Bayesian active learning line sampling with log-normal process for rare-event probability estimation

Chao Dang[a,*], Marcos A. Valdebenito[b], Pengfei Wei[c], Jingwen Song[d], Michael Beer[a,e,f]

[a] *Institute for Risk and Reliability, Leibniz University Hannover, Callinstr. 34, Hannover 30167, Germany*
[b] *Chair for Reliability Engineering, TU Dortmund University, Leonhard-Euler-Str. 5, Dortmund 44227, Germany*
[c] *School of Power and Energy, Northwestern Polytechnical University, Xi'an 710072, PR China*
[d] *School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an 710072, PR China*
[e] *Institute for Risk and Uncertainty, University of Liverpool, Liverpool L69 7ZF, United Kingdom*
[f] *International Joint Research Center for Resilient Infrastructure & International Joint Research Center for Engineering Reliability and Stochastic Mechanics, Tongji University, Shanghai 200092, PR China*

Abstract

Line sampling (LS) is a powerful stochastic simulation method for structural reliability analysis, especially for assessing small failure probabilities. To further improve the performance of traditional LS, a Bayesian active learning idea has been successfully pursued. This work presents another Bayesian active learning alternative, called 'Bayesian active learning line sampling with log-normal process' (BAL-LS-LP), to traditional LS. In this method, we assign an LP prior instead of a Gaussian process prior over the distance function so as to account for its non-negativity constraint. Besides, the approximation error between the logarithmic approximate distance function and the logarithmic true distance function is assumed to follow a zero-mean normal distribution. The approximate posterior mean and variance of the failure probability are derived accordingly. Based on the posterior statistics of the failure probability, a learning function and a stopping criterion are developed to enable Bayesian active learning. In the numerical implementation of the proposed BAL-LS-LP method, the important direction can be updated on the fly without re-evaluating the distance function. Four numerical examples are studied to demonstrate the proposed method. Numerical results show that the proposed method can estimate extremely small failure probabilities with desired efficiency and accuracy.

Keywords: Structural reliability analysis; Line sampling; Bayesian active learning; Numerical uncertainty; Log-normal process; Gaussian process

1. Introduction

Probabilistic structural reliability analysis is concerned with the calculation of the failure probability, which is defined by a multiple integral of the form:

$$P_f = \int_{\mathbb{X}} I(g(\boldsymbol{x})) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{1}$$

where $\boldsymbol{X} = [X_1, X_2, \cdots, X_d]^\top \in \mathbb{X} \subseteq \mathbb{R}^d$ is a vector of d random variables; $f_{\boldsymbol{X}}(\boldsymbol{x})$ denotes the joint probability density function (PDF) of $\boldsymbol{X}$, which is assumed to be known; $g(\boldsymbol{X}) : \mathbb{X} \to \mathbb{R}$ is the so-called performance function (also known as limit state function) such that g takes negative values when the underlying system behaves unacceptably and vice versa; $I(\cdot)$ is the indicator function: $I(g(\boldsymbol{x})) = 1$ if $g(\boldsymbol{x}) < 0$ and $I(g(\boldsymbol{x})) = 0$ otherwise. Typically, Eq. (1) is not analytically tractable, leading to the development of various numerical methods over the years. One of the major challenges arises in assessing extremely low failure probabilities for computationally demanding problems, a situation commonly encountered in real-world scenarios.

Stochastic simulation techniques occupy a prominent position among the existing methods to estimate failure probabilities. As the most representative example, Monte Carlo simulation (MCS) has proved to be a universal method for reliability analysis. In many practical cases, however, the use of MCS is ruled out due to its low sampling efficiency, especially when the g-function is expensive-to-evaluate and the failure probability is extremely small. This leads to the development of more advanced stochastic simulation techniques that require less performance function evaluations. A partial list of such techniques includes importance sampling [1–3], subset simulation [4, 5], directional simulation [6, 7] and line sampling (LS) [8, 9]. Among these methods, the LS technique has attracted growing attention, especially when dealing with the challenging task of evaluating very small failure probabilities.

As a stand-alone simulation method, the invention of LS is attributed to the work of Koutsourelakis et al. [8, 10]. However, a similar but slightly different idea was exposed early in [11]. In the standard normal space, LS first identifies a unit vector that points towards the failure domain, which is the so-called important

*Corresponding author
Email address: chao.dang@irz.uni-hannover.de (Chao Dang)

direction $\boldsymbol{\alpha}$. Then, the d-dimensional failure probability integral is reformulated into a nested integral, with the inner being a one-dimensional conditional integral along $\boldsymbol{\alpha}$, and the outer being a $(d-1)$-dimensional integral over the hyperplane orthogonal to $\boldsymbol{\alpha}$. In practice, the inner integral conditional on a point on the hyperplane is solved by means of a root-finding algorithm, while the outer integral is approximated by the MCS. The basic idea of LS can be understood as follows: to explore the failure domain by using random but parallel lines instead of random points. As a result, the simulation can be focused on the region where failure is most likely to occur. This makes it possible to provide an accurate estimate for the failure probability with less g-function calls than the crude MCS. The LS method has been shown to be particularly suitable for assessing small failure probabilities of weakly and moderately nonlinear problems.

The traditional LS has been enhanced in various ways to improve its performance and applicability. In [12–14], efforts have been made to efficiently adjust the important direction and/or process lines. These methods still rely on the direct use of MCS to address the outer integral, which can lead to unnecessary computational cost. To alleviate this problem, LS can be used in combination with active-learning-driven surrogate models [15, 16]. Beyond its original purpose, the application scope of the traditional LS has also been expanded greatly. Examples include but not limited to reliability sensitivity analysis [17–19], imprecise reliability analysis [12, 20–24], reliability-based design optimization [25] and system reliability analysis [26].

More recently, the first author and his collaborators have attempted to interpret the reliability analysis problem as a Bayesian inference problem and then to further frame reliability analysis in a Bayesian active learning setting [27–29]. Compared with the existing active learning reliability methods, the developed Bayesian active learning methods put more emphasis on using Bayesian principles, and hence have many promising advantages. For example, the uncertainty about the failure probability estimate can be modeled explicitly, based on which two critical components for active learning, i.e., learning function and stopping criterion, can be developed. The Bayesian active learning idea has also been pursued in the context of LS for reliability analysis. In [30], a method, called 'partially Bayesian active learning line sampling' (PBAL-LS), has been developed. This is a first attempt to approach the failure probability integral in LS from a Bayesian active learning perspective, where the posterior mean and an upper bound of the posterior variance of the

failure probability are available. The exact expression of the posterior variance of the failure probability is then given in [31], which allows for a more complete uncertainty characterization of the failure probability in terms of second-order statistics. The resulting method is termed 'Bayesian active learning line sampling' (BAL-LS), which can be regarded as an enhanced version of PBAL-LS. However, both PBAL-LS and BAL-LS only account for the discretization error, which is only one source of uncertainty than preventing from learning the true value of the failure probability. Actually, there is another kind of numerical uncertainty, i.e., the approximation error, due to the numerical approximation of the inner integral. In addition, the non-negativity constraint of the distance function is disregarded both in PBAL-LS, as well as in BAL-LS. Ignoring these two factors (i.e., approximation error and non-negativity constraint) may lead to a less accurate failure probability estimate.

The goal of this work is to simultaneously consider the discretization error, the approximation error, and the non-negativity constraint in a strategic manner when approaching the Bayesian active learning idea in the context of LS for structural reliability analysis. For this purpose, the distance function associated with the inner integral of LS is assigned to a log-normal process (LP) prior in order to explicitly express the non-negativity constraint, instead of a Gaussian process (GP) as used in PBAL-LS and BAL-LS. Using a trick, the prior assumption can be equivalent to placing a GP prior over the logarithmic distance function. Further, the approximation error between the logarithmic distance function and the logarithmic true distance function is assumed to follow a zero-mean normal distribution. Conditional on some observations arising from evaluating the logarithmic distance function at several locations, the posterior distribution of the logarithmic distance function follows a GP. This implies that the posterior distribution of the distance function follows an LP. The posterior mean and variance of the failure probability can be derived based on a moment-matched GP approximation of the LP posterior of the distance function. To enable Bayesian active learning, a learning function and a stopping criterion are developed in light of the uncertainty representation of the failure probability.

The rest of this paper is structured as follows. In Section 2, two related methods are briefly reviewed. The proposed method is presented in Section 3. Four numerical examples are investigated in Section 4 to

demonstrate the proposed method. Section 5 gives some concluding remarks.

2. Brief review of two related methods

In this section, two methods in close relation to our development, i.e., traditional LS [8] and BAL-LS [31], are briefly introduced. To do so, we first reformulate our reliability analysis problem in the standard normal space. Assume that a reversible transformation T can be applied to transforming the basic random vector $\boldsymbol{X}$ into a standard normal vector $\boldsymbol{U} = [U_1, U_2, \cdots, U_d]^\top$, i.e., $\boldsymbol{U} = T(\boldsymbol{X})$. This makes it possible to define a transformed performance function $\mathcal{G}(\boldsymbol{U}) := g(T^{-1}(\boldsymbol{U}))$.

2.1. Traditional line sampling

Traditional LS begins by identifying an important direction $\boldsymbol{\alpha}$, see Fig. 1. It is a unit vector pointing to the failure domain in the standard normal space, i.e., $\mathcal{F} = \{\boldsymbol{u} \in \mathcal{U} : \mathcal{G}(\boldsymbol{u}) < 0\}$. The identification of $\boldsymbol{\alpha}$ can be achieved by using the, e.g., gradient information of $\mathcal{G}$ at a certain point [12], design point by the first-order reliability method [32], or failure samples generated by the Markov Chain Monte Carlo [32].

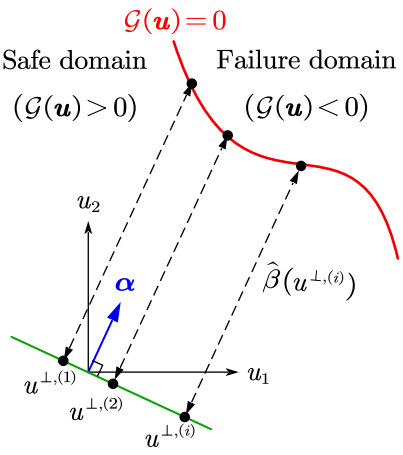


Figure 1: Illustration of the traditional LS in two dimensions.

Under the premise that the failure domain $\mathcal{F}$ is a half-open region, the failure probability can be formulated as:

$$P_f = \int_{\mathbb{R}^{d-1}} \Phi(-\beta(\boldsymbol{u}^\perp))\phi_{\boldsymbol{U}^\perp}(\boldsymbol{u}^\perp)\mathrm{d}\boldsymbol{u}^\perp, \tag{2}$$

where $\boldsymbol{u}^{\perp}$ denotes a realization of a $(d-1)$-dimensional standard normal vector $\boldsymbol{U}^{\perp} = [U_1^{\perp}, U_2^{\perp}, \cdots, U_{d-1}^{\perp}]^{\top}$ such that $\boldsymbol{U} = \boldsymbol{\alpha}U^{\parallel} + \boldsymbol{B}\boldsymbol{U}^{\perp}$; $U^{\parallel}$ is a standard normal variable parallel to $\boldsymbol{\alpha}$; $\boldsymbol{B}$ is a $d \times (d-1)$ matrix containing $(d-1)$ orthogonal basis vectors for the hyperplane perpendicular to $\boldsymbol{\alpha}$; $\beta(\boldsymbol{u}^{\perp})$ returns the Euclidean distance between $\boldsymbol{u}^{\perp}$ and the limit state surface $\mathcal{G} = 0$ along $\boldsymbol{\alpha}$; $\Phi(\cdot)$ is the cumulative distribution function (CDF) of the standard normal distribution; $\phi_{\boldsymbol{U}^{\perp}}(\cdot)$ is the joint PDF of $\boldsymbol{U}^{\perp}$. The standard normal vector $\boldsymbol{U}' = [U^{\parallel}; \boldsymbol{U}^{\perp}]$ can be interpreted as a rotated counterpart of $\boldsymbol{U}$, and the matrix $\boldsymbol{R} = [\boldsymbol{\alpha}, \boldsymbol{B}]$ turns out to be the rotational matrix such that $\boldsymbol{U} = \boldsymbol{R}\boldsymbol{U}'$.

In traditional LS, the failure probability integral defined in Eq. (2) is solved by the crude MCS in conjugation with a root-finding technique. The MCS estimator of P_f is given by:

$$\hat{P}_f = \frac{1}{N} \sum_{i=1}^{N} \Phi(-\hat{\beta}(\boldsymbol{u}^{\perp,(i)})), \tag{3}$$

where $\left\{\boldsymbol{u}^{\perp,(i)}\right\}_{i=1}^{N}$ is a set of N random samples generated according to $\phi_{\boldsymbol{U}^{\perp}}(\cdot)$; $\hat{\beta}(\boldsymbol{u}^{\perp,(i)})$ denotes the approximate result of $u^{\parallel}$ subject to $\mathcal{G}(\boldsymbol{\alpha}u^{\parallel} + \boldsymbol{B}\boldsymbol{u}^{\perp,(i)}) = 0$ (see Fig. 1), which can be obtained by a suitable root-finding algorithm such as polynomial interpolation [8] and Newton's method [12]. The crude MCS method is a robust technique for approximating the integral (Eq. (2)). However, its convergence rate is quite low. In addition, the approximation error of $\hat{\beta}(\boldsymbol{u}^{\perp,(i)})$ is not considered when forming the estimate for the failure probability.

2.2. Bayesian active learning line sampling

BAL-LS provides a Bayesian active learning alternative to the traditional LS described above. The basic ideas of BAL-LS are as follows. In contrast to frequentist inference, estimating the failure probability integral defined in Eq. (2) is first treated as a Bayesian inference problem, where the discretzation error is considered as a kind of epistemic uncertainty. Then, the induced probabilistic uncertainty in the failure probability allows the development of an active learning scheme so as to reduce the epistemic uncertainty.

Following a Bayesian approach, BAL-LS places a GP prior over the β-function:

$$\beta_0(\boldsymbol{u}^{\perp}) \sim \mathcal{GP}(m_{\beta_0}(\boldsymbol{u}^{\perp}), k_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{4}$$

where β_0 denotes the prior distribution of β; $m_{\beta_0}(\boldsymbol{u}^{\perp})$ is the prior mean function; $k_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$ is the prior covariance function. The prior mean and covariance functions are assumed to be a constant and squared exponential kernel, respectively.

Suppose that we now obtain a training dataset $\boldsymbol{\mathcal{D}} = \left\{\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{\mathcal{Y}}\right\}$ by evaluating the β-function, where $\boldsymbol{\mathcal{U}}^{\perp} = \left\{\boldsymbol{u}^{\perp,(j)}\right\}_{j=1}^{n}$ is a $(d-1) \times n$ design matrix with its j-th column being a observation point $\boldsymbol{u}^{\perp,(j)}$, and $\boldsymbol{\mathcal{Y}} = \left\{y^{(j)}\right\}_{j=1}^{n}$ is a column vector with its j-th element being $y^{(j)} = \beta(\boldsymbol{u}^{\perp,(j)})$. Conditioning the GP prior on the data $\boldsymbol{\mathcal{D}}$ gives a GP posterior of β:

$$\beta_n(\boldsymbol{u}^{\perp}) \sim \mathcal{GP}(m_{\beta_n}(\boldsymbol{u}^{\perp}), k_{\beta_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{5}$$

where β_n denotes the posterior distribution of β conditional on $\boldsymbol{\mathcal{D}}$; $m_{\beta_n}(\boldsymbol{u}^{\perp})$ and $k_{\beta_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$ are the posterior mean and covariance functions respectively, which can be expressed in closed form [33]:

$$m_{\beta_n}(\boldsymbol{u}^{\perp}) = m_{\beta_0}(\boldsymbol{u}^{\perp}) + \boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{\mathcal{U}}^{\perp})^{\top} \boldsymbol{K}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{\mathcal{U}}^{\perp})^{-1} (\boldsymbol{\mathcal{Y}} - \boldsymbol{m}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp})), \tag{6}$$

$$k_{\beta_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) = k_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) - \boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{\mathcal{U}}^{\perp})^{\top} \boldsymbol{K}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{\mathcal{U}}^{\perp})^{-1} \boldsymbol{k}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{u}^{\perp\prime}), \tag{7}$$

where $\boldsymbol{m}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp}) = [m_{\beta_0}(\boldsymbol{u}^{\perp,(1)}), m_{\beta_0}(\boldsymbol{u}^{\perp,(2)}), \cdots, m_{\beta_0}(\boldsymbol{u}^{\perp,(n)})]^{\top}$; $\boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{\mathcal{U}}^{\perp}) = [k_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp,(1)}), k_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp,(2)}), \cdots, k_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp,(n)})]^{\top}$; $\boldsymbol{k}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{u}^{\perp\prime}) = [\boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp,(1)}, \boldsymbol{u}^{\perp\prime}), \boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp,(2)}, \boldsymbol{u}^{\perp\prime}), \cdots, \boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp,(n)}, \boldsymbol{u}^{\perp\prime})]^{\top}$; $\boldsymbol{K}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{\mathcal{U}}^{\perp})$ is an $n \times n$ covariance matrix with (i, j)-th entry being $k_{\beta_0}(\boldsymbol{u}^{\perp,(i)}, \boldsymbol{u}^{\perp,(j)})$.

Conditional on $\boldsymbol{\mathcal{D}}$, the posterior mean and covariance functions of $\Phi(-\beta(\boldsymbol{u}^{\perp}))$ can also be derived as [30, 31]:

$$m_{\Phi_n(-\tilde{\beta})}\left(\boldsymbol{u}^{\perp}\right) = \Phi\left(\frac{-m_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}}\right), \tag{8}$$

$$\begin{aligned} k_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) =& \Psi\left(\begin{bmatrix} m_{\beta_n}\left(\boldsymbol{u}^{\perp}\right) \\ m_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right) \end{bmatrix}; \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)+1 & k_{\beta_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) \\ k_{\beta_n}(\boldsymbol{u}^{\perp\prime}, \boldsymbol{u}^{\perp}) & \sigma^2_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)+1 \end{bmatrix}\right) \\ & - \Phi\left(\frac{m_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}}\right) \Phi\left(\frac{m_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)}{\sqrt{1+\sigma^2_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)}}\right), \end{aligned} \tag{9}$$

where $\sigma^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)$ is the posterior variance function of β, i.e., $\sigma^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right) = k_{\beta_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp})$; Ψ denotes the bivariate normal CDF.

The posterior mean and variance of the failure probability conditional on $\boldsymbol{\mathcal{D}}$ turn out to be:

$$m_{P_{f,n}} = \int_{\mathbb{R}^{d-1}} m_{\Phi_n(-\beta)}\left(\boldsymbol{u}^{\perp}\right) \phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}) \mathrm{d}\boldsymbol{u}^{\perp}, \tag{10}$$

$$\sigma^2_{P_{f,n}} = \int_{\mathbb{R}^{d-1}} \int_{\mathbb{R}^{d-1}} k_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) \phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}) \phi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime}) \mathrm{d}\boldsymbol{u}^{\perp} \mathrm{d}\boldsymbol{u}^{\perp\prime}. \tag{11}$$

Note that the posterior distribution of the failure probability (denoted as $P_{f,n}$) reflects our uncertainty about the true failure probability value, where the uncertainty is due to the discretization of the β-function. The posterior mean $m_{P_{f,n}}$ can be used as a point estimate of the failure probability, while the posterior variance $\sigma^2_{P_{f,n}}$ lends itself as a natural convergence diagnostic. Due to their analytical intractability, $m_{\tilde{P}_{f,n}}$ and $\sigma^2_{\tilde{P}_{f,n}}$ have to be numerically approximated.

Based on the uncertainty representation of the failure probability, the above Bayesian inference framework can also be equipped with the use of active learning, which is the so-called Bayesian active learning. The stopping criterion for active learning is defined as:

$$\frac{\sigma_{P_{f,n}}}{m_{P_{f,n}}} < \delta, \tag{12}$$

where δ is a user-specified tolerance value. If the stopping criterion is not satisfied, the next best point to query the β-function can be identified by maximizing the following learning function, called 'posterior standard deviation contribution' (PSDC):

$$\mathrm{PSDC}\left(\boldsymbol{u}^{\perp}\right) = \phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}) \times \int_{\mathcal{U}^{\perp}} k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) \phi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime}) \mathrm{d}\boldsymbol{u}^{\perp\prime}, \tag{13}$$

where the integral term is estimated by means of a numerical integration scheme.

In addition, another salient feature of BAL-LS is that it can adjust the important direction on the fly during its course. This means that it is not necessary to specify an optimal important direction at the very beginning, which is usually difficult or expensive to obtain. The reader is referred to [31] for more information about BAL-LS.

However, the BAL-LS method also has some limitations that motivate the present work. First, BAL-LS directly places a GP prior over the β-function. This can be a poor choice as it is unable to express the non-negativity of β. Second, the numerical error introduced by the numerical approximation of $y^{(j)} = \beta(\boldsymbol{u}^{\perp,(j)})$ is also ignored in BAL-LS, which may result in a poor failure probability estimate.

3. Bayesian active learning line sampling with log-normal process

This section introduces another Bayesian active learning alternative, i.e., BAL-LS-LP, to the traditional LS, in order to address the aforementioned limitations of BAL-LS. The proposed method starts by assigning an LP prior, instead of a GP prior, over the β-function, which allows explicitly taking into account its non-negativity constraint. Furthermore, to account for the approximation error of the β-function resulting from the root-finding procedure, the error term between the log approximate distance function and the log true distance function is assumed to follow a zero-mean normal distribution. The approximate posterior mean and variance of the failure probability are obtained by using a moment-matched GP approximation of the LP posterior of the distance function. Based on the quantified uncertainty, two critical components for active learning, i.e., stopping criterion and learning function, are proposed accordingly.

3.1. Theoretical development

3.1.1. Prior distributions

Let $\hat{\beta}(\boldsymbol{u}^{\perp})$ denote the approximation of $\beta(\boldsymbol{u}^{\perp})$. In this study, we assume that the error between $\log\left(\hat{\beta}(\boldsymbol{u}^{\perp})\right)$ and $\log\left(\beta(\boldsymbol{u}^{\perp})\right)$ is additive:

$$\log\left(\hat{\beta}(\boldsymbol{u}^{\perp})\right) = \log\left(\beta(\boldsymbol{u}^{\perp})\right) + \varepsilon, \tag{14}$$

where ε represents the error term. For notational simplicity, we denote $\log\left(\hat{\beta}(\boldsymbol{u}^{\perp})\right)$ and $\log\left(\beta(\boldsymbol{u}^{\perp})\right)$ as $\hat{l}(\boldsymbol{u}^{\perp})$ and $l(\boldsymbol{u}^{\perp})$ respectively. It follows that Eq. (14) can be rewritten as:

$$\hat{l}(\boldsymbol{u}^{\perp}) = l(\boldsymbol{u}^{\perp}) + \varepsilon. \tag{15}$$

Considering the non-negativity of β, our prior beliefs about it are encoded by an LP model:

$$\beta_0(\boldsymbol{u}^{\perp}) \sim \mathcal{LP}(\boxed{m}_{\beta_0}(\boldsymbol{u}^{\perp}), \boxed{k}_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{16}$$

where $\boxed{m}_{\beta_0}(\boldsymbol{u}^{\perp})$ and $\boxed{k}_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$ denote the prior mean and covariance functions receptively, which can completely characterize the LP model. By using a trick, we equate the LP prior over β to a GP prior over $l(\boldsymbol{u}^{\perp})$:

$$l_0(\boldsymbol{u}^{\perp}) \sim \mathcal{GP}(m_{l_0}(\boldsymbol{u}^{\perp}), k_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{17}$$

where l_0 denotes the prior distribution of l; $m_{l_0}(\boldsymbol{u}^{\perp})$ and $k_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$ are the prior mean and covariance functions, respectively. Without loss of generality, the prior mean and covariance functions are chosen as a constant and as a squared exponential kernel, respectively:

$$m_{l_0}(\boldsymbol{u}^{\perp}) = b, \tag{18}$$

$$k_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) = \sigma_k^2 \exp\left(-\frac{1}{2}(\boldsymbol{u}^{\perp} - \boldsymbol{u}^{\perp\prime})^{\top} \boldsymbol{\Sigma}^{-1} (\boldsymbol{u}^{\perp} - \boldsymbol{u}^{\perp\prime})\right), \tag{19}$$

where $b \in \mathbb{R}$; $\sigma_k > 0$ is the process standard deviation; $\boldsymbol{\Sigma} = \operatorname{diag}\left(w_1^2, w_2^2, \cdots, w_{d-1}^2\right)$ with $w_i > 0$ being the length scale in the i-th dimension.

In order to account for the difference between l and $\hat{l}$, the error term should also be properly modeled. In this study, we assume that the additive error ε follows a zero-mean normal distribution:

$$\varepsilon \sim \mathcal{N}(0, \sigma_{\varepsilon}^2), \tag{20}$$

where $\sigma_{\varepsilon} > 0$ is the standard deviation of ε. The mean is taken as zero because we believe that the average error over the location $\boldsymbol{u}^{\perp}$ is not very biased.

3.1.2. Hyper-parameters tuning

Our prior assumptions expressed in Eqs. (18)-(20) depend on a set of $d+2$ parameters $\boldsymbol{\Omega} = \{b, \sigma_k, w_1, w_2, \cdots, w_{d-1}, \sigma_{\varepsilon}\}^{\top}$ which are referred as hyper-parameters. Given a noisy training dataset $\tilde{\boldsymbol{\mathcal{D}}} = \left\{\boldsymbol{\mathcal{U}}^{\perp}, \tilde{\boldsymbol{\mathcal{Z}}}\right\}$, where $\boldsymbol{\mathcal{U}}^{\perp} = \left\{\boldsymbol{u}^{\perp,(j)}\right\}_{j=1}^{n}$ is a $(d-1)\times n$ design matrix with its j-th column being a design point $\boldsymbol{u}^{\perp,(j)}$, and $\tilde{\boldsymbol{\mathcal{Z}}} = \left\{\tilde{z}^{(j)}\right\}_{j=1}^{n}$ is a column vector with its j-th element being $\tilde{z}^{(j)} = \log\left(\tilde{\beta}(\boldsymbol{u}^{\perp,(j)})\right)$. The hyper-parameters can be tuned by maximizing the log marginal likelihood:

$$\boldsymbol{\Omega} = \arg\max \log p(\tilde{\boldsymbol{\mathcal{Z}}} | \boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{\Omega}), \tag{21}$$

in which

$$\log p(\tilde{\boldsymbol{\mathcal{Z}}} | \boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{\Omega}) = -\frac{1}{2}\left[\log\left(|\boldsymbol{K}_{l_0} + \sigma_{\varepsilon}\boldsymbol{I}|\right) + \left(\tilde{\boldsymbol{\mathcal{Z}}} - b\right)^{\top} \left(\boldsymbol{K}_{l_0} + \sigma_{\varepsilon}\boldsymbol{I}\right)^{-1} \left(\tilde{\boldsymbol{\mathcal{Z}}} - b\right) + n\log(2\pi)\right], \tag{22}$$

where $\boldsymbol{K}_{l_0}$ is an $n \times n$ matrix whose (i,j)-th entry is $k_{l_0}(\boldsymbol{u}^{\perp,(i)}, \boldsymbol{u}^{\perp,(j)})$; $\boldsymbol{I}$ is an $n \times n$ identity matrix.

3.1.3. Posterior distributions

The posterior distribution of l conditional on $\tilde{\boldsymbol{\mathcal{D}}}$ is also a GP:

$$l_n(\boldsymbol{u}^{\perp}) \sim \mathcal{GP}(m_{l_n}(\boldsymbol{u}^{\perp}), k_{l_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{23}$$

where l_n denotes the posterior distribution of l after seeing n noisy observations; $m_{l_n}(\boldsymbol{u}^{\perp})$ and $k_{l_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$ are the posterior mean and covariance functions respectively, which can be further expressed as [33]:

$$m_{l_n}(\boldsymbol{u}^{\perp}) = m_{l_0}(\boldsymbol{u}^{\perp}) + \boldsymbol{k}_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{\mathcal{U}}^{\perp})^{\top} \left(\boldsymbol{K}_{l_0} + \sigma_{\varepsilon}\boldsymbol{I}\right)^{-1} \left(\tilde{\boldsymbol{\mathcal{Z}}} - \boldsymbol{m}_{l_0}(\boldsymbol{\mathcal{U}}^{\perp})\right), \tag{24}$$

$$k_{l_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) = k_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) - \boldsymbol{k}_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{\mathcal{U}}^{\perp})^{\top} \left(\boldsymbol{K}_{l_0} + \sigma_{\varepsilon}\boldsymbol{I}\right)^{-1} \boldsymbol{k}_{l_0}(\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{u}^{\perp,\prime}), \tag{25}$$

where $\boldsymbol{m}_{l_0}(\boldsymbol{\mathcal{U}}^{\perp}) = [m_{l_0}(\boldsymbol{u}^{\perp,(1)}), m_{l_0}(\boldsymbol{u}^{\perp,(2)}), \cdots, m_{l_0}(\boldsymbol{u}^{\perp,(n)})]^{\top}$; $\boldsymbol{k}_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{\mathcal{U}}^{\perp}) = [k_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp,(1)}), k_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp,(2)}), \cdots, k_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp,(n)})]^{\top}$; $\boldsymbol{k}_{l_0}(\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{u}^{\perp\prime}) = [\boldsymbol{k}_{l_0}(\boldsymbol{u}^{\perp,(1)}, \boldsymbol{u}^{\perp\prime}), \boldsymbol{k}_{l_0}(\boldsymbol{u}^{\perp,(2)}, \boldsymbol{u}^{\perp\prime}), \cdots, \boldsymbol{k}_{l_0}(\boldsymbol{u}^{\perp,(n)}, \boldsymbol{u}^{\perp\prime})]^{\top}$.

It is readily noticed that the induced posterior distribution for β conditional on $\tilde{\boldsymbol{\mathcal{D}}}$ follows an LP:

$$\beta_n(\boldsymbol{u}^{\perp}) \sim \mathcal{LP}(\boxed{m}_{\beta_n}(\boldsymbol{u}^{\perp}), \boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{26}$$

where β_n denotes the posterior distribution of β; $\boxed{m}_{\beta_n}(\boldsymbol{u}^{\perp})$ and $\boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$ are the posterior mean and covariance functions respectively, which can be derived as:

$$\boxed{m}_{\beta_n}(\boldsymbol{u}^{\perp}) = \exp\left(m_{l_n}(\boldsymbol{u}^{\perp}) + \frac{1}{2}\sigma^2_{l_n}(\boldsymbol{u}^{\perp})\right), \tag{27}$$

$$\boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) = \left[\exp\left(k_{l_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})\right) - 1\right] \exp\left(m_{l_n}(\boldsymbol{u}^{\perp}) + m_{l_n}(\boldsymbol{u}^{\perp\prime}) + \frac{1}{2}\left(\sigma^2_{l_n}(\boldsymbol{u}^{\perp}) + \sigma^2_{l_n}(\boldsymbol{u}^{\perp\prime})\right)\right), \tag{28}$$

where $\sigma^2_{l_n}(\cdot) = k_{l_n}(\cdot, \cdot)$.

With the LP posterior of β, it is challenging to derive the resulting posterior distribution of $\Phi(-\beta)$ and even its posterior mean and covariance functions. This in turn prevents us from obtaining the posterior statistics of the failure probability P_f. Inspired by [34, 35], we adopt an approximation scheme for β_n in order to avoid the lack of traceability. Specifically, the GP posterior $\mathcal{LP}(\boxed{m}_{\beta_n}(\boldsymbol{u}^{\perp}), \boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}))$ is approximated by a moment-matched GP, i.e., $\mathcal{GP}(\boxed{m}_{\beta_n}(\boldsymbol{u}^{\perp}), \boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}))$. Note that the accuracy of the approximation depends on the specific characteristics of the LP. If the LP deviates significantly from a GP, the moment-matched GP approximation may become less accurate. However, according to our experience,

this approximation can provide fairly good results in most cases. Besides, another advantage of such an approximation is that we can directly exploit the previous results given in BAL-LS [31] when inferring the posterior statistics of both $\Phi(-\beta)$ and P_f.

Under the Gaussian approximation, the approximate posterior mean and covariance functions of $\Phi(-\beta)$ conditional on $\tilde{\boldsymbol{\mathcal{D}}}$ can be given by:

$$\boxed{m}_{\tilde{\Phi}_n(-\beta)}\left(\boldsymbol{u}^{\perp}\right) = \Phi\left(\frac{-\boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\boxed{\sigma}^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}}\right), \tag{29}$$

$$\begin{aligned}\boxed{k}_{\tilde{\Phi}_n(-\beta)}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) =& \Psi\left(\begin{bmatrix}\boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)\\ \boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)\end{bmatrix};\begin{bmatrix}0\\0\end{bmatrix},\begin{bmatrix}\boxed{\sigma}^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)+1 & \boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime})\\ \boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp\prime},\boldsymbol{u}^{\perp}) & \boxed{\sigma}^2_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)+1\end{bmatrix}\right)\\ &-\Phi\left(\frac{\boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\boxed{\sigma}^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)\Phi\left(\frac{\boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)}{\sqrt{1+\boxed{\sigma}^2_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)}}\right),\end{aligned} \tag{30}$$

where $\boxed{\sigma}^2_{\beta_n}(\cdot) = \boxed{k}_{\beta_n}(\cdot,\cdot)$. For proofs of Eqs. (29) and (30), please refer to [31]. Note that Eqs. (29) and (30) are respectively different from Eqs. (8) and (9) in essence due to the differences in the mean, variance and covariance functions involved.

As a consequence, we can approximate the posterior mean and variance of P_f by:

$$\boxed{m}_{P_{f,n}} = \int_{\mathbb{R}^{d-1}} \boxed{m}_{\Phi_n(-\beta)}\left(\boldsymbol{u}^{\perp}\right)\phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}\boldsymbol{u}^{\perp}, \tag{31}$$

$$\boxed{\sigma}^2_{P_{f,n}} = \int_{\mathbb{R}^{d-1}}\int_{\mathbb{R}^{d-1}} \boxed{k}_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime})\phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\phi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime})\mathrm{d}\boldsymbol{u}^{\perp}\mathrm{d}\boldsymbol{u}^{\perp\prime}. \tag{32}$$

Eqs. (31) and (32) can be proved by using the Fubini's theorem, hence the proofs are omitted. It is noted that Eqs. (31) and (32) are essentially different from Eqs. (10) and (11) respectively due to the differences in the integrands involved. The uncertainty in the failure probability summarizes the numerical uncertainty resulting from both the discretization error (i.e., discretizing the l-function at discrete locations) and the approximation error (i.e., approximating the value $l(\boldsymbol{u}^{\perp})$). The approximate posterior mean $\boxed{m}_{P_{f,n}}$ can be used as a point estimate of the failure probability, while the approximate posterior variance $\boxed{\sigma}^2_{P_{f,n}}$ provides a measure for the uncertainty.

3.1.4. Estimating the approximate posterior mean and variance of the failure probability

The approximate posterior mean and variances of the failure probability defined in (31) and (32) have to be numerically approximated due to their analytical intractability. Following the same way in BAL-LS, we employ the standard deviation-amplified importance sampling (SDA-IS) originally developed in [29]. The SDA-IS estimators of $\boxed{m}_{P_{f,n}}$ and $\boxed{\sigma}^2_{P_{f,n}}$ can be given by:

$$\hat{\boxed{m}}_{P_{f,n}} = \frac{1}{N}\sum_{q=1}^{N} \Phi\left(\frac{-\boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp,(q)}\right)}{\sqrt{1+\boxed{\sigma}^2_{\beta_n}\left(\boldsymbol{u}^{\perp,(q)}\right)}}\right)\frac{\phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp,(q)})}{\phi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp,(q)})}, \tag{33}$$

$$\hat{\boxed{\sigma}}^2_{P_{f,n}} = \frac{1}{N}\sum_{i=1}^{N} \boxed{k}_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp,(q)},\boldsymbol{u}^{\perp\prime,(q)})\frac{\phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp,(q)})\phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp\prime,(q)})}{\phi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp,(q)})\phi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp\prime,(q)})}, \tag{34}$$

where $\left\{\boldsymbol{u}^{\perp,(q)}\right\}_{q=1}^{N}$ and $\left\{\boldsymbol{u}^{\perp\prime,(q)}\right\}_{q=1}^{N}$ are two sets of N random samples generated according to $\phi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp})$ and $\phi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp\prime})$, respectively; $\phi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp})$ is the SDA-IS density of the form $\phi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp}) = \prod_{i=1}^{d-1}\phi_{U_i^{\perp},\lambda}(u_i^{\perp})$, in which

$$\phi_{U_i^{\perp},\lambda}(u_i^{\perp}) = \frac{1}{\lambda\sqrt{2\pi}}\exp\left(-\frac{u_i^{\perp,2}}{2\lambda^2}\right), \tag{35}$$

where $\lambda > 1$ is the amplification factor.

The corresponding variances of the above two estimators can be expressed as:

$$\mathbb{V}\left[\hat{\boxed{m}}_{P_{f,n}}\right] = \frac{1}{N(N-1)}\sum_{q=1}^{N}\left[\Phi\left(\frac{-\boxed{m}_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(q)}\right)}{\sqrt{1+\boxed{\sigma}^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(q)}\right)}}\right)\frac{\phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp,(q)})}{\phi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp,(q)})} - \hat{\boxed{m}}_{P_{f,n}}\right]^2, \tag{36}$$

$$\mathbb{V}\left[\hat{\boxed{\sigma}}^2_{P_{f,n}}\right] = \frac{1}{N(N-1)}\sum_{q=1}^{N}\left[\boxed{k}_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp,(q)},\boldsymbol{u}^{\perp\prime,(q)})\frac{\phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp,(q)})\phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp\prime,(q)})}{\phi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp,(q)})\phi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp\prime,(q)})} - \hat{\boxed{\sigma}}^2_{P_{f,n}}\right]^2. \tag{37}$$

In order to reduce the computational burden and guarantee the accuracy of the results, the SDA-IS is implemented in a step-by-step manner, rather than all at once. That is, we generate samples incrementally (e.g., 1×10^4 at once) until $\sqrt{\mathbb{V}\left[\hat{\boxed{m}}_{P_{f,n}}\right]}/\hat{\boxed{m}}_{P_{f,n}} < \tau_1$ and $\sqrt{\mathbb{V}\left[\hat{\boxed{\sigma}}^2_{P_{f,n}}\right]}/\hat{\boxed{\sigma}}^2_{P_{f,n}} < \tau_2$ are satisfied, where τ_1 and τ_2 are two user-specified thresholds.

3.1.5. Stopping criterion and learning function

The above Bayesian framework can be further cast in an active learning setting based on the uncertainty modeling of the failure probability. Two principal components for active learning are the stopping criterion and learning function.

Supposing that we are at the stage with n noisy observations, the stopping criterion can be defined in terms of the estimated COV of the posterior failure probability such that:

$$\frac{\hat{\boxed{\sigma}}_{P_{f,n}}}{\hat{\boxed{m}}_{P_{f,n}}} < \eta, \tag{38}$$

where η is a tolerance value. The stopping criterion in Eq. (38) should be met twice in a row in order to avoid fake convergence.

If the stopping criterion is not reached, then the training dataset should be enriched so as to further reduce the epistemic uncertainty in the failure probability. For this propose, a learning function, called 'approximate posterior standard deviation contribution' (APSDC), is first introduced:

$$\mathrm{APSDC}\left(\boldsymbol{u}^{\perp}\right) = \phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}) \times \int_{\mathbb{R}^{d-1}} \boxed{k}_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})\phi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime})\mathrm{d}\boldsymbol{u}^{\perp\prime}. \tag{39}$$

Note that $\int_{\mathbb{R}^{d-1}} \mathrm{APSDC}\left(\boldsymbol{u}^{\perp}\right)\mathrm{d}\boldsymbol{u}^{\perp} = \boxed{\sigma}^2_{\tilde{P}_{f,n}}$ holds true. Hence, the APSDC function provides a measure of the contribution of the epistemic uncertainty at site $\boldsymbol{u}^{\perp}$ to the approximate posterior variance (or standard deviation) of the failure probability. The intractable integral term involved in the APSDC function can be approximated by a numerical integration scheme such that:

$$\widehat{\mathrm{APSDC}}\left(\boldsymbol{u}^{\perp}\right) = \phi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\frac{1}{M}\sum_{p=1}^{M} \boxed{k}_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime,(p)}), \tag{40}$$

where $\left\{\boldsymbol{u}^{\perp\prime,(p)}\right\}_{p=1}^{M}$ is a set of M integration points, which are generated according to $\phi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime})$ using Sobol sequence in this study. To obtain good results, the number of integration points M should be as large as possible. However, a too large M will result in a non-negligible computational load when optimizing the learning function.

The next best point $\boldsymbol{u}^{\perp,(n+1)}$ to query the l-function can be identified by maximizing the estimated APSDC function such that:

$$\boldsymbol{u}^{\perp,(n+1)} = \underset{\boldsymbol{u}^{\perp}\in\mathbb{R}^{d-1}}{\arg\max}\, \widehat{\mathrm{APSDC}}\left(\boldsymbol{u}^{\perp}\right), \tag{41}$$

where a global optimization algorithm, i.e., particle swarm optimization, can be used. As soon as $\boldsymbol{u}^{\perp,(n+1)}$ is selected, $\tilde{l}(\boldsymbol{u}^{\perp,(n+1)})$ should be evaluated by an appropriate algorithm.

3.2. Step-by-step procedure

During the theoretical development of the proposed BAL-LS-LP method, the important direction is assumed to be fixed. However, it is not necessary to do so and the important direction can be updated as well. To be specific, the BAL-LS-LP algorithm will start with a sub-optimal important direction, and then update to a new one once a more probable is found during the active learning phase. In addition, how to evaluate the l function is another important aspect that remains unmentioned. These issues will be addressed as the steps of the proposed method are presented.

The procedure for implementing the proposed BAL-LS-LP method is summarized below in six main steps, and illustrated with a flowchart in Fig. 2.

Step 1: Specifying an initial important direction

The proposed method is initialized with an important direction $\boldsymbol{\alpha}^{(0)}$, which can be a rough guess and does not need to be optimal. In this study, the initial important direction is chosen as the negative normalized gradient of the $\mathcal{G}$-function at the origin:

$$\boldsymbol{\alpha}^{(0)} = -\frac{\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0})}{||\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0})||}, \tag{42}$$

where $\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0}) = \left[\frac{\partial\mathcal{G}(\mathbf{0})}{\partial u_1}, \frac{\partial\mathcal{G}(\mathbf{0})}{\partial u_2}, \cdots, \frac{\partial\mathcal{G}(\mathbf{0})}{\partial u_d}\right]^\top$; $||\cdot||$ is the Euclidean norm. The gradient vector $\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0})$ may not be analytically available in most cases. To this end, the forward difference method is used to provide a numerical approximation at the cost of $(d+1)$ $\mathcal{G}$-function evaluations. Given $\boldsymbol{\alpha}^{(0)}$, it is in principle not possible to uniquely determine the corresponding matrix $\boldsymbol{B}^{(0)}$ that describes the hyperplane orthogonal to $\boldsymbol{\alpha}^{(0)}$. However, this does not impose severe restrictions in practice because one can simply employ, e.g., the Gram–Schmidt orthonormalization, to specify an admissible $\boldsymbol{B}^{(0)}$.

Step 2: Generating an initial training dataset and updating the important direction

In this step, an initial training dataset needs to be generated and the initial important direction can be updated. First, we draw a small set of samples $\underline{\boldsymbol{\mathcal{U}}}^{\perp} = \left\{\underline{\boldsymbol{u}}^{\perp,(j)}\right\}_{j=1}^{n_0}$ uniformly distributed within a hyper-rectangle $[-r, r]^{d-1}$ on the hyperplane orthogonal to $\boldsymbol{\alpha}^{(0)}$, using Sobol sequence. As a convenient

rule of thumb, the two parameters n_0 and r are specified as 5 and 3.5, respectively. Second, for each sample $\underline{\boldsymbol{u}}^{\perp,(j)}$, one has to compute the Euclidean distance between $\underline{\boldsymbol{u}}^{\perp,(j)}$ and the limit state surface $\mathcal{G}=0$ along $\boldsymbol{\alpha^{(0)}}$. This is equivalent to finding the root of $\mathcal{G}(\boldsymbol{\alpha}^{(0)}u^{\parallel}+\boldsymbol{B}^{(0)}\underline{\boldsymbol{u}}^{\perp,(j)})=0$, which can be solved by using the adaptive inverse interpolation method [31]. The approximate roots corresponding to $\underline{\boldsymbol{\mathcal{U}}}^{\perp}$ are denoted as $\underline{\tilde{\boldsymbol{\mathcal{Y}}}}=\left\{\underline{\tilde{y}}^{(j)}\right\}_{j=1}^{n_0}$ with $\underline{\tilde{y}}^{(j)}=\tilde{\beta}(\underline{\boldsymbol{u}}^{\perp,(j)})$. Besides, it is also important to record each approximate intersection $\boldsymbol{\alpha}^{(0)}\underline{\tilde{y}}^{(j)}+\boldsymbol{B}^{(0)}\underline{\boldsymbol{u}}^{\perp,(j)}$ of the line $\boldsymbol{\alpha}^{(0)}u^{\parallel}+\boldsymbol{B}^{(0)}\underline{\boldsymbol{u}}^{\perp,(j)}$ and $\mathcal{G}=0$. Third, a new important direction $\boldsymbol{\alpha}^{(1)}$ can be set as the normalized vector of the approximate intersection with the shortest distance to the origin, i.e., $\boldsymbol{\alpha}^{(1)}=\frac{\boldsymbol{\alpha}^{(0)}\underline{\tilde{y}}^{(j^\star)}+\boldsymbol{B}^{(0)}\underline{\boldsymbol{u}}^{\perp,(j^\star)}}{||\boldsymbol{\alpha}^{(0)}\underline{\tilde{y}}^{(j^\star)}+\boldsymbol{B}^{(0)}\underline{\boldsymbol{u}}^{\perp,(j^\star)}||}$ with $j^\star=\arg\min_{1\le j\le n_0}||\boldsymbol{\alpha}^{(0)}\underline{\tilde{y}}^{(j)}+\boldsymbol{B}^{(0)}\underline{\boldsymbol{u}}^{\perp,(j)}||$. The matrix $\boldsymbol{B}^{(1)}$ corresponding to $\boldsymbol{\alpha}^{(1)}$ can be specified by means of the Gram-Schmidt process. Fourth, by projecting those n_0 approximate intersections onto the hyperplane perpendicular to $\boldsymbol{\alpha}^{(1)}$, one can simply obtain the projection points $\boldsymbol{\mathcal{U}}^{\perp}=\left\{\boldsymbol{u}^{\perp,(j)}\right\}_{j=1}^{n_0}$ and distances $\tilde{\boldsymbol{\mathcal{Y}}}=\left\{\tilde{y}^{(j)}\right\}_{j=1}^{n_0}$. The initial training dataset is obtained as $\tilde{\boldsymbol{\mathcal{D}}}=\left\{\boldsymbol{\mathcal{U}}^{\perp},\tilde{\boldsymbol{\mathcal{Z}}}\right\}$ with $\tilde{\boldsymbol{\mathcal{Z}}}=\log\tilde{\boldsymbol{\mathcal{Y}}}$. Let $n=n_0$ and $q=1$.

Step 3: Inferring the posterior statistics of the failure probability

The approximate posterior mean and variance of the failure probability can be inferred based on data $\tilde{\boldsymbol{\mathcal{D}}}$. First, we make an inference about the GP posterior of the l-function, as defined in Eq. (23). This can be achieved by using, e.g., the *fitrgp* function in Statistics and Machine Learning Toolbox of Matlab. Second, via the relationship between the l-function and the β-function, it is straightforward to obtain the LP posterior of the β-function, as given by Eq. (26). Third, with the help of the moment-matched GP approximation, we can finally arrive at the approximate posterior mean and variance of the failure probability (as shown in Eqs. (31) and (32)). Fourth, one can obtain the approximate mean estimate $\hat{\boxed{m}}_{P_{f,n}}$ and the approximate variance estimate $\hat{\boxed{\sigma}}^2_{P_{f,n}}$ by using the sequential SDA-IS method described in Section 3.1.4. The sequential method ($\lambda=1.5$) is stopped until $\sqrt{\mathbb{V}\left[\hat{\boxed{m}}_{P_{f,n}}\right]}/\hat{\boxed{m}}_{P_{f,n}}<\tau_1$ and $\sqrt{\mathbb{V}\left[\hat{\boxed{\sigma}}^2_{P_{f,n}}\right]}/\hat{\boxed{\sigma}}^2_{P_{f,n}}<\tau_2$ are met ($\tau_1=0.01$ and $\tau_2=0.05$).

Step 4: Checking the stopping criterion

If the stopping criterion $\frac{\hat{\boxed{\sigma}}_{P_{f,n}}}{\hat{\boxed{m}}_{P_{f,n}}}<\eta$ is reached twice in a row, go to **Step 6**; Otherwise, go to **Step 5**. In this study, the threshold η takes the value of 0.05.

Step 5: Enriching the training dataset and updating the important direction

This step involves enriching the previous training dataset by identifying a new promising location at which to query the l-function, and updating the important direction once a more probable one is found. First, the next best point $\underline{\boldsymbol{u}}^{\perp,(n+1)}$ is determined by maximizing the learning function (Eq. (40)), where $M = 20$ is adopted. Second, the approximate distance $\tilde{\underline{y}}^{(n+1)}$ between $\underline{\boldsymbol{u}}^{\perp,(n+1)}$ and the limit state surface $\mathcal{G} = 0$ is solved by using the Newton's method. As a guess, $\boxed{m}_{\beta_n}(\underline{\boldsymbol{u}}^{\perp,(n+1)})$ can be taken as the starting point. An approximate intersection is recorded as $\boldsymbol{\alpha}^{(q)}\tilde{\underline{y}}^{(n+1)} + \boldsymbol{B}^{(q)}\underline{\boldsymbol{u}}^{\perp,(n+1)}$. Third, if the new intersection does not have the shortest distance to the origin among all the available approximate intersections, the previous training dataset $\tilde{\boldsymbol{\mathcal{D}}}$ is directly enriched with $\left\{\underline{\boldsymbol{u}}^{\perp,(n+1)}, \log \tilde{\underline{y}}^{(n+1)}\right\}$. Otherwise, the previous important direction is then updated to a new one, i.e., $\boldsymbol{\alpha}^{(q+1)} = \frac{\boldsymbol{\alpha}^{(0)}\tilde{\underline{y}}^{(n+1)} + \boldsymbol{B}^{(q)}\underline{\boldsymbol{u}}^{\perp,(n+1)}}{||\boldsymbol{\alpha}^{(0)}\tilde{\underline{y}}^{(n+1)} + \boldsymbol{B}^{(q)}\underline{\boldsymbol{u}}^{\perp,(n+1)}||}$. Accordingly, a new matrix $\boldsymbol{B}^{(q+1)}$ can be specified and $q = q + 1$. Projecting all the available approximate intersections on the latest hyperplane yields the enriched training dataset $\tilde{\boldsymbol{\mathcal{D}}}$. Let $n = n + 1$ and go to **Step 3**.

Step 6: Stopping the algorithm

The latest $\hat{\boxed{m}}_{P_{f,n}}$ and $\hat{\boxed{\sigma}}^2_{P_{f,n}}$ are returned and the algorithm is stopped.

4. Numerical examples

In this section, we illustrate the proposed BAL-LP-LS method on four numerical examples. Although some examples have explicit performance functions, they are all treated as implicit. In all cases, the crude MCS method is employed to provide the reference failure probabilities whenever possible. For comparison purposes, several existing methods, i.e., first-order reliability method with sequential quadratic programming (FORM-SQP) [36], traditional LS [8], combination line sampling (CLS) [14], active learning reliability method in UQLab version 2.0 (denoted as ALR in UQLab) [37] and BAL-LS [31], are also implemented. In FORM-SQP, the starting point is set as the point of origin and the SQP method adopts the one available in Matlab R2022b with its default settings. The important direction in traditional LS is specified by FORM-SQP, and the Newton's method is employed to process lines. For CLS, the initial important direction uses the same as the proposed method (Eq. (42)). The ALR in UQLab employs the Kriging model with Gaussian

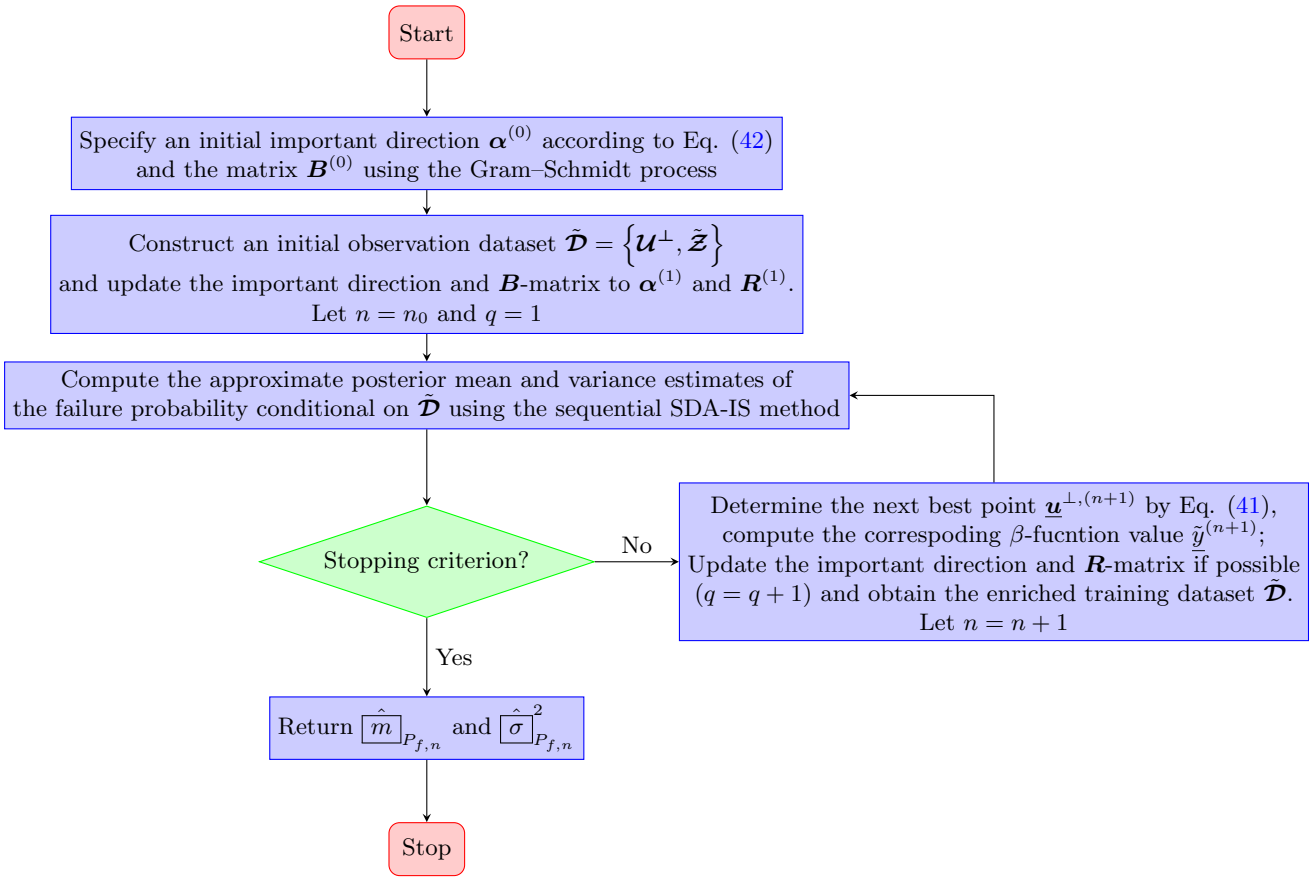


Figure 2: Flowchart of the proposed BAL-LS-LP method.

kernel instead of its default polynomial chaos-Kriging. For ALR in UQLab, BAL-LS and BAL-LS-LP, 20 independent runs are performed for the first three examples in order to test their robustness. Therefore, we only report the mean and/or variability of the quantities of interest.

4.1. Example 1: A test function

For the first example, let us consider a test function taking the form [30]:

$$Y = g(\boldsymbol{X}) = a - X_2 + bX_1^3 + c\sin\left(dX_1\right), \tag{43}$$

where a, b, c and d are four parameters that can influence the non-linearity of the problem and the level of failure probability, which are specified as: $a = 5.5$, $b = 0.02$, $c = \frac{5}{6}$, $d = \frac{\pi}{3}$; X_1 and X_2 are two standard normal variables.

The results of the proposed BAL-LS-LP method and several existing methods are summarized in Table 1. The reference failure probability is taken as 3.54×10^{-7}, which is provided by MCS with 10^{11} samples. The estimated failure probability from FORM-SQP (i.e., 7.19×10^{-7}) differs significantly from the reference value, mainly due to the violation of the linearity assumption in FORM. In two cases, $N_{line} = 100, 200$, both traditional LS and CLS can produce more accurate results than FORM-SQP. However, in order to have a small COV, both methods require a large number of $\mathcal{G}$ function evaluations. ALR in UQLab only needs 16.00 performance function evaluations on average, but it results in obvious bias in the mean of 20 failure probability estimates (say 3.95×10^{-7}). The BAL-LS method gives an average failure probability of 3.50×10^{-7} with a COV of 3.50%, which are at a cost of 9.25 lines and 35.50 $\mathcal{G}$-function evaluations on average. The proposed BAL-LS-LP can further reduce the average number of N_{line} and N_{call}, while producing a fairly good failure probability mean (i.e., 3.59×10^{-7}) with a sufficiently small variability (COV $\left[\hat{P}_f\right] = 0.30\%$).

To provide a schematic illustration of the proposed method, Fig. 3 shows some of the results obtained from an exemplary run. It can be observed from Fig. 3(a) that the initial important direction is far from optimal, but still informative. After five approximate intersections are obtained, the initial important direction is immediately updated to a new one. After three additional intersections are available, the proposed method stops as the stopping criterion is satisfied. As seen from Fig. 3(b), the final important direction is almost optimal.

4.2. Example 2: A non-linear oscillator

The second example consists of a non-linear oscillator subject to a rectangular-pulse load [38], as shown in Fig. 4. The performance function is defined by:

$$Z = g\left(m, k_1, k_2, r, F_1, t_1\right) = 3r - \left| \frac{2F_1}{k_1 + k_2} \sin\left(\frac{t_1}{2} \sqrt{\frac{k_1 + k_2}{m}} \right) \right|, \tag{44}$$

where m, k_1, k_2, r, F_1 and t_1 are six random variables, as detailed in Table 2.

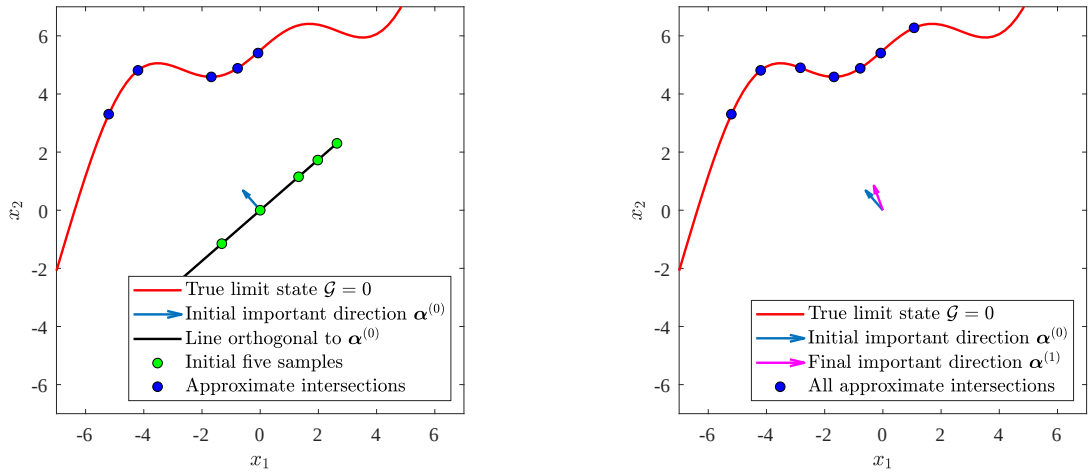
In Table 3, we summarize the results of several methods, including MCS, FORM-SQP, traditional LS, CLS, ALR in UQLab, BAL-LS and BAL-LS-LP. The reference value for the failure probability is 4.01×10^{-8}

Table 1: Results of Example 1 by several methods.

Method	N_{line}	N_{call}	$\hat{P}_f$	COV $\left[\hat{P}_f\right]$
MCS	-	10^{11}	3.54×10^{-7}	0.53%
FORM-SQP	-	28	7.19×10^{-7}	-
Traditional LS	100	366	3.74×10^{-7}	7.01%
	200	714	3.33×10^{-7}	5.24%
CLS	100	490	3.79×10^{-7}	6.91%
	200	964	3.37×10^{-7}	5.74%
ALR in UQLab	-	16.00	3.95×10^{-7}	6.30%
BAL-LS	9.25	35.50	3.50×10^{-7}	3.50%
Proposed BAL-LS-LP	7.00	30.00	3.59×10^{-7}	0.30%

Note: N_{line} = the total number of lines; N_{call} = the total number of $\mathcal{G}$-function calls (including the number of $\mathcal{G}$-function calls to find the roots, if applicable).

with a COV of 0.50%, provided by MCS with 10^{12} samples. At the cost of 176 $\mathcal{G}$-function evaluations, FORM-SQP provides a failure probability estimate of 4.88×10^{-8}, which is not that close to the reference value. The accuracy of FORM-SQP can be further improved by the traditional LS with some extra lines (e.g., 100), which, in turn, leads to a significant increase in $\mathcal{G}$-function calls. Compared to the traditional LS, CLS needs more lines and $\mathcal{G}-$function evaluations to yield a reasonable result. ALR in UQLab is able to reduce the number of $\mathcal{G}$-function evaluations to 46.55 on average. Nevertheless, the mean value of 20 failure probability estimates (say 4.75×10^{-8}) appears to be biased and relatively larger than the reference value. At the cost of 12.65 lines and 43.25 $\mathcal{G}$-function calls on average, BAL-LS produces a failure probability mean of 3.73×10^{-8} with a COV of 30.52%. Compared to BAL-LS, BAL-LS-LP requires on average slightly more lines and $\mathcal{G}$-function calls, but produces a almost unbiased result with a significantly small COV, say 0.92%.



(a) Initial stage (b) Final stage

Figure 3: Schematic illustration of the proposed BAL-LS-LP method for Example 1.

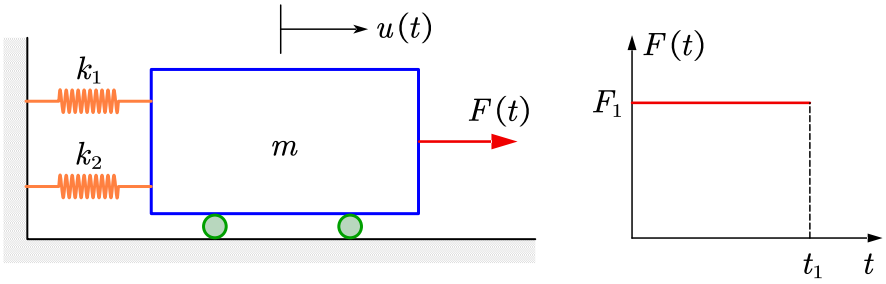


Figure 4: A nonlinear oscillator driven by a rectangular pulse load.

4.3. Example 3: An I beam

As a third example, we consider a simply-supported I beam subject to a concentrated force [39], as depicted in Fig. 5. The performance function is expressed as:

$$Y = g(\boldsymbol{X}) = S - \sigma_{\max}, \tag{45}$$

in which

$$\sigma_{\max} = \frac{Pa(L-a)d}{2LI}, \tag{46}$$

with

$$I = \frac{b_f d^3 - (b_f - t_w)(d - 2t_f)^3}{12}. \tag{47}$$

Table 2: Random variables for Example 2.

Variable	Description	Distribution	Mean	COV
m	Mass	Lognormal	1.0	0.05
k_1	Stiffness	Lognormal	1.0	0.10
k_2	Stiffness	Lognormal	0.2	0.10
r	Yield displacement	Lognormal	0.5	0.10
F_1	Load amplitude	Lognormal	0.4	0.20
t_1	Load duration	Lognormal	1.0	0.20

A total number of eight random variables $\boldsymbol{X} = [P, L, a, S, d, b_f, t_w, f_f]^\top$ are involved in this example, as listed in Table 4.

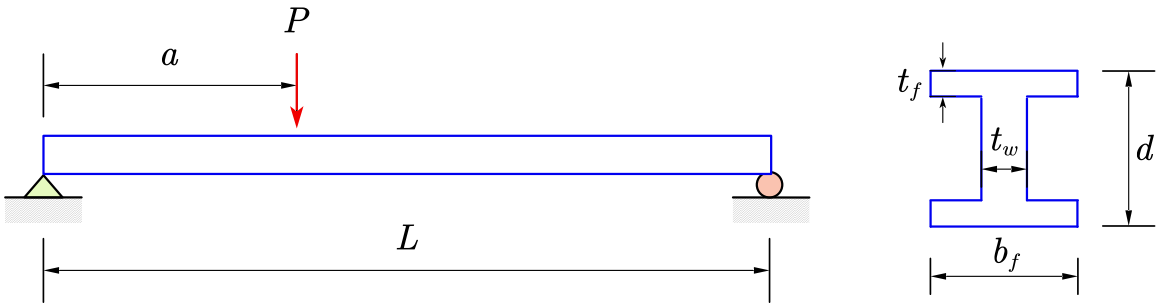


Figure 5: A simply-supported I beam.

The results obtained from several methods are reported in Table 5. MCS with 10^{11} samples produces a reference failure probability of 1.69×10^{-7} with a COV being 0.77%. FORM-SQP gives a result (say 1.48×10^{-7}) that is slightly smaller than the reference one. However, it necessitates a large number (i.e., 1511) of performance function evaluations. In order to achieve a failure probability estimate with a COV less than 5%, traditional LS may require more than 100 additional lines. Even with 200 lines, the failure probability given by CLS still has a large COV, i.e., 7.40%. At the cost of 93.10 $\mathcal{G}$-function calls on average, the result from ALR in UQLab is still biased and tends to be larger than the reference value. The average numbers of lines and $\mathcal{G}$-function calls required by BAL-LP-LS are less than those of BAL-LS, but can still

Table 3: Results of Example 2 by several methods.

Method	N_{line}	N_{call}	$\hat{P}_f$	COV $\left[\hat{P}_f\right]$
MCS	-	10^{12}	4.01×10^{-8}	0.50%
FORM-SQP	-	176	4.88×10^{-8}	-
Traditional LS	50	376	4.16×10^{-8}	2.93%
	100	576	4.09×10^{-8}	1.92%
CLS	200	868	4.87×10^{-8}	7.91%
	300	1,329	4.65×10^{-8}	6.79%
ALR in UQLab	-	46.55	4.75×10^{-8}	11.62%
BAL-LS	12.65	43.25	3.73×10^{-8}	30.52%
Proposed BAL-LS-LP	13.65	46.20	4.02×10^{-8}	0.92%

give a failure probability mean that is closed to the reference one and with a smaller COV.

4.4. Example 4: A space truss structure

The last example involves a 120-bar space truss structure subject to seven vertical loads [27, 28], as shown in Fig. 6. The structure is modeled as a three-dimensional truss using an open-source finite element analysis software, OpenSees. The established model consists of 49 nodes and 120 truss elements. It is assumed that all elements have the same cross-sectional area, A, and the same modulus of elasticity, E. The thirteen vertical loads (as depicted in Fig. 6) are denoted as $P_0 \sim P_{12}$. The performance function is defined as:

$$Y = g(\boldsymbol{X}) = \Delta - V_0(A, E, P_0 \sim P_{12}), \tag{48}$$

where V_0 is the vertical displacement of node 0; Δ is a threshold, which is specified as 100 mm; A, E, $P_0 \sim P_{12}$ are fifteen random variables, as described in Table 6.

In this example, we cannot afford to run the crude MCS in order to provide a reference solution because the target failure probability is quite small. To this end, the importance sampling (IS) available in UQLab [37] is then implemented as an alternative, where the importance sampling density is chosen as Gaussian

Table 4: Random variables for Example 3.

Variable	Distribution	Mean	COV
P	Lognormal	1500	0.20
L	Normal	120	0.05
a	Normal	72	0.10
S	Normal	200,000	0.15
d	Normal	2.3	0.05
b_f	Normal	2.3	0.05
t_w	Normal	0.16	0.05
t_f	Normal	0.26	0.05

centered on the most probable point. The failure probability given by IS is 1.90×10^{-9} with a COV of 1.97%. The results of IS and several other methods are compared in Table 7. FORM-SQP converges to an infeasible point after one iteration. Therefore, the traditional LS also cannot work because it is based on the FORM-SQP in our setting. ALR in UQLab produces a wrong result for the failure probability due to it is premature in most trials. Although the CLS method is workable, its variability is quite large even using 1,000 lines. At the cost of 25 lines and 144 performance function evaluations, BAL-LS gives a failure probability estimate of 2.24×10^{-9} with a COV of 2.69%. Remarkably, the proposed BAL-LS-LP method can produce a much better estimate with less $\mathcal{G}$-functions calls compared to BAL-LS.

Table 5: Results of Example 3 by several methods.

Method	N_{line}	N_{call}	$\hat{P}_f$	COV $\left[\hat{P}_f\right]$
MCS	-	10^{11}	1.69×10^{-7}	0.77%
FORM-SQP	-	1,511	1.48×10^{-7}	-
Traditional LS	100	1,859	1.89×10^{-7}	7.08%
	200	2,195	1.62×10^{-7}	2.43%
CLS	100	504	1.62×10^{-7}	10.27%
	200	993	1.50×10^{-7}	7.40%
ALR in UQLab	-	85.95	2.01×10^{-7}	14.88%
BAL-LS	17.20	59.30	1.61×10^{-7}	10.19%
Proposed BAL-LS-LP	11.35	40.70	1.62×10^{-7}	8.88%

Table 6: Random variables for Example 4.

Variable	Distribution	Mean	COV
A	Normal	2,000 mm^2	0.10
E	Normal	200 GPa	0.10
P_0	Lognormal	400 kN	0.20
$P_1 \sim P_{12}$	Lognormal	50 kN	0.15

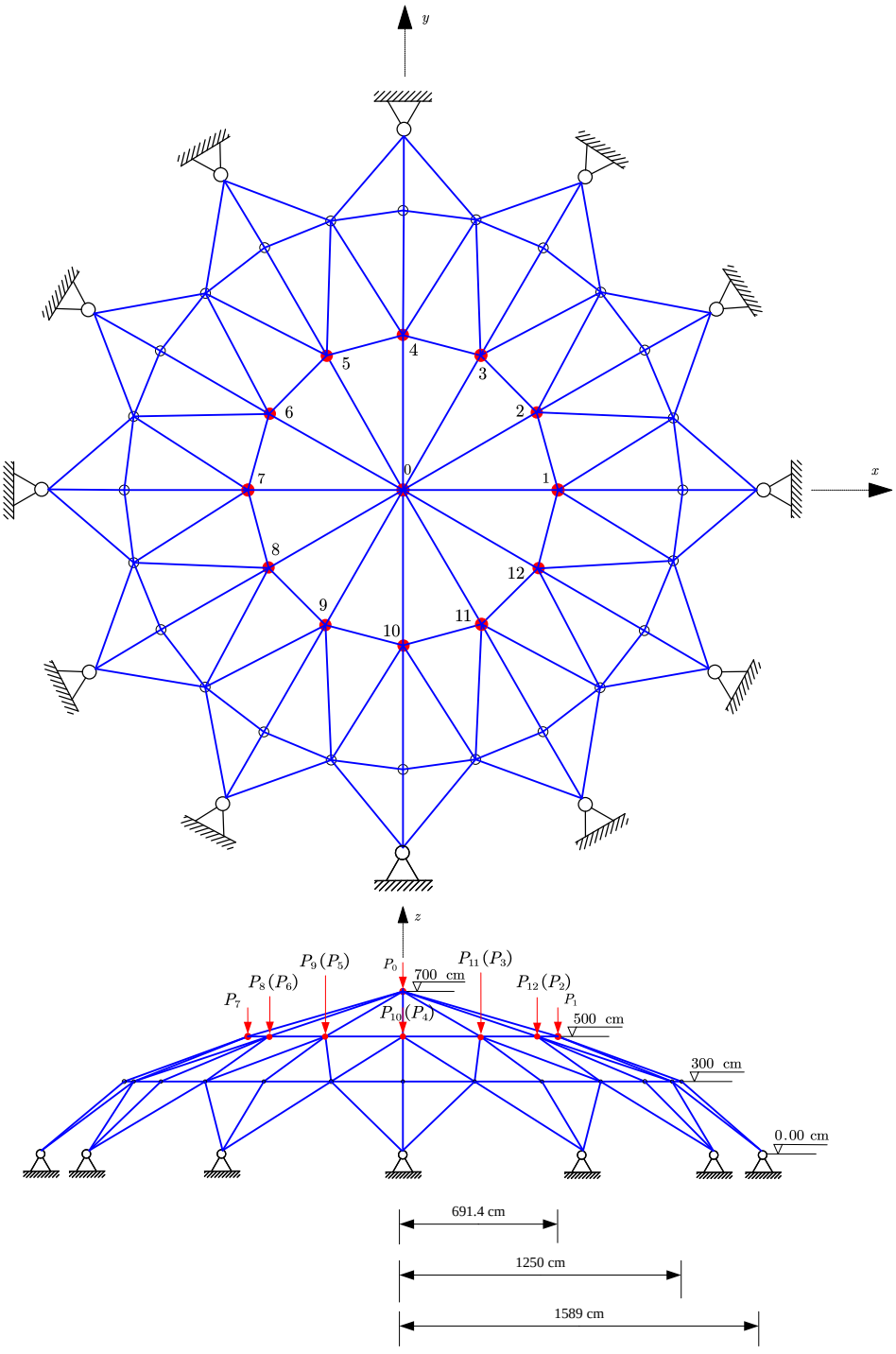


Figure 6: A 120-bar space truss structure subject to thirteen vertical loads.

Table 7: Results of Example 4 by several methods.

Method	N_{line}	N_{call}	$\hat{P}_f$	COV $\left[\hat{P}_f\right]$
IS	-	25,141	1.90×10^{-9}	1.97%
FORM-SQP	-	-	-	-
Traditional LS	-	-	-	-
CLS	500	3,001	1.02×10^{-9}	15.20%
	1,000	5,926	1.82×10^{-9}	14.12%
ALR in UQLab	-	-	-	-
BAL-LS	25	144	2.24×10^{-9}	2.69%
Proposed BAL-LS-LP	26	102	1.90×10^{-9}	2.39%

5. Concluding remarks

This paper presents a new Bayesian active learning alternative, called 'Bayesian active learning line sampling with log-normal process' (BAL-LS-LP), to the traditional line sampling for structural reliability analysis, especially for assessing small failure probabilities. First, we treat the estimation of the failure probability in LS with Bayesian inference. By using an LP prior instead of a GP prior, it is possible to simultaneously consider the discretization error of the distance function, as well as its non-negativity constraint that is ignored in both PBAL-LS and BAL-LS. In addition, the approximation error of the distance function is taken into account by assuming a zero-mean normal distribution. The approximate posterior mean and variance of the failure probability are derived based on the use of a moment-matched GP approximation of the posterior distribution of the distance function. Second, two essential components for active learning, i.e., learning function and stopping criterion, are developed using the posterior statistics of the failure probability. Third, the important direction can be automatically updated on the fly during the simulation from an initial rough guess. By means of four numeral examples, it is demonstrated that the proposed method is able to assess extremely small failure probabilities (e.g., an order of magnitude $10^{-7} \sim 10^{-9}$) with reasonable accuracy and efficiency.

Note that the BAL-LS-LP method is suitable for weakly and moderately nonlinear problems with a single half-open failure domain. The authors suggest potential improvements for the method in the following directions. Firstly, optimizing the learning function using a nature-inspired global optimization algorithm can be time-consuming as the dimensions increase. This reduces the efficiency of the proposed method in higher dimensions. The problem may be solved by simplifying the learning function or using a more efficient optimization algorithm. Secondly, approximating the posterior variance of the failure probability using the SDA-IS method can be challenging. One solution could be to simplify the approximation or develop a more efficient numerical integrator.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Chao Dang is mainly supported by China Scholarship Council (CSC). Pengfei Wei is grateful to the support from the National Natural Science Foundation of China (grant no. 51905430 and 72171194). Jingwen Song acknowledges the financial support from the National Natural Science Foundation of China (grant no. 12202358 and 12220101002). Michael Beer would like to thank the support of the National Natural Science Foundation of China under grant number 72271025.

Data availability

Data will be made available on request.

References

[1] R. Melchers, Importance sampling in structural systems, Structural Safety 6 (1) (1989) 3–10. doi:https://doi.org/10.1016/0167-4730(89)90003-9.

[2] S.-K. Au, J. L. Beck, A new adaptive importance sampling scheme for reliability calculations, Structural Safety 21 (2) (1999) 135–158. doi:https://doi.org/10.1016/S0167-4730(99)00014-4.

[3] N. Kurtz, J. Song, Cross-entropy-based adaptive importance sampling using gaussian mixture, Structural Safety 42 (2013) 35–44. doi:https://doi.org/10.1016/j.strusafe.2013.01.006.

[4] S.-K. Au, J. L. Beck, Estimation of small failure probabilities in high dimensions by subset simulation, Probabilistic Engineering Mechanics 16 (4) (2001) 263–277. doi:https://doi.org/10.1016/S0266-8920(01)00019-4.

[5] S.-K. Au, Y. Wang, Engineering risk assessment with subset simulation, John Wiley & Sons, 2014.

[6] O. Ditlevsen, R. E. Melchers, H. Gluver, General multi-dimensional probability integration by directional simulation, Computers & Structures 36 (2) (1990) 355–368. doi:https://doi.org/10.1016/0045-7949(90)90134-N.

[7] J. Nie, B. R. Ellingwood, Directional methods for structural reliability analysis, Structural Safety 22 (3) (2000) 233–249. doi:https://doi.org/10.1016/S0167-4730(00)00014-X.

[8] P.-S. Koutsourelakis, H. J. Pradlwarter, G. I. Schueller, Reliability of structures in high dimensions, part i: algorithms and applications, Probabilistic Engineering Mechanics 19 (4) (2004) 409–417. doi:https://doi.org/10.1016/j.probengmech.2004.05.001.

[9] H. Pradlwarter, G. I. Schueller, P.-S. Koutsourelakis, D. C. Charmpis, Application of line sampling simulation method to reliability benchmark problems, Structural Safety 29 (3) (2007) 208–221. doi:https://doi.org/10.1016/j.strusafe.2006.07.009.

[10] P.-S. Koutsourelakis, Reliability of structures in high dimensions. part ii. theoretical validation, Probabilistic engineering mechanics 19 (4) (2004) 419–423. doi:https://doi.org/10.1016/j.probengmech.2004.05.002.

[11] M. Hohenbichler, R. Rackwitz, Improvement of second-order reliability estimates by importance sampling, Journal of Engineering Mechanics 114 (12) (1988) 2195–2199. doi:https://doi.org/10.1061/(ASCE)0733-9399(1988)114:12(2195).

[12] M. de Angelis, E. Patelli, M. Beer, Advanced line sampling for efficient robust reliability analysis, Structural Safety 52 (2015) 170–182. doi:https://doi.org/10.1016/j.strusafe.2014.10.002.

[13] M. A. Shayanfar, M. A. Barkhordari, M. Barkhori, M. Rakhshanimehr, An adaptive line sampling method for reliability analysis, Iranian Journal of Science and Technology, Transactions of Civil Engineering 41 (2017) 275–282. doi:https://doi.org/10.1007/s40996-017-0070-3.

[14] I. Papaioannou, D. Straub, Combination line sampling for structural reliability analysis, Structural Safety 88 (2021) 102025. doi:https://doi.org/10.1016/j.strusafe.2020.102025.

[15] I. Depina, T. M. H. Le, G. Fenton, G. Eiksund, Reliability analysis with metamodel line sampling, Structural Safety 60 (2016) 1–15. doi:https://doi.org/10.1016/j.strusafe.2015.12.005.

[16] J. Song, P. Wei, M. Valdebenito, M. Beer, Active learning line sampling for rare event analysis, Mechanical Systems and Signal Processing 147 (2021) 107113. doi:https://doi.org/10.1016/j.ymssp.2020.107113.

[17] Z. Lu, S. Song, Z. Yue, J. Wang, Reliability sensitivity method by line sampling, Structural Safety 30 (6) (2008) 517–532. doi:https://doi.org/10.1016/j.strusafe.2007.10.001.

[18] M. A. Valdebenito, H. A. Jensen, H. Hernández, L. Mehrez, Sensitivity estimation of failure probability applying line sampling, Reliability Engineering & System Safety 171 (2018) 99–111. doi:https://doi.org/10.1016/j.ress.2017.11.010.

[19] M. A. Valdebenito, H. B. Hernández, H. A. Jensen, Probability sensitivity estimation of linear stochastic finite element models applying line sampling, Structural Safety 81 (2019) 101868. doi:https://doi.org/10.1016/j.strusafe.2019.06.002.

[20] J. Song, M. Valdebenito, P. Wei, M. Beer, Z. Lu, Non-intrusive imprecise stochastic simulation by line sampling, Structural Safety 84 (2020) 101936. doi:https://doi.org/10.1016/j.strusafe.2020.101936.

[21] J. Song, P. Wei, M. Valdebenito, M. Beer, Adaptive reliability analysis for rare events evaluation with global imprecise line sampling, Computer Methods in Applied Mechanics and Engineering 372 (2020) 113344. doi:https://doi.org/10.

1016/j.cma.2020.113344.

[22] J. Wang, Z. Lu, L. Wang, An efficient method for estimating failure probability bounds under random-interval mixed uncertainties by combining line sampling with adaptive kriging, International Journal for Numerical Methods in Engineering 124 (2) (2023) 308–333. doi:https://doi.org/10.1002/nme.7122.

[23] J. Wang, Z. Lu, Y. Cheng, L. Wang, An efficient method for estimating failure probability bound functions of composite structure under the random-interval mixed uncertainties, Composite Structures 298 (2022) 116011. doi:https://doi.org/10.1016/j.compstruct.2022.116011.

[24] X. Zhang, Z. Lu, W. Yun, K. Feng, Y. Wang, Line sampling-based local and global reliability sensitivity analysis, Structural and Multidisciplinary Optimization 61 (2020) 267–281. doi:https://doi.org/10.1007/s00158-019-02358-9.

[25] X. Yuan, Z. Zheng, B. Zhang, Augmented line sampling for approximation of failure probability function in reliability-based analysis, Applied Mathematical Modelling 80 (2020) 895–910. doi:https://doi.org/10.1016/j.apm.2019.11.009.

[26] M. A. Valdebenito, P. Wei, J. Song, M. Beer, M. Broggi, Failure probability estimation of a class of series systems by multidomain line sampling, Reliability Engineering & System Safety 213 (2021) 107673. doi:https://doi.org/10.1016/j.ress.2021.107673.

[27] C. Dang, P. Wei, J. Song, M. Beer, Estimation of failure probability function under imprecise probabilities by active learning–augmented probabilistic integration, ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering 7 (4) (2021) 04021054. doi:https://doi.org/10.1061/AJRUA6.0001179.

[28] C. Dang, P. Wei, M. G. Faes, M. A. Valdebenito, M. Beer, Parallel adaptive Bayesian quadrature for rare event estimation, Reliability Engineering & System Safety 225 (2022) 108621. doi:https://doi.org/10.1016/j.ress.2022.108621.

[29] C. Dang, M. A. Valdebenito, M. G. Faes, P. Wei, M. Beer, Structural reliability analysis: A Bayesian perspective, Structural Safety 99 (2022) 102259. doi:https://doi.org/10.1016/j.strusafe.2022.102259.

[30] C. Dang, M. A. Valdebenito, J. Song, P. Wei, M. Beer, Estimation of small failure probabilities by partially Bayesian active learning line sampling: Theory and algorithm, Computer Methods in Applied Mechanics and Engineering 412 (2023) 116068. doi:https://doi.org/10.1016/j.cma.2023.116068.

[31] C. Dang, M. A. Valdebenito, M. G. Faes, J. Song, P. Wei, M. Beer, Structural reliability analysis by line sampling: A Bayesian active learning treatment, Structural Safety 104 (2023) 102351. doi:https://doi.org/10.1016/j.strusafe.2023.102351.

[32] G. I. Schuëller, H. J. Pradlwarter, P.-S. Koutsourelakis, A critical appraisal of reliability estimation procedures for high dimensions, Probabilistic Engineering Mechanics 19 (4) (2004) 463–474. doi:https://doi.org/10.1016/j.probengmech.2004.05.004.

[33] C. K. Williams, C. E. Rasmussen, Gaussian processes for machine learning, Vol. 2, MIT press Cambridge, MA, 2006.

[34] T. Gunter, M. A. Osborne, R. Garnett, P. Hennig, S. J. Roberts, Sampling for inference in probabilistic models with fast bayesian quadrature, Advances in Neural Information Processing Systems 27 (2014).

[35] H. R. Chai, R. Garnett, Improving quadrature for constrained integrands, in: The 22nd International Conference on Artificial Intelligence and Statistics, PMLR, 2019, pp. 2751–2759.

[36] P.-L. Liu, A. Der Kiureghian, Optimization algorithms for structural reliability, Structural Safety 9 (3) (1991) 161–177. doi:https://doi.org/10.1016/0167-4730(91)90041-7.

[37] M. Moustapha, S. Marelli, B. Sudret, UQLab user manual – Active learning reliability, Tech. rep., Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland, report UQLab-V2.0-117 (2022).

[38] C. G. Bucher, U. Bourgund, A fast and efficient response surface approach for structural reliability problems, Structural Safety 7 (1) (1990) 57–66. doi:https://doi.org/10.1016/0167-4730(90)90012-E.

[39] B. Huang, X. Du, Uncertainty Analysis by Dimension Reduction Integration and Saddlepoint Approximations, Journal of Mechanical Design 128 (1) (2005) 26–33. doi:https://doi.org/10.1115/1.2118667.