Adaptive reliability analysis for rare events evaluation with global imprecise line sampling

Jingwen Song [a,b], Pengfei Wei [a,b,*], Marcos Valdebenito [c], Michael Beer [b,d,e]

[a]*Northwestern Polytechnical University, West Youyi Road 127, Xi'an 710072, China*
[b]*Institute for Risk and Reliability, Leibniz Universität Hannover, Callinstr.34, Hannover 30167, Germany*
[c]*Departmento de Obras Civiles, Universidad Tecnica Federico Santa Maria, Av. España 1680, Valparaiso, Chile*
[d]*Institute for Risk and Uncertainty, University of Liverpool, Peach Street, L69 7ZF Liverpool,United Kingdom*
[e]*International Joint Research Center for Engineering Reliability and Stochastic Mechanics, Tongji University, Shanghai 200092, China*

Abstract: The efficient estimation of the failure probability function of rare failure events is a challenging task in the structural safety analysis when the input variables are characterized by imprecise probability models due to insufficient information on these variables. The recently developed non-intrusive imprecise stochastic simulation (NISS) provides a general, yet competitive, framework for dealing with this type of problems, and it has been shown that many classical stochastic simulation techniques, with suitable adequations, can be injected into this framework for tackling different types of problems in uncertainty quantification. This work aims at investigating the rare failure event analysis based on the global version of NISS and line sampling. A new method, called global imprecise line sampling (GILS), is firstly proposed, to efficiently estimate failure probability function with the same computational cost as classical line sampling. By joint sampling from both the aleatory and epistemic spaces, the GILS provides elegant estimators for the functional components of the failure probability. Then, to further reduce the computational cost, and improve its suitability for nonlinear problems, an imprecise active learning line sampling procedure is established by combining GILS with Gaussian process regression (GPR) with the target of adaptively exploring the aleatory and epistemic spaces within the framework of line sampling. Two analytical examples and two engineering applications demonstrate the efficiency and accuracy of the proposed method.

Keywords: Uncertainty quantification; Imprecise probability; Line sampling; Sensitivity analysis; Active learning; Gaussian process regression.

1. Introduction

Estimating failure probability is one of the most important tasks in structural engineering with the consideration of the intrinsic randomness present in structural parameters, initial/boundary conditions, and environmental excitations. Although it has been widely studied for many decades, it is continually receiving great attention, especially when the available information on those random input parameters is imperfect, and thus there exist both aleatory uncertainty (intrinsic randomness) and epistemic uncertainty (caused by lack of information). These two sources of uncertainties have totally different effects on the reliability of structural systems. The aleatory uncertainty results in random failure, thus leading to the intrinsic probability of failure, of the structural systems, while the existence of epistemic uncertainty prevents us from learning the true value of this probability. It has been widely

* Corresponding author at: School of Mechanics, Civil Engineering and Architecture, Northwestern Polytechnical University, Xi'an, 710072, PR China, pengfeiwei@nwpi.edu.cn (Pengfei Wei)

realized that, without properly distinguishing and modeling these two types of uncertainties, the failure probability can be underestimated or overestimated to a significant extent [1].

With only aleatory uncertainty being present, precise probability models (random variables, random fields, and stochastic processes) are mostly utilized, and the resultant failure probability is a deterministic value. Since the 70s of last century, the problem of estimating this deterministic probability of failure has been widely investigated, and plenty of methods such as approximate analytical methods (e.g., first-order and second reliability methods [2], moment methods [3], etc.), probability conservation based methods (e.g., probability density evolution method [4]), stochastic simulation methods (e.g., importance sampling [5], subset simulation [6], line sampling [7] and directional sampling [8]) and surrogate model methods (e.g., support vector machine [9], neural network [10], radial basis function and Kriging model [11]), have been developed for tackling different types of challenges such as problems with high-dimensional inputs, highly nonlinear performance function, extremely small failure probability, etc. Specifically, the active learning algorithms combining the Kriging (also called Gaussian Process Regression, GPR) model and the stochastic simulation techniques have received increased attention (see e.g., Refs. [12]-[15]), and have been shown to be especially effective for dealing with large structural systems due to their perfect balance of accuracy and efficiency. Overall, estimating the deterministic failure probability has been comprehensively addressed, although pursuing algorithms with even higher efficiency is still a subject of active research.

With both aleatory and epistemic uncertainties being present, the precise probability models are no longer sufficient for modeling the input parameters due to their incapability of distinguishing these two types of uncertainties. The imprecise probability models, such as the probability-box (*p*-box) model [16], the evidence theory (or called Dempster-Shafer theory) [17][18], and the fuzzy probability model [19], as natural extensions of the precise probability models, have been regarded as the most appealing model frameworks for addressing the mixed uncertainties [20], since they can provide reasonable (solid or soft) bounds for bounding the true probability distribution, and for properly modeling the aleatory and epistemic uncertainties separately under hierarchical model frameworks. Thus, propagating these imprecise probability models through computer simulators of structures, and estimating the resultant non-deterministic failure probability, have become the new challenge to be solved for structural reliability analysis.

Traditional algorithms for addressing this challenge involve a double-loop numerical process, and three different stochastic simulation strategies have been developed. The first strategy involves estimating the deterministic failure probability in the inner loop, given fixed probability distribution of inputs, and calculating the bounds of the failure probability in the outer-loop, by performing numerical optimization [21]. The second strategy is based on performing sampling in the outer-loop to create a set of interval samples for the input parameters, and then in the inner loop, estimating the bounds of the model response subjected to each interval sample of inputs, by using, e.g. interval finite element analysis or numerical optimization, and in this way, estimate the bounds of failure probability based on the resultant interval samples of model response[22][23]. Both of the above two strategies can be computationally intractable for large structures due to the tremendous number of response function calls required. The third strategy then aims to reduce the computational cost by reusing the response function calls in each inner-loop iteration, and typical methods include advanced line sampling [24] and the extended Monte Carlo simulation (EMCS)[25]. The EMCS method requires only one stochastic simulation, thus its theoretical computational cost is the

same as that for estimating the deterministic failure probability; however, the variation of the EMCS estimators can be quite large for problems with high-dimensional epistemic parameters. For overcoming this disadvantage, a general methodology framework, named as Non-intrusive Imprecise Stochastic Simulation (NISS), has been established by combining EMCS, high-dimensional model representation (HDMR) decomposition and sensitivity analysis[26][27], where the local NISS methods are established based on the cut-HDMR decomposition [28], and the global NISS methods are developed based on the Random Sampling- (RS-) HDMR decomposition [29]. In our previous work, two strategies have been developed for injecting the classical line sampling to the local NISS framework for efficiently estimating the imprecise failure probability with only one line sampling simulation, and shown to be effective for rare failure events[30]. Alternatively, the surrogate model methods have also been introduced for efficiently propagating the mixed uncertainties or for estimating the imprecise (or interval) failure probability[31]-[33].

The contributions of this paper are twofold. First, to further refine the NISS framework for rare event analysis, the failure probability function is reformulated by injecting probability density function (PDF) weight into the integral of classic line sampling, which is actually a combination of a (*n*-1)-dimensional integral orthogonal to the important direction and a one-dimensional integral along each line that is parallel to the aforementioned important direction. Then, the failure probability function is formulated as a series of one-dimensional failure probability functions, and the computational cost is shown to be the same as classical line sampling. Since this method can globally investigate the whole space of imprecise distribution parameters, we denote it as global imprecise line sampling (GILS). As this scheme involves implementing the simulation in the joint space of both aleatory variables and epistemic distribution parameters, its convergence rate can be slightly slower than classical line sampling. Then, for further reduction of the computational cost, an active learning algorithm based on GPR model is proposed and injected into the GILS method, which can actively construct a computationally less expensive GPR model to approximate the true failure surface of the performance function, such that the intersection points for all lines can be accurately calculated.

The rest of this paper is organized as follows. Section 2 provides a brief review of the global NISS, followed by the detailed development of the GILS method in section 3. In section 4, the imprecise active learning ling sampling is combined with GILS to further reduce the computational cost. Section 5 introduces two numerical examples and two engineering problems for demonstrating the effectiveness of the proposed methods. Section 6 concludes the paper.

2. Non-intrusive imprecise stochastic simulation

Let $g(\boldsymbol{x})$ denote the performance function of a structure or an engineering system, where $\boldsymbol{x}=(x_1,\cdots,x_d,\cdots,x_n)^T$ denotes the *n*-dimensional random input variables with joint PDF $f_{\mathbf{X}}(\boldsymbol{x}|\boldsymbol{\theta})$, where $\boldsymbol{\theta}=(\theta_1,\cdots,\theta_i,\cdots,\theta_m)^T$ denotes the vector of all the distribution parameters associated with $\boldsymbol{x}$. In the framework of precise probability models, the PDF of $\boldsymbol{x}$ is precisely known and the distribution parameters are constant. In this paper, the distribution parameters are assumed to be imprecisely known due to the lack of information on $\boldsymbol{x}$. For

simplification, the uncertainty of $\boldsymbol{\theta}$ is considered to be characterized by intervals, i.e., $\boldsymbol{\theta} \in \left[\underline{\boldsymbol{\theta}}, \overline{\boldsymbol{\theta}}\right]$, so the input variables are modeled by a parameterized *p*-box.

This paper focuses on the reliability analysis of models with imprecise probability, thus the failure domain is defined as $F = \{\boldsymbol{x}: g(\boldsymbol{x}) \leqslant 0\}$, and the corresponding indicator function is formulated by $I_F(\boldsymbol{x}) = 1$ if $\boldsymbol{x} \in F$, and $I_F(\boldsymbol{x}) = 0$ if $\boldsymbol{x} \notin F$. Since the distribution parameters $\boldsymbol{\theta}$ of input variables are uncertain, the failure probability becomes a function of $\boldsymbol{\theta}$, denoted as failure probability function $p_f(\boldsymbol{\theta})$ with the formulation expressed by

$$p_f(\boldsymbol{\theta}) = \int_{g(\boldsymbol{x}) \leqslant 0} f_{\boldsymbol{X}}(\boldsymbol{x}|\boldsymbol{\theta}) \mathrm{d}\boldsymbol{x} = \int_{\mathbb{R}^n} I_F(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}|\boldsymbol{\theta}) \mathrm{d}\boldsymbol{x}. \tag{1}$$

The HDMR decomposition of the failure probability function represents $p_f(\boldsymbol{\theta})$ as the sum of functional components of increasing order, i.e.

$$p_f(\boldsymbol{\theta}) = p_{f0} + \sum_{i=1}^{m} p_{fi}(\theta_i) + \sum_{i=1}^{m} \sum_{j=i+1}^{m} p_{fij}(\theta_i, \theta_j) + \cdots + p_{f1,\ldots,m}(\boldsymbol{\theta}) \tag{2}$$

, where p_{f0} denotes a constant term, $p_{fi}(\theta_i)$ is a function of the distribution parameter θ_i, $p_{fij}(\theta_i, \theta_j)$ is a function of the distribution parameters θ_i and θ_j, etc. The formulations of component functions are based on the way of how the contributions of $\boldsymbol{\theta}$ are apportioned to the components. If the cut-HDMR method is utilized, the failure probability function is expanded at a fixed point $\boldsymbol{\theta}^*$, which is the basis of the local NISS method that has been investigated in Ref. [30]. However, the implementation of the local method demands to determine an expansion point $\boldsymbol{\theta}^*$ within the support domain, and generally, the estimation accuracy will decrease for values of the distribution parameters which are far from $\boldsymbol{\theta}^*$. To improve the global performance and avoid determining a fixed value that acts as an expansion point, this paper focuses on global NISS by using random sampling (RS)-HDMR method [29], and the functional components in Eq.(2) are defined as

$$\begin{aligned} p_{f0} &= \int p_f(\boldsymbol{\theta}) f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} \\ p_{fi}(\theta_i) &= \int p_f(\boldsymbol{\theta}) f_{\boldsymbol{\Theta}_{-i}}(\boldsymbol{\theta}_{-i}) \mathrm{d}\boldsymbol{\theta}_{-i} - p_{f0} \\ p_{fij}(\theta_i, \theta_j) &= \int p_f(\boldsymbol{\theta}) f_{\boldsymbol{\Theta}_{-ij}}(\boldsymbol{\theta}_{-ij}) \mathrm{d}\boldsymbol{\theta}_{-ij} - p_{fi}(\theta_i) - p_{fj}(\theta_j) - p_{f0} \end{aligned} \tag{3}$$

, where $\boldsymbol{\theta}_{-i}$ denotes the set of all distribution parameters except for θ_i while $\boldsymbol{\theta}_{-ij}$ denotes the set of all distribution parameters except for θ_i and θ_j, and where $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$, $f_{\boldsymbol{\Theta}_{-i}}(\boldsymbol{\theta}_{-i})$ and $f_{\boldsymbol{\Theta}_{-ij}}(\boldsymbol{\theta}_{-ij})$ indicate the instrumental PDF w.r.t. $\boldsymbol{\theta}$, $\boldsymbol{\theta}_{-i}$ and $\boldsymbol{\theta}_{-ij}$ within the support domain $\boldsymbol{\Theta}$, respectively. Generally, the instrumental PDF can be assumed to be of uniform type within $\left[\underline{\boldsymbol{\theta}}, \overline{\boldsymbol{\theta}}\right]$. One note that, by introducing this instrumental probability distribution, we don't mean that the distribution parameter $\boldsymbol{\theta}$ are random variables in nature. Instead, the auxiliary distribution is introduced for defining the HDMR decomposition of the failure probability function, and for making it possible to estimate these HDMR components in the joint space of $\boldsymbol{x}$ and $\boldsymbol{\theta}$. This is the vital difference between global NISS and local NISS. One can also use other types of auxiliary distribution, and theoretically, this will not affect the behavior of the failure

probability function, but will affect the behavior of its HDMR components as well as the estimation performance. For improving the estimation accuracy around the bounds of $\boldsymbol{\theta}$, the support of the instrumental distribution can be relaxed as $\left[\underline{\boldsymbol{\theta}}-\epsilon\Delta\boldsymbol{\theta},\overline{\boldsymbol{\theta}}+\epsilon\Delta\boldsymbol{\theta}\right]$, where $\Delta\boldsymbol{\theta}=\overline{\boldsymbol{\theta}}-\underline{\boldsymbol{\theta}}$, and ϵ can be set as $3-5\%$.

According to the NISS framework, only one set of samples is needed for estimating all the component functions in Eq.(3). According to Ref. [26], by generating a set of joint samples $(\boldsymbol{x}^{(s)},\boldsymbol{\theta}^{(s)})$ $(s=1,\cdots,N)$ from the joint PDF $f_{\mathbf{X},\Theta}(\boldsymbol{x},\boldsymbol{\theta})=f_{\mathbf{X}}(\boldsymbol{x}|\boldsymbol{\theta})f_{\Theta}(\boldsymbol{\theta})$, the unbiased estimators for the constant, first-order and second-order component functions are formulated as

$$\begin{aligned}\hat{p}_{f0}&=\frac{1}{N}\sum_{s=1}^{N}I_F(\boldsymbol{x}^{(s)})\\ \hat{p}_{fi}(\theta_i)&=\frac{1}{N}\sum_{s=1}^{N}I_F(\boldsymbol{x}^{(s)})r_i(\boldsymbol{x}^{(s)}|\theta_i,\boldsymbol{\theta}^{(s)})\\ \hat{p}_{fij}(\theta_i,\theta_j)&=\frac{1}{N}\sum_{s=1}^{N}I_F(\boldsymbol{x}^{(s)})r_{ij}(\boldsymbol{x}^{(s)}|\theta_i,\theta_j,\boldsymbol{\theta}^{(s)})\end{aligned}\tag{4}$$

, where $r_i(\boldsymbol{x}^{(s)}|\theta_i,\boldsymbol{\theta}^{(s)})$ and $r_{ij}(\boldsymbol{x}^{(s)}|\theta_i,\theta_j,\boldsymbol{\theta}^{(s)})$ are regarded as weight coefficients defined as

$$\begin{aligned}r_i(\boldsymbol{x}^{(s)}|\theta_i,\boldsymbol{\theta}^{(s)})&=\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(s)}|\theta_i,\boldsymbol{\theta}_{-i}^{(s)})}{f_{\boldsymbol{X}}(\boldsymbol{x}^{(s)}|\boldsymbol{\theta}^{(s)})}-1\\ r_{ij}(\boldsymbol{x}^{(s)}|\theta_i,\theta_j,\boldsymbol{\theta}^{(s)})&=\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(s)}|\theta_i,\theta_j,\boldsymbol{\theta}_{-ij}^{(s)})}{f_{\boldsymbol{X}}(\boldsymbol{x}^{(s)}|\boldsymbol{\theta}^{(s)})}-\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(s)}|\theta_i,\boldsymbol{\theta}_{-i}^{(s)})}{f_{\boldsymbol{X}}(\boldsymbol{x}^{(s)}|\boldsymbol{\theta}^{(s)})}-\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(s)}|\theta_j,\boldsymbol{\theta}_{-j}^{(s)})}{f_{\boldsymbol{X}}(\boldsymbol{x}^{(s)}|\boldsymbol{\theta}^{(s)})}-1.\end{aligned}\tag{5}$$

The variances of the above estimators of component functions can be easily derived referring to the basic Monte Carlo algorithm. From the cases studied in the available literature [28][29], commonly, an HDMR decomposition with second-order truncation can lead to a satisfactory approximation of the original model, hence the estimator of failure probability function $\hat{p}_f(\boldsymbol{\theta})$ can be expressed by:

$$\hat{p}_f(\theta)\approx\hat{p}_{f0}+\sum_{i=1}^{m}\hat{p}_{fi}(\theta_i)+\sum_{i=1}^{m}\sum_{j=i+1}^{m}\hat{p}_{fij}(\theta_i,\theta_j).\tag{6}$$

However, the above method will be less efficient especially for rare failure events, and one needs to investigate more advanced stochastic simulation techniques to improve the performance of global NISS for solving those problems. In the next section, the line sampling method is injected into global NISS so as to provide a more efficient scheme to estimate failure probability considering imprecise probabilities.

3. Global imprecise line sampling

The classical line sampling method for reliability analysis is well-known for its high efficiency for problems involving a linear or moderately nonlinear performance function [34]. It is an advanced stochastic simulation technique that operates by formulating the original reliability problem into a series of one-dimensional reliability problems solved along randomly sampled lines. This motivates us to combine it with the global NISS to efficiently estimate the failure probability function of rare failure events by only one set of samples. The proposed method is called global imprecise line sampling (GILS).

First, a general derivation of failure probability function $p_f(\boldsymbol{\theta})$ is proposed. Similar to Ref. [27], let $\boldsymbol{\theta}'$ denote an independent random replicate of $\boldsymbol{\theta}$, and its joint PDF is denoted as $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}')$. Then the failure probability function can be represented by the following formulation

$$p_f(\boldsymbol{\theta}) = \int_{g(\boldsymbol{x})\leqslant 0} f_{\mathbf{X}}(\boldsymbol{x}|\boldsymbol{\theta}) f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}') \mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{\theta}' = \int_{g(\boldsymbol{x})\leqslant 0} \frac{f_{\mathbf{X}}(\boldsymbol{x}|\boldsymbol{\theta})}{f_{\mathbf{X}}(\boldsymbol{x}|\boldsymbol{\theta}')} f_{\mathbf{X}}(\boldsymbol{x}|\boldsymbol{\theta}') f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}') \mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{\theta}' . \tag{7}$$

Line sampling is implemented in standard normal space. Let $\boldsymbol{z} = (z_1, \cdots, z_n)$ denote standard normal variables corresponding to $\boldsymbol{x}$, and the transformation relationship of them are expressed as $\boldsymbol{z} = T(\boldsymbol{x}|\boldsymbol{\theta}')$ and $\boldsymbol{x} = T^{-1}(\boldsymbol{z}|\boldsymbol{\theta}')$, which are actually affected by the value of $\boldsymbol{\theta}'$. Then $p_f(\boldsymbol{\theta})$ can be rewritten as

$$p_f(\boldsymbol{\theta}) = \int_{g(T^{-1}(\boldsymbol{z}|\boldsymbol{\theta}'))\leqslant 0} \frac{f_{\mathbf{X}}(T^{-1}(\boldsymbol{z}|\boldsymbol{\theta}')|\boldsymbol{\theta})}{f_{\mathbf{X}}(T^{-1}(\boldsymbol{z}|\boldsymbol{\theta}')|\boldsymbol{\theta}')} \phi_n(\boldsymbol{z}) f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}') \mathrm{d}\boldsymbol{z}\,\mathrm{d}\boldsymbol{\theta}' \tag{8}$$

, where $\phi_n(\cdot)$ is the PDF of the standard normal distribution in *n* dimensions. According to the classical line sampling [7], $\boldsymbol{z}$ can be orthogonally decomposed as $\boldsymbol{z} = \boldsymbol{z}^{\perp} + \boldsymbol{e}_{\alpha}\overline{z}$, where $\boldsymbol{z}^{\perp}$ indicates a vector that belongs to an $(n-1)$ -dimensional hyperplane, $\overline{z}$ indicates a one-dimensional standard normal variable, and $\boldsymbol{e}_{\alpha}$ denotes the normalized important direction in standard normal space. In this paper, we assume that the important direction is fixed and can be chosen considering any value of $\boldsymbol{\theta}$ within its distribution range; details about its selection are discussed in Section 4.3. Hence, $p_f(\boldsymbol{\theta})$ can be further derived as

$$\begin{aligned} p_f(\boldsymbol{\theta}) &= \int_{g(T^{-1}(\boldsymbol{z}^{\perp}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}'))\leqslant 0} \frac{f_{\mathbf{X}}(T^{-1}(\boldsymbol{z}^{\perp}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}')|\boldsymbol{\theta})}{f_{\mathbf{X}}(T^{-1}(\boldsymbol{z}^{\perp}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}')|\boldsymbol{\theta}')} \phi(\overline{z})\mathrm{d}\overline{z}\,\phi_{n-1}(\boldsymbol{z}^{\perp})\mathrm{d}\boldsymbol{z}^{\perp} f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}')\mathrm{d}\boldsymbol{\theta}' \\ &= \int_{g(T^{-1}(\boldsymbol{z}^{\perp}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}'))\leqslant 0} \omega(\boldsymbol{\theta},\boldsymbol{\theta}',\boldsymbol{z}^{\perp},\overline{z})\phi(\overline{z})\mathrm{d}\overline{z}\,\phi_{n-1}(\boldsymbol{z}^{\perp})\mathrm{d}\boldsymbol{z}^{\perp} f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}')\mathrm{d}\boldsymbol{\theta}' \end{aligned} \tag{9}$$

, where $\omega(\boldsymbol{\theta},\boldsymbol{\theta}',\boldsymbol{z}^{\perp},\overline{z})$ denotes the PDF weight function.

Generating N_G samples from both $\phi_{n-1}(\boldsymbol{z}^{\perp})$ and $f_{\boldsymbol{\Theta}'}(\boldsymbol{\theta}')$ simultaneously, which are denoted by $\boldsymbol{S} = (\boldsymbol{z}^{\perp(s)}, \boldsymbol{\theta}^{(s)})(s = 1, \cdots, N_G)$, the Monte Carlo estimator of Eq.(9) is expressed as

$$p_f(\boldsymbol{\theta}) \approx \hat{p}_f(\boldsymbol{\theta}) = \frac{1}{N_G}\sum_{s=1}^{N_G} \int_{g(T^{-1}(\boldsymbol{z}^{\perp(s)}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}^{(s)}))\leqslant 0} \omega(\boldsymbol{\theta},\boldsymbol{\theta}^{(s)},\boldsymbol{z}^{\perp(s)},\overline{z})\phi(\overline{z})\mathrm{d}\overline{z} . \tag{10}$$

Then the estimation of failure probability function reduces to the estimation of the N_G one-dimensional integrals within the failure domain. Taking the *s*-th sample for instance, Figure 1 shows the sketch of estimating the one-dimensional integral associated with the corresponding line. In this figure, the two-dimensional standard normal space is shown in the horizontal plane where the axis (z_1, z_2) rotates to $(\boldsymbol{z}^{\perp}, \overline{z})$, and the vertical axis represents the value of the performance function, denoted as *g*-function in the remaining part of this work. Note that, the transformation formulation $T^{-1}(\boldsymbol{z}|\boldsymbol{\theta}^{(s)})$ associated with the evaluation of the *g*-function is dependent on the value of sample $\boldsymbol{\theta}^{(s)}$, as indicated in the vertical axis of the Figure. From the Figure, it is noted that the line associated with the one-dimensional integral of Eq. (10) passes through the sample $\boldsymbol{z}^{\perp(s)}$ and is parallel to the important direction $\boldsymbol{e}_{\alpha}$. Furthermore, the integration of the function $\omega(\boldsymbol{\theta},\boldsymbol{\theta}^{(s)},\boldsymbol{z}^{\perp(s)},\overline{z})\phi(\overline{z})$ is carried out exclusively over the portion of the

line that lies within the failure domain F. Assuming a monotonic behavior of the performance function along the line, it becomes clear that such integration is carried out between $\nu^{(s)}$ and infinity, where $\nu^{(s)}$ denotes the Euclidean distance between the sample $\boldsymbol{z}^{\perp(s)}$ and the intersection of the line with the limit state function (that is, $g=0$). The distance $\nu^{(s)}$ can be determined by means of the following procedure [34]: select three points along the *s*-th sample line using an appropriate criterion (see, e.g. Ref. [34]) and evaluate the corresponding values of the performance function, as suggested in Figure 1. Then a curve can be drawn by using a second-order polynomial interpolation method for approximating the performance function, and the intersection point between the sample line and failure surface is actually the point of this curve where $g\left(T^{-1}\left(\boldsymbol{z}^{\perp(s)}+\boldsymbol{e}_{\alpha}\nu^{(s)}|\boldsymbol{\theta}^{(s)}\right)\right)=0$.

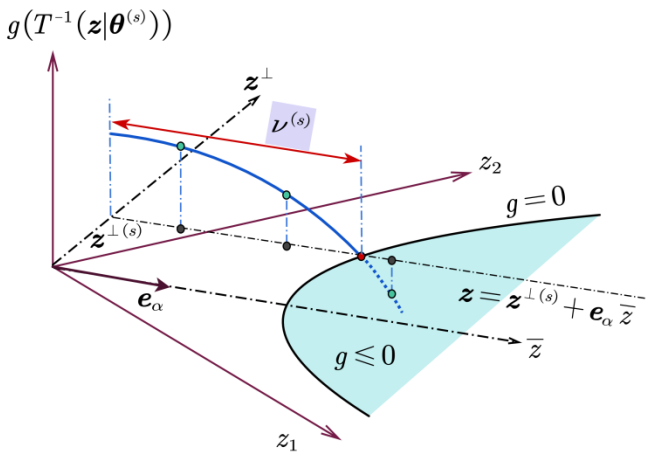

Figure 1 Sketch for estimating the one-dimensional integral corresponding to the *s*-th sample line

Based on the samples of $\left(\boldsymbol{z}^{\perp(s)},\boldsymbol{\theta}^{(s)}\right)$ as well as the estimated value of $\nu^{(s)}$, the probability estimator becomes

$$\hat{p}_f(\boldsymbol{\theta})=\frac{1}{N}\sum_{s=1}^{N}\int_{\nu^{(s)}}^{+\infty}\omega(\boldsymbol{\theta},\boldsymbol{\theta}^{(s)},\boldsymbol{z}^{\perp(s)},\overline{z})\phi(\overline{z})\mathrm{d}\overline{z}=\frac{1}{N}\sum_{s=1}^{N}L^{(s)}(\boldsymbol{\theta}) \tag{11}$$

, where $L^{(s)}(\boldsymbol{\theta})$ denotes the integral of the PDF weight times the standard normal distribution along the corresponding line, which is indeed a function of $\boldsymbol{\theta}$. The variance of the above estimator is

$$var\left(\hat{p}_f(\boldsymbol{\theta})\right)=\frac{1}{N(N-1)}\sum_{s=1}^{N}\left(L^{(s)}(\boldsymbol{\theta})-\hat{p}_f(\boldsymbol{\theta})\right)^2. \tag{12}$$

There may exist two potential disadvantages of using the above estimators: one is that the estimation error (variance) introduced by the PDF weight ω will conspicuously increase when the dimension of distribution parameters increases [35]; the other one is that it's not easy to identify the individual effect of each parameter on the estimation of p_f. Hence, we follow the idea of global NISS reviewed in section 2, and the RS-HDMR is applied here for both improving the estimation accuracy and implementing sensitivity analysis of distribution parameters.

The constant, first-order, and second-order components of RS-HDMR decomposition are derived separately based on the definitions in Eq. (3). For ease of presentation, the derivation details are shown in **Appendix A**, only the estimators and the corresponding variances of these estimators are listed here. The component estimators are

$$\hat{p}_{f0} = \frac{1}{N}\sum_{s=1}^{N}\Phi(-\nu^{(s)})$$
$$\hat{p}_{fi}(\theta_i) = \frac{1}{N}\sum_{s=1}^{N}L^{(s)}(\theta_i) - \hat{p}_{f0} \tag{13}$$
$$\hat{p}_{fij}(\theta_i,\theta_j) = \frac{1}{N}\sum_{s=1}^{N}L^{(s)}(\theta_i,\theta_j) - \hat{p}_{fi}(\theta_i) - \hat{p}_{fj}(\theta_j) - \hat{p}_{f0}$$

, where $\Phi(\cdot)$ indicates the CDF of the standard normal distribution, $L^{(s)}(\theta_i)$ and $L^{(s)}(\theta_i,\theta_j)$ denote functions of θ_i and (θ_i,θ_j), respectively, which are defined as the integrals of their corresponding PDF weight. The detailed formulations are also shown in **Appendix A**. Accordingly, the variances of these estimators are

$$Var(\hat{p}_{f0}) = \frac{1}{N(N-1)}\sum_{s=1}^{N}\left(\Phi(-\nu^{(s)}) - \hat{p}_{f0}\right)^2$$
$$Var(\hat{p}_{fi}(\theta_i)) = \frac{1}{N(N-1)}\sum_{s=1}^{N}\left(L^{(s)}(\theta_i) - \Phi(-\nu^{(s)}) - \hat{p}_{fi}\right)^2 \quad . \tag{14}$$
$$Var(\hat{p}_{fij}(\theta_i,\theta_j)) = \frac{1}{N(N-1)}\sum_{s=1}^{N}\left(L^{(s)}(\theta_i,\theta_j) - L^{(s)}(\theta_i) - L^{(s)}(\theta_j) + \Phi(-\nu^{(s)}) - \hat{p}_{fij}\right)^2$$

Note that the estimators in Eqs. (13) and (14) share the same set of N samples with those used in Eqs. (11) and (12). For the commonly used normal distribution type, closed-form expressions of $L^{(s)}(\boldsymbol{\theta})$, $L^{(s)}(\theta_i)$ and $L^{(s)}(\theta_i,\theta_j)$ in the estimators of failure probability function can be analytically derived, thus no estimation error will be introduced for the calculation of those (line) integrals. The analytical derivations of normal distribution case are provided in **Appendix B**. For lognormal distribution, the analytical derivation process is almost the same as that of the normal case except that the distribution parameters correspond to the lognormal case.

The above component functions allow us to investigate the functional behavior between the parameters and the failure probability, and also to estimate sensitivity indices without the necessity to call the *g*-function. The sensitivity index of uncertain distribution parameters associated with component function $p_{f i_1,\cdots,i_k}(\theta_{i_1},\cdots,\theta_{i_k})$ is defined as

$$S_{i_1,\cdots,i_k} = \frac{var\left(p_{f i_1,\cdots,i_k}(\theta_{i_1},\cdots,\theta_{i_k})\right)}{var(p_f(\boldsymbol{\theta}))}. \tag{15}$$

Since the component functions are derived based on the RS-HDMR method, the interpretation of the above definition is consistent with the widely used Sobol's variance-based sensitivity indices, hence $S_{i_1,\cdots,i_k}$ inherits the well-known properties of Sobol's method. Thus, a large value of $S_{i_1,\cdots,i_k}$ indicates a more influential effect of the uncertainties in $(\theta_{i_1},\cdots,\theta_{i_k})$ on the uncertainty of failure probability.

To sum up, in the computational procedure of the above GILS method, a sample set with size N_G is firstly generated, denoted by $(\boldsymbol{z}^{(s)},\boldsymbol{\theta}^{(s)})(s=1,\cdots,N_G)$. For each sample line, the intersection point between this line and the true failure surface is searched by any efficient numerical method for determining the roots of an equation. Note that the sample $\boldsymbol{\theta}^{(s)}$ is used when the value of $\boldsymbol{z}$ is transformed back to the original space of $\boldsymbol{x}$. If the nonlinearity of the performance function along this line is moderate, then the three-point-second-order polynomial interpolation method described above can provide a satisfactory estimation of the value $\nu^{(s)}$ at the intersection point; otherwise,

four or more points of *g*-function evaluations are needed for each line analyzed. The flowchart of estimating failure probability function by GILS is shown in Figure 2, together with the advanced active learning based method to be developed in the next section.

One notes that the above GILS method is recommended for problems with approximately linear and/or moderately nonlinear *g*-functions. For highly non-linear problems, the required number of *g*-function calls for estimating the intersection point accurately for each line can be significant, and also an increasing number of lines are required for promising the small variation of the GILS estimators. The above two factors jointly result in the high computational cost for highly nonlinear problems. In the next section, an improvement by using the active learning GPR model is introduced for overcoming the above two obstacles.

4. Improvement of GILS by active learning

The proposed GILS method can be applied with only one set of samples, which can greatly relieve the computational burden for estimating the failure probability function. However, it can still be computationally too expensive for complex structures with time-consuming computer simulators. In such a context, surrogate models have been extensively investigated in recent years. Among them, the active learning GPR methodology has received increased attention for estimating failure probability and it has been proved to be an efficient way for structural reliability analysis in the standard probabilistic safety analysis framework. Here, we develop a scheme for injecting the active learning GPR (AGPR) model into the GILS method, and the developed technique is denoted as GILS-AGPR. Before the development of this method, it is helpful to briefly review the classical GPR model.

4.1 Gaussian process regression (GPR)

Consider that a GPR model $\hat{g}(\boldsymbol{x})$ serves as the surrogate for the functional relationship between $\boldsymbol{x}$ and $g(\boldsymbol{x})$. A Gaussian process is fully specified by its mean function $m(\boldsymbol{x})$ and covariance function $c(\boldsymbol{x},\boldsymbol{x}')$ (also known as kernel function), and the GPR model (denoted as $\mathcal{GP}$) can be regarded as a Gaussian process with mean function m and covariance function c, i.e.,

$$\hat{g}(\boldsymbol{x}) \sim \mathcal{GP}(m,c). \tag{16}$$

The mean function $m(\boldsymbol{x})$ can be assumed to be zero, constant, linear or other higher-order polynomial functions, which reflects the analysts' prior knowledge on the *g*-function, and $c(\boldsymbol{x},\boldsymbol{x}')$ represents the covariance between two realizations $\boldsymbol{x}$ and $\boldsymbol{x}'$. The formulations of the mean and covariance functions reflect our prior knowledge of the true performance function. As explained in Ref.[36], any positive definite function can be used as the covariance function, and understanding the properties of particular covariance functions is an important research goal. In this paper, we use the squared exponential kernel function with different scale parameters in each dimension [36].

The next step is to derive the posterior GPR model by updating the prior in the light of the training data set. It is assumed that there is a training sample set $(\boldsymbol{X},\boldsymbol{G})$, where $\boldsymbol{X}$ is a $N_c \times n$ matrix with N_c samples of $\boldsymbol{x}$ contained in each of its rows, $\boldsymbol{G}$ is a column-wise vector with N_c training samples of $g(\boldsymbol{x})$ in which the s-th component

corresponds to the s-th sample site in $\boldsymbol{X}$. Then the hyper-parameters in the above mean function $m(\boldsymbol{x})$ and covariance function $c(\boldsymbol{x},\boldsymbol{x}')$ can be estimated based on the maximum likelihood method. After these hyper-parameters are evaluated, the posterior GPR model $\hat{g}(\boldsymbol{x})$ can be utilized for prediction at any new realization $\boldsymbol{x}$. Since the posterior is also a Gaussian process, $\hat{g}(\boldsymbol{x})$ predicted at a new realization $\boldsymbol{x}$ can be regarded as a Gaussian variable whose expectation and prediction variance are expressed as

$$\hat{\mu}_g(\boldsymbol{x}) = m(\boldsymbol{x}) + c(\boldsymbol{x},\boldsymbol{X})^{\mathrm{T}} C^{-1}(\boldsymbol{G} - m(\boldsymbol{X})) \tag{17}$$

and

$$\hat{\sigma}_g^2(\boldsymbol{x}) = c(\boldsymbol{x},\boldsymbol{x}) - c(\boldsymbol{x},\boldsymbol{X})^{\mathrm{T}} C^{-1} c(\boldsymbol{x},\boldsymbol{X}) \tag{18}$$

, where $c(\boldsymbol{x},\boldsymbol{X})$ is a column-wise vector, the s-th component of which is the covariance between $\boldsymbol{x}$ and the s-th row-wise training sample in $\boldsymbol{X}$, C is a $N_c \times N_c$ matrix with the (s,r)-th component being the covariance between the s-th and r-th training samples in $\boldsymbol{X}$, and $m(\boldsymbol{X})$ is a column-wise vector with the s-th component being the value of mean function located at the s-th training sample.

We can see that the expectation $\hat{\mu}_g(\boldsymbol{x})$ is composed of two terms: the first term reflects prior information of $g(\boldsymbol{x})$ and the second term contains the information learned from training data. And the variance $\hat{\sigma}_g^2(\boldsymbol{x})$ equals to the prior variance minus a combination of covariance matrix between the new realization and all training samples, indicating that larger training data size will lead to a reduction of $\hat{\sigma}_g^2(\boldsymbol{x})$. In fact, this variance measures the prediction accuracy of the posterior GPR model at the location of $\boldsymbol{x}$, and this property is of major benefit for establishing the active learning algorithm in the next subsection.

4.2 GILS-AGPR algorithm

Recently, we developed an active learning line sampling algorithm for standard probabilistic reliability analysis [37], which is extremely efficient for problems with rare failure events, and the proposed GILS-AGPR algorithm can be regarded as an extension of this method from the probabilistic framework into imprecise probability. The basic idea is to actively construct a computationally less expensive GPR model to approximate the true failure surface of performance function by learning from both the space of input variables and the distribution parameter space. After the GPR model is properly trained, all the intersection distances $\nu^{(s)}$ between the candidate sample lines and the failure surface are accurately estimated at the same time. Consequently, the failure probability function, as well as its HDMR component functions, can be directly estimated based on those distances provided by the GPR model without calling the performance functions.

To make the relationship between the proposed GILS and GILS-AGPR procedures more clear, a flowchart of both the proposed methods is given in Figure 2. For GILS-AGPR, the most important element is the learning function, which serves as an engine of the algorithm to adaptively select training points. In this contribution, we use the learning function originally devised in our previous work [37], which is formulated as:

$$\eta(\boldsymbol{x}) = \int_{-\epsilon+\hat{\mu}_g(\boldsymbol{x})}^{\epsilon+\hat{\mu}_g(\boldsymbol{x})} \phi\big(\hat{g}(\boldsymbol{x}) | \hat{\mu}_g(\boldsymbol{x}), \hat{\sigma}_g^2(\boldsymbol{x})\big) \mathrm{d}\hat{g}(\boldsymbol{x}) \tag{19}$$

, where $\phi\left(\cdot|\hat{\mu}_g(\boldsymbol{x}),\hat{\sigma}_g^2(\boldsymbol{x})\right)$ indicates the normal PDF with the mean and variance being the posterior mean $\hat{\mu}_g(\boldsymbol{x})$ and posterior variance $\hat{\sigma}_g^2(\boldsymbol{x})$ of the GPR model, and ϵ indicates a pre-specified error tolerance determining the accuracy of estimating the interaction point for each line. Once the intersection point for one line is specified, e.g., as $\boldsymbol{x}^*$, based on the trained GPR model $\hat{g}(\boldsymbol{x})$, then $\eta(\boldsymbol{x}^*)$ measures the probability that the real value $g(\boldsymbol{x}^*)$ is covered by the small interval $[-\epsilon,\epsilon]$. Obviously, $\eta(\boldsymbol{x}) \in [0,1]$, and high values of $\eta(\boldsymbol{x}^*)$ indicate high accuracy in estimating the intersection point for this line. The above property of the learning function makes it possible to accurately estimate the intersection points for all lines used in the GILS algorithm with a small number of training samples, and thus g-function calls, even for highly nonlinear problems. For an in-depth interpretation of the learning function from the perspective of geometry and the specification of the error tolerance ϵ, one can refer to our previous work [37].

The steps of the GILS-AGPR algorithm shown in Figure 2 are explained as follows in detail.

Step 1: Initialize the total number of candidate sample lines N_A, the number of initial lines N_0 for training the initial GPR model, the threshold η^*, the error tolerance ϵ, the initial training sample set $\boldsymbol{S} = (\boldsymbol{X}, \boldsymbol{G})$. Let $N_c = 0$. Compute the important direction $\boldsymbol{e}_\alpha$ using any appropriate procedure [34].

Step 2: Generate N_A candidate samples $(\boldsymbol{z}^{(s)}, \boldsymbol{\theta}^{(s)})$ $(s = 1, \cdots, N_A)$. For each sample, generate a candidate line that is parallel to the important direction $\boldsymbol{e}_\alpha$. Then randomly select N_0 initial lines from the N_A candidate lines.

Step 3: For each initial line, estimate the value of $\nu^{(s)}$ at the intersection point by solving the equation $g(T^{-1}(\boldsymbol{z}^{\perp(s)} + \boldsymbol{e}_\alpha \overline{z}|\boldsymbol{\theta}^{(s)})) = 0$. Note that the estimation of the s-th line is based on the sample pair $(\boldsymbol{z}^{(s)}, \boldsymbol{\theta}^{(s)})$, which means the sample $\boldsymbol{\theta}^{(s)}$ is only used for the s-th line. A second-order polynomial interpolation constructed considering three evaluation points (that is, $\overline{z}$ is specified with three constant value c_1, c_2, c_3) is utilized for solving this equation. In turn, such a procedure involves three g-function evaluations for each line. Add $T^{-1}(\boldsymbol{z}^{\perp(s)} + \boldsymbol{e}_\alpha c_r|\boldsymbol{\theta}^{(s)})$ into $\boldsymbol{X}$ and $g(T^{-1}(\boldsymbol{z}^{\perp(s)} + \boldsymbol{e}_\alpha c_r|\boldsymbol{\theta}^{(s)}))$ into $\boldsymbol{G}$, where $r = 1, 2, 3$.

Step 4: Evaluate the g-function value for each intersection point found in the previous step, add $T^{-1}(\boldsymbol{z}^{\perp(s)} + \boldsymbol{e}_\alpha \nu^{(s)}|\boldsymbol{\theta}^{(s)})$ into $\boldsymbol{X}$ and $g(T^{-1}(\boldsymbol{z}^{\perp(s)} + \boldsymbol{e}_\alpha \nu^{(s)}|\boldsymbol{\theta}^{(s)}))$ into $\boldsymbol{G}$. So far, the training data set includes $3N_0$ samples from **Step 3**, N_0 samples at the intersection point evaluated at this step, as well as $N_{\boldsymbol{e}_\alpha}$ g-function evaluations for determining the important direction as described in **Step 1**. Thus, the size of the training data set is $N_c = N_{\boldsymbol{e}_\alpha} + 3N_0 + N_0$.

Step 5: Train or update the GPR model $\hat{g}(\boldsymbol{x})$ with the training sample set $\boldsymbol{S}$.

Step 6: For all the N_A candidate sample lines (including N_0 initial lines), search for the intersection point between those lines and the failure surface specified by the GPR model $\hat{g}(\boldsymbol{x})$ so that $\hat{\mu}_g(T^{-1}(\boldsymbol{z}^{\perp(s)} + \boldsymbol{e}_\alpha \nu^{(s)}|\boldsymbol{\theta}^{(s)})) = 0$ is satisfied. Because of the smoothness of the GPR model for the line search, any advanced numerical rooting method can be used to reach an accurate calculation of $\nu^{(s)}$.

Step 7: Predict the posterior variance $\hat{\sigma}_g(T^{-1}(\boldsymbol{z}^{\perp(s)}+\boldsymbol{e}_\alpha\nu^{(s)}|\boldsymbol{\theta}^{(s)}))$ at the N_A intersection points; note that the posterior means at these points are all equal to zero. Compute the value of the learning function $\eta^{(s)}$ for each intersection point to judge the accuracy of the intersection point estimated by the GPR model. High values of $\eta^{(s)}$ (close to 1) indicate a more accurate estimation of that intersection point.

Step 8: Find the minimum value $\eta_{\min}=\min_{s=1}^{N_A}(\eta^{(s)})$, if $\eta_{\min}<\eta^*$, then add this intersection point corresponding to $\eta_{\min}$ into training sample set $\boldsymbol{X}$, evaluate the corresponding *g*-function value and add it into $\boldsymbol{G}$, let $N_c=N_c+1$, go to back to **Step 5**. Otherwise, go to **Step 9**.

Step 9: Estimate the constant component of failure probability function $\hat{p}_{f0}$ with the estimator in Eq. (13) as well as its variance in Eq. (14), compute the coefficient of variation (COV) with $\sqrt{var(\hat{p}_{f0})}/\hat{p}_{f0}$. If the COV is larger than 0.05, add more candidate lines with $N_A=N_A+N_{add}$, where commonly, $N_{add}=50\sim200$ is recommended. Then go back to **Step 6**. If the COV is smaller than 0.05, a converged estimation of $\hat{p}_{f0}$ has been achieved, finish the iteration procedure.

Till now, the GPR model $\hat{g}(\boldsymbol{x})$ has been properly trained to approximate the true failure surface by exploring the candidate samples of both input and distribution parameters. In the following task of estimating the failure probability function, the proposed GILS-AGPR algorithm shares the same estimators with GILS method as proposed in section 3, in which no more *g*-function evaluations are involved. The values of $\boldsymbol{z}^{\perp(s)}$, $\boldsymbol{\theta}^{(s)}$ and $\nu^{(s)}$ $(s=1,\cdots,N_G$ or $N_A)$ for all the sample lines are extracted, then the failure probability function and the component failure probability functions are calculated employing the estimators in Eq. (13). In the meantime, their corresponding variances are computed using the estimators in Eq. (14) to judge their quality.

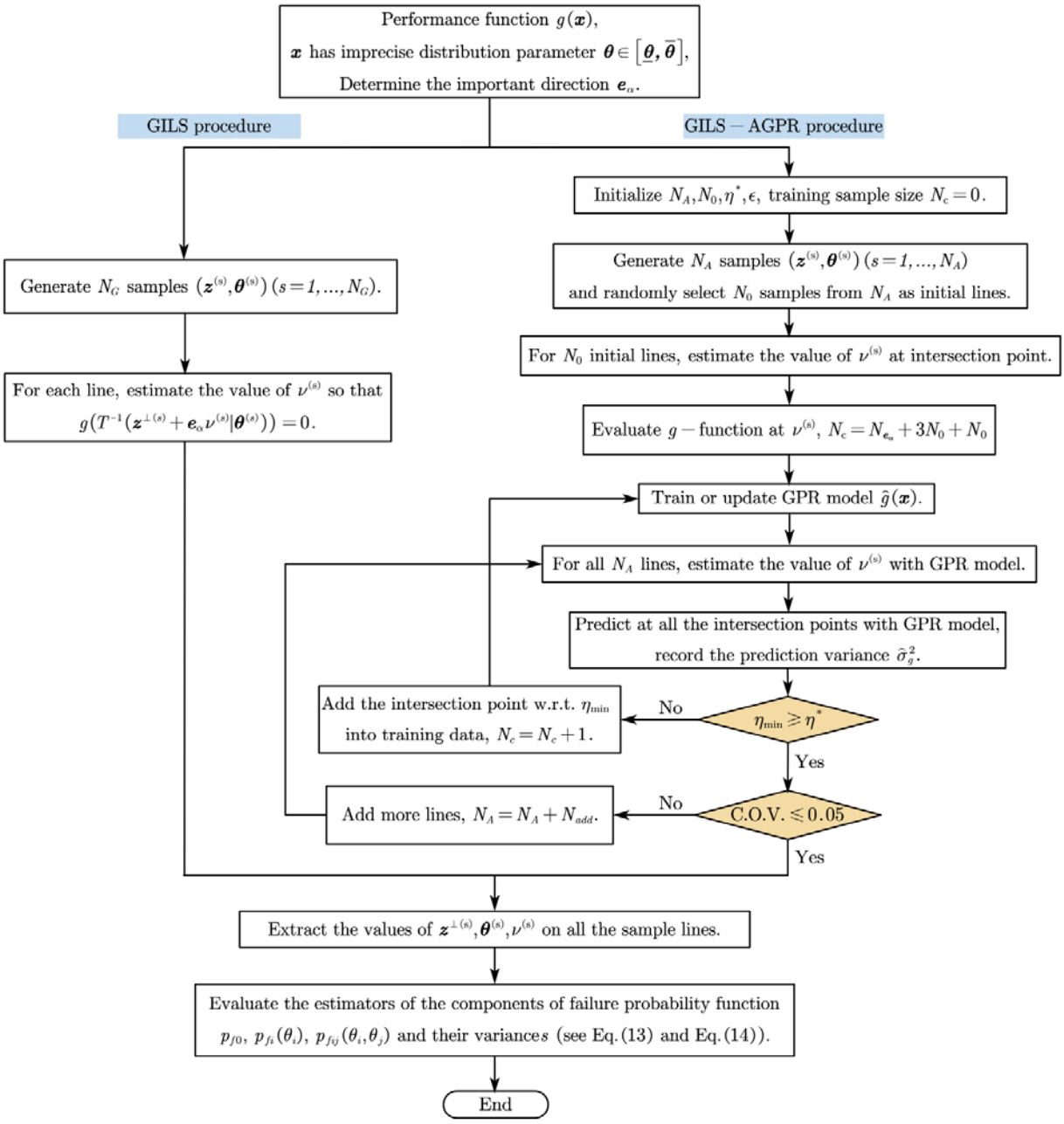

Figure 2 Flowchart of the proposed GILS and GILS-AGPR algorithm.

4.3 Discussions of GILS-AGPR algorithm

This subsection provides some discussions on the details of implementing the GILS-AGPR algorithm for better use of it in practical applications.

■ **The important direction**

In this paper, the important direction $\boldsymbol{e}_\alpha$ in the standard normal space is constant, and it is determined by fixing imprecise distribution parameters $\boldsymbol{\theta}$ at a pre-specified point so that the traditional methods for searching an important direction can be directly used. Generally, $\boldsymbol{\theta}$ can be fixed at any point within the interval of $[\underline{\boldsymbol{\theta}}, \overline{\boldsymbol{\theta}}]$, and it

is found that the generated importance direction may vary slightly while the different fixed point is utilized, and no thumb rule is available for selecting the optimal fixed point which produces the best important direction. Fortunately, as revealed by our previous work [37], compared with the classical line sampling utilized in GILS, the performance of the active learning line sampling is less sensitive to the important direction. Even when the specified important direction is not the optimal one, the active learning line sampling, and thus also the GILS-AGPR algorithm will adaptively produce more lines to promise the estimation accuracy, without a huge increment of *g*-functions, since the intersection points can be efficiently and accurately learned by the adaptive GPR model. In all the test examples of this paper, the important direction is generated by fixing $\boldsymbol{\theta}$ at the middle point of its support.

For a linear or moderately nonlinear performance function, the important direction can be determined by many mature techniques. One straightforward way is to firstly search for the design point (or the most probable point, MPP) using the strategy of Advanced Second Moment (ASM) method [38] or the Hasofer-Lindt-Rackwitz-Fiessler (HLRF) algorithm [39][40] and then the important direction points from the origin to that design point. Another less computationally expensive way is to use the negative gradient of the performance function at the original point in the standard normal space as the important direction. For highly nonlinear problems, only the first scheme is recommended since the second scheme may result in a large error. In this paper, the ASM method is utilized for the case studies.

- **Training the GPR model**

As mentioned in subsection 4.1, the formulations of mean function $m(\boldsymbol{x})$ and covariance function $k(\boldsymbol{x},\boldsymbol{x}')$ need to be specified before training. Usually zero or constant mean function is applicable for linear or moderately nonlinear performance function, and linear or quadratic polynomial mean function is applicable for highly nonlinear cases. Note that, if the nonlinearity is high, the number of initial lines N_0 should be larger accordingly. In this paper, the squared exponential kernel [36] is used as the covariance function, and the Matlab function "fitrgp" in the Matlab Statistic and Machine Learning Toolbox is utilized for training the GPR model.

- **The role of GPR model in estimating failure probability function**

The active learning GPR model plays a role to update the simulation process for efficiently figuring out the site of $\boldsymbol{x}$ with the maximum contribution that improves the accuracy of failure probability estimation. The well-trained model itself provides a good approximation for the intersection points of each candidate line with the true failure surface, but cannot guarantee the approximate accuracy to the *g*-function across the full support of $\boldsymbol{x}$. Hence, the candidate samples $(\boldsymbol{x},\boldsymbol{\theta})$ used in the GILS-AGPR procedure should be definitely the same set of samples used to estimate the component failure probability functions.

- **Improvements of GILS-AGPR compared to GILS**

For GILS, one needs to take several (at least three) *g*-function calls for estimating the interaction point for each line, and for highly nonlinear problems, the required number of interpolation points, and thus the g-function calls, will increase largely to ensure the estimation accuracy. However, the GILS-AGPR algorithm can accurately estimate the intersection points by calling the GPR model which has been properly trained for predicting the intersection points, thus the total number of g-function calls for estimating the intersection points for all lines is much smaller than that of the GILS method. This benefit comes from the active learning scheme and the spatial correlation

information revealed by the GPR model. The GPR model for accurately approximating the failure surface also benefits from the high efficiency of the one-dimensional search of the line sampling scheme. Besides, increasing the initial number of lines N_A of GILS-AGPR will not directly increase the number of *g*-function evaluations, as one can set a much larger sample size N_A than the sample size N_G in the GILS procedure, so as to produce a high estimation accuracy for highly nonlinear problems without considerably increasing the computational cost. For highly nonlinear problems, the accurate estimation of the important direction may require more *g*-function calls. However, these *g*-function calls can be reused for training the initial GPR model, thus will not be wasted. While the important direction is estimated with poor quality, the GILS-AGPR algorithm will adaptively increase the number of lines for promising the estimation accuracy without significantly increasing the required number of *g*-function calls. The above features of GILS-AGPR make it extremely efficient for even highly nonlinear problems.

■ **Comments on estimation errors**

The error sources for estimating the failure probability function include (i) the statistical error due to the limited number of lines, which can be controlled with enough lines for training, (ii) the truncation error caused by HDMR truncation, which can be measured by sensitivity indices, and (iii) the numerical error for estimating the intersection surface, controlled by carefully selecting the threshold η^* and the error tolerance ϵ. With all these three types of errors being properly addressed, the method is believed to be of wide applicability.

5. Case studies

5.1 Analytical example

Consider a simple analytical example where the failure surface associated with the performance function is a parabola,

$$g(x_1,x_2) = \beta - x_1 + \gamma x_2^2 \tag{20}$$

, where $x_1 \sim N(\mu_1,\sigma_1^2)$, $x_2 \sim N(\mu_2,\sigma_2^2)$. The constant β controls the failure probability level and γ controls the degree of nonlinearity of the performance function (which in turn also affects the failure probability). Let $\beta = 3.5$ and $\gamma = 0.2$ such that the performance function is moderately nonlinear, and the failure probability is rather small. The failure probability function can be analytically calculated with the expression below

$$p_f(\mu_1,\mu_2,\sigma_1,\sigma_2) = \int_{-\infty}^{+\infty} \Phi\left(-\frac{\beta-\mu_1}{\sigma_1} - \frac{\gamma}{\sigma_1}(\mu_2+\sigma_2 z)^2\right)\phi(z)\mathrm{d}z. \tag{21}$$

The imprecise distribution parameters are defined as $\mu_1 \in [-0.5, 0.5]$, $\mu_2 \in [-0.5, 0.5]$, $\sigma_1 \in [0.6, 1]$, $\sigma_2 \in [0.8, 1]$.

According to the analytical formulation of the performance function, the important direction is assumed to be fixed at $\boldsymbol{e}_\alpha = (1, 0)^T$. The proposed GILS and GILS-AGPR procedures are both implemented with this important direction, and the results are compared with the analytical solutions for illustrating the effectiveness of these two methods. For implementing the GILS-AGPR method, the number of initial lines used for training the initial GPR model is set to be $N_0 = 4$, the total number of candidate sample lines is set to be 3,000.

The training process of the GPR model constructed within the GILS-AGPR procedure is schematically shown in Figure 3. It can be seen that the GPR failure surface matches well with the true failure surface, and the 17 training samples are also plotted. Figure 4 shows the plots of the first-order component functions estimated by the proposed GILS and GILS-AGPR procedures, as well as the analytical results with the help of Eq. (21). For the GILS procedure, $N_G = 200$ lines are used to reach a converged estimation, and the total number of *g*-function evaluations is 600. It is shown that all the first-order plots with the two proposed methods have good consistency with the analytical results, and in the second row of this figure, the small standard deviations (SDs, estimated by Eq. (14)) also indicate good precision. Moreover, compared with the GILS method, the estimations of the GILS-AGPR method is more accurate since its plots are much closer to the analytical plots, and the corresponding SDs are much smaller. This is because the number of sample lines N_A is much larger than that involved in the GILS method, although the total number of *g*-function evaluations of GILS-AGPR is only 17. It indicates that the GILS-AGPR method can overcome the deficiency of a slow convergence speed of the proposed estimators of $p_f(\boldsymbol{\theta})$, and provides an accurate estimation with much less computational cost. Additionally, the SDs of GILS-AGPR remain at a low value within the whole uncertain region, demonstrating a better global performance.

The first- and second-order sensitivity indices of the uncertain distribution parameters as well as the constant component p_{f0} are listed in Table 1, with the SDs shown in the superscripts. Higher-order sensitivity indices are all close to zero and are omitted here. Note that all the sensitivity results related to either GILS or GILS-AGPR procedure don't require additional *g*-function evaluations since the component functions can be regarded as explicitly known. It shows that the results of the two proposed methods are in good agreement with the analytical results, illustrating the effectiveness of the proposed methods for sensitivity analysis. The GILS-AGPR method shows a relatively better performance than the GILS method, which is consistent with the conclusion drawn from the plots in Figure 4. It is interesting that all the indices relating to μ_2 or σ_2 are close to zero, which means the distribution parameters associated with x_2 are non-influential for estimating failure probability. On the other hand, the first-order index of σ_1 is almost twice the value of μ_1, so σ_1 is the most influential distribution parameter on the uncertainty of failure probability. The second-order index of (μ_1, σ_1) is close to 0.39, showing a strong interaction effect between μ_1 and σ_1 in the failure probability function.

Based on the estimated influential component functions, the lower and upper bounds of the failure probability is calculated, and the results are compared in the fourth row of Table 1. As can be seen, both methods compute the upper bounds with satisfactory accuracy when compared to the analytical result, but the lower bound is not correctly computed by both methods due to the extremely small value, i.e., 4.99e-12. This indicates that the developed methods are not suitable for estimating the lower bound when the span of the failure probability covers several orders of magnitude. In this example, the exact value of the lower bound is 4.99e-12, however, the SDs of the estimate lower bounds by the two methods are both much higher than this value, indicating the estimates are not accurate. Fortunately, in real-world engineering applications, the upper bound is of most concern as it provides a conservative measure of structural safety, which is of great significance.

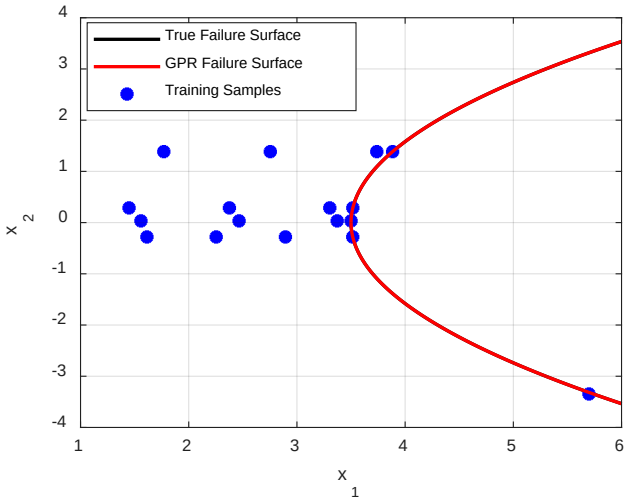

Figure 3 Training details of GPR model in GILS-AGPR procedure for the analytical example.

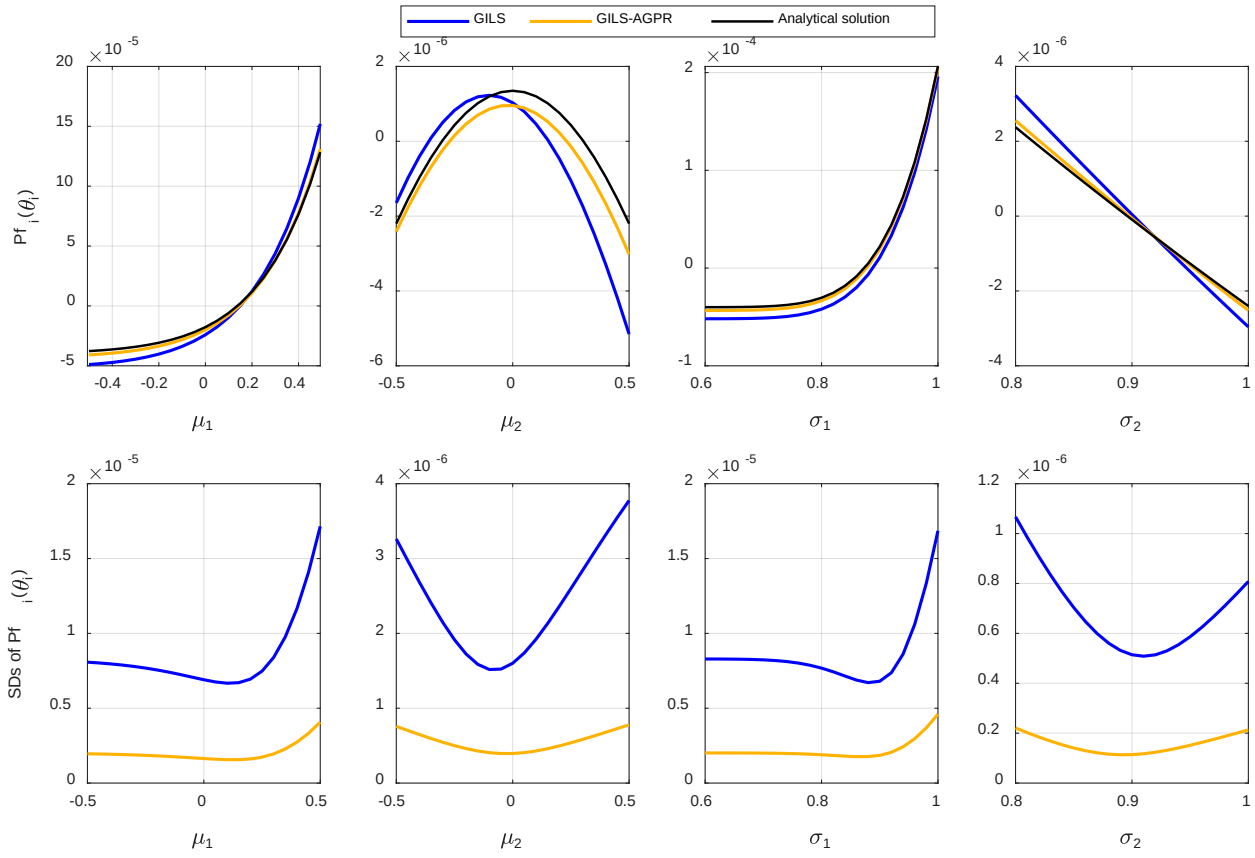

Figure 4 Plots of the first-order component failure probability functions and their corresponding SDs in the analytical example.

Table 1 First- and second-order sensitivity indices for the analytical example. Standard deviations of estimates are shown as superscripts in parenthesis.

Methods		GILS	GILS-AGPR	Analytical results
Number of *g*-function calls		600	17	/
p_{f0}		5.19e-5$^{(8.3e-6)}$	4.34e-5$^{(2.0e-6)}$	4.39e-5
Lower bound of p_f		-3.11e-6$^{(8.3e-6)}$	3.17e-6$^{(4.0e-6)}$	4.99e-12
Upper bound of p_f		9.24e-4$^{(2.1e-5)}$	9.37e-4$^{(9.5e-6)}$	9.88e-4
S_i	μ_1	0.2658$^{(0.0067)}$	0.2131$^{(0.0014)}$	0.2066
	μ_2	0.0003$^{(0.0005)}$	0.0001$^{(0.0000)}$	0.0001
	σ_1	0.3665$^{(0.0068)}$	0.3971$^{(0.0009)}$	0.3928
	σ_2	0.0003$^{(0.0001)}$	0.0002$^{(0.0000)}$	0.0001
S_{ij}	(μ_1,μ_2)	0.0003$^{(0.0005)}$	0.0001$^{(0.0000)}$	0.0001
	(σ_1,σ_2)	0.0005$^{(0.0000)}$	0.0005$^{(0.0000)}$	0.0005
	(μ_1,σ_1)	0.3651$^{(0.0071)}$	0.3883$^{(0.0005)}$	0.3901
	(μ_1,σ_2)	0.0004$^{(0.0001)}$	0.0003$^{(0.0000)}$	0.0003
	(μ_2,σ_1)	0.0009$^{(0.0005)}$	0.0003$^{(0.0000)}$	0.0003
	(μ_2,σ_2)	0.0000$^{(0.0000)}$	0.0000$^{(0.0000)}$	0.0000

5.2 Parallel system

Considering a highly nonlinear parallel system (as proposed in Ref. [41]) to investigate the performance of the proposed methods, and the performance function is defined as

$$g(x_1,x_2)=\max\begin{cases}2-x_2+\exp(-0.1x_1^2)+(0.2x_1)^4\\ 4.5-x_1x_2\end{cases} \tag{22}$$

, where x_1 and x_2 are two independent normal variables following $x_1\sim N(\mu_1,1^2)$ and $x_2\sim N(\mu_2,1^2)$, respectively. The two imprecise distribution parameters μ_1 and μ_2 both range within the intervals $[-0.5,0.5]$. The design point is featured as $\boldsymbol{x}_p=(1.6148,2.7806)^{\mathrm{T}}$ when considering given value $\mu_1=\mu_2=0$.

The important direction is $\boldsymbol{e}_\alpha=(0.5030,0.8643)^T$, which is set according to the given design point. The two-dimensional failure surface, the design point as well as all the training samples generated in GILS-AGPR procedure are shown in Figure 5, and the training sample size is $N_c=51$. We can see that most of the training samples are located close to the true failure surface, hence the failure surface constructed by GPR model comes out as a good approximation of the true failure surface by a small set of training samples.

Figure 6 shows the plots of the first-order components $p_{f1}(\mu_1)$ and $p_{f2}(\mu_2)$ estimated by GILS and GILS-AGPR procedures, and estimations of double-loop crude Monte Carlo estimations (denoted as DL) are also plotted in the same figure. For GILS-AGPR method, set the size of sample lines as $N_A=3000$ in order to get estimations with small SDs, although its total number of *g*-function evaluations is 51. In GILS procedure, $N_G=1000$ sample lines

are used so as to produce comparable results with GILS-AGPR. Because of the high nonlinearity of the parallel system, advanced optimization methods, specifically, the Matlab function "fsolve" is adopted to search an accurate intersection point along each line, so the number of *g*-function evaluations is 7954. In the first row of Figure 6, it is shown that the plots have a good consistency with the DL method, indicating the effectiveness and high efficiency of the proposed methods in estimating first-order component functions. In the second row, the small SDs indicate the small derivations of the estimations. Moreover, both μ_1 and μ_2 have a monotonic increasing relationship with failure probability. Table 2 lists the first- and second-order sensitivity indices of μ_1 and μ_2, which are regarded as the by-product of proposed methods. We can see that indices estimated by both GILS and GILS-AGPR methods match well with the DL method with small SDs. Since the first-order index of μ_2 is more than twice of the first-order index of μ_1, μ_2 shows an obviously influential effect on the uncertainty of failure probability. Based on the estimated influential component functions, the bounds of the failure probability are computed by the two methods, and the results are compared in Table 2. As can be seen, the upper bounds are accurately estimated by both methods are accurate, while for the lower bound, the SDs are even larger than the mean estimates although the mean estimates are both close to the reference solution.

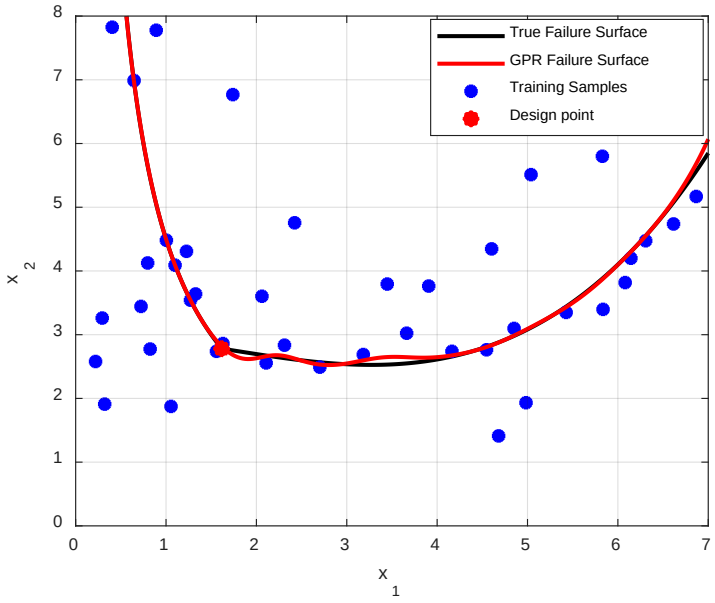

Figure 5 Training details of GILS-AGPR procedure for the parallel system

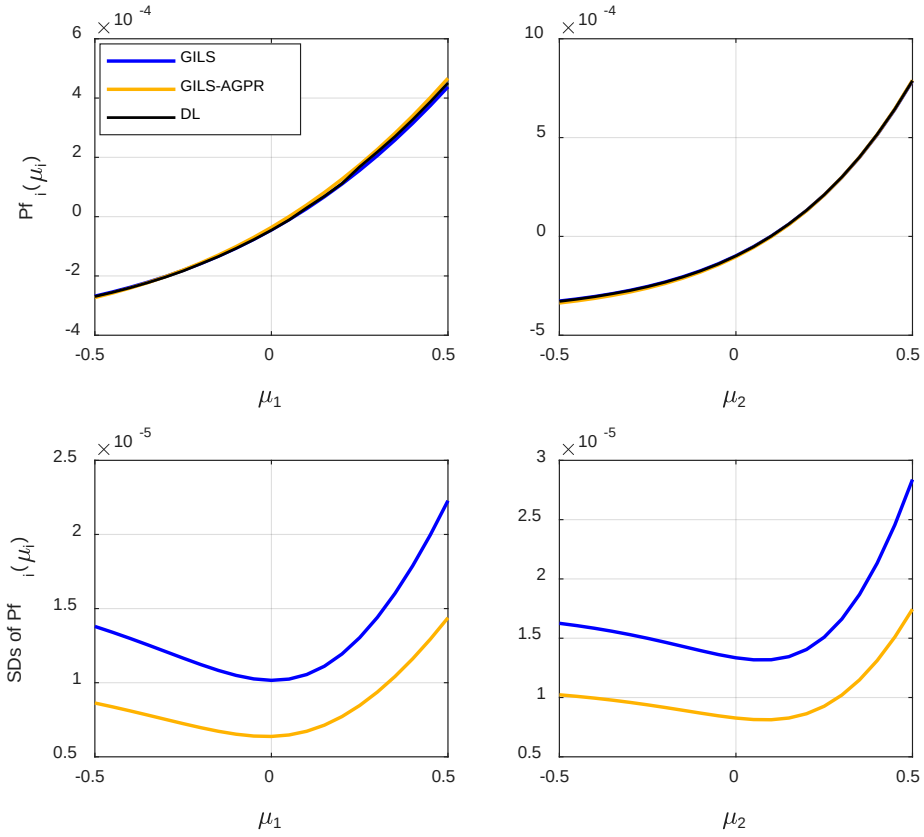

Figure 6 Plots of the first-order component failure probability functions and their corresponding SDs for the parallel system.

Table 2 Results of first- and second-order sensitivity indices for the parallel system. Standard deviations of estimates are shown as superscripts in parenthesis.

Methods		GILS	GILS-AGPR	DL
Function evaluations		7954	51	/
p_{f0}		3.82e-4 (1.7e-5)	3.94e-4(1.1e-5)	3.78e-4
Lower bound of p_f		1.59e-5(3.0e-5)	1.70e-5(1.9e-5)	1.61e-5
Upper bound of p_f		2.57e-3(4.9e-5)	2.54e-3(3.2e-5)	2.44e-3
S_i	μ_1	0.2537(0.0011)	0.2660(0.0004)	0.2637
	μ_2	0.5879(0.0016)	0.5731(0.0006)	0.5832
S_{ij}	(μ_1,μ_2)	0.1584(0.0008)	0.1609(0.0003)	0.1531

5.3 A simplified riveting model

In the aircraft industry, riveting is the most common approach to assemble sheet metal parts such as the wing covering, so the quality of rivets in the riveting process is quite important for the safety of the whole aircraft structure. Since the true riveting process is complex, a simplified headless rivet model [42] is introduced here for illustrating the proposed method. Here the riveting process is simply divided into three states, as shown in Figure 7. In the initial

state, the rivet is without any deformation and the diameter is denoted as d. Then the rivet is punched to the intermediate state in which the clearance between the rivet and the hole is zero. At last, the rivet heads are further punched to the final state and the rivet heads are formed. In this process, the maximum squeeze stress $\sigma_{\max}$ of the rivet head in the *y*-direction is one of the most significant factors affecting the quality of rivets, and the rivet will crack when $\sigma_{\max}$ exceeds the allowable squeeze strength.

For establishing the mathematical formulation between the squeeze stress and the geometric parameters, assumptions of some ideal conditions during the riveting process are required, namely: a. The hole diameter keeps unchanged; b. The whole rivet volume stays the same; c. The rivet head in the final state has a cylindrical shape; d. The material of the rivet is isotropic. Based on the power hardening theory, the maximum squeeze stress in the *y*-direction can be expressed as $\sigma_{\max} = K(\varepsilon_y)^{n_{\text{SHE}}}$, where K is the strength coefficient, n_{SHE} is the strain hardening exponent of the rivet material, and ε_y is the true strain in the *y*-direction of the rivet head. In the simplified model, ε_y is composed of two parts, i.e., $\varepsilon_y = \varepsilon_{y1} + \varepsilon_{y2}$, and $\varepsilon_{y1} = \ln(h/h_1)$ is caused by punching from the initial state to intermediate state, $\varepsilon_{y2} = \ln\left(\frac{h_1 - t}{2H}\right)$ is caused by punching from intermediate state to the final state. Based on those above-mentioned assumptions, one can obtain the performance function associated with the maximum squeeze stress of the riveting process:

$$g = \sigma_{sq} - K\left(\ln\frac{d^2h - D_0^2 t}{2Hd^2}\right)^{n_{\text{SHE}}}. \tag{23}$$

In this paper, the material of the rivet is 2017-T4, its strain hardening exponent is $n_{\text{SHE}} = 0.15$, and the allowable squeeze strength is $\sigma_{sq} = 580$ MPa. The height of the rivet head is assumed to be constant, i.e., $H = 2.2$ mm. The other five input variables are all supposed to follow (truncated) normal distribution $N(\mu, \sigma^2)$, where all σ are set to be uncertain parameters, as listed in Table 3.

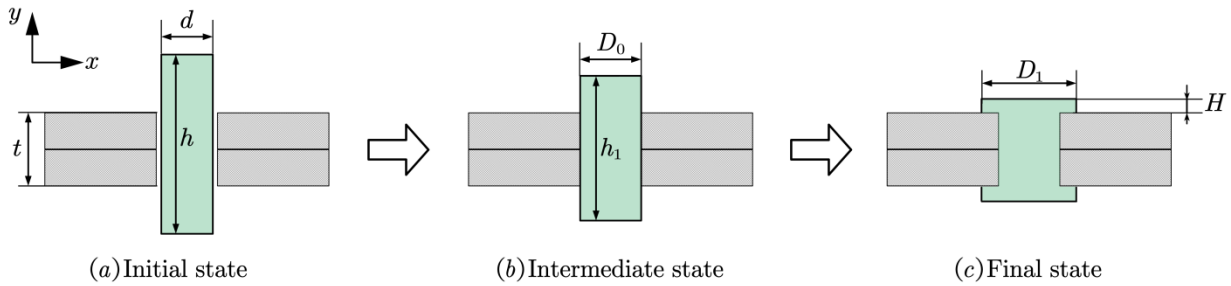

Figure 7 Simplified riveting process.

Table 3 Probabilistic description of five random input variables in the simplified riveting model.

Variables	Description	Distribution	μ	σ
d (mm)	The rivet diameter in initial state.	(truncated) Normal	4.45	[0.07,0.09]
h (mm)	The rivet height in initial state.	(truncated) Normal	18	[0.3,0.36]
D_0 (mm)	The rivet diameter in intermediate state.	(truncated) Normal	5.2	[0.21,0.26]

t (mm)	The thickness of the two sheets.	(truncated) Normal	5	[0.32,0.4]
K (MPa)	The strength coefficient.	(truncated) Normal	547.5	[8.8,11]

The important direction is determined considering that the imprecise parameters are fixed at their midpoint values of the intervals. ASM method [36] method is used to search the MPP with a total number of 18 *g*-function evaluations, and the important direction is specified as $\boldsymbol{e}_{\alpha} = (0.1596, 0.2145, -0.4011, -0.3195, 0.8158)^T$ in the standard normal space. Then GILS and GILS-AGPR procedures are applied on the riveting model. For the GILS method, the size of the sample lines is $N_G = 400$; for the GILS-AGPR method, the training sample size is $N_c = 45$ for establishing the GPR model of the performance function, and the number of sample lines is $N_A = 2000$. The first-order component failure probability functions as well as the SDs are plotted in Figure 8, and DL Monte Carlo method is also utilized to provide comparative results. It is shown that all the plots are consistent with the comparative plots; GILS-AGPR shows a better performance than GILS especially when the values of parameters are far from the midpoint values. As we know, the midpoint values are chosen to determine the important direction; there may be slower convergence speed for the estimators in those regions far away from those midpoint values, as the important direction becomes less representative in those regions. Nonetheless, in such a situation GILS-AGPR can still provide estimates with high precision, as the number of lines involved in the analysis procedure N_A can be large, without affecting the overall numerical analysis cost. The latter assertion can be verified by verifying the small SDs of GILS-AGPR in the second row of plots in Figure 8.

The results of the bounds of the failure probability are reported in Table 4, and it is shown that both methods produce accurate estimates for the upper bound. For the lower bound, the coefficients of variation of the estimates can be latrger due to the small value of mean estimate, but comparably, the accuracy is also acceptable. Table 4 also provides the first- and second-order sensitivity indices of the distribution parameters, and those second-order indices close to zero are omitted here. We can see that all the indices match well with each other among the three methods, indicating converged and accurate estimations of the first- and second-order component functions. Compared with the GILS method, the results of GILS-AGPR also show better consistency with the DL method. According to the values of indices, σ_K is the most influential parameter and all the remaining first-order and second-order indices are quite small, hence the imprecision of these remaining parameters has little effect on the uncertainty of failure probability.

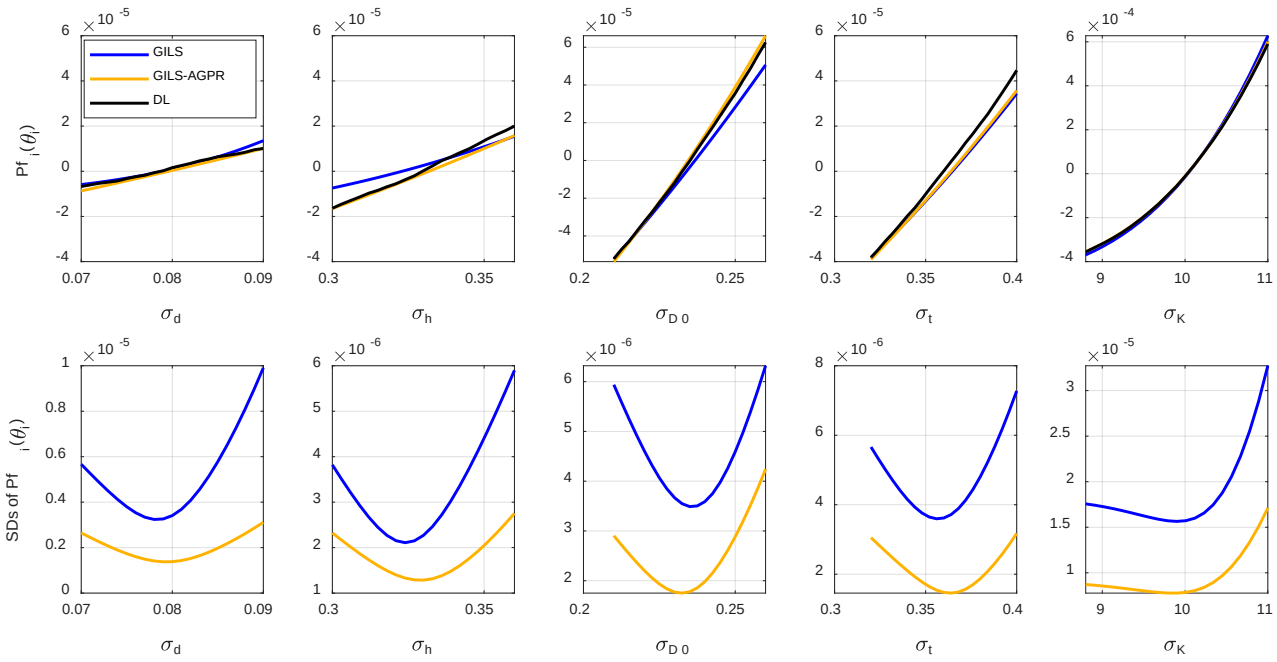

Figure 8 Plots of the first-order component failure probability functions and their corresponding SDs for the simplified riveting model

Table 4 Results of first- and second-order sensitivity indices for the simplified riveting model. Standard deviations of estimates are shown as superscripts in parenthesis.

Methods		GILS	GILS-AGPR	DL
Function evaluations		18+3×400	18+45	/
p_{f0}		5.08e-4$^{(1.8e-5)}$	4.92e-4$^{(1.3e-5)}$	4.89e-4
Upper bound of p_f		9.92e-5$^{(2.8e-5)}$	9.58e-5$^{(1.4e-5)}$	9.33e-5
Upper bound of p_f		1.35e-3$^{(4.3e-5)}$	1.32e-3$^{(2.2e-5)}$	1.32e-3
S_i	σ_{d}	0.0004$^{(0.0003)}$	0.0004$^{(0.0001)}$	0.0012
	σ_{h}	0.0006$^{(0.0001)}$	0.0011$^{(0.0000)}$	0.0021
	σ_{D_0}	0.0106$^{(0.0002)}$	0.0156$^{(0.0001)}$	0.0154
	σ_t	0.0054$^{(0.0003)}$	0.0062$^{(0.0001)}$	0.0086
	σ_{K}	0.9794$^{(0.0043)}$	0.9727$^{(0.0012)}$	0.9684
S_{ij}	$(\sigma_{\mathrm{d}},\sigma_{\mathrm{K}})$	0.0001$^{(0.0002)}$	0.0001$^{(0.0000)}$	0.0001
	$(\sigma_{\mathrm{h}},\sigma_{\mathrm{K}})$	0.0002$^{(0.0001)}$	0.0001$^{(0.0000)}$	0.0003
	$(\sigma_{\mathrm{D}_0},\sigma_{\mathrm{K}})$	0.0020$^{(0.0001)}$	0.0027$^{(0.0001)}$	0.0026
	$(\sigma_{\mathrm{t}},\sigma_{\mathrm{K}})$	0.0012$^{(0.0002)}$	0.0011$^{(0.0000)}$	0.0007

5.4 Confined seepage model

Consider a confined seepage model (taken from Ref. [43]). The elevation of the dam is shown in Figure 9. The dam rests over soil which is composed of two permeable layers and one impermeable layer. The water height in the

upstream of the dam is denoted by h_D (m) following the uniform distribution $U(7,10)$. Over segment *AB*, the hydraulic head h_W with respect to the impermeable layer is $h_W = h_D + 20$. The water flows through two permeable soil layers towards the downstream over the segment CD. Assume that no water flows on any of the boundaries excepted the segments AB and CD. The first permeable layer is silty sand, while the second one is composed of silty gravel. The permeability of the two layers are modeled as anisotropic and characterized by lognormal random variables, the means and COVs of the horizontal and vertical permeability of the two soil layers are listed in Table 5, and the mean values are modeled as imprecise distribution parameters.

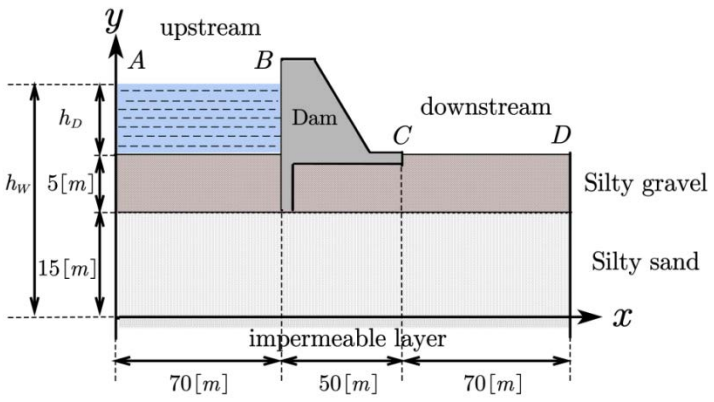

Figure 9 The elevation of the dam.

Table 5 Description of input parameters for confined seepage model.

Input variables	Description	Distribution type	Mean	COV
$k_{xx,1}$ (10^{-7}m/s)	Horizontal permeability of sand soil layer	lognormal	[4.5,5.5]	100%
$k_{yy,1}$ (10^{-7}m/s)	Vertical permeability of sand soil layer	lognormal	[1.8,2.2]	100%
$k_{xx,2}$ (10^{-6}m/s)	Horizontal permeability of gravel soil layer	lognormal	[4.5,5.5]	100%
$k_{yy,2}$ (10^{-6}m/s)	Vertical permeability of gravel soil layer	lognormal	[1.8,2.2]	100%

We assume that all the COVs of permeability equal to 100%, indicating a high degree of dispersion in characterizing the permeability. The seepage q at the downstream side is measured by volume over time (hour) over distance (meter) with the unit as $L/h/m$, which is calculated by the integral $q = -\int_{CD} k_{yy,2}\,\partial h_W/\partial y\,\mathrm{d}x$. The failure event of interest is defined when seepage q exceeds a prescribed threshold 33. Summarily, the four permeability parameters of the permeable layers are modeled as imprecise random variables, while water height h_D is modeled as a precise uniform random variable, and limit state function is $g = 33 - q$. The seepage flow q is calculated by solving the associated Laplace partial differential equation with the help of a finite element model comprising more than 1500 quadratic triangular elements.

Firstly, we calculate the MPP on the condition that the imprecise parameters are fixed at the midpoint values, so the important direction is determined as $\boldsymbol{e}_\alpha = (0.8080, 0.3930, 0.2782, 0.2814, 0.1899)^T$ with 64 g-function calls. Then the two proposed procedures are applied to estimate the failure probability function of the seepage model,

and the first-order component functions are plotted in Figure 10. As for the GILS procedure, the two cases when the sample size $N_G = 500$ and $N_G = 2000$ are plotted separately. In the GILS-AGPR procedure, 54 training samples are involved to establish the GPR model so as to accurately perform one-dimensional reliability analysis along 2000 candidate sample lines. In Figure 10, the three curves for each component function coincide well except that there is a slight difference for the mean value of $k_{xx,2}$. This slight difference may be introduced due to the approximation error from the established GPR model in the GILS-AGPR procedure. All the first-order components are monotonically increasing w.r.t. the imprecise parameters, so the upper and lower bounds of the uncertain intervals correspond to the maximum and minimum values of component failure probability.

Table 6 shows the results of the constant component, the bounds of the failure probability as well as the first- and second-order sensitivity indices of the four uncertain mean values. It is seen that for both lower and upper bounds, the two proposed methods produce accurate estimate with small variation. This is due to the relatively small gap between the lower and upper bounds. Note that in the estimations of GILS, 500 sample lines used in Figure 10 are chosen and a total of 1500 *g*-function evaluations are carried out. We can see that the results of both methods match well and lead to the same importance ranking of the components. By comparing the values of the indices, one can find that the uncertain mean value of $k_{xx,1}$ and $k_{yy,1}$ are the first and second most influential parameters in failure probability function, and all the second-order indices are quite small, indicating that the uncertain mean values have a weak interaction effect on the uncertainty of the failure probability.

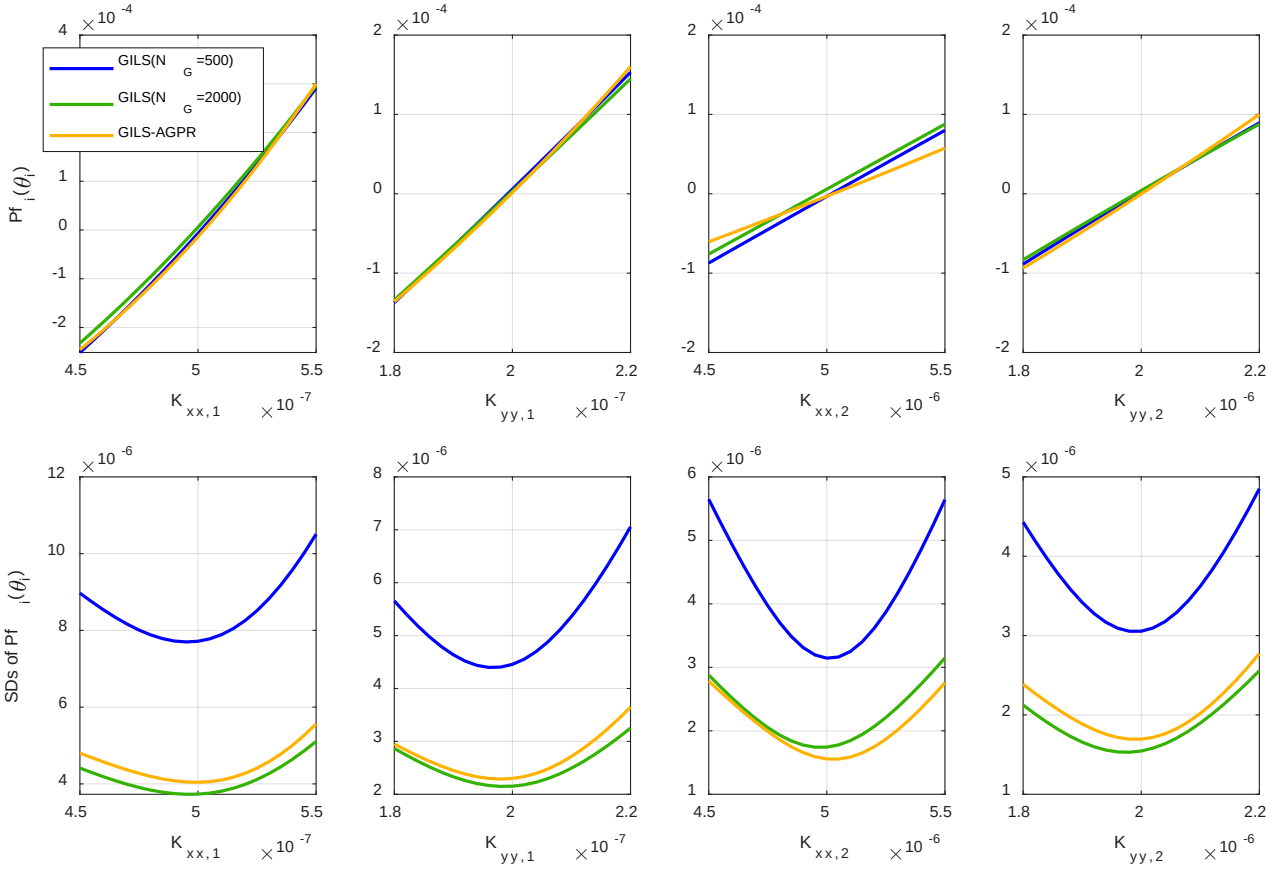

Figure 10 Plots of the first-order component functions and their corresponding SDs for confined seepage model.

Table 6 First- and second-order sensitivity indices of the confined seepage model. Standard deviations of estimates are shown as superscripts in parenthesis.

Methods		GILS	GILS-AGPR
Function evaluations		64+1500	64+54
p_{f0}		8.29e-4$^{(1.7e-5)}$	8.20e-4$^{(9.3e-6)}$
Lower bound of p_f		3.92e-4$^{(2.19e-5)}$	3.75e-4$^{(9.4e-6)}$
Upper bound of p_f		1.59e-3$^{(2.32e-5)}$	1.48e-3$^{(9.8e-6)}$
S_i	$k_{xx,1}$	0.6624$^{(0.0019)}$	0.6723$^{(0.0005)}$
	$k_{yy,1}$	0.1901$^{(0.0007)}$	0.1992$^{(0.0002)}$
	$k_{xx,2}$	0.0640$^{(0.0005)}$	0.0317$^{(0.0001)}$
	$k_{yy,2}$	0.0716$^{(0.0004)}$	0.0851$^{(0.0001)}$
S_{ij}	$(k_{xx,1},k_{yy,1})$	0.0062$^{(0.0001)}$	0.0066$^{(0.0000)}$
	$(k_{xx,1},k_{xx,2})$	0.0018$^{(0.0000)}$	0.0009$^{(0.0000)}$
	$(k_{xx,1},k_{yy,2})$	0.0023$^{(0.0000)}$	0.0028$^{(0.0000)}$
	$(k_{yy,1}k_{xx,2})$	0.0006$^{(0.0000)}$	0.0003$^{(0.0000)}$
	$(k_{yy,1},k_{yy,2})$	0.0007$^{(0.0000)}$	0.0009$^{(0.0000)}$
	$(k_{xx,2},k_{yy,2})$	0.0003$^{(0.0000)}$	0.0001$^{(0.0000)}$

6. Conclusions

This paper develops two algorithms based on line sampling for rare failure event analysis subjected to imprecise probabilities. The first one is the GILS algorithm, which allows us to estimate the failure probability function with only one set of samples, globally exploring both the space of input variables and imprecise parameters at the same time. It is designed to inject PDF weight into the integral of classic line sampling, then the weighted integral formulation of the failure probability function is decomposed by the RS-HDMR method, and the constant, first-order and second-order estimators of component failure probability functions are analytically and separately derived. Additionally, the analytical formulations for the normal and lognormal case are also derived. The second algorithm formulated is the GILS-AGPR method, which is developed to further reduce the number of *g*-function calls and improve the accuracy of estimations. The steps of GPR model training and estimation of the component functions, as well as the relationship with the first algorithm, have been interpreted in detail. With only one stochastic simulation, the two proposed methods produce three important products, i.e., (i) the HDMR component functions for visibly learning the relationship between the failure probability and epistemic distribution parameters, (ii) the sensitivity indices for quantifying the contribution of the epistemic uncertainty in each input variable to that of the failure

probability, and (iii) the upper bound of failure probability which is served as a conservative estimation of the structural safety.

The results of case studies show that the proposed algorithms have good global performance in estimating failure probability function for problems with rare events as well as the upper bound of the failure probability, and the computational cost is proved to be much lower. It is also concluded that, for the cases where the span of the bounds of failure probability covers one or two orders of magnitude, the two proposed methods produce satisfactory estimates for the lower bound, however, if it covers several orders (e.g., three or more) of magnitude, the upper bound can still be accurately estimated, but the estimates for the lower bound can be of low quality. GILS-AGPR can well make up the deficiency of a slow convergence speed of the GILS procedure with a reduced numerical cost. From the case studies, we can observe that the convergence speed of the estimators is influenced by two sources: one is the spread of the samples of uncertain distribution parameters in the estimators; the other one is that, the important direction is determined at a fixed point of imprecise parameters, which may cause that the estimators become less efficient in those areas far from this fixed direction. There also exist limitations of the line sampling algorithm. It relies on the correct identification of the important direction associated with the failure surface and is mostly applicable for problems with moderately nonlinear performance functions. As for models with high-dimensional input variables and multiple failure modes, one needs to perform additional investigations by considering other advanced approaches.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (grant number NSFC 51905430) and ANID (National Agency for Research and Development, Chile) under its program FONDECYT, grant number 1180271. The first author is supported by the program of China Scholarships Council (CSC). The second and third authors are both supported by the Alexander von Humboldt Foundation of Germany.

Appendix A

Derivations of the constant, first-order and second-order component functions in Eqs. (13) and (14)

a. Constant component p_{f0}

Replace all the vectors $\boldsymbol{\theta}$ in the integral formulation of p_{f0} with $\boldsymbol{\theta}'$ (independent and identically distributed with $\boldsymbol{\theta}$), and transform the original space into standard normal space with the formulation $\boldsymbol{x}=T^{-1}(\boldsymbol{z}^{\perp}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}')$ based on line sampling procedure,

$$\begin{aligned} p_{f0} &= \int_{g(\boldsymbol{x})\leqslant 0} f_{\boldsymbol{X}}(\boldsymbol{x}|\boldsymbol{\theta}')f_{\boldsymbol{\Theta}'}(\boldsymbol{\theta}')\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{\theta}' \\ &= \int_{g(T^{-1}(\boldsymbol{z}^{\perp}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}'))\leqslant 0} \phi(\overline{z})\mathrm{d}\overline{z}\phi_{n-1}(\boldsymbol{z}^{\perp})\mathrm{d}\boldsymbol{z}^{\perp}f_{\boldsymbol{\Theta}'}(\boldsymbol{\theta}')\mathrm{d}\boldsymbol{\theta}' \end{aligned}. \tag{A1}$$

Utilizing the same set of samples $\boldsymbol{S}=(\boldsymbol{z}^{\perp(s)},\boldsymbol{\theta}^{(s)})(s=1,\cdots,N)$ and the corresponding value of $\nu^{(s)}$ at the intersection point along each sample line, the constant component can be estimated by

$$\hat{p}_{f0}=\frac{1}{N}\sum_{s=1}^{N}\int_{g(T^{-1}(\boldsymbol{z}^{\perp(s)}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}'^{(s)}))\leqslant 0}\phi(\overline{z})\mathrm{d}\overline{z}=\frac{1}{N}\sum_{s=1}^{N}\Phi(-\nu^{(s)}). \tag{A2}$$

Additionally, the variance of the above estimator is

$$Var(\hat{p}_{f0})=\frac{1}{N(N-1)}\sum_{s=1}^{N}\left(\Phi(-\nu^{(s)})-\hat{p}_{f0}\right)^2. \tag{A3}$$

b. The first-order component function $p_{fi}(\theta_i)$

The vector $\boldsymbol{\theta}$ is reconsidered by separating it into two parts, θ_i and $\boldsymbol{\theta}_{-i}'$, then the integral is derived as

$$\begin{aligned}p_{fi}(\theta_i)&=\int_{g(\boldsymbol{x})\leqslant 0}f_{\boldsymbol{X}}(\boldsymbol{x}|\boldsymbol{\theta})f_{\boldsymbol{\Theta}_{-i}}(\boldsymbol{\theta}_{-i})\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{\theta}_{-i}-p_{f0}\\&=\int_{g(\boldsymbol{x})\leqslant 0}f_{\boldsymbol{X}}(\boldsymbol{x}|\theta_i,\boldsymbol{\theta}_{-i}')f_{\boldsymbol{\Theta}_{-i}'}(\boldsymbol{\theta}_{-i}')\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{\theta}_{-i}'-p_{f0}\end{aligned}. \tag{A4}$$

Then $f_{\Theta_i}(\theta_i')\mathrm{d}\theta_i'$ is introduced into the above integral as well as the PDF weight with $f_{\boldsymbol{X}}(\boldsymbol{x}|\boldsymbol{\theta}')$,

$$\begin{aligned}p_{fi}(\theta_i)&=\int_{g(\boldsymbol{x})\leqslant 0}f_{\boldsymbol{X}}(\boldsymbol{x}|\theta_i,\boldsymbol{\theta}_{-i}')f_{\boldsymbol{\Theta}_{-i}'}(\boldsymbol{\theta}_{-i}')f_{\Theta_i'}(\theta_i')\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{\theta}_{-i}'\mathrm{d}\theta_i'-p_{f0}\\&=\int_{g(\boldsymbol{x})\leqslant 0}\frac{f_{\boldsymbol{X}}(\boldsymbol{x}|\theta_i,\boldsymbol{\theta}_{-i}')}{f_{\boldsymbol{X}}(\boldsymbol{x}|\boldsymbol{\theta}')}f_{\boldsymbol{X}}(\boldsymbol{x}|\boldsymbol{\theta}')f_{\boldsymbol{\Theta}'}(\boldsymbol{\theta}')\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{\theta}'-p_{f0}\end{aligned}. \tag{A5}$$

Similarly, transform the original space into standard normal space so that

$$\begin{aligned}p_{fi}(\theta_i)&=\int_{g(T^{-1}(\boldsymbol{z}^{\perp}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}'))\leqslant 0}\frac{f_{\boldsymbol{X}}(T^{-1}(\boldsymbol{z}^{\perp}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}')|\theta_i,\boldsymbol{\theta}_{-i}')}{f_{\boldsymbol{X}}(T^{-1}(\boldsymbol{z}^{\perp}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}')|\boldsymbol{\theta}')}\phi(\overline{z})\mathrm{d}\overline{z}\phi_{n-1}(\boldsymbol{z}^{\perp})\mathrm{d}\boldsymbol{z}^{\perp}f_{\boldsymbol{\Theta}'}(\boldsymbol{\theta}')\mathrm{d}\boldsymbol{\theta}'-p_{f0}\\&=\int_{g(T^{-1}(\boldsymbol{z}^{\perp}+\boldsymbol{e}_{\alpha}\overline{z}|\boldsymbol{\theta}'))\leqslant 0}\omega(\theta_i,\boldsymbol{\theta}',\boldsymbol{z}^{\perp},\overline{z})\phi(\overline{z})\mathrm{d}\overline{z}\phi_{n-1}(\boldsymbol{z}^{\perp})\mathrm{d}\boldsymbol{z}^{\perp}f_{\boldsymbol{\Theta}'}(\boldsymbol{\theta}')\mathrm{d}\boldsymbol{\theta}'-p_{f0}\end{aligned} \tag{A6}$$

, where $\omega(\theta_i,\boldsymbol{\theta}',\boldsymbol{z}^{\perp},\overline{z})$ is the PDF weight function. Finally, the Monte Carlo estimator of $p_{fi}(\theta_i)$ obtained considering the set of samples $\boldsymbol{S}=(\boldsymbol{z}^{\perp(s)},\boldsymbol{\theta}^{(s)})(s=1,\cdots,N)$, leading to:

$$\begin{aligned}\hat{p}_{fi}(\theta_i)&=\frac{1}{N}\sum_{s=1}^{N}\int_{\nu^{(s)}}\omega(\theta_i,\boldsymbol{\theta}^{(s)},\boldsymbol{z}^{\perp(s)},\overline{z})\phi(\overline{z})\mathrm{d}\overline{z}-\hat{p}_{f0}\\&=\frac{1}{N}\sum_{s=1}^{N}L^{(s)}(\theta_i)-\hat{p}_{f0}\end{aligned} \tag{A7}$$

, where $L^{(s)}(\theta_i)=\int_{\nu^{(s)}}\omega(\theta_i,\boldsymbol{\theta}^{(s)},\boldsymbol{z}^{\perp(s)},\overline{z})\phi(\overline{z})\mathrm{d}\overline{z}$ denotes the integral which can be regarded as a function of θ_i.

$\hat{p}_{fi}(\theta_i)$ can be further derived as $\hat{p}_{fi}(\theta_i)=\frac{1}{N}\sum_{s=1}^{N}\left(L^{(s)}(\theta_i)-\Phi(-\nu^{(s)})\right)$, so the variance of this estimator is

$$Var(\hat{p}_{fi}(\theta_i))=\frac{1}{N(N-1)}\sum_{s=1}^{N}\left(L^{(s)}(\theta_i)-\Phi(-\nu^{(s)})-\hat{p}_{fi}\right)^2. \tag{A8}$$

c. The second-order RS-HDMR component function $p_{fij}(\theta_i,\theta_j)$

The derivation for the second-order component function is quite similar with the one of the first-order component. Firstly, the vector $\boldsymbol{\theta}$ is rewritten as $\boldsymbol{\theta} = (\theta_i, \theta_j, \boldsymbol{\theta}_{-ij}')$, and the first term (denoted by L_{ij}) in $p_{fij}(\theta_i, \theta_j)$ formulation is derived as

$$L_{ij} = \int_{g(\boldsymbol{x}) \leqslant 0} f_{\boldsymbol{X}}(\boldsymbol{x}|\boldsymbol{\theta}) f_{\boldsymbol{\Theta}_{-ij}}(\boldsymbol{\theta}_{-ij}) \mathrm{d}\boldsymbol{x} \, \mathrm{d}\boldsymbol{\theta}_{-ij} = \int_{g(\boldsymbol{x}) \leqslant 0} f_{\boldsymbol{X}}(\boldsymbol{x}|\theta_i, \theta_j, \boldsymbol{\theta}_{-ij}') f_{\boldsymbol{\Theta}_{-ij}'}(\boldsymbol{\theta}_{-ij}') \mathrm{d}\boldsymbol{x} \, \mathrm{d}\boldsymbol{\theta}_{-ij}' . \tag{A9}$$

Then, $f_{\Theta_i}(\theta_i') f_{\Theta_j}(\theta_j') \mathrm{d}\theta_i' \mathrm{d}\theta_j'$ and the PDF weight of $f_{\boldsymbol{X}}(\boldsymbol{x}|\boldsymbol{\theta}')$ are introduced into the above integral,

$$\begin{aligned} L_{ij} &= \int_{g(\boldsymbol{x}) \leqslant 0} f_{\boldsymbol{X}}(\boldsymbol{x}|\theta_i, \theta_j, \boldsymbol{\theta}_{-ij}') f_{\boldsymbol{\Theta}_{-ij}'}(\boldsymbol{\theta}_{-ij}') f_{\Theta_i}(\theta_i') f_{\Theta_j}(\theta_j') \mathrm{d}\boldsymbol{x} \, \mathrm{d}\boldsymbol{\theta}_{-ij}' \mathrm{d}\theta_i' \mathrm{d}\theta_j' \\ &= \int_{g(\boldsymbol{x}) \leqslant 0} \frac{f_{\boldsymbol{X}}(\boldsymbol{x}|\theta_i, \theta_j, \boldsymbol{\theta}_{-ij}')}{f_{\boldsymbol{X}}(\boldsymbol{x}|\boldsymbol{\theta}')} f_{\boldsymbol{X}}(\boldsymbol{x}|\boldsymbol{\theta}') f_{\boldsymbol{\Theta}'}(\boldsymbol{\theta}') \mathrm{d}\boldsymbol{x} \, \mathrm{d}\boldsymbol{\theta}' \end{aligned} . \tag{A10}$$

The transformation formulation $\boldsymbol{x} = T^{-1}(\boldsymbol{z}^{\perp} + \boldsymbol{e}_{\alpha} \overline{z} | \boldsymbol{\theta}')$ is also applied on the above integral, and we denote the PDF weight as $\omega(\theta_i, \theta_j, \boldsymbol{\theta}', \boldsymbol{z}^{\perp}, \overline{z})$, hence, for the s-th sample $(\boldsymbol{z}^{\perp(s)}, \boldsymbol{\theta}^{(s)})$ the PDF weight becomes

$$\omega(\theta_i, \theta_j, \boldsymbol{\theta}^{(s)}, \boldsymbol{z}^{\perp(s)}, \overline{z}) = \frac{f_{\boldsymbol{X}}\left(T^{-1}(\boldsymbol{z}^{\perp(s)} + \boldsymbol{e}_{\alpha} \overline{z} | \boldsymbol{\theta}^{(s)}) | \theta_i, \theta_j, \boldsymbol{\theta}_{-ij}^{(s)}\right)}{f_{\boldsymbol{X}}\left(T^{-1}(\boldsymbol{z}^{\perp(s)} + \boldsymbol{e}_{\alpha} \overline{z} | \boldsymbol{\theta}^{(s)}) | \boldsymbol{\theta}^{(s)}\right)} . \tag{A11}$$

Let $L^{(s)}(\theta_i, \theta_j) = \int_{\nu^{(s)}} \omega(\theta_i, \theta_j, \boldsymbol{\theta}^{(s)}, \boldsymbol{z}^{\perp(s)}, \overline{z}) \phi(\overline{z}) \mathrm{d}\overline{z}$ denote the integral w.r.t. $\overline{z}$, then the estimator of $p_{fij}(\theta_i, \theta_j)$ is derived as

$$\hat{p}_{fij}(\theta_i, \theta_j) = \frac{1}{N} \sum_{s=1}^{N} L^{(s)}(\theta_i, \theta_j) - \hat{p}_{fi}(\theta_i) - \hat{p}_{fj}(\theta_j) - \hat{p}_{f0} . \tag{A12}$$

$\hat{p}_{fij}(\theta_i, \theta_j)$ can be further derived as

$$\begin{aligned} \hat{p}_{fij}(\theta_i, \theta_j) &= \frac{1}{N} \sum_{s=1}^{N} \left(L^{(s)}(\theta_i, \theta_j) - \left(L^{(s)}(\theta_i) - \Phi(-\nu^{(s)})\right) - \left(L^{(s)}(\theta_j) - \Phi(-\nu^{(s)})\right) - \Phi(-\nu^{(s)})\right) \\ &= \frac{1}{N} \sum_{s=1}^{N} \left(L^{(s)}(\theta_i, \theta_j) - L^{(s)}(\theta_i) - L^{(s)}(\theta_j) + \Phi(-\nu^{(s)})\right) \end{aligned} \tag{A13}$$

, so the variance of the above estimator is

$$Var\left(\hat{p}_{fij}(\theta_i, \theta_j)\right) = \frac{1}{N(N-1)} \sum_{s=1}^{N} \left(L^{(s)}(\theta_i, \theta_j) - L^{(s)}(\theta_i) - L^{(s)}(\theta_j) + \Phi(-\nu^{(s)}) - \hat{p}_{fij}\right)^2 \tag{A14}$$

Appendix B

Analytical derivation of $L^{(s)}(\boldsymbol{\theta})$ (in Eq. (11)) as well as $L^{(s)}(\theta_i)$ and $L^{(s)}(\theta_i, \theta_j)$ (in Eq.(13)) for normal distribution case

Let x_d be the *d*-th component of vector $\boldsymbol{x}$. For the case where x_d follows normal distribution, $x_d \sim N(\mu_d, \sigma_d^2)$, the transformation of that component associated with the s-th sample set to the rotated standard normal space will be specified as

$$x_d = \mu_d^{(s)} + \sigma_d^{(s)} z_d = \mu_d^{(s)} + \sigma_d^{(s)} z_d^{\perp(s)} + \sigma_d^{(s)} \overline{z} e_{\alpha,d} \tag{B1}$$

, where $\mu_d^{(s)}$ and $\sigma_d^{(s)}$ denote the sample of the distribution parameters while $z_d^{\perp(s)}$ denotes the *d*-th component of sample $\boldsymbol{z}^{\perp(s)}$. Under the assumption that all components of $\boldsymbol{x}$ follow Gaussian distribution, it can be shown that the PDF weight in Eq.(11) is analytically expressed as

$$\omega(\boldsymbol{\theta},\boldsymbol{\theta}^{(s)},\boldsymbol{z}^{\perp(s)},\overline{z}) = \prod_{d=1}^{n}\frac{\sigma_d^{(s)}}{\sigma_d}\exp\left(\sum_{d=1}^{n}\left(\frac{\left(z_d^{\perp(s)}+\overline{z}e_{\alpha,d}\right)^2}{2} - \frac{\left(\mu_d^{(s)}+\sigma_d^{(s)}z_d^{\perp(s)}+\sigma_d^{(s)}\overline{z}e_{\alpha,d}-\mu_d\right)^2}{2\sigma_d^2}\right)\right) \quad \text{(B2)}$$

, where $\boldsymbol{\theta}=[\mu_1,\ldots,\mu_n,\sigma_1,\ldots,\sigma_n]$ and $\boldsymbol{\theta}^{(s)}=[\mu_1^{(s)},\ldots,\mu_n^{(s)},\sigma_1^{(s)},\ldots,\sigma_n^{(s)}]$. For simplification, four parameters $\xi^{(s)}$, $\zeta^{(s)}$, $\lambda^{(s)}$ and $\kappa^{(s)}$ are introduced with the expressions below,

$$\begin{aligned}
\xi^{(s)} &= \prod_{d=1,\cdots,n}\sigma_d^{(s)}/\sigma_d \\
\zeta^{(s)} &= \sum_{d=1}^{n}\frac{1}{2}e_{\alpha,d}^2\left(1-\sigma_d^{(s)2}/\sigma_d^2\right) \\
\lambda^{(s)} &= \sum_{d=1}^{n}\left(\frac{1}{2}\left(z_d^{\perp(s)}\right)^2-\left(\mu_d^{(s)}+\sigma_d^{(s)}z_d^{\perp(s)}-\mu_d\right)^2/(2\sigma_d^2)\right) \\
\kappa^{(s)} &= \sum_{d=1}^{n}e_{\alpha,d}\left(z_d^{\perp(s)}-\left(\mu_d^{(s)}+\sigma_d^{(s)}z_d^{\perp(s)}-\mu_d\right)\sigma_d^{(s)}/\sigma_d^2\right)
\end{aligned} \quad \text{(B3)}$$

The above four parameters are not only functions of distribution parameters but also vary according to the values of the s th sample. The four parameters are inserted into Eq.(B2) so as to rewrite the PDF weight as

$$\omega(\boldsymbol{\theta},\boldsymbol{\theta}^{(s)},\boldsymbol{z}^{\perp(s)},\overline{z}) = \xi^{(s)}\exp\left(\lambda^{(s)}+\overline{z}\kappa^{(s)}+\overline{z}^2\zeta^{(s)}\right) \quad \text{(B4)}$$

The following steps for deriving the integral of $\omega(\boldsymbol{\theta},\boldsymbol{\theta}^{(s)},\boldsymbol{z}^{\perp(s)},\overline{z})$ times $\phi(\overline{z})$ w.r.t. $\overline{z}$ along the line are the same as those in Appendix A of Ref.[30] and thus, the analytical expression of $L^{(s)}(\boldsymbol{\theta})$ is derived as

$$L^{(s)}(\boldsymbol{\theta}) = L^{(s)}(\boldsymbol{\mu},\boldsymbol{\sigma}) = \frac{\xi^{(s)}}{\sqrt{1-2\zeta^{(s)}}}\exp\left(\lambda^{(s)}+\frac{\left(\kappa^{(s)}\right)^2}{2-4\zeta^{(s)}}\right)\Phi\left(\frac{\kappa^{(s)}-(1-2\zeta^{(s)})\nu^{(s)}}{\sqrt{1-2\zeta^{(s)}}}\right) \quad \text{(B5)}$$

Based on the above derivations, the integral functions $L^{(s)}(\theta_i)$ and $L^{(s)}(\theta_i,\theta_j)$ in the estimators of the first-order and second-order components can be easily derived accordingly. Their functional form is the same as the one shown in eq. (B5), except for the fact that the definition of the parameters $\xi^{(s)}$, $\zeta^{(s)}$, $\lambda^{(s)}$ and $\kappa^{(s)}$ is different. The corresponding formulations of the parameters $\xi^{(s)}$, $\zeta^{(s)}$, $\lambda^{(s)}$ and $\kappa^{(s)}$ for each of the functions $L^{(s)}(\theta_i)$ and $L^{(s)}(\theta_i,\theta_j)$ are listed in Table B1.

The expressions in Table B1 have been deduced for the case where each component of vector $\boldsymbol{x}$ follows a normal distribution. For the case where each component of $\boldsymbol{x}$ follows a lognormal distribution, it can be shown that the expressions for $L^{(s)}(\boldsymbol{\theta})$, $L^{(s)}(\theta_i)$ and $L^{(s)}(\theta_i,\theta_j)$ are identical to those shown in Eq. (B5) and Table B.1, except for the fact that one must consider the mean and standard deviation of the logarithm of $\boldsymbol{x}$ [30]. In case the components of vector $\boldsymbol{x}$ follow distributions different from normal or lognormal, the associated line integrals must be solved considering analytic or numerical schemes, as discussed in Ref. [30].

Table B1 Analytical formulations of $\xi^{(s)}$, $\zeta^{(s)}$, $\lambda^{(s)}$ and $\kappa^{(s)}$ in integral functions $L^{(s)}(\theta_i)$ and $L^{(s)}(\theta_i,\theta_j)$

Integral functions	$\xi^{(s)}$	$\zeta^{(s)}$	$\lambda^{(s)}$	$\kappa^{(s)}$
$L^{(s)}(\mu_i)$	1	0	$\frac{1}{2}(z_i^{\perp(s)})^2-\frac{(\mu_i^{(s)}+\sigma_i^{(s)}z_i^{\perp(s)}-\mu_i)^2}{2\sigma_i^{(s)2}}$	$(\mu_i-\mu_i^{(s)})e_{\alpha,i}/\sigma_i^{(s)}$
$L^{(s)}(\sigma_i)$	$\sigma_i^{(s)}/\sigma_i$	$\frac{1}{2}e_{\alpha,i}^2(1-\sigma_i^{(s)2}/\sigma_i^2)$	$\frac{1}{2}(z_i^{\perp(s)})^2(1-\sigma_i^{(s)2}/\sigma_i^2)$	$e_{\alpha,i}z_i^{\perp(s)}(1-\sigma_i^{(s)2}/\sigma_i^2)$
$L^{(s)}(\mu_i,\mu_j)$	1	0	$\sum_{k=i,j}\left(\frac{1}{2}(z_k^{\perp(s)})^2-\frac{(\mu_k^{(s)}+\sigma_k^{(s)}z_k^{\perp(s)}-\mu_k)^2}{2\sigma_k^{(s)2}}\right)$	$\sum_{k=i,j}(\mu_k-\mu_k^{(s)})e_{\alpha,k}/\sigma_k^{(s)}$
$L^{(s)}(\sigma_i,\sigma_j)$	$\sigma_i^{(s)}\sigma_j^{(s)}/\sigma_i\sigma_j$	$\sum_{k=i,j}\frac{1}{2}e_{\alpha,k}^2(1-\sigma_k^{(s)2}/\sigma_k^2)$	$\sum_{k=i,j}\frac{1}{2}(z_k^{\perp(s)})^2(1-\sigma_k^{(s)2}/\sigma_k^2)$	$\sum_{k=i,j}e_{\alpha,k}z_k^{\perp(s)}(1-\sigma_k^{(s)2}/\sigma_k^2)$
$L^{(s)}(\mu_i,\sigma_i)$	$\sigma_i^{(s)}/\sigma_i$	$\frac{1}{2}e_{\alpha,i}^2(1-\sigma_i^{(s)2}/\sigma_i^2)$	$\frac{1}{2}(z_i^{\perp(s)})^2-(\mu_i^{(s)}+\sigma_i^{(s)}z_i^{\perp(s)}-\mu_i)^2/(2\sigma_i^2)$	$e_{\alpha,i}\left(\begin{matrix}z_i^{\perp(s)}-\\(\mu_i^{(s)}+\sigma_i^{(s)}z_i^{\perp(s)}-\mu_i)\sigma_i^{(s)}/\sigma_i^2\end{matrix}\right)$
$L^{(s)}(\mu_i,\sigma_j)$ $(i\neq j)$	$\sigma_j^{(s)}/\sigma_j$	$\frac{1}{2}e_{\alpha,j}^2(1-\sigma_j^{(s)2}/\sigma_j^2)$	$\frac{1}{2}(z_i^{\perp(s)})^2-(\mu_i^{(s)}+\sigma_i^{(s)}z_i^{\perp(s)}-\mu_i)^2/(2\sigma_i^{(s)2})$ $+\frac{1}{2}(z_j^{\perp(s)})^2(1-\sigma_j^{(s)2}/\sigma_j^2)$	$(\mu_i-\mu_i^{(s)})e_{\alpha,i}/\sigma_i^{(s)}+$ $e_{\alpha,j}z_j^{\perp(s)}(1-\sigma_j^{(s)2}/\sigma_j^2)$

References

[1] Der Kiureghian A, Ditlevsen O. Aleatory or epistemic? Does it matter?. Structural Safety, 2009, 31(2): 105-112.

[2] Hohenbichler M, Gollwitzer S, Kruse W, et al. New light on first-and second-order reliability methods. Structural Safety, 1987, 4(4): 267-284.

[3] Lu Z, Cai C, Zhao Y, et al. Normalization of correlated random variables in structural reliability analysis using fourth-moment transformation. Structural Safety, 2020, 82: 101888.

[4] Li J. Probability density evolution method: background, significance and recent developments. Probabilistic Engineering Mechanics, 2016, 44: 111-117.

[5] Au S K, Beck J L. A new adaptive importance sampling scheme for reliability calculations. Structural Safety, 1999, 21(2): 135-158.

[6] Au S K, Beck J L. Estimation of small failure probabilities in high dimensions by subset simulation. Probabilistic Engineering Mechanics, 2001, 16(4): 263-277.

[7] Schuëller G I, Pradlwarter H J, Koutsourelakis P S. A critical appraisal of reliability estimation procedures for high dimensions. Probabilistic Engineering Mechanics, 2004, 19(4): 463-474.

[8] Nie J, Ellingwood B R. Directional methods for structural reliability analysis. Structural Safety, 2000, 22(3): 233-249

[9] Pan Q, Dias D. An efficient reliability method combining adaptive support vector machine and Monte Carlo simulation. Structural Safety, 2017, 67: 85-95.

[10] Chojaczyk A A, Teixeira A P, Neves L C, et al. Review and application of artificial neural networks models in reliability analysis of steel structures. Structural Safety, 2015, 52: 78-89.

[11] Bichon B J, Eldred M S, Swiler L P, et al. Efficient global reliability analysis for nonlinear implicit performance functions. AIAA Journal, 2008, 46(10): 2459-2468.

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

[13] Li M, Wang Z. Active Resource Allocation for Reliability Analysis With Model Bias Correction. Journal of Mechanical Design, 2019, 141(5).

[14] Wang Z, Wang P. Accelerated failure identification sampling for probability analysis of rare events. Structural and Multidisciplinary Optimization, 2016, 54(1): 137-149.

[15] Li M, Wang Z. Surrogate model uncertainty quantification for reliability-based design optimization. Reliability Engineering & System Safety, 2019, 192: 106432.

[16] Karanki D R, Kushwaha H S, Verma A K, et al. Uncertainty analysis based on probability bounds (p-box) approach in probabilistic safety assessment. Risk Analysis: An International Journal, 2009, 29(5): 662-675.

[17] Ferson S, Kreinovich V, Ginzburg L, et al. Constructing probability boxes and Dempster-Shafer structures. Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2003.

[18] Wang C, Matthies H G. Epistemic uncertainty-based reliability analysis for engineering system with hybrid evidence and fuzzy variables. Computer Methods in Applied Mechanics and Engineering, 2019, 355: 438-455

[19] Stein M, Beer M, Kreinovich V. Bayesian approach for inconsistent information. Information Sciences, 2013, 245: 96-111.

[20] Beer M, Ferson S, Kreinovich V. Imprecise probabilities in engineering analyses. Mechanical Systems and Signal Processing, 2013, 37(1-2): 4-29.

[21] Pedroni N, Zio E. Hybrid uncertainty and sensitivity analysis of the model of a twin-jet aircraft. Journal of Aerospace Information Systems, 2015, 12(1): 73-96

[22] Zhang H, Dai H, Beer M, et al. Structural reliability analysis on the basis of small samples: an interval quasi-Monte Carlo method. Mechanical Systems and Signal Processing, 2013, 37(1-2): 137-151

[23] Alvarez D A, Uribe F, Hurtado J E. Estimation of the lower and upper bounds on the probability of failure using subset simulation and random set theory. Mechanical Systems and Signal Processing, 2018, 100: 782-801.

[24] de Angelis M, Patelli E, Beer M. Advanced line sampling for efficient robust reliability analysis. Structural Safety, 2015, 52: 170-182.

[25] Wei P, Lu Z, Song J. Extended Monte Carlo simulation for parametric global sensitivity analysis and optimization. AIAA Journal, 2014, 52(4): 867-878.

[26] Wei P, Song J, Bi S, et al. Non-intrusive stochastic analysis with parameterized imprecise probability models: I. Performance estimation. Mechanical Systems and Signal Processing, 2019, 124: 349-368.

[27] Wei P, Song J, Bi S, et al. Non-intrusive stochastic analysis with parameterized imprecise probability models: II. Reliability and rare events analysis. Mechanical Systems and Signal Processing, 2019, 126: 227-247.

[28] Li G, Wang S W, Rosenthal C, et al. High dimensional model representations generated from low dimensional data samples. I. mp-Cut-HDMR. Journal of Mathematical Chemistry, 2001, 30(1): 1-30.

[29] Li G, Artamonov M, Rabitz H, et al. High-dimensional model representations generated from low order terms—lp-RS-HDMR. Journal of Computational Chemistry, 2003, 24(5): 647-656.

[30] Song J, Valdebenito M, Wei P, et al. Non-intrusive imprecise stochastic simulation by line sampling. Structural Safety, 2020, 84: 101936.

[31] Sofi A, Muscolino G, Giunta F. Propagation of uncertain structural properties described by imprecise Probability Density Functions via response surface method. Probabilistic Engineering Mechanics, 2020, 60: 103020.

[32] Schöbi R, Sudret B. Structural reliability analysis for p-boxes using multi-level meta-models. Probabilistic Engineering Mechanics, 2017, 48: 27-38.

[33] Freitag S, Edler P, Kremer K, et al. Multilevel surrogate modeling approach for optimization problems with polymorphic uncertain parameters. International Journal of Approximate Reasoning, 2020, 119:81-91.

[34] Koutsourelakis P S, Pradlwarter H J, Schuëller G I. Reliability of structures in high dimensions, part I: algorithms and applications. Probabilistic Engineering Mechanics, 2004, 19(4):409-417.

[35] Troffaes M. Imprecise Monte Carlo simulation and iterative importance sampling for the estimation of lower previsions. International Journal of Approximate Reasoning, 2018, 101: 31-48.

[36] Rasmussen C E, Williams CKI. Gaussian processes for machine learning. Cambridge: The MIT Press, 2006.

[37] Song J, Wei P, Valdebenito M, et al. Active learning line sampling for rare event analysis. Mechanical Systems and Signal Processing, under review.

[38] Bilal M A. Risk analysis in engineering and economics. Chapman and Hall/CRC, 2014.

[39] Hasofer A M and Lind N C. Exact and invariant second-moment code format. J. Engineering Mechanics Division, ASCE, 1974, 100(1), 111-121.

[40] Rackwitz R and Fiessler B. Structural reliability under combined load sequences. Computers & Structures, 1978, 9, 489-494.

[41] Depina I, Le TM, Fenton G, Eiksund G. Reliability analysis with metamodel line sampling. Structural Safety, 2016, 60:1-5.

[42] Tian L, Lu Z, Hao W. Moving least squares based sensitivity analysis for models with dependent variables. Applied Mathematical Modelling, 2013, 37(8):6097-6109.

[43] Valdebenito M A, Jensen H A, Hernandez H B, et al. Sensitivity estimation of failure probability applying line sampling. Reliability Engineering & System Safety, 2018, 171:99-111.