognormal	$2.1 \times 10^7(\mathrm{N/m})$	0.1
$\mathrm{E}_5 - \mathrm{E}_{20}$	Lognormal	$1.8 \times 10^7(\mathrm{N/m})$	0.1
$\mathrm{E}_{21} - \mathrm{E}_{23}$	Lognormal	$1.6 \times 10^7(\mathrm{N/m})$	0.1
$\mathrm{E}_{24} - \mathrm{E}_{35}$	Lognormal	$1.4 \times 10^7(\mathrm{N/m})$	0.1
$\mathrm{I}_1 - \mathrm{I}_4$	Lognormal	$1.5 \times 10^{-2}(\mathrm{m}^4)$	0.1
$\mathrm{I}_5 - \mathrm{I}_{20}$	Lognormal	$1.2 \times 10^{-2}(\mathrm{m}^4)$	0.1
$\mathrm{I}_{21} - \mathrm{I}_{23}$	Lognormal	$0.9 \times 10^{-2}(\mathrm{m}^4)$	0.1
$\mathrm{I}_{24} - \mathrm{I}_{35}$	Lognormal	$0.7 \times 10^{-2}(\mathrm{m}^4)$	0.1
$\mathrm{A}_1 - \mathrm{A}_4$	Lognormal	$4.0 \times 10^{-1}(\mathrm{m}^2)$	0.1
$\mathrm{A}_5 - \mathrm{A}_{20}$	Lognormal	$3.2 \times 10^{-1}(\mathrm{m}^2)$	0.1
$\mathrm{A}_{21} - \mathrm{A}_{23}$	Lognormal	$2.7 \times 10^{-1}(\mathrm{m}^2)$	0.1
$\mathrm{A}_{24} - \mathrm{A}_{35}$	Lognormal	$2.3 \times 10^{-1}(\mathrm{m}^2)$	0.1

Fig. 11 shows the convergence history of the estimated intermediate thresholds $\hat{t}_i$. Specifically, the identified intermediate thresholds $\hat{t}_1$, $\hat{t}_2$ and $\hat{t}_3$ are exactly the same as the true ones $t_1 = 0.0060$, $t_2 = 0.0018$ and $t_3 = -0.0019$. However, considering Eq. (20), $COV_{\hat{P}_f^{ss}}$ is estimated as 0.0639. According to

Algorithm 3, the minimum required N_{ss} is 9.07×10^4, if COV_{thr} is set as 0.05. However, N_{ss}^{max} is evaluated and set as 1×10^4 for a regular computer such as the one specified earlier in this paper. According to Eq. (24) for this N_{ss}^{max}, the minimum $COV_{\hat{P}_f^{ss}}$ that RASA can reach is 0.024, which is also illustrated in Fig. 12. Thus, p_0 is set to 0.08 to satisfy the requirement of $COV_{thr} = 0.05$. Moreover, the updated intermediate failure probability thresholds are summarized in Table 12. As p_0 has changed from 0.1 to 0.08, the estimate probability of failure $\hat{P}_f^{ss}$ is reevaluated as 3.013×10^{-3} with the corresponding $COV_{\hat{P}_f^{ss}} = 0.0451$.

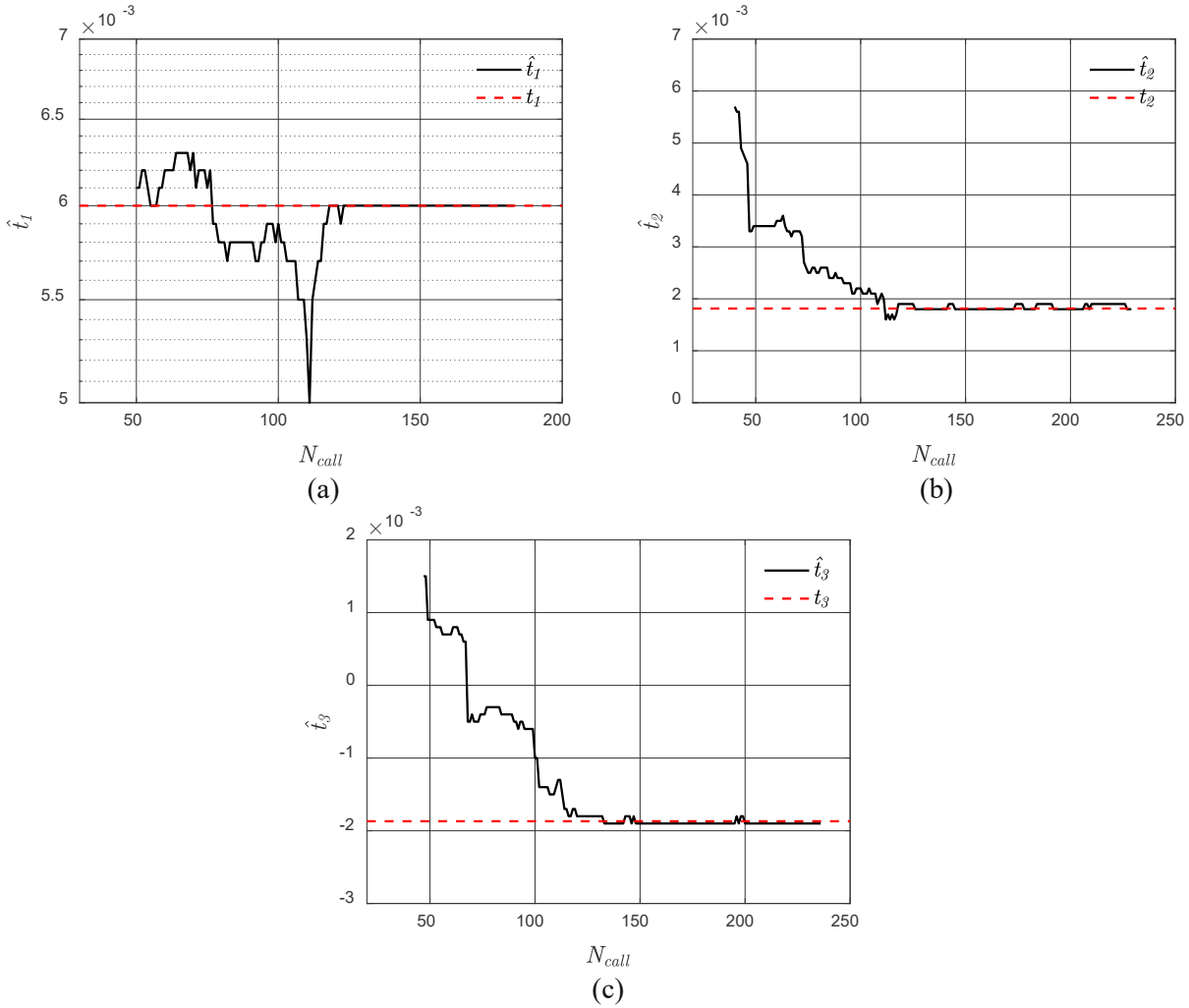

Fig. 11. Convergence history of intermediate failure thresholds for (a) $\hat{t}_1$, (b) $\hat{t}_2$, and (c) $\hat{t}_3$ in Example 3 starting with 50 initial training samples ($p_0 = 0.1$ $N_{ss} = 5000$).

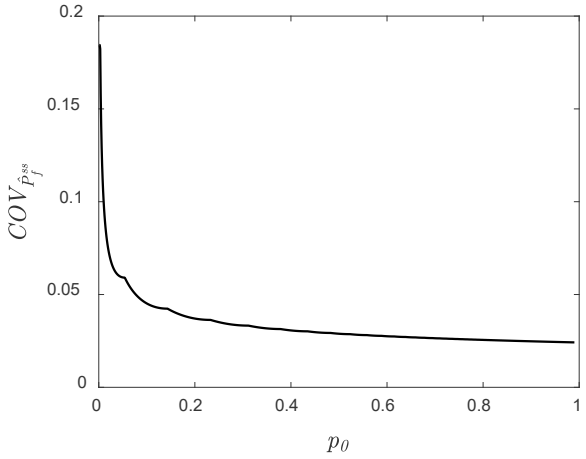

Fig. 12 $COV_{\hat{P}_f^{ss}}$ vs p_0 in Example 3.

Table 11. Identified intermediate failure thresholds using $g(x)$ and $\hat{g}(x)$ for Example 3.

Performance function	t_1 or $\hat{t}_1(N_{call})$	t_2 or $\hat{t}_2(\mathrm{N_{call}})$	t_3 or $\hat{t}_3(N_{call})$	$\hat{P}_f^{ss}$
$g_{(x)}$	0.0060	0.0018	-0.0019	2.932×10^{-3}
$\hat{g}_{(x)}$	$0.0060(50+137)$	$0.0018(194)$	$-0.0019(196)$	2.932×10^{-3}

Table 12. Intermediate failure thresholds using updated N_{ss} and p_0 for Example 3.

p_0	t_1 *or* $\hat{t}_1$	t_2 *or* $\hat{t}_2$	t_3 *or* $\hat{t}_3$	$\hat{P}_f^{ss}$
0.1	0.0060	0.0018	-0.0019	2.932×10^{-3}
0.08	0.0055	0.0012	-	3.013×10^{-3}

For problems with a large number of random inputs, identification of accurate hyperparameters can be a challenge. To examine the computational performance of the method and the change in failure probability estimate for a different number of candidate design samples, the analysis results and the convergence history of the method with 150 initial training points are summarized in Table 13 and Fig. 13. It is shown that the intermediate failure threshold in the first subset converges very fast to the true one due to the large information provided by the initial training samples. However, for the rest of the intermediate failure thresholds, results follow a similar trend to the case with 50 initial training samples as shown in Fig. 11. It is recommended to start with a sufficient number of initial training samples e.g., $n_{in} > N_d$ to guarantee that the problem is not ill-conditioned. Results in Tables 11-13 point to high capabilities of RASA in handling high-dimensional reliability problems. By controlling the number of candidate design samples generated in each subset, RASA can successfully estimate the probability of failure without requiring sensitivity analysis or other dimension-reduction methods.

Table 13. Identified intermediate failure thresholds using $g(x)/\hat{g}(x)$ with 110 initial training samples for Example 3.

Performance function	t_1 or $\hat{t}_1(N_{call})$	t_2 or $\hat{t}_2(N_{call})$	t_3 or $\hat{t}_3(N_{call})$	$\hat{P}_f^{ss}$
$g_{(x)}$	0.0056	0.0011	-0.0028	3.205×10^{-3}
$\hat{g}_{(x)}$	$0.0056(150+82)$	$0.0011(165)$	$-0.0028(196)$	3.206×10^{-3}

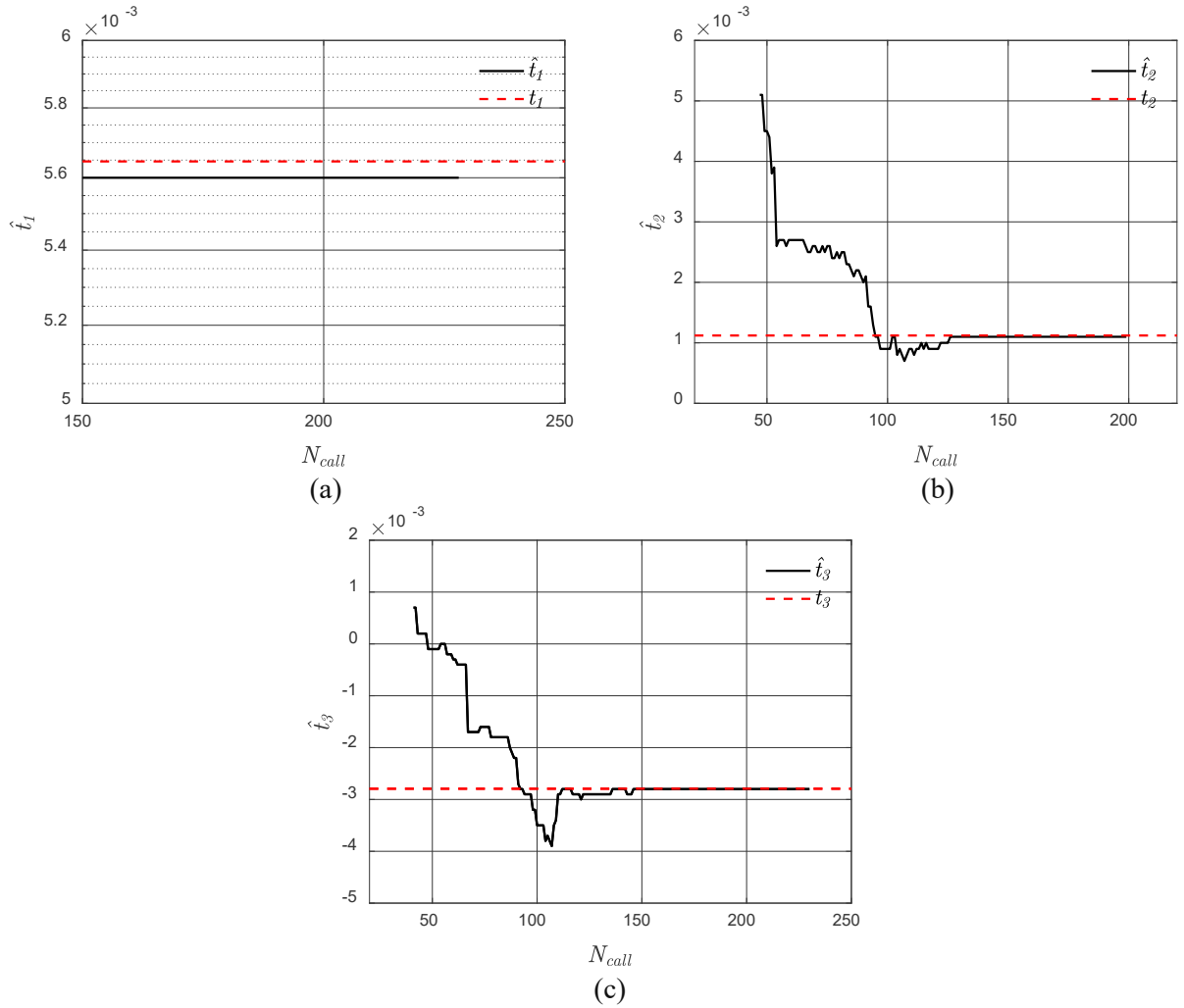

Fig. 13. Convergence history of intermediate failure thresholds for (a) $\hat{t}_1$, (b) $\hat{t}_2$, and (c) $\hat{t}_3$ in Example 3 starting with 150 initial training samples ($p_0 = 0.08$ $N_{ss} = 5000$).

Moreover, the computational results for RASA, AK-SS and AK-SS (Ling et al.) are summarized in Table 14. As discussed in section 6.1.3, Kriging-based reliability methods such as EGRA, AK-MCS, and REAK that need a large pool of candidate design samples can hardly meet the computational limitations for this problem. Based on Eq. (16), the required number of candidate design samples should be as large as 1.4×10^5 for the level of $COV_{thr} = 0.05$. Direct implementation of crude Kriging-based MCS will exceed the computational limit of $N_{ss}^{max} = 1 \times 10^4$. Other Kriging-based Subset Simulation methods such as AK-SS [26] and the approach by Ling et al. [40] cannot guarantee reliable estimates of failure probability. For example, the estimated failure probability can be unreliable if the initial number of candidate design samples is smaller than 1×10^4, and fails to satisfy the requirement that the COV of estimated failure probability should be smaller than COV_{thr}. In fact, the relative errors in failure probability estimates by AK-SS, AK-SS (Ling et al.) and RASA are 26.32%, 9.94% and 1.1%, respectively.

Table 14. The comparison of computational performance among AK-SS, AK-SS (Ling et al.) and RASA for Example 3.

Methodologies	$\hat{P}_f$	$\bar{N}_{call}$	ECC	Accounts for COV	Is the estimate reliable	Unbiased estimator
MCS	3.017×10^{-3}	1×10^{8}	-	-	-	-
AK-SS ($p_o = 0.1$& $N_{ss} = 10000$)	2.232×10^{-3}	>700	Depends on p_o and N_{ss}	Yes	Depends on the definition of proposal sampling for MCMC	No
AK-SS(Ling) ($p_o = 0.1$& $N_{ss} = 10000$)	2.716×10^{-3}	421	Depends on p_o and N_{ss}	No	Depends on the definition of proposal sampling for MCMC	No
RASA	3.013×10^{-3}	523	No	Yes	Depends on N_{ss}	Yes

*ECC means the specified algorithm will exceed the computational capacity of the computer built with Intel(R) Core(TM) i5-6300HQ CPU, RAM 16.00GB.

7. Conclusion

Metamodel-based reliability analysis methods face significant challenge for computationally demanding problems such as high-dimensional or rare event reliability problems. To enable tackling such problems using Kriging-based reliability analysis techniques, a novel method called RASA is proposed here, which integrates Subset Simulation and Kriging surrogate modeling. The main idea of RASA is to control the required computational demand by adaptively adjusting the intermediate failure probabilities and the number of candidate design samples in each subset. Toward this goal, two new concepts are proposed in this paper to adaptively identify the intermediate failure thresholds: Conditional Failure Probability Curve (CFPC) and Dynamic Learning Function (DLF). CFPC and DLF enable the process of adaptively identifying the threshold value for each limit state function with corresponding intermediate probability of failure. Since the number of candidate design samples in each subset is significantly small compared to regular Kriging-based reliability approaches such as AK-MCS, RASA has the capability to handle complex reliability problems. Three numerical examples are investigated to examine the application and performance of RASA. Results confirm the capabilities of RASA in tackling large, complex reliability problems and offering reliable estimates of failure probability.

Acknowledgements

This research was supported in part by the U.S. National Science Foundation (NSF) through awards CMMI-1762918 and 2000156 as well as 'Shuimu Tsinghua Scholar' Plan by Tsinghua University. The first author also thanks Prof. Quanwang Li for the support of postdoctoral research at Tsinghua university. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of these supports. These supports are greatly appreciated.

Compliance with ethical standards

Conflict of interests

On behalf of all authors, the corresponding author states that there is no conflict of interest.

Replication of results

The algorithms and step-by-step implementation approach for the proposed reliability analysis with Subset Simulation using adaptive Kriging (RASA) are presented in Algorithms 1 to 5 in the paper. Readers can use MATLAB and UQLab Kriging package to implement the algorithms and generate the results presented in the paper.

Reference

[1] J. H. Choi, N.-H. Kim, and N. Kogiso, "Reliability for aerospace systems: Methods and applications," *Adv. Mech. Eng.*, vol. 8, no. 11, p. 1687814016677092, Nov. 2016, doi: 10.1177/1687814016677092.

[2] D. Kececioglu, "Reliability analysis of mechanical components and systems," *Nucl. Eng. Des.*, vol. 19, no. 2, pp. 259–290, May 1972, doi: 10.1016/0029-5493(72)90133-1.

[3] W. Kuo, "Reliability and Nuclear Power," *IEEE Trans. Reliab.*, vol. 60, no. 2, pp. 365–367, Jun. 2011, doi: 10.1109/TR.2011.2152430.

[4] Z. Wang and A. Shafieezadeh, "REAK: Reliability analysis through Error rate-based Adaptive Kriging," *Reliab. Eng. Syst. Saf.*, vol. 182, pp. 33–45, Feb. 2019, doi: 10.1016/j.ress.2018.10.004.

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

[6] G. Fishman, *Monte Carlo: concepts, algorithms, and applications*. Springer Science & Business Media, 2013.

[7] B. Echard, N. Gayton, M. Lemaire, and N. Relun, "A combined importance sampling and kriging reliability method for small failure probabilities with time-demanding numerical models," *Reliab. Eng. Syst. Saf.*, vol. 111, pp. 232–240, 2013.

[8] S.-K. Au and J. L. Beck, "Estimation of small failure probabilities in high dimensions by subset simulation," *Probabilistic Eng. Mech.*, vol. 16, no. 4, pp. 263–277, 2001.

[9] R. Rackwitz and B. Flessler, "Structural reliability under combined random load sequences," *Comput. Struct.*, vol. 9, no. 5, pp. 489–494, Nov. 1978, doi: 10.1016/0045-7949(78)90046-9.

[10] Kiureghian Armen Der and Stefano Mario De, "Efficient Algorithm for Second-Order Reliability Analysis," *J. Eng. Mech.*, vol. 117, no. 12, pp. 2904–2923, Dec. 1991, doi: 10.1061/(ASCE)0733-9399(1991)117:12(2904).

[11] V. J. Romero, L. P. Swiler, and A. A. Giunta, "Construction of response surfaces based on progressive-lattice-sampling experimental designs with application to uncertainty propagation," *Struct. Saf.*, vol. 26, no. 2, pp. 201–219, 2004.

[12] A. A. Giunta, J. M. McFarland, L. P. Swiler, and M. S. Eldred, "The promise and peril of uncertainty quantification using response surface approximations," *Struct. Infrastruct. Eng.*, vol. 2, no. 3–4, pp. 175–189, 2006.

[13] W. Zhao, F. Fan, and W. Wang, "Non-linear partial least squares response surface method for structural reliability analysis," *Reliab. Eng. Syst. Saf.*, vol. 161, pp. 69–77, May 2017, doi: 10.1016/j.ress.2017.01.004.

[14] G. Blatman and B. Sudret, "An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis," *Probabilistic Eng. Mech.*, vol. 25, no. 2, pp. 183–197, 2010.

[15] H. Dai, H. Zhang, W. Wang, and G. Xue, "Structural reliability assessment by local approximation of limit state functions using adaptive Markov chain simulation and support vector regression," *Comput.-Aided Civ. Infrastruct. Eng.*, vol. 27, no. 9, pp. 676–686, 2012.

[16] J.-M. Bourinet, "Rare-event probability estimation with adaptive support vector regression surrogates," *Reliab. Eng. Syst. Saf.*, vol. 150, pp. 210–221, 2016.

[17] B. Echard, N. Gayton, and M. Lemaire, "AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation," *Struct. Saf.*, vol. 33, no. 2, pp. 145–154, 2011.

[18] W. Fauriat and N. Gayton, "AK-SYS: An adaptation of the AK-MCS method for system reliability," *Reliab. Eng. Syst. Saf.*, vol. 123, pp. 137–144, 2014.

[19] Z. Wang and A. Shafieezadeh, "Highly efficient Bayesian updating using metamodels: An adaptive Kriging-based approach," *Struct. Saf.*, vol. 84, p. 101915, May 2020, doi: 10.1016/j.strusafe.2019.101915.

[20] Z. Wang and A. Shafieezadeh, "ESC: an efficient error-based stopping criterion for kriging-based reliability analysis methods," *Struct. Multidiscip. Optim.*, vol. 59, no. 5, Art. no. 5, May 2019, doi: 10.1007/s00158-018-2150-9.

[21] I. Kaymaz, "Application of kriging method to structural reliability problems," *Struct. Saf.*, vol. 27, no. 2, pp. 133–151, 2005.

[22] B. Gaspar, A. P. Teixeira, and C. G. Soares, “Assessment of the efficiency of Kriging surrogate models for structural reliability analysis,” *Probabilistic Eng. Mech.*, vol. 37, pp. 24–34, Jul. 2014, doi: 10.1016/j.probengmech.2014.03.011.
[23] B. J. Bichon, M. S. Eldred, L. P. Swiler, S. Mahadevan, and J. M. McFarland, “Efficient global reliability analysis for nonlinear implicit performance functions,” *AIAA J.*, vol. 46, no. 10, pp. 2459–2468, 2008.
[24] M. Balesdent, J. Morio, and J. Marzat, “Kriging-based adaptive Importance Sampling algorithms for rare event estimation,” *Struct. Saf.*, vol. 44, pp. 1–10, Sep. 2013, doi: 10.1016/j.strusafe.2013.04.001.
[25] V. Dubourg, B. Sudret, and F. Deheeger, “Metamodel-based importance sampling for structural reliability analysis,” *Probabilistic Eng. Mech.*, vol. 33, pp. 47–57, Jul. 2013, doi: 10.1016/j.probengmech.2013.02.002.
[26] X. Huang, J. Chen, and H. Zhu, “Assessing small failure probabilities by AK–SS: an active learning method combining Kriging and subset simulation,” *Struct. Saf.*, vol. 59, pp. 86–95, 2016.
[27] V. Dubourg, B. Sudret, and J.-M. Bourinet, “Reliability-based design optimization using Kriging surrogates and subset simulation,” *Struct. Multidiscip. Optim.*, vol. 44, no. 5, pp. 673–690, Nov. 2011, doi: 10.1007/s00158-011-0653-8.
[28] N. Pedroni and E. Zio, “An Adaptive Metamodel-Based Subset Importance Sampling approach for the assessment of the functional failure probability of a thermal-hydraulic passive system,” *Appl. Math. Model.*, vol. 48, pp. 269–288, Aug. 2017, doi: 10.1016/j.apm.2017.04.003.
[29] V. J. Romero, L. P. Swiler, and A. A. Giunta, “Construction of response surfaces based on progressive-lattice-sampling experimental designs with application to uncertainty propagation,” *Struct. Saf.*, vol. 26, no. 2, Art. no. 2, 2004.
[30] “UQLab Kriging (Gaussian process modelling) manual,” *UQLab, the Framework for Uncertainty Quantification*. http://www.uqlab.com/userguidekriging (accessed May 13, 2017).
[31] B. Bichon, M. Eldred, L. Swiler, S. Mahadevan, and J. McFarland, “Multimodal reliability assessment for complex engineering applications using efficient global optimization,” in *48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2007, p. 1946, Accessed: May 12, 2017. [Online]. Available: http://arc.aiaa.org/doi/pdf/10.2514/6.2007-1946.
[32] J. Wang, Z. Sun, Q. Yang, and R. Li, “Two accuracy measures of the Kriging model for structural reliability analysis,” *Reliab. Eng. Syst. Saf.*, vol. 167, pp. 494–505, Nov. 2017, doi: 10.1016/j.ress.2017.06.028.
[33] Z. Wang and A. Shafieezadeh, “On confidence intervals for failure probability estimates in Kriging-based reliability analysis,” *Reliab. Eng. Syst. Saf.*, vol. 196, p. 106758, Apr. 2020, doi: 10.1016/j.ress.2019.106758.
[34] Z. Wang and A. Shafieezadeh, “Confidence Intervals for Failure Probability Estimates in Adaptive Kriging-based Reliability Analysis,” *Reliab. Eng. Syst. Saf.*
[35] J. Guo and X. Du, “Reliability sensitivity analysis with random and interval variables,” *Int. J. Numer. Methods Eng.*, vol. 78, no. 13, pp. 1585–1617, 2009.
[36] Z. Wen, H. Pei, H. Liu, and Z. Yue, “A Sequential Kriging reliability analysis method with characteristics of adaptive sampling regions and parallelizability,” *Reliab. Eng. Syst. Saf.*, vol. 153, pp. 170–179, 2016.
[37] B. Echard, N. Gayton, and M. Lemaire, “AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation,” *Struct. Saf.*, vol. 33, no. 2, Art. no. 2, 2011.
[38] B. J. Bichon, M. S. Eldred, L. P. Swiler, S. Mahadevan, and J. M. McFarland, “Efficient global reliability analysis for nonlinear implicit performance functions,” *AIAA J.*, vol. 46, no. 10, Art. no. 10, 2008.
[39] Z. Wang and A. Shafieezadeh, “REAK: Reliability analysis through Error rate-based Adaptive Kriging,” *Reliab. Eng. Syst. Saf.*, vol. 182, pp. 33–45, Feb. 2019, doi: 10.1016/j.ress.2018.10.004.
[40] C. Ling, Z. Lu, K. Feng, and X. Zhang, “A coupled subset simulation and active learning kriging reliability analysis method for rare failure events,” *Struct. Multidiscip. Optim.*, vol. 60, no. 6, pp. 2325–2341, Dec. 2019, doi: 10.1007/s00158-019-02326-3.