INSTITUTE FOR RISK AND RELIABILITY

Leibniz Universität Hannover

Forward Uncertainty Quantification with Special Emphasis on a Bayesian Active Learning Perspective

Von der Fakultät für Bauingenieurwesen und Geodäsie

der Gottfried Wilhelm Leibniz Universität Hannover

zur Erlangung des Grades

Doktor-Ingenieur

Dr.-Ing.

genehmigte Dissertation

von

Chao Dang, M.Eng.

2023

Referent: Prof. Dr.-Ing. Michael Beer

Korreferent: Prof. Dr.-Ing. habil. Fadi Aldakheel

Prof. Dr. sc. nat. ETH Insa Neuweiler

Prof. Dr. Bruno Sudret (ETH Zurich)

Tag der Promotion: 25.08.2023

To my parents

Erklärung zum Promotionsgesuch

Ich erkläre, dass ich die Regeln der geltenden Promotionsordnung kenne und eingehalten habe und mit einer Prüfung nach den Bestimmungen der Promotionsordnung einverstanden bin. Ich habe die Dissertation selbst verfasst, keine Textabschnitte von Dritten oder eigener Prüfungsarbeiten ohne Kennzeichnung übernommen und alle benutzten Hilfsmittel und Quellen in der Arbeit angegeben. Weiterhin habe ich Dritten weder unmittelbar noch mittelbar geldwerte Leistungen für Vermittlungstätigkeiten oder für die inhaltliche Ausarbeitung der Dissertation erbracht und die Dissertation noch nicht als Prüfungsarbeit für eine staatliche oder andere wissenschaftliche Prüfung eingereicht. Die vorliegende Arbeit wurde ebenso nicht bei einer anderen Hochschule als Dissertation eingereicht. Ich bin damit einverstanden, dass die Dissertation auch zum Zwecke der Überprüfung der Einhaltung allgemein geltender wissenschaftlicher Standards genutzt wird, insbesondere auch unter Verwendung elektronischer Datenverarbeitungsprogramme.

Hannover, am 07.09.2023 Chao Dang

Datum Unterschrift

Acknowledgments

First of all, I would like to express my deepest gratitude to my supervisor Prof. Dr.-Ing. Michael Beer. Three years ago, it was he who offered me the opportunity to start my Ph.D. program at the Institute for Risk and Reliability at Leibniz University Hannover. Throughout my Ph.D. journey, his careful guidance and strong support have been instrumental in my academic achievements. Besides, much of my success is also attributed to the academic freedom he gave me.

I acknowledge the financial support that I received from the China Scholarship Council (CSC). Without this generous grant, I would not have been able to conduct this research.

I would also like to thank my collaborators, Dr. Pengfei Wei, Dr. Jingwen Song, Dr. Marcos Valdebenito and Prof. Dr. Matthias Faes. I have benefited greatly from the scientific discussions and their professional feedback while working with them.

My sincere thanks also go to various colleagues who worked at or visited the Institute for Risk and Reliability during my stay. They are, Dr. Matteo Broggi, Dr. Masaru Kitahara, Dr. Jiashu Yang, Dr. Kang Liao, Dr. Guan Chen, Dr. Marco Behrendt, Dr. Alice Cicirello, Peihua Ni, Zhuo Hu, Chengxin Feng, Chen Ding, Jingxin Liu, Yi Luo, Marius Bittner, Thomas Potthast, to just name a few. Not only did I enjoy working with you, but I also learned a lot from you.

Finally, I want to give a special thanks to my parents for their unconditional love. I am so lucky to have you as my parents. Thank you for always being there for me.

Abstract

Uncertainty quantification (UQ) in its broadest sense aims at quantitatively studying all sources of uncertainty arising from both computational and real-world applications. Although many subtopics appear in the UQ field, there are typically two major types of UQ problems: forward and inverse uncertainty propagation. The present study focuses on the former, which involves assessing the effects of the input uncertainty in various forms on the output response of a computational model. In total, this thesis reports nine main developments in the context of forward uncertainty propagation, with special emphasis on a Bayesian active learning perspective.

The first development is concerned with estimating the extreme value distribution and small first-passage probabilities of uncertain nonlinear structures under stochastic seismic excitations, where a moment-generating function-based mixture distribution approach (MGF-MD) is proposed. As the second development, a triple-engine parallel Bayesian global optimization (T-PBGO) method is presented for interval uncertainty propagation. The third contribution develops a parallel Bayesian quadrature optimization (PBQO) method for estimating the response expectation function, its variable importance and bounds when a computational model is subject to hybrid uncertainties in the form of random variables, parametric probability boxes (p-boxes) and interval models. In the fourth research, of interest is the failure probability function when the inputs of a performance function are characterized by parametric p-boxes. To do so, an active learning augmented probabilistic integration (ALAPI) method is proposed based on offering a partially Bayesian active learning perspective on failure probability estimation, as well as the use of high-dimensional model representation (HDMR) technique. Note that in this work we derive an upper-bound of the posterior variance of the failure probability, which bounds our epistemic uncertainty about the failure probability due to a kind of numerical uncertainty, i.e., discretization error. The fifth contribution further strengthens the previously developed active learning probabilistic integration (ALPI) method in two ways, i.e., enabling the use of parallel computing and enhancing the capability of assessing small failure probabilities. The resulting method is called parallel adaptive Bayesian quadrature (PABQ). The sixth research presents a principled Bayesian failure probability inference (BFPI) framework,

where the posterior variance of the failure probability is derived (not in closed form). Besides, we also develop a parallel adaptive-Bayesian failure probability learning (PA-BFPI) method upon the BFPI framework. For the seventh development, we propose a partially Bayesian active learning line sampling (PBAL-LS) method for assessing extremely small failure probabilities, where a partially Bayesian active learning insight is offered for the classical LS method and an upper-bound for the posterior variance of the failure probability is deduced. Following the PBAL-LS method, the eighth contribution finally obtains the expression of the posterior variance of the failure probability in the LS framework, and a Bayesian active learning line sampling (BALLS) method is put forward. The ninth contribution provides another Bayesian active learning alternative, Bayesian active learning line sampling with log-normal process (BAL-LS-LP), to the traditional LS. In this method, the log-normal process prior, instead of a Gaussian process prior, is assumed for the beta function so as to account for the non-negativity constraint. Besides, the approximation error resulting from the root-finding procedure is also taken into consideration.

In conclusion, this thesis presents a set of novel computational methods for forward UQ, especially from a Bayesian active learning perspective. The developed methods are expected to enrich our toolbox for forward UQ analysis, and the insights gained can stimulate further studies.

Keywords: Uncertainty propagation; Structural reliability analysis; Bayesian active learning; Bayesian inference; Bayesian quadrature; Bayesian optimization; Active learning; Gaussian process; Numerical uncertainty; Line sampling

Zusammenfassung

Die Unsicherheitsquantifizierung (UQ) dient im weitesten Sinne der quantitativen Untersuchung von Quellen von Unsicherheiten in rechnergestützten und realen Applikationen. Obwohl das Feld der UQ in viele Unterthemen gegliedert ist, lassen sich zwei typische Hauptprobleme benennen: vorwärts und inverse Unsicherheitspropagierung. Im Fokus der vorliegenden Thesis sind Vorwärtsprobleme, bei denen die Auswirkungen unterschiedlicher Formen von Unsicherheit in Inputparametern auf den Output von rechnergestützten Modellen untersucht werden. Insgesamt beinhaltet diese Arbeit neun wesentliche Weiterentwicklungen zur Vorwärts-UQ-Propagierung mit dem Schwerpunkt auf Methoden des Baysian-active-learnings.

Die erste Entwicklung befasst sich mit der Schätzung von Extremwertverteilung und kleinen First-Passage Wahrscheinlichkeiten von unsicheren nichtlinearen Strukturen unter stochastischer seismischer Anregung. Hierfür wird ein momenterzeugender funktionsbasierter Mischverteilungsansatz (MGF-MD) vorgestellt. Die zweite Entwicklung beinhaltet eine Triple-Engine-Parallel-Baysian-Global-Optimization-Methode (T-PBGO) zur Propagierung von Intervallunsicherheiten. In dem dritten Beitrag wird eine Parallel-Baysian-Quadrature-Optimization-Methode (PBQO) beschrieben. Diese dient zur Schätzung der Erwartungswertfunktion sowie den dazugehörigen signifikantesten Variablen und Grenzen, unter der Annahme von rechnergestützten Modellen die hybriden Unsicherheiten in Form von Zufallsvariablen, parametrischen Probability-Boxes (P-Boxes) und Intervallmodelle als Input beinhalten. Die vierte Veröffentlichung behandelt die Funktion der Versagenswahrscheinlichkeit, wenn Inputparameter der Grenzzustandsfunktion durch parametrische P-Boxes charakterisiert werden. Zu diesem Zweck wird die Active-learning-augmented-probabilistic-Integration-Methode (ALAPI) vorgeschlagen. Diese basiert auf einer partiellen Baysian-active-learning-Perspektive zur Schätzung von Ausfallwahrscheinlichkeit sowie auf der Verwendung der High-dimensional-Model-Representation (HDMR). Hierzu wird eine Obergrenze für die a posteriori Varianz der Ausfallwahrscheinlichkeit abgeleitet. Diese erzeugt eine Limitierung der epistemischen Unsicherheit über die Ausfalwahrscheinlichkeit, aufgrund einer Art numerischer Unsicherheit, d.h. Diskretisierungsfehler. Der fünfte Beitrag stärkt die zuvor entwickelte ALPI-Methode (Active-

learning-probabilistic-Integration) in zweierlei Hinsicht: Es wird die Nutzung paralleler Berechnungen ermöglicht und die Fähigkeit verbessert, kleine Ausfallwahrscheinlichkeiten zu bewerten. Die daraus resultierende Methode wird als Parallel-adaptive-Baysian-Quadrature (PABQ) bezeichnet. In der sechsten Forschungsarbeit wird eine prinzipientreue Baysian-Failure-Probability-Inference (BFPI) vorgestellt, bei dem die a posteriori Varianz der Ausfallwahrscheinlichkeit abgeleitet wird (nicht in geschlossener Form). Außerdem wird auf der Grundlage des BFPI-Frameworks eine Parallel-adaptive-Baysian-Failure-Probability-learning-Methode (PA-BFPI) entwickelt. Als siebte Weiterentwicklung wird eine Partially-Baysian-active-learning-Line-Sampling-Methode (PBAL-LS) für die Bewertung extrem kleiner Ausfallwahrscheinlichkeiten vorgeschlagen. Hierbei wird eine Partially-Baysian-active-learning-Methode für die klassische LS-Methode eingeführt und eine Obergrenze für die a posteriori Varianz der Ausfallwahrscheinlichkeit abgeleitet. Im Anschluss an die PBAL-LS-Methode liefert die achte Entwicklung den Ausdruck für die a posteriori Varianz der Ausfallwahrscheinlichkeit im LS-Framework. Ferner wird eine Bayesian-active-learning-Line-Sampling-Methode (BALLS) vorgestellt. Der neunte Beitrag bietet eine weitere Alternative zum traditionellen Line-Sampling: Bayesian-active-learning-Line-Sampling-with-Log-Normal-Process (BAL-LS-LP). Bei dieser Methode wird anstelle eines Gauß-Prozesses ein logarithmisch normalverteilter a priori Prozess für die Betafunktion angenommen, um die Nicht-Negativitätsbeschränkung zu berücksichtigen. Außerdem wird der Approximationsfehler, resultierend aus dem Verfahren zur Findung der Nullstelle, berücksichtigt.

Zusammenfassend wird in dieser Arbeit eine Reihe neuartiger Berechnungsmethoden für Vorwärtsprobleme in der UQ vorgestellt. Insbesondere aus der Active-Learning-Perspektive nach Bayes. Die entwickelten Methoden sollen die Verfahren zur Vorwärts-UQ-Analyse ergänzen, wobei die gewonnenen Erkenntnisse weitere Studien anregen können.

Stichworte: Uncertainty-Propagation; Strukturzuverlässigkeitsanalyse; Bayesian-active-learning; Bayesian-Inference; Bayesian-Quadrature; Bayesian-Optimization; Active-learning; Gauß-Prozesse; Numerische Unsicherheiten; Line-Sampling

Contents

Chapter 1

Introduction

1.1 Opening section

Forward uncertainty quantification (UQ) involves quantifying the effects of input uncertainties on the output responses of a computational model encountered in computational science and engineering. Considerable efforts have been devoted to this area for promoting the development of versatile techniques over the past several decades. However, it is still highly desired to have more advanced methods for large-scale science and engineering applications across many diverse fields. The aim of this thesis is to develop a wide range of methodologies that can contribute to the state-of-the-art of forward UQ, especially from a Bayesian active learning perspective. This chapter will provide an introduction to the study by first discussing the research background, followed by the problem statement, aim and objectives, original contributions and structure of the thesis.

1.2 Research background

UQ in its broadest sense is a large field of science that identifies, characterizes and manages uncertainties in both computational and real world systems. It can be dated back to the 1950s and 1960s when the need to account for uncertainty in mathematical models used in various fields, such as engineering and physics, became increasingly recognized. From then on, in order to systematically account for all relevant uncertainties arising from different situations, UQ has evolved into an independent but interdisciplinary field of science. As illustrated in Fig. 1.1, such interdisciplinary nature is resulted by the interaction and collaboration of a number of disciplines, such as applied mathematics, probability and statistics, computational science and engineering, etc. As an independent discipline with distinct characteristics, UQ seeks to integrate multidisciplinary knowledge into the study of complex models in the face of uncertainty, primarily through computer simulation. Through many years, UQ has become an important aspect of the development and use of predictive computational simulation tools.

In a typical UQ problem, there could be a variety of specific tasks depending on the context. In general, the main tasks include but not limited to uncertainty identification, uncertainty characterization, forward and inverse uncertainty propagation, model calibration and validation, sensitivity analysis and decision making under uncertainty, as summarized in Fig. 1.2. To better understand the overall theme of this thesis, only three aspects among them, i.e., sources and types

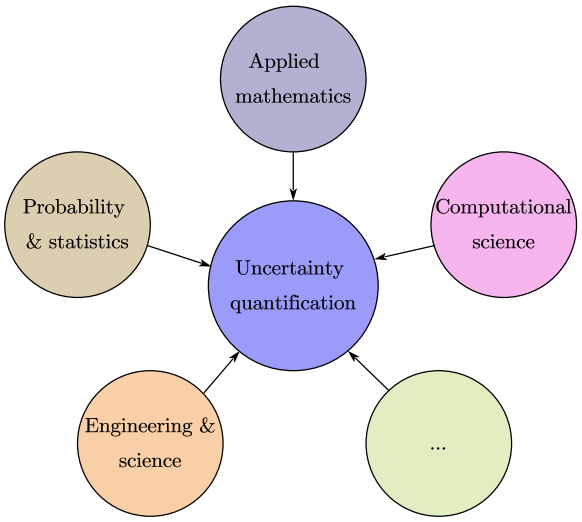


Figure 1.1: Illustration of the interdisciplinary nature of UQ.

of uncertainty, mathematical representation of uncertainty and forward uncertainty quantification, are briefly introduced in the following.

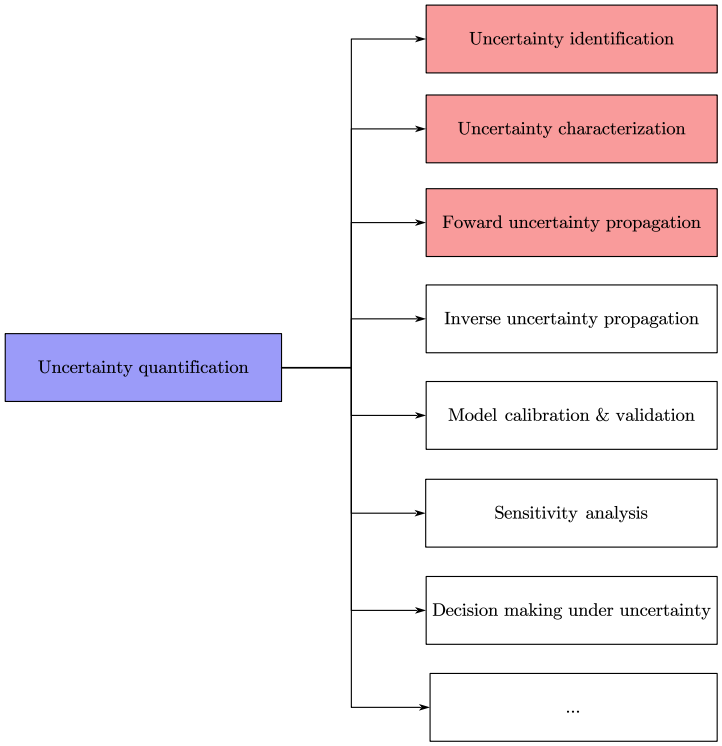


Figure 1.2: Typical tasks included in UQ.

1.2.1 Sources and types of uncertainty

Uncertainty seems to be everywhere, from everyday life to the natural sciences and engineering. Simply put, it arises whenever there is a lack of certainty. This implies that uncertainty can come from diverse sources. Therefore, identifying the sources of uncertainty is a relevant task in the practical application of UQ, and ignoring some important sources may lead to misleading and even catastrophic results in the subsequent analysis and decision making. Note that such a task in a broad sense may vary from problem to problem. In the context of predictive model building, some common sources of uncertainties include, as illustrated in Fig. 1.3, uncertainties in the mathematical forms of the models (i.e., model bias), uncertainties coming from the model parameters (e.g., model initial and boundary conditions), uncertainties arising from the experimental data (e.g., sparse data and measurement error) used to calibrate the models, as well as uncertainties stemming from numerical solving (e.g., discretization error and approximation error).

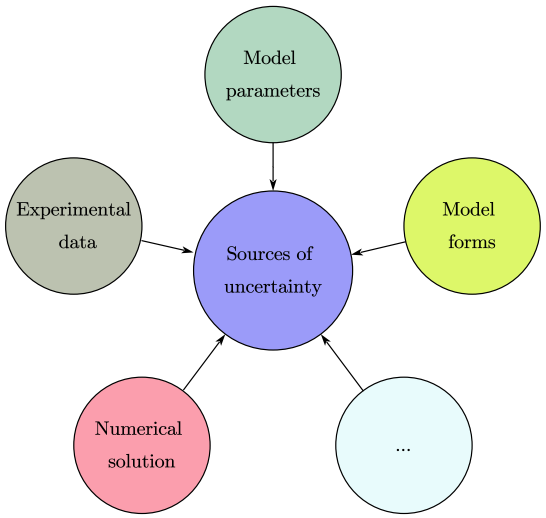


Figure 1.3: Some common sources of uncertainties in the context of predictive model developing.

In general, uncertainty can be grouped in many different ways, e.g., [1–3]. However, it is more common to divide uncertainty into two types, i.e., aleatory and epistemic uncertainty [4–6]. Aleatory uncertainty (also known as statistical, stochastic, objective or irreducible uncertainty) refers to the inherent randomness of a phenomena. It is thus objective, and cannot in principle be reduced by acquiring more knowledge or information. Epistemic uncertainty (also known as ignorance, subjective or reducible uncertainty) relates to the uncertainty caused by any lack of knowledge. Therefore, it is subjective and likely to be reduced or eliminated if more relevant knowledge or

information is available. It is worth noting that aleatory uncertainty and epistemic uncertainty do not always exist independently, but can also occur simultaneously in a single term. We refer to the latter as mixed aleatory-epistemic uncertainty, and classify uncertainty into three categories in this study, as shown in Fig. 1.4.

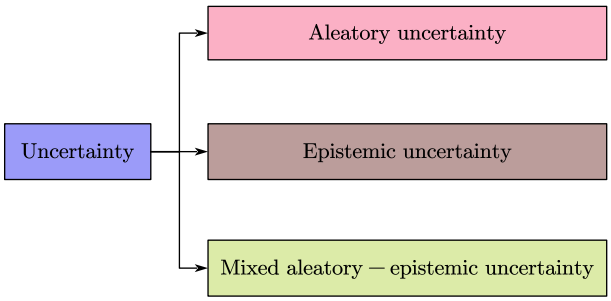


Figure 1.4: Three types of uncertainty.

1.2.2 Mathematical representation of uncertainty

The quantitative characterization of uncertainty by mathematical models is recognized as the basis for most, if not all, of UQ tasks. Many different models have been available for representing uncertainty in different situations. These models can be broadly grouped into three categories (as shown in Fig. 1.5): probabilistic models, non-probabilistic models and imprecise probabilistic models.

1.2.2.1 Probabilistic models

Probabilistic models are the most popular tool in the quantitative mathematical treatment of uncertainty. They are deeply grounded on the classical probability theory, which is a branch of mathematics concerned with probability. The term probability has many interpretations, among which two prevailing are the frequentist and Bayesian perspectives. The frequentist thinks of probability as the relative frequency of a repeated event, whereas the Bayesian interprets probability as a measure of the belief or confidence. As a result, both aleatory and epistemic uncertainty can be modelled by using probabilistic models.

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, where Ω denotes a sample space containing all possible outcomes of a random event, $\mathcal{F}$ is a σ-algebra of mensurable subsets (events) of Ω, and $\mathbb{P}$ is a

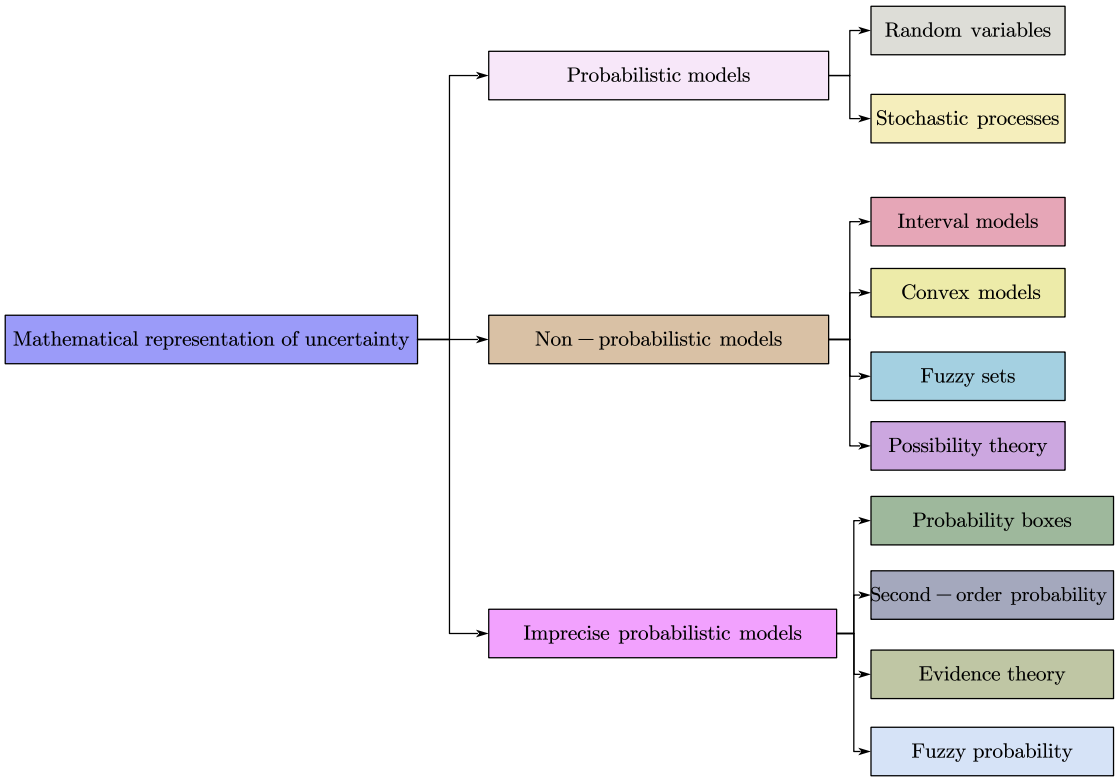


Figure 1.5: Some typical uncertainty representation models.

probability measure function that assigns a probability (between 0 and 1) to each event belonging to $\mathcal{F}$. Note that the probability of an empty set is zero (i.e., $\mathbb{P}(\emptyset) = 0$), while the probability of all possible outcomes is one (i.e., $\mathbb{P}(\Omega) = 1$).

In the above setting, a random variable (denoted as X for convenience) is then introduced to assign numbers to outcomes in the sample space. More formally, X is defined as a measurable function $X(\omega)$: $\Omega \to \mathcal{X}$, where $\omega \in \Omega$ represents an elementary event and $\mathcal{X} \subseteq \mathbb{R}$ is the domain of X. A realization or sample of X is usually denoted by the corresponding lowercase letter x. A random variable is said to be discrete (respectively continuous) if it has a countable (respectively uncountable) number of possible values.

The distribution of a random variable X can be fully characterized by its cumulative distribution function (CDF) $F_X : \mathcal{X} \to [0, 1]$, which is defined by:

$$F_X(x) = \mathbb{P}\left\{\omega \in \Omega | X(\omega) \leq x\right\}. \tag{1.1}$$

This is often written as $F_X(x) = \mathbb{P}\{X \leq x\}$. The probability mass and density functions can also be used to describe a discrete and a continuous random variable, respectively. The probability mass function (PMF) of a discrete random variable X is given by:

$$f_X(x_i) = \mathbb{P}\{\omega \in \Omega | X(\omega) = x_i\}. \tag{1.2}$$

In a more concise notation, $f_X(x_i) = \mathbb{P}\{X = x_i\}$. For a continuous random variable X, the probability density function (PDF) is defined as:

$$f_X(x) = \frac{dF_X(x)}{dx}. \tag{1.3}$$

The aforementioned concepts (i.e., CDF, PMF and PDF) are schematically illustrated in Fig. 1.6.

The summary statistics, such as the expectation and variance, of a random variable are often of interest. The expectation μ of a random variable X is defined as:

$$\mu = E(X) = \begin{cases} \sum_i x_i f_X(x_i), X \text{ is discrete} \\ \int_{\mathcal{X}} x f_X(x) dx, X \text{ is continuous} \end{cases}. \tag{1.4}$$

The variance σ^2 of a random variable X is given by:

$$\sigma^2 = \mathrm{Var}(X) = \begin{cases} \sum_i (x_i - \mu)^2 f_X(x_i), X \text{ is discrete} \\ \int_{\mathcal{X}} (x - \mu)^2 f_X(x) dx, X \text{ is continuous} \end{cases}. \tag{1.5}$$

By definition, the expectation describes the average value and the variance measures the dispersion around the expectation.

When dealing with multiple random variables, we can define a random vector $\boldsymbol{X} = [X_1, X_2, \cdots, X_n]^T$ on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, where each individual component X_i represents a random variable. The random vector $\boldsymbol{X}$ can be characterized by its joint CDF:

$$F_{\boldsymbol{X}}(\boldsymbol{x}) = \mathbb{P}\{X_1 \leq x_1, X_2 \leq x_2, \cdots, X_n \leq x_n\}, \tag{1.6}$$

where $\boldsymbol{x} = [x_1, x_2, \cdots, x_n]^T$. In case of n discrete random variables, the joint PMF of $\boldsymbol{X}$ can be

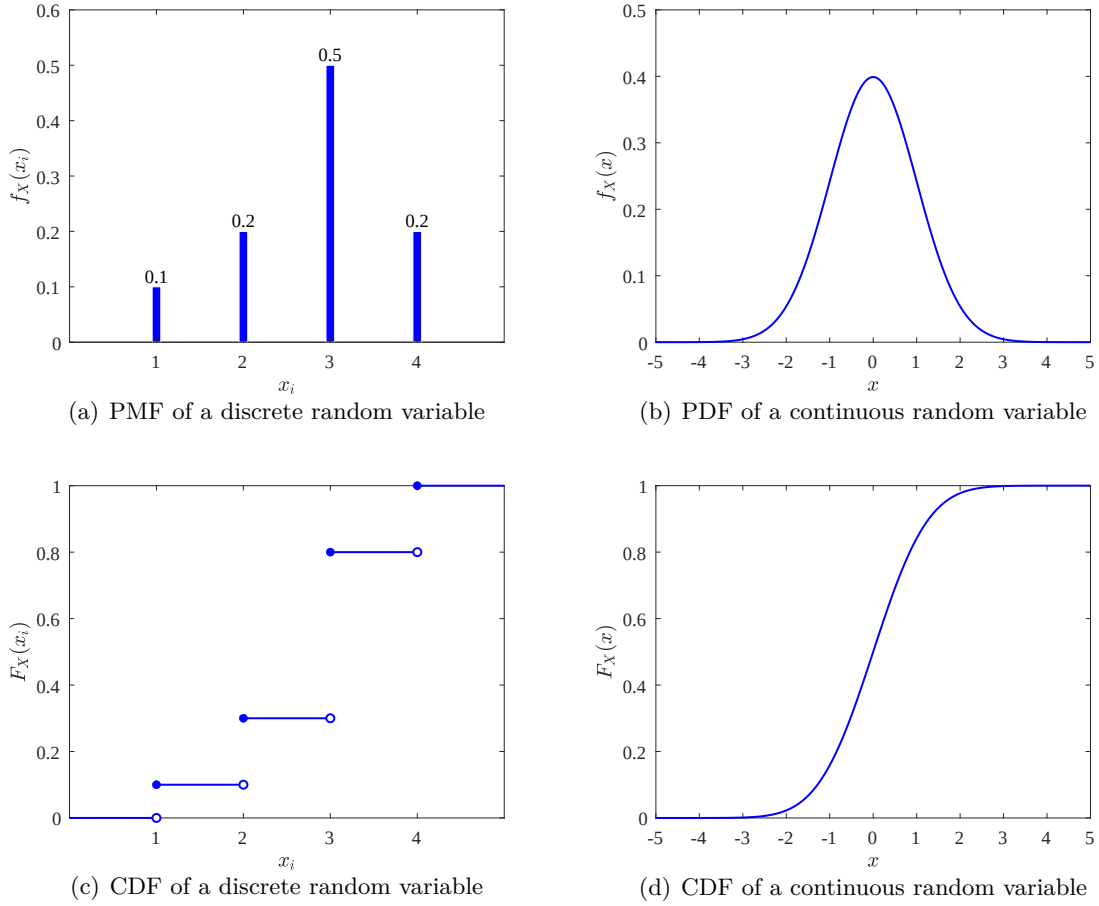
defined as:

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \mathbb{P}\{X_1 = x_1, X_2 = x_2, \cdots, X_n = x_n\}. \tag{1.7}$$

In case of n continuous random variables, the joint PDF of $\boldsymbol{X}$ can be expressed as:

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \frac{\partial^n}{\partial x_1 \cdots \partial x_n} F_{\boldsymbol{X}}(\boldsymbol{x}), \tag{1.8}$$

given that $F_{\boldsymbol{X}}(\boldsymbol{x})$ is differentiable.



(a) PMF of a discrete random variable

(b) PDF of a continuous random variable

(c) CDF of a discrete random variable

(d) CDF of a continuous random variable

Figure 1.6: Examples of probability distributions of discrete and continuous random variables.

Now, we move on to the discussion of a stochastic process (also known as random process), which is defined as a collection of random variables indexed by some mathematical set. Given the

same probability space $(\Omega, \mathcal{F}, \mathbb{P})$, a stochastic process can be written as:

$$\{X(t,\omega) : t \in T, \omega \in \Omega\}, \tag{1.9}$$

where T is an arbitrary set, but often totally ordered. A stochastic process $\{X(t,\omega) : t \in T, \omega \in \Omega\}$ is usually denoted as $\{X(t), t \in T\}$ or simply as $X(t)$ omitting its dependence on ω. For a fixed $t \in T$, $X(\cdot,\omega)$ is a random variable, whereas for a given $\omega \in \Omega$, $X(t,\cdot)$ is a realization or sample of the stochastic process (as schematically shown in Fig. 1.7).

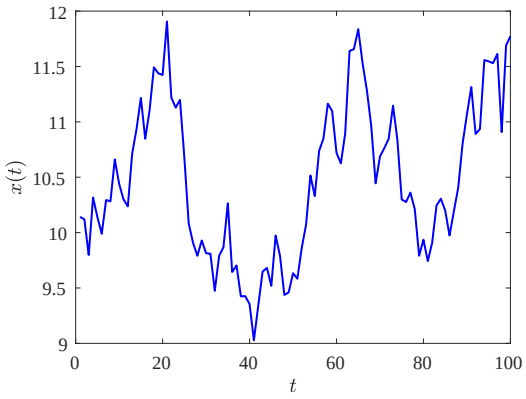


Figure 1.7: A realization of a stochastic process $X(t,\omega)$.

A stochastic process can be fully described by its finite-dimensional distributions, while they are not given here for the sake of simplicity. Alternatively, we present the more commonly-used summary statistics, i.e., expectation and covariance functions, which are defined as follows:

$$\mu(t) = E_\Omega \left[X(t,\omega)\right], \tag{1.10}$$

$$c(t_1,t_2) = E_\Omega \left[\left(X(t_1,\omega) - \mu(t_1)\right)\left(X(t_2,\omega) - \mu(t_2)\right)\right], \tag{1.11}$$

where E_Ω denotes the expectation taken over the sample space Ω. From the covariance function, we can obtain a variance function as $\sigma^2(t) = c(t,t)$. It should be noted that these moments are only partial characterizations for a general stochastic process.

Stochastic processes can be categorized into several types based on their properties and behavior. In terms of stationarity, a stochastic process can be classified into either stationary or non-stationary. A stochastic process is said to be stationary if its statistical properties (such as the

mean and variance) do not change over time, while a non-stationary stochastic process does the opposite.

1.2.2.2 Non-probabilistic models

In contrast to the probabilistic models, non-probabilistic models typically treat uncertainty with set-theoretic approaches, rather than precise probability values. Representative examples of such models include interval models, convex models, fuzzy models.

Interval models [7, 8] represent the most basic type of mathematical models for describing uncertainty in a non-probabilistic way. In this context, an interval scalar is defined as a bounded set of real numbers. Depending on whether the set includes its limit points, there exist four types of intervals (i.e., closed, open, left-closed and right-open, and left-open and right-closed), which are defined as:

$$[x] = [\underline{x}, \overline{x}] = \{x \in \mathbb{R} : \underline{x} \leq x \leq \overline{x}\}, \tag{1.12}$$

$$]x[= (\underline{x}, \overline{x}) = \{x \in \mathbb{R} : \underline{x} < x < \overline{x}\}, \tag{1.13}$$

$$[x[= [\underline{x}, \overline{x}) = \{x \in \mathbb{R} : \underline{x} \leq x < \overline{x}\}, \tag{1.14}$$

$$]x] = (\underline{x}, \overline{x}] = \{x \in \mathbb{R} : \underline{x} < x \leq \overline{x}\}, \tag{1.15}$$

where $\underline{x} \in \mathbb{R}$ and $\overline{x} \in \mathbb{R}$ denote respectively the left and right endpoints of the set, and $\underline{x} < \overline{x}$. See Fig. 1.8 for an illustration of a closed interval scalar. In a large body of literature, intervals are defined by default as closed. For this reason, when it is referred to an interval in the following, we mean a closed interval.

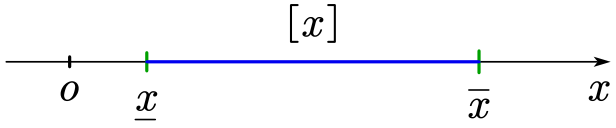


Figure 1.8: Illustration of an interval scalar.

An interval vector is an array of which each element is an interval, that is:

$$[\boldsymbol{x}] = \begin{bmatrix} [x]_1 \\ [x]_2 \\ \vdots \\ [x]_n \end{bmatrix} = \{\boldsymbol{x} \in \mathbb{R}^n : \underline{x}_i \leqslant x_i \leqslant \overline{x}_i\}, \tag{1.16}$$

where x_i denotes the i-th interval with lower and upper bounds $\underline{x}_i$ and $\overline{x}_i$, $i = 1, 2, \cdots, n$. For illustrating an interval vector in two dimensions, one can refer to Fig. 1.9. An n-dimensional interval vector describes a hyperrectangle in an n-dimensional Euclidean space.

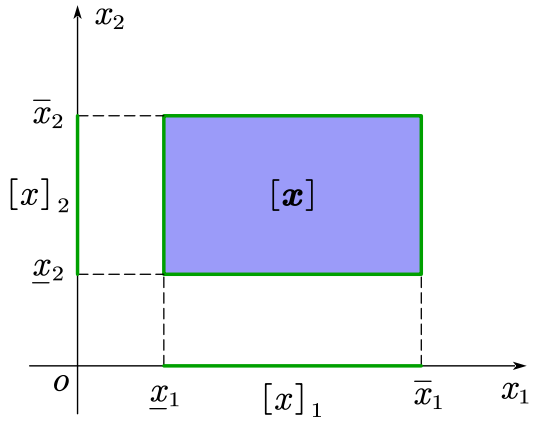


Figure 1.9: Illustration of an interval vector in two dimensions.

Similarly, an interval matrix is a matrix whose entries are intervals, that is:

$$[\boldsymbol{X}] = \begin{bmatrix} [x]_{11} & [x]_{12} & \cdots & [x]_{1n} \\ [x]_{21} & [x]_{22} & \cdots & [x]_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ [x]_{m1} & [x]_{m2} & \cdots & [x]_{mn} \end{bmatrix} = \left\{\boldsymbol{X} \in \mathbb{R}^{m\times n} : \underline{x}_{ij} \leq x_{ij} \leq \overline{x}_{ij}\right\}, \tag{1.17}$$

where x_{ij} represents the (i, j)-th entry with lower and upper bounds $\underline{x}_{ij}$ and $\overline{x}_{ij}$, $i = 1, 2, \cdots, m$, $j = 1, 2, \cdots, n$.

In addition, the interval concept can also be extended to interval processes or fields, see for instance [9, 10]. It should be pointed out that the elements of an interval vector or an interval matrix are assumed to be mutually independent by definition. However, the validity of this assumption has

been questioned and some techniques have been developed to account for the dependence between multiple intervals as in the context of modelling interval processes or fields.

Convex models [11, 12] are another family of non-probabilistic uncertainty models. A convex model is defined by a convex set, i.e., a collection of points in which the line segment AB connecting any two points A, B in the set lies completely within the set. Thereby, dependence between multiple uncertain variables can be considered in convex models and interval models can be seen a special case of convex models. Various types of convex models have been developed so far, such as the ellipsoid convex model [11], super ellipsoid convex model [13], multi-ellipsoid convex model [14], multidimensional parallelepiped convex model [15] and exponential convex model [16]. Among these models, the ellipsoid convex model is the most classical and widely used model, and it will be introduced below.

An n-dimensional ellipsoidal convex model is defined as:

$$\boldsymbol{Y} = \left\{ \boldsymbol{y} \in \mathbb{R}^{\boldsymbol{n}} : \left(\boldsymbol{y} - \boldsymbol{y}^0\right)^{\mathrm{T}} \boldsymbol{G} \left(\boldsymbol{y} - \boldsymbol{y}^0\right) \leq 1 \right\}, \tag{1.18}$$

where $\boldsymbol{y}^0 = \left[y_1^0, y_2^0, \cdots, y_n^0\right]^{\mathrm{T}}$ denotes the central point of the ellipsoid, hence determining its location; $\boldsymbol{G}$ is the characteristic matrix of the ellipsoid:

$$\boldsymbol{G} = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ g_{n1} & g_{n2} & \cdots & g_{nn} \end{bmatrix}, \tag{1.19}$$

which determines the size and the orientation of the ellipsoid. Fig. 1.10 shows a 2-dimensional ellipsoidal convex model, where the marginal model of each dimension can be interpreted as an interval model $[y]_i$.

The convex model theory has also found applications in modeling spatial or temporal uncertainty, see for example [17, 18].

Fuzzy models [19, 20] appear to be a body of non-probabilistic uncertainty models that are based on the fuzzy set theory. A fuzzy set can be seen as an extension of the classical notation of set (known as crisp set). In a crisp set, an element is either a member of the set or not, while a fuzzy set assigns to each element a grade of membership ranging between zero and one. In this

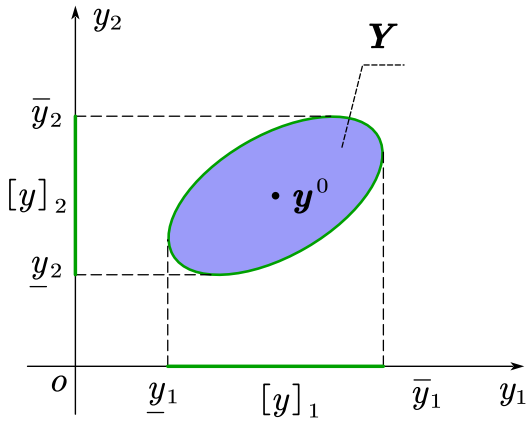


Figure 1.10: Illustration of an ellipsoidal convex model in two dimensions.

context, a fuzzy variable $\tilde{x}$ is define as:

$$\tilde{x} = \{(x, \mu(x)) : x \in \mathcal{D}_{\tilde{x}}, \mu(x) \in [0,1]\}, \tag{1.20}$$

where $\mu(x)$ is the membership function (also known as characteristic function); $\mathcal{D}_{\tilde{x}} \subseteq \mathbb{R}$ is the domain of $\tilde{x}$. If $\mu(x) = 0$, x is definitely not a member of $\tilde{x}$. On the contrary, x is definitely a member of $\tilde{x}$ if $\mu(x) = 1$. In case $0 < \mu(x) < 1$, the membership is not certain. A fuzzy variable with a triangular membership function is schematically shown in Fig. 1.11. Given a membership level $\alpha \in (0, 1]$, we can obtain a crisp set from $\tilde{x}$:

$$\tilde{x}_\alpha = \{x \in \mathcal{D}_{\tilde{x}} :, \mu(x) \geq \alpha\}, \tag{1.21}$$

which is called α-level set.

Likewise, an n-dimensional fuzzy vector can be expressed as:

$$\tilde{\boldsymbol{x}} = \{(\boldsymbol{x}, \mu(\boldsymbol{x})) : \boldsymbol{x} \in \mathcal{D}_{\tilde{\boldsymbol{x}}}, \mu(\boldsymbol{x}) \in [0,1]\}. \tag{1.22}$$

where $\mathcal{D}_{\tilde{\boldsymbol{x}}} \subseteq \mathbb{R}^n$ is the domain of $\tilde{\boldsymbol{x}}$.

For fuzzy processes and fuzzy fields, one can refer to, e.g., [21].

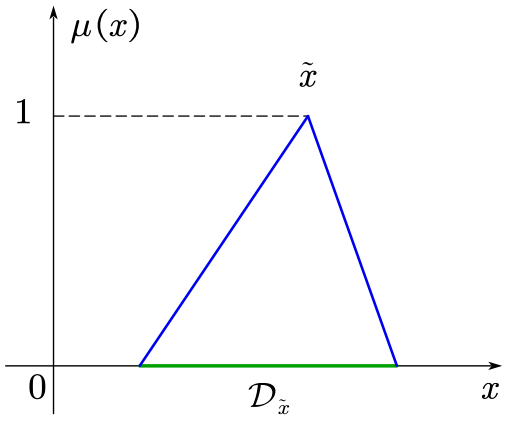


Figure 1.11: Illustration of a fuzzy variable with a triangular membership function.

1.2.2.3 Imprecise probabilistic models

Imprecise probabilistic models are another class of uncertainty models that extends and generalizes the classical precise probabilistic models, allowing for partial probability specifications. The basis for imprecise probabilistic models is the theory of imprecise probability [22, 23]. Rather than only a single probability measure in probability theory, the core idea of imprecise probability theory is to represent uncertainty using a set of probability measures. In doing so, an imprecise probabilistic model can admit imprecision in probabilistic uncertainty modeling, allowing for the simultaneous consideration of aleatory and epistemic uncertainty. This is very useful when the amount of information available is not sufficient enough to specify a precise probabilistic model. Among many imprecise probabilistic models that have been developed so far, the probability box (p-box) models, evidence theory-based models and fuzzy probability models are briefly described below.

P-box models [24, 25] impose interval-type bounds on admissible CDFs, and hence can be seen as an expressive generalization of both probabilistic models and interval models. Recall that in the context of probabilistic models a random variable can be fully characterized by its CDF. However, this CDF may not be precisely known due to our limited knowledge. P-box models can release the requirement of an exact CDF. To be specific, the true-but-unknown CDF $F(x)$ of a random variable X is supposed to lie within some interval such that:

$$\underline{F}(x) \leq F(x) \leq \overline{F}(x), \tag{1.23}$$

where $\underline{F}(x)$ and $\overline{F}(x)$ are two admissible CDFs and satisfy $\underline{F}(x) \leq \overline{F}(x)$ for all $x \in \mathcal{X}$. Usually,

we refer to $\underline{F}(x)$ and $\overline{F}(x)$ as the lower (or right) and upper (or left) CDF bounds receptively. Such a pair $\left[\underline{F}(x), \overline{F}(x)\right]$ formally defines a p-box model. The difference between $\underline{F}(x)$ and $\overline{F}(x)$ can reflect the amount of our epistemic uncertainty about the true CDF of X, which describes its aleatory uncertainty. In this regard, a p-box model can account for the mixed aleatory-epistemic uncertainty to a certain degree. This property can be illustrated by considering two extreme cases. In an extreme case that the epistemic uncertainty disappears (i.e., $\underline{F}(x) = \overline{F}(x)$) but the aleatory uncertainty still exists, a p-box model reduces to a single CDF, and hence a probabilistic model. In another extreme case that the aleatory uncertainty disappears (i.e., $X = x^\star$, where $x^\star \in \mathcal{X}$ is a constant) but the epistemic uncertainty still exists, the p-box model degenerates into an interval model.

In general, p-box models can be classified into two types, namely distribution-free p-box models and distributional p-box models. A distribution-free p-box model does not need to make any assumption about the distribution type pertaining to the set of possible CDFs. That is to say, any valid CDF is allowed within the given CDF bounds, even non-smooth one. Fig. 1.12(a) depicts a distribution-free p-box model, including its upper and lower CDF bounds and four realizations. On the contrary, a distributional p-box model assumes that the set of possible CDFs comes from a certain distribution family with interval parameters:

$$F(x) = \left\{ F(x|\boldsymbol{\theta}) : \boldsymbol{\theta} \in \left[\underline{\boldsymbol{\theta}}, \overline{\boldsymbol{\theta}}\right] \right\}, \tag{1.24}$$

where $\boldsymbol{\theta} = [\theta_1, \theta_2, \cdots, \theta_{n_{\boldsymbol{\theta}}}]$ is a vector of $n_{\boldsymbol{\theta}}$ interval variables; $\underline{\boldsymbol{\theta}}$ and $\overline{\boldsymbol{\theta}}$ represent the lower and upper bounds of $\boldsymbol{\theta}$, receptively. In this context, the lower and upper CDF bounds of a distributional p-box model can be expressed as:

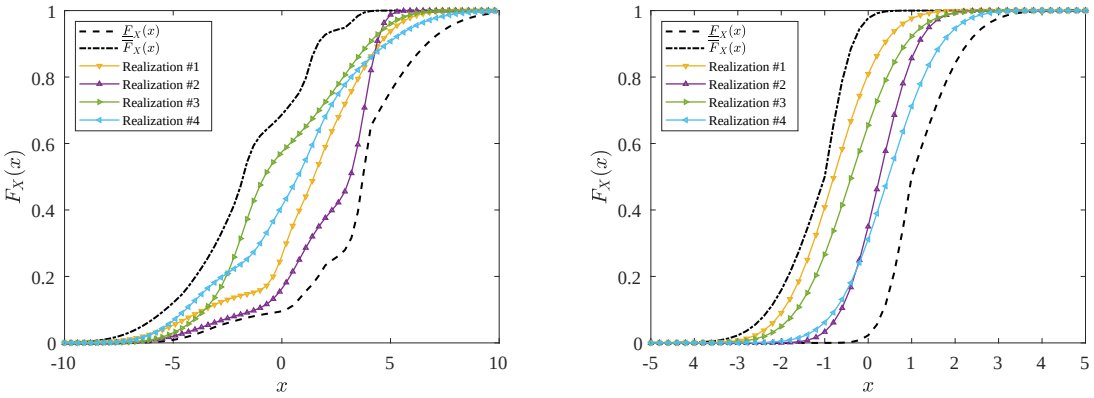
$$\underline{F}(x) = \min_{\boldsymbol{\theta} \in \left[\underline{\boldsymbol{\theta}}, \overline{\boldsymbol{\theta}}\right]} F(x|\boldsymbol{\theta}), \tag{1.25}$$

$$\overline{F}(x) = \max_{\boldsymbol{\theta} \in \left[\underline{\boldsymbol{\theta}}, \overline{\boldsymbol{\theta}}\right]} F(x|\boldsymbol{\theta}). \tag{1.26}$$

Fig. 1.12(b) illustrates a distributional p-box model defined by a Gaussian family $\Phi(x; \mu, \sigma^2)$, where $\mu = [-1.0, 1.0]$ and $\sigma = [0.5, 1.0]$.

A vector containing multiple p-box variables can be defined without or with considering their dependency [26, 27]. Besides, it is also possible to define p-box processes (or fields) [28, 29].

Evidence theory-based models are established based on a mathematical theory of evidence, called



(a) Distribution-free p-box model

(b) Distributional p-box model

Figure 1.12: Examples of distribution-free and distributional p-box models.

Dempster-Shafer theory [30, 31] (also referred to as evidence theory or theory of belief functions). Unlike in probability theory, evidence theory introduces a set of two measures of uncertainty, i.e., belief and plausibility. Let us consider an evidence space $(\Omega, \mathcal{S}, m)$, where Ω is the universal set that contains all possible outcomes, $\mathcal{S}$ denotes the countable collection of subsets of Ω (which is not required to be a σ-algebra as is the case in probability theory), and m represents the basic probability assignment (BPA). For a given subset $\mathcal{A} \in \Omega$, the BPA defines a mapping of $\mathcal{A}$ onto the interval $[0, 1]$, i.e., $m : \mathcal{A} \rightarrow [0, 1]$, such that:

$$m(\mathcal{A}) \geq 0, \forall \mathcal{A} \in \mathcal{S}, \tag{1.27}$$

$$m(\emptyset) = 0, \tag{1.28}$$

$$\sum_{\mathcal{A} \in \mathcal{S}} m(\mathcal{A}) = 1. \tag{1.29}$$

Note that the term BPA does not refer to the probability measure any more, but reflects the degree of belief that can be assigned to $\mathcal{A}$, but to no subset of $\mathcal{A}$. The belief measure of $\mathcal{A}$ is defined as the sum of all the probabilities of the subsets of $\mathcal{A}$, while the plausibility of $\mathcal{A}$ is defined as the sum of all the probabilities of the sets that intersect the set $\mathcal{A}$:

$$Bel(\mathcal{A}) = \sum_{\mathcal{B} | \mathcal{B} \in \mathcal{A}} m(\mathcal{B}), \tag{1.30}$$

$$Pl(\mathcal{A}) = \sum_{\mathcal{B}|\mathcal{B}\cap\mathcal{A}\neq\emptyset} m(\mathcal{B}). \tag{1.31}$$

It is obvious that $Bel(\mathcal{A}) \leq Pl(\mathcal{A})$. The belief and plausibility measures can be regarded as the lower and upper bounds of the probability measure of $\mathcal{A}$ respectively, i.e., $Bel(\mathcal{A}) \leq \mathbb{P}(\mathcal{A}) \leq Pl(\mathcal{A})$. The belief interval $[Bel(\mathcal{A}), Pl(\mathcal{A})]$ reflects the amount of epistemic uncertainty about the true probability $\mathbb{P}(\mathcal{A})$. The probability theory can thus be considered as a special case of the evidence theory when $Bel(\mathcal{A}) = Pl(\mathcal{A})$. For an illustration, an evidence theory based model is displayed in Fig. 1.13.

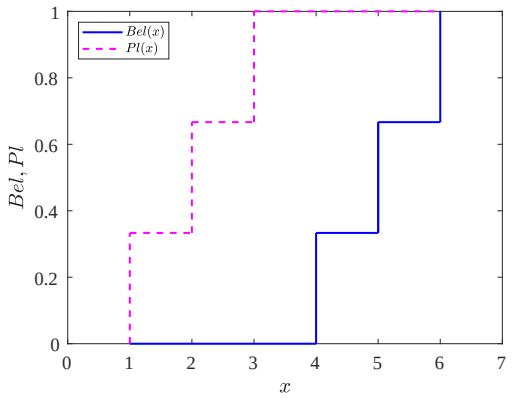


Figure 1.13: Illustration of an evidence theory based model.

For more information about the evidence theory, one can refer to, e.g., [32].

Fuzzy probability models [21] are based on the fuzzy probability theory, which can be regarded as a marriage between fuzzy set theory and probability theory. A fuzzy probability model allows for the consideration of a fuzzy set of probabilistic models, and hence mixed aleatory-epistemic uncertainty. The aleatory uncertainty is captured by a probabilistic model, while the epistemic uncertainty regarding the probabilistic model specification is reflected by a fuzzy set. A formal definition of a fuzzy random variable entails the construction of a fuzzy probability space. However, there are many ways to formalize a fuzzy probability space and there is no consensus yet. For this reason, we only introduce a fuzzy random variable in a less formal but more convenient manner. A fuzzy random variable $\tilde{X}$ can be defined by its fuzzy CDF $\tilde{F}(x)$ such that:

$$\tilde{F}(x) = \{(F(x), \mu(F(x))) : F(x) \in \mathbb{F}, \mu(F(x)) \in [0,1]\}, \tag{1.32}$$

where $\mathbb{F}$ denotes the set of admissible CDFs. See Fig. 1.14 for an illustration of a fuzzy random variable. An α-level set of $\tilde{F}(x)$ can be expressed as:

$$\tilde{F}_\alpha(x) = \{F(x) \in \mathbb{F} : \mu(F(x)) \geq \alpha\}, \tag{1.33}$$

where $\alpha \in (0, 1]$. In fact, each α-level set $\tilde{F}_\alpha(x)$ can lead to a distribution-free p-box model.

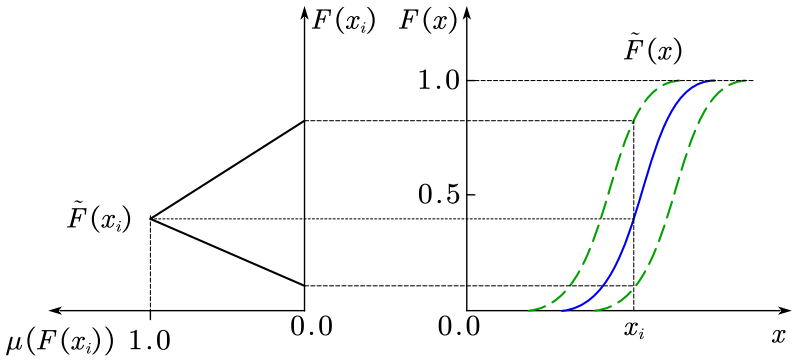


Figure 1.14: Illustration of a fuzzy probability model.

1.2.3 Forward uncertainty quantification

As a main task of UQ, forward UQ (also known as uncertainty propagation) aims at quantifying the output uncertainty of a computational model subject to input uncertainty. This can be conceptually illustrated by Fig. 1.15. A core component lies in a computational model, which is a mathematical representation of a real system. A computational model can be either deterministic or stochastic, depending on whether it always produces the same outputs whenever evaluated on the same inputs. In this thesis, we restrict our focus to only deterministic computational models. They are represented by an input-output mapping $\mathcal{M}(\cdot) : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$, where $d_1 \in \mathbb{N}_1$ and $d_2 \in \mathbb{N}_1$ denote the dimensions of the input and output spaces receptively. The model inputs specify some proprieties of the system under consideration, while the model outputs are constituted by some quantities of interest. Let us denote the inputs and outputs as $\boldsymbol{X} = [X_1, X_2, \cdots, X_{d_1}] \in \mathbb{R}^{d_1}$ and $\boldsymbol{Y} = [Y_1, Y_2, \cdots, Y_{d_2}] \in \mathbb{R}^{d_2}$ respectively. In our context, the inputs of $\mathcal{M}(\cdot)$ are not deterministic values, but subject to uncertainty. Assume that the uncertain model inputs have been well described by suitable mathematical uncertainty models. The uncertainty in model inputs propagated through a computational model leads to the uncertainty of the model outputs $\boldsymbol{Y} = \mathcal{M}(\boldsymbol{X})$. The objective

of forward UQ is to understand how uncertainty in the model inputs $\boldsymbol{X}$ affects the uncertainty in the model outputs $\boldsymbol{Y}$.

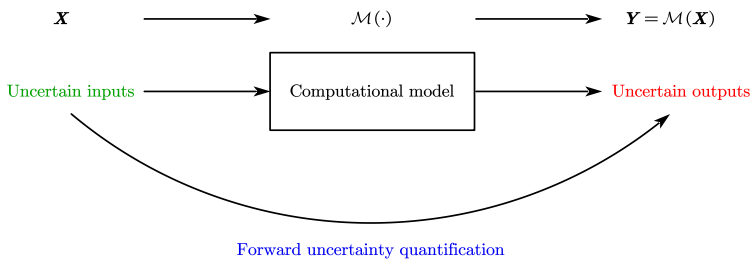


Figure 1.15: Schematic illustration of forward UQ.

The content of UQ is very rich, and it is continuously enriched. This makes it unrealistic to give a comprehensive overview. In the following, we only give a brief introduction along each class of input uncertainty models as well as a combination thereof. For the convenience of description, a scalar model output Y is considered.

1.2.3.1 Propagation of probabilistic uncertainty

In case that the model inputs are fully described by probabilistic uncertainty models (i.e., $\boldsymbol{X}$ is a vector of d_1 random variables), the scalar model output Y should also be a random variable. The ultimate goal of a forward UQ analysis is to obtain certain statistics of the random variable Y. Various methods have been developed to obtain different types of statistics:

- Probability distribution of Y. The uncertainty of Y can be fully characterized once its PDF $f_Y(y)$ or CDF $F_Y(y)$ is derived. For this purpose, typical methods include but are not limited to Monte Carlo simulation (MCS) [33], probability density evolution method [34], probability transformation method [35, 36], direct probability integral method [37] and maximum entropy method [38, 39].

- Statistical moments of Y, e.g., mean and variance, which can be defined by:

$$\mu_Y = \int_{\mathbb{R}^{d_1}} \mathcal{M}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{1.34}$$

$$\sigma_Y^2 = \int_{\mathbb{R}^{d_1}} \left[\mathcal{M}(\boldsymbol{x}) - \mu_Y\right]^2 f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}. \tag{1.35}$$

In a general setting, the statistical moments of Y can only characterize Y partially. One can resort to MCS [40], sparse grid integration [41], cubature formula [42], Bayesian quadrature [43, 44], univariate-dimension reduction method [45], etc.

- Probabilistic reliability analysis. The central task is to compute the complement of the reliability, that is the so-called failure probability. If we assume that a failure occurs whenever Y exceeds a certain threshold b, then the failure probability P_f can be defined by:

$$P_f = \mathbb{P}\left\{Y > b\right\} = \int_{\left\{\boldsymbol{x}\in\mathbb{R}^{d_1}:\mathcal{M}(\boldsymbol{x})>b\right\}} f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}. \tag{1.36}$$

In this context, a considerably large number of methods are available in the literature. Examples of such methods include MCS and its variants [46, 47], first-/second-order reliability method [48, 49], AK-MCS [50], high-order moment method [51].

1.2.3.2 Propagation of non-probabilistic uncertainty

When the model inputs are described by non-probabilistic uncertainty models, the quantity of interest Y should also be non-probabilistic. For instance, if $\boldsymbol{X}$ is a vector of d_1 interval (fuzzy) variables, Y should be a interval (fuzzy) variable as well. Under non-probabilistic inputs, many research efforts have been devoted to the forward UQ analysis along the following directions:

- Non-probabilistic characterization of Y. This is aimed at giving a full account for the non-probabilistic content of Y. In case that Y is an interval variable, of interest are its lower and upper bounds that completely characterize Y. Given that Y is identified as a fuzzy variable, a full characterization could be its fuzzy set or α-discretization description. One can refer to [52] for an overview of recent advances in the field of non-probabilistic uncertainty propagation.

- Non-probabilistic reliability analysis. This mainly involves defining non-probabilistic reliability measures and developing efficient methods for computing those measures. Existing studies include, e.g., [53–56].

1.2.3.3 Propagation of imprecise probabilistic uncertainty

Supposing that the model inputs are considered as imprecise probabilistic models, the model output might be imprecise probabilistic as well. For example, whenever the computational model

is subject to a vector of d_1 p-box variables, the model output should also be a p-box variable. For propagating imprecise probabilistic uncertainty, the following aspects have received particular attention:

- Complete characterization of Y. In order to obtain the output p-box when the inputs are p-boxes, there are several existing methods as documented in [57–60]. In case that one is interested in the output evidence theory-based model when the inputs are modeled by evidence theory, the relevant methods can be found in [61–65]. When it comes to propagating the inputs modeled by fuzzy probability theory to a fuzzy random output, the reader is referred to [21].

- Partial characterization of Y. Like in probabilistic uncertainty propagation, it is also possible to consider the summary statistics of Y in certain situations. For example, if the inputs are described by distributional p-box models, the output variable should also be a distributional p-box variable and we can define its moments, e.g., mean and variance. However, they are not crisp values but functions with respect to interval distribution parameters of the inputs. In this context, some research efforts have been made to obtain the response moment functions and/or their bounds [59, 60, 66].

- Imprecise probabilistic reliability analysis. In contrast to probabilistic reliability analysis, the likelihood of failure usually cannot be measured by a crisp failure probability in imprecise probabilistic reliability analysis. For example, assuming that the model inputs are distributional p-boxes, the failure probability turns out to be a function with respect to interval distribution parameters of the inputs. In order to derive the failure probability bounds, it is referred to [25] for an overview of existing computational methods. For reliability analysis under evidence theory, recent progresses can also be found in a review [67]. When it comes to fuzzy probabilistic reliability analysis, the interested reader is referred to [21].

1.2.3.4 Propagation of polymorphic uncertainties

For a single computational model, it is not always possible or reasonable to describe the model inputs by the same type of uncertainty models. This promotes the need to take different types of uncertainty models into consideration, and hence to propagate polymorphic uncertainties. Of course, various situations may occur due largely to the diversity of uncertainty models. Therefore,

it is unrealistic for us to provide a comprehensive overview. For this reason, we only discuss the propagation of probability-interval hybrid uncertainties as a simple illustration. This means that the model inputs $\boldsymbol{X}$ are assumed to have only random and interval variables. Under this assumption, the model output Y can be regarded as a p-box variable. A forward UQ analysis in the context of probability-interval hybrid uncertainties is typically related to the following topics:

- Complete characterization of Y. As Y can be regarded as a p-box variable, a complete characterization of it involves determining the lower and upper CDF bounds, see for example [68].
- Partial characterization of Y. A usual practice is to consider the moments (such as mean and variance) of Y, which are functions with respect to the input interval variables. Thus, efforts have been made to estimate response moment bounds [68–70].
- Probability-interval hybrid reliability analysis. The failure probability is no longer a crisp value, but a function of the input interval variables. In view of this, there is increasing attention paid to deriving the failure probability bounds [71–75].

1.2.4 Bayesian active learning and Bayesian probabilistic numerics

In order to better understand what Bayesian active learning is, let us first introduce active learning. The term active learning can refer to a method of learning in which students are actively rather than passively involved in the learning process. Besides, it is more commonly known as a form of machine learning, and this is exactly what we want to talk about. In the context of machine learning, active learning refers to any learning algorithm that can interactively query the user (or some other information source) in order to obtain the desired outputs at new data points. This allows the learning algorithm to learn more efficiently by focusing on the most informative data points, rather than just training on a randomly selected subset of the data. Active learning can find applications in a wide range of fields, including forward UQ. Two pioneering methods that make use of active learning for probabilistic reliability analysis are efficient global reliability analysis (EGRA) [76] and AK-MCS [50]. After that, significant progress has been made to develop active learning methods for forward UQ, especially for probabilistic reliability analysis, see, e.g., [77, 78].

Bayesian active learning is a specific method for active learning that takes advantage of Bayesian principles. It is typically used in the context of Bayesian models. In a Bayesian model, our prior

beliefs about the model parameters are represented by prior distributions. Once data is observed, these prior distributions are updated to posterior distributions, which reflect the updated beliefs about the model parameters given the observed data. The updating process is typically done by using Bayes' theorem. A Bayesian active learning algorithm selects the most informative data points to label based on the expected improvement of the model's posterior distribution. This allows one to incorporate prior knowledge and model uncertainty in the process of selecting the most informative samples. In contrast, traditional active learning methods more rely on some model-agnostic criteria, such as uncertainty sampling, query-by-committee, or expected error reduction. Due to its attractive features, Bayesian active learning has been successfully applied in many fields, e.g., machine learning [79], natural language processing [80], computer vision [81], robotics [82] and so on.

Along with or even before the Bayesian active learning in machine learning, a branch of numerical analysis, known as Bayesian probabilistic numerics [83, 84], emerged to use Bayesian statistical techniques to solve numerical problems, e.g., integration, optimization and differential equations. Compared to deterministic numerics, Bayesian probabilistic numerics allow for developing numerical methods that have a probabilistic interpretation. Although Bayesian active learning and Bayesian probabilistic numerics were developed in different fields for different purposes, they share very similar ideas (simply being referred as Bayesian active learning in this study). However, these ideas have only been pursued by few studies [44, 66] in the field of forward UQ.

1.3 Problem statement

Forward UQ plays an important role in understanding the effects of uncertainty in the model inputs on the model outputs, which in turn enables more informed decision-making, risk management, and improved predictions or results. As discussed above, numerous efforts have been dedicated to the development of computational methods for different purposes. However, it can be still challenging to apply those existing methods to many computational models of real-world systems, even using today's supercomputers. This can be due to, e.g., ever-increasing complexity of computational models and diversity of uncertainty models. The former is to capture the real behavior of the associated system, while the latter is to account for different kinds of uncertainties. Furthermore, in addition to the uncertainty in the model inputs, another type of uncertainty, i.e., numerical error (e.g., discretization error and approximation error), is largely ignored in the exist-

ing numerical methods for forward UQ. However, numerical error should be properly treated in the process of forward UQ analysis because it can lead to inaccurate or unreliable results. Last but not least, although some machine learning techniques (such as active learning) have been successfully used for forward UQ, their potential still needs to be fully exploited in a strategic manner.

1.4 Aim and objectives

This thesis aims to develop a wide range of forward UQ methods that can help narrow the research gap, especially from a Bayesian active learning perspective. The aim is detailed by the following objectives:

(i) To develop a dynamic reliability analysis method for nonlinear structures with uncertain parameters under stochastic seismic excitations, where the dimension of model inputs is more than one thousand;

(ii) To present a method for interval uncertainty propagation based on the use of Bayesian global optimization, which is able to generate simultaneously the upper and lower bounds of a model output, and support parallel distributed processing;

(iii) To propose a parallel Bayesian active learning method for propagating hybrid uncertainties in the form of precise probability models, distributional p-box models and interval models, where the quantities of interest are the response expectation function, its variable importance and bounds;

(iv) To provide a Bayesian active learning method for propagating distributional p-box models, where the failure probability function is of interest and discretization error should be handled in a proper way;

(v) To establish a parallel Bayesian active learning method for probabilistic reliability analysis with small failure probabilities;

(vi) To offer a Bayesian perspective, as opposed to the classical frequentist perspective, on the problem of failure probability integral estimation;

(vii) To treat the traditional line sampling for probabilistic reliability analysis with a Bayesian active learning perspective, where the numerical error is properly addressed.

1.5 Original contributions

The main work of this thesis is the development of several innovative approaches for forward UQ analysis, as well as offering some new perspectives. The main contributions can be summarized as follows:

(a) For objective (i), we develop a dynamic reliability analysis method, called 'moment-generating function based mixture distribution' (MGF-MD). The main idea lies in estimating the extreme value distribution from a small number of simulations by using the moment-generating function and a proposed mixture distribution model. A key advantage of the proposed MGF-MD method is that it is capable of handling high-dimensional nonlinear stochastic dynamic systems;

(b) For objective (ii), we present an interval uncertainty propagation method, termed 'triple-engine parallel Bayesian global optimization' (T-PBGO). In contrast to the existing BGO methods for interval uncertainty propagation, the proposed T-PBGO method can produce the lower and upper bounds of a model output of interest in a single run. In addition, it can identify multiple points to query the response function at each iteration, and hence support parallel distributed processing;

(c) For objective (iii), we propose a parallel Bayesian active learning method, called 'parallel Bayesian quadrature optimization' (PBQO), for propagating hybrid uncertainties in the form of precise probability models, distributional p-box models and interval models. In this method, estimation of response expectation function is treated as a Bayesian active learning problem, and discretization error is regarded as a kind of epistemic uncertainty;

(d) For objective (iv), we provide a Bayesian active learning method, named 'active learning augmented probabilistic integration' (ALAPI), for propagating distributional p-box models. To do so, an 'active learning probabilistic integration' (ALPI) method is first developed for probabilistic reliability analysis. In ALPI, we seek to treat the estimation of failure probability integral as a Bayesian active learning problem in which the discretization error is properly treated. Then, the ALAPI method is built upon a combination of ALPI and high-dimensional model representation in the augmented space;

(e) For objective (v), we establish a parallel Bayesian active learning method, termed 'parallel adaptive Bayesian quadrature' (PABQ), for probabilistic reliability analysis with small failure probabilities. The established method is based on our previously developed ALPI, but with two important enhancements. The first is to enable the estimation of small failure probabilities, while the second consists in facilitating parallel computing;

(f) For objective (vi), we offer a Bayesian perspective, as opposed to the classical frequentist perspective, on the problem of failure probability integral estimation. For this purpose, a principled 'Bayesian failure probability inference' (BFPI) framework is advocated based on assigning a Gaussian process prior over the performance function. In addition, a 'parallel adaptive-Bayesian failure probability learning' (PA-BFPL) method is also developed for probabilistic reliability analysis;

(g) For objective (vii), we put forward a 'partially Bayesian active learning line sampling' (PBAL-LS) method for estimating small failure probabilities. First, estimation of the failure probability integral in the classical LS is treated as a Bayesian active learning problem, which allows to incorporate our prior knowledge and model the discretization error. The proposed method is called 'partially' because we only derive an upper bound of the posterior variance for the failure probability. In the numerical implementation of PBAL-LS, the important direction is allowed to be updated on the fly without re-evaluating the beta function;

(h) For objective (vii), in addition to the PBAL-LS method, we also come up with a more complete Bayesian active learning treatment of the traditional LS, leading to the 'Bayesian active learning line sampling' (BAL-LS) method. In this method, the exact posterior variance of the failure probability is derived in analytic form. Based on the posterior statistics of the failure probability, a learning function and a stopping criterion are proposed accordingly;

(i) For objective (vii), other than the PBAL-LS and BAL-LS methods, we introduce another Bayesian active alternative, called 'Bayesian active learning line sampling with log-normal process' (BAL-LS-LP), to the traditional LS. Instead of a Gaussian process prior, we assign a LP prior over the beta function, which can explicitly express its non-negativity. Besides, the approximation error introduced by the root-finding procedure is also accounted for.

The above contributions correspond to nine peer-reviewed journal articles, among which the

first six have been published and the remaining three are under review or to be submitted. A complete list of these articles is given as follows:

[a] Dang C., Wei P., and Beer M. An approach to evaluation of EVD and small failure probabilities of uncertain nonlinear structures under stochastic seismic excitations. Mechanical Systems and Signal Processing 152 (2021): 107468. DOI: https://doi.org/10.1016/j.ymssp.2020.107468

[b] Dang C., Wei P., Faes M.G., Valdebenito M.A. and Beer M. Interval uncertainty propagation by a parallel Bayesian global optimization method. Applied Mathematical Modelling 108 (2022): 220-235. DOI: https://doi.org/10.1016/j.apm.2022.03.031

[c] Dang C., Wei P., Faes M.G., and Beer M. Bayesian probabilistic propagation of hybrid uncertainties: Estimation of response expectation function, its variable importance and bounds. Computers & Structures 270 (2022): 106860. DOI: https://doi.org/10.1016/j.compstruc.2022.106860

[d] Dang C., Wei P., Song J., and Michael 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

[e] Dang C., Wei P., Faes M.G., Valdebenito M.A. and Beer M. 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

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

[g] Dang C., Valdebenito M.A., Song J., Wei P., and Beer M. 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

[h] Dang C., Valdebenito M.A., Faes M.G., Song J., Wei P., and Beer M. 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`

[i] Dang C., Valdebenito M.A., Song J., Wei P., and Beer M. Bayesian active learning line sampling with log-normal process for rare event estimation. Reliability Engineering & System Safety (Under Review)

1.6 Structure of the thesis

This thesis is organized into eleven chapters in total. Following this introduction, Chapter 2 - 10 report article [a] - [i], respectively. Chapter 11 concludes main findings of the work reported in this thesis, and provides some future research directions.

Bibliography

[1] William D Rowe. Understanding uncertainty. *Risk Analysis*, 14(5):743–750, 1994.

[2] Ralph C Smith. *Uncertainty quantification: theory, implementation, and applications*, volume 12. SIAM, 2013.

[3] Peter F Pelz, Peter Groche, Marc E Pfetsch, and Maximilian Schaeffner. *Mastering uncertainty in mechanical engineering*. Springer Nature, 2021.

[4] Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural Safety*, 31(2):105–112, 2009.

[5] Timothy John Sullivan. *Introduction to uncertainty quantification*, volume 63. Springer, 2015.

[6] Ryan G McClarren, Penrose McClarren, and Penrose. *Uncertainty quantification and predictive computational science*. Springer, 2018.

[7] Ramon E Moore. *Methods and applications of interval analysis*. SIAM, 1979.

[8] Ramon E Moore, R Baker Kearfott, and Michael J Cloud. *Introduction to interval analysis*. SIAM, 2009.

[9] David Moens, Maarten De Munck, Wim Desmet, and Dirk Vandepitte. Numerical dynamic analysis of uncertain mechanical structures based on interval fields. In *IUTAM symposium on the vibration analysis of structures with uncertainties*, pages 71–83. Springer, 2011.

[10] Chao Jiang, BY Ni, NY Liu, Xu Han, and Jie Liu. Interval process model and non-random vibration analysis. *Journal of Sound and Vibration*, 373:104–131, 2016.

[11] Yakov Ben-Haim and Isaac Elishakoff. *Convex models of uncertainty in applied mechanics*. Elsevier, 2013.

[12] Isaac Elishakoff, Pierre Elisseeff, and Stewart AL Glegg. Nonprobabilistic, convex-theoretic modeling of scatter in material properties. *AIAA Journal*, 32(4):843–849, 1994.

[13] Isaac Elishakoff and Yannis Bekel. Application of lamé's super ellipsoids to model initial imperfections. *Journal of Applied Mechanics*, 80(6), 2013.

[14] Yangjun Luo, Zhan Kang, Zhen Luo, and Alex Li. Continuum topology optimization with non-probabilistic reliability constraints based on multi-ellipsoid convex model. *Structural and Multidisciplinary Optimization*, 39(3):297–310, 2009.

[15] C Jiang, QF Zhang, X Han, J Liu, and DA Hu. Multidimensional parallelepiped model—a new type of non-probabilistic convex model for structural uncertainty analysis. *International Journal for Numerical Methods in Engineering*, 103(1):31–59, 2015.

[16] Zeng Meng, Zhuohui Zhang, and Huanlin Zhou. A novel experimental data-driven exponential convex model for reliability assessment with uncertain-but-bounded parameters. *Applied Mathematical Modelling*, 77:773–787, 2020.

[17] C Jiang, QF Zhang, X Han, and YH Qian. A non-probabilistic structural reliability analysis method based on a multidimensional parallelepiped convex model. *Acta Mechanica*, 225(2):383–395, 2014.

[18] Yangjun Luo, Junjie Zhan, Jian Xing, and Zhan Kang. Non-probabilistic uncertainty quantification and response analysis of structures with a bounded field model. *Computer Methods in Applied Mechanics and Engineering*, 347:663–678, 2019.

[19] Lotfi A Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.

[20] Hans-Jürgen Zimmermann. *Fuzzy set theory—and its applications*. Springer Science & Business Media, 2011.

[21] Bernd Möller and Michael Beer. *Fuzzy randomness: uncertainty in civil engineering and computational mechanics*. Springer Science & Business Media, 2004.

[22] Peter Walley. *Statistical reasoning with imprecise probabilities*. London: Chapman and Hall, 1991.

[23] Thomas Augustin, Frank PA Coolen, Gert De Cooman, and Matthias CM Troffaes. *Introduction to imprecise probabilities*. John Wiley & Sons, 2014.

[24] Scott Ferson, Vladik Kreinovich, Lev Ginzburg, Davis Myers, and Kari Sentz. Constructing probability boxes and dempster-shafer structures. Technical report, Sandia National Laboratories (United States), 2003.

[25] Matthias GR Faes, Marco Daub, Stefano Marelli, Edoardo Patelli, and Michael Beer. Engineering analysis with probability boxes: A review on computational methods. *Structural Safety*, 93:102092, 2021.

[26] Z Xiao and G Yang. Structural response analysis under dependent variables based on probability boxes. *Mathematical Problems in Engineering*, 2015, 2015.

[27] Haibo Liu, Ming Chen, Chong Du, Jiachang Tang, Chunming Fu, and Guilin She. A copula-based uncertainty propagation method for structures with correlated parametric p-boxes. *International Journal of Approximate Reasoning*, 138:89–104, 2021.

[28] Matthias GR Faes, Matteo Broggi, Guan Chen, Kok-Kwang Phoon, and Michael Beer. Distribution-free p-box processes based on translation theory: Definition and simulation. *Probabilistic Engineering Mechanics*, 69:103287, 2022.

[29] Jinwu LI and Jiang Chao. A novel imprecise stochastic process model for time-variant or dynamic uncertainty quantification. *Chinese Journal of Aeronautics*, 35(9):255–267, 2022.

[30] Dempster AP. Upper and lower probabilities induced by a multivalued mapping. *The Annals of Mathematical Statistics*, 38(2):325–339, 1967.

[31] Glenn Shafer. *A mathematical theory of evidence*, volume 42. Princeton university press, 1976.

[32] Kari Sentz and Scott Ferson. Combination of evidence in dempster-shafer theory. Technical report, Sandia National Laboratories (United States), 2002.

[33] Pierre L'Ecuyer and Florian Puchhammer. Density estimation by monte carlo and quasi-monte carlo. In *International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, pages 3–21. Springer, 2022.

[34] Jianbing Chen and Jie Li. Extreme value distribution and reliability of nonlinear stochastic structures. *Earthquake Engineering and Engineering Vibration*, 4(2):275–286, 2005.

[35] G Falsone and D Settineri. Explicit solutions for the response probability density function of linear systems subjected to random static loads. *Probabilistic Engineering Mechanics*, 33:86–94, 2013.

[36] G Falsone and D Settineri. Explicit solutions for the response probability density function of nonlinear transformations of static random inputs. *Probabilistic Engineering Mechanics*, 33:79–85, 2013.

[37] Guohai Chen and Dixiong Yang. Direct probability integral method for stochastic response analysis of static and dynamic structural systems. *Computer Methods in Applied Mechanics and Engineering*, 357:112612, 2019.

[38] Gang Li and Kai Zhang. A combined reliability analysis approach with dimension reduction method and maximum entropy method. *Structural and Multidisciplinary Optimization*, 43(1):121–134, 2011.

[39] Xufang Zhang and Mahesh D Pandey. Structural reliability analysis based on the concepts of entropy, fractional moment and dimensional reduction method. *Structural Safety*, 43:28–40, 2013.

[40] Christian P Robert, George Casella, and George Casella. *Monte Carlo statistical methods*, volume 2. Springer, 1999.

[41] Sergei Abramovich Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. In *Doklady Akademii Nauk*, volume 148, pages 1042–1045. Russian Academy of Sciences, 1963.

[42] Nicolas Victoir. Asymmetric cubature formulae with few points in high dimension for symmetric measures. *SIAM Journal on Numerical Analysis*, 42(1):209–227, 2004.

[43] Carl Edward Rasmussen and Zoubin Ghahramani. Bayesian monte carlo. *Advances in Neural Information Processing Systems*, pages 505–512, 2003.

[44] Pengfei Wei, Xing Zhang, and Michael Beer. Adaptive experiment design for probabilistic integration. *Computer Methods in Applied Mechanics and Engineering*, 365:113035, 2020.

[45] Sharif Rahman and Heqin Xu. A univariate dimension-reduction method for multi-dimensional integration in stochastic mechanics. *Probabilistic Engineering Mechanics*, 19(4):393–408, 2004.

[46] Zio Enrico. *The Monte Carlo Simulation Method for System Reliability and Risk Analysis*. Springer, 2013.

[47] Maurice Lemaire. *Structural reliability*. John Wiley & Sons, 2013.

[48] Rüdiger Rackwitz and Bernd Flessler. Structural reliability under combined random load sequences. *Computers & structures*, 9(5):489–494, 1978.

[49] Zhangli Hu, Rami Mansour, Mårten Olsson, and Xiaoping Du. Second-order reliability methods: a review and comparative study. *Structural and Multidisciplinary Optimization*, 64(6):3233–3263, 2021.

[50] Benjamin Echard, Nicolas Gayton, and Maurice Lemaire. Ak-mcs: an active learning reliability method combining kriging and monte carlo simulation. *Structural Safety*, 33(2):145–154, 2011.

[51] Yan-Gang Zhao and Zhao-Hui Lu. Fourth-moment standardization for structural reliability assessment. *Journal of Structural Engineering*, 133(7):916–924, 2007.

[52] Matthias Faes and David Moens. Recent trends in the modeling and quantification of non-probabilistic uncertainty. *Archives of Computational Methods in Engineering*, 27(3):633–671, 2020.

[53] Yakov Ben-Haim. A non-probabilistic concept of reliability. *Structural safety*, 14(4):227–245, 1994.

[54] Zhan Kang, Yangjun Luo, and Alex Li. On non-probabilistic reliability-based design optimization of structures with uncertain-but-bounded parameters. *Structural Safety*, 33(3):196–205, 2011.

[55] Chao Jiang, RG Bi, GY Lu, and X Han. Structural reliability analysis using non-probabilistic convex model. *Computer Methods in Applied Mechanics and Engineering*, 254:83–98, 2013.

[56] Zeng Meng, Dequan Zhang, Gang Li, and Bo Yu. An importance learning method for non-probabilistic reliability analysis and optimization. *Structural and Multidisciplinary Optimization*, 59(4):1255–1271, 2019.

[57] Z Xiao, X Han, C Jiang, and G Yang. An efficient uncertainty propagation method for parameterized probability boxes. *Acta Mechanica*, 227(3):633–649, 2016.

[58] Roland Schöbi and Bruno Sudret. Uncertainty propagation of p-boxes using sparse polynomial chaos expansions. *Journal of Computational Physics*, 339:307–327, 2017.

[59] HB Liu, C Jiang, XY Jia, XY Long, Zhaoyong Zhang, and FJ Guan. A new uncertainty propagation method for problems with parameterized probability-boxes. *Reliability Engineering & System Safety*, 172:64–73, 2018.

[60] HB Liu, C Jiang, J Liu, and JZ Mao. Uncertainty propagation analysis using sparse grid technique and saddlepoint approximation based on parameterized p-box representation. *Structural and Multidisciplinary Optimization*, 59(1):61–74, 2019.

[61] Ha-Rok Bae, Ramana V Grandhi, and Robert A Canfield. Epistemic uncertainty quantification techniques including evidence theory for large-scale structures. *Computers & Structures*, 82(13-14):1101–1112, 2004.

[62] Michael S Eldred, Laura Painton Swiler, and Gary Tang. Mixed aleatory-epistemic uncertainty quantification with stochastic expansions and optimization-based interval estimation. *Reliability Engineering & System Safety*, 96(9):1092–1113, 2011.

[63] Z Zhang, C Jiang, Xue Han, Dean Hu, and S Yu. A response surface approach for structural reliability analysis using evidence theory. *Advances in Engineering Software*, 69:37–45, 2014.

[64] Hesheng Tang, Dawei Li, Jingjing Li, and Songtao Xue. Epistemic uncertainty quantification in metal fatigue crack growth analysis using evidence theory. *International Journal of Fatigue*, 99:163–174, 2017.

[65] Z Zhang, XX Ruan, MF Duan, and C Jiang. An efficient epistemic uncertainty analysis method using evidence theory. *Computer Methods in Applied Mechanics and Engineering*, 339:443–466, 2018.

[66] Pengfei Wei, Fangqi Hong, Kok-Kwang Phoon, and Michael Beer. Bounds optimization of model response moments: a twin-engine bayesian active learning method. *Computational Mechanics*, 67(5):1273–1292, 2021.

[67] Z Zhang and C Jiang. Evidence-theory-based structural reliability analysis with epistemic uncertainty: a review. *Structural and Multidisciplinary Optimization*, 63(6):2935–2953, 2021.

[68] Kais Zaman, Mark McDonald, and Sankaran Mahadevan. Probabilistic framework for uncertainty propagation with both probabilistic and interval variables. *Journal of Mechanical Design*, 133(2), 2011.

[69] Wei Gao, Chongmin Song, and Francis Tin-Loi. Probabilistic interval analysis for structures with uncertainty. *Structural Safety*, 32(3):191–199, 2010.

[70] Wei Gao, Di Wu, Chongmin Song, Francis Tin-Loi, and Xiaojing Li. Hybrid probabilistic interval analysis of bar structures with uncertainty using a mixed perturbation monte-carlo method. *Finite Elements in Analysis and Design*, 47(7):643–652, 2011.

[71] Xiaoping Du. Interval reliability analysis. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 48078, pages 1103–1109, 2007.

[72] C Jiang, GY Lu, X Han, and LX Liu. A new reliability analysis method for uncertain structures with random and interval variables. *International Journal of Mechanics and Materials in Design*, 8(2):169–182, 2012.

[73] Umberto Alibrandi and CG Koh. First-order reliability method for structural reliability analysis in the presence of random and interval variables. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems Part B: Mechanical Engineering*, 1(4):041006, 2015.

[74] Xufeng Yang, Yongshou Liu, Yi Gao, Yishang Zhang, and Zongzhan Gao. An active learning kriging model for hybrid reliability analysis with both random and interval variables. *Structural and Multidisciplinary Optimization*, 51(5):1003–1016, 2015.

[75] Jinhao Zhang, Mi Xiao, Liang Gao, and Junjian Fu. A novel projection outline based active learning method and its combination with kriging metamodel for hybrid reliability analysis with random and interval variables. *Computer Methods in Applied Mechanics and Engineering*, 341:32–52, 2018.

[76] Barron J Bichon, Michael S Eldred, Laura Painton Swiler, Sandaran Mahadevan, and John M McFarland. Efficient global reliability analysis for nonlinear implicit performance functions. *AIAA Journal*, 46(10):2459–2468, 2008.

[77] Rui Teixeira, Maria Nogal, and Alan O'Connor. Adaptive approaches in metamodel-based reliability analysis: A review. *Structural Safety*, 89:102019, 2021.

[78] Maliki Moustapha, Stefano Marelli, and Bruno Sudret. Active learning for structural reliability: Survey, general framework and benchmark. *Structural Safety*, 96:102174, 2022.

[79] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

[80] Aditya Siddhant and Zachary C Lipton. Deep bayesian active learning for natural language processing: Results of a large-scale empirical study. *arXiv preprint arXiv:1808.05697*, 2018.

[81] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International Conference on Machine Learning*, pages 1183–1192. PMLR, 2017.

[82] Shervin Javdani, Yuxin Chen, Amin Karbasi, Andreas Krause, Drew Bagnell, and Siddhartha Srinivasa. Near optimal bayesian active learning for decision making. In *Artificial Intelligence and Statistics*, pages 430–438. PMLR, 2014.

[83] Philipp Hennig, Michael A Osborne, and Mark Girolami. Probabilistic numerics and uncertainty in computations. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2179):20150142, 2015.

[84] Chris J Oates and Timothy John Sullivan. A modern retrospective on probabilistic numerics. *Statistics and Computing*, 29(6):1335–1351, 2019.

Chapter 2

Dynamic reliability analysis of nonlinear structures

An approach to evaluation of EVD and small failure probabilities of uncertain nonlinear structures under stochastic seismic excitations

Chao Dang[a], Pengfei Wei[a,b,⋆], Michael Beer[a,c,d]

[a] *Institute for Risk and Reliability, Leibniz University Hannover, Callinstr. 34, Hannover 30167, Germany*

[b] *School of Mechanics, Civil Engineering and Architecture, Northwestern Polytechnical University, Xi'an 710072, PR China*

[c] *Institute for Risk and Uncertainty, University of Liverpool, Liverpool L69 7ZF, United Kingdom*

[d] *International Joint Research Center for Engineering Reliability and Stochastic Mechanics, Tongji University, Shanghai 200092, PR China*

Published in Mechanical Systems and Signal Processing in May 2021

Abstract: Efficient assessment of small first-passage failure probabilities of nonlinear structures with uncertain parameters under stochastic seismic excitations is an important but still challenging problem. In principle, the first-passage failure probabilities can be evaluated once the extreme value distribution (EVD) of studied structural response becomes available. With this in mind, this study presents a novel approach, termed as moment-generating function based mixture distribution (MGF-MD), for evaluation of the EVD. In this method, the MGF is firstly introduced to characterize the EVD, and the advantages of this characterization are highlighted. To calculate the MGF defined by a high-dimensional expectation integral, a low-discrepancy sampling technique, named Latinized partially stratified sampling (LPSS), is employed with a small sample size. Besides, the unbiasedness of the estimator is proven and the confidence interval is given. Then, a mixture of two generalized inverse Gaussian distributions (MTGIGD) with a closed-form MGF is proposed to approximate the EVD from the knowledge of its estimated MGF. The parameter estimation is conducted by matching the MGF of MTGIGD with seven values of the estimated one. Three numerical examples, including the EVD of random variables and reliability evaluations of two uncertain nonlinear structures subjected to fully non-stationary stochastic ground motions, are studied. Results indicate that the proposed approach can provide reasonable accuracy and efficiency and is applicable to very high-dimensional systems with small failure probabilities. The source code is readily available at: `https:`

*Corresponding Author
E-mail address: pengfeiwei@nwpu.edu.cn (P. Wei)

//github.com/Chao-Dang/Moment-generating-function-based-mixture-distribution.

Keywords: Extreme value distribution, Small first-passage probability, Moment-generating function, Mixture distribution, Generalized inverse Gaussian distribution, Nonlinear structure, Stochastic seismic excitation

2.1 Introduction

Efficient assessment of reliability of engineering structures in the presence of various uncertainties is an important task not only in the entire design process, but also in the whole service life. In order to ensure the seismic safety, the dynamic reliability analysis of structures under stochastic seismic excitations is usually formulated as the well-known first-passage failure problem [1, 2]. That is, the failure occurs once the studied stochastic structural response exceeds the prescribed threshold during a given time interval. Many efforts have been devoted to evaluating the first-passage probability of stochastic dynamic systems, such as the out-crossing rate based methods [3, 4], stochastic averaging [5, 6], path integration [7, 8], etc. The first-passage problem nevertheless still remain one of the most difficult issues in stochastic dynamics due to the existence of several challenges. First of all, uncertainties arising from both the earthquake loads and structural properties should be reasonably addressed simultaneously, which commonly results in high dimensions of the reliability problem to be solved. The performance functions of structures may also exhibit highly nonlinear behavior especially when subjected to strong earthquakes. Besides, as the structural systems become increasingly complex, the existence of multiple underlying failure modes with dependencies further increases the complexity of the problem. Apart from those challenges, the failure probabilities are typically expected to be very small, e.g., the order of magnitude 10^{-4} or smaller.

The extreme value distribution (EVD) of stochastic structural response has attracted considerable attention since the first-passage failure probabilities under different prescribed thresholds can be obtained equivalently through the associated EVD. Further, with the EVD, the time-dependent dynamic reliability problem can be conveniently converted into a time-invariant counterpart. In this setting, some well-known reliability analysis methods could be directly invoked to address the above-mentioned challenges, e.g., the Monte Carlo simulation (MCS) and Subset Simulation (SS) [9, 10] owing to their robustness to the dimension and complexity of the problem. However, the MCS is largely restricted by its computational cost, especially for a expensive-to-evaluate model with a low

level of failure probability. The SS offers noticeably improved efficiency compared to the MCS, but is still far away from desirable for real-world applications. The present study particularly focuses on the evaluation of the entire EVD with efficiency and accuracy. Usually, two kinds of approaches have been developed for deriving the EVD of a stochastic structural response, i.e., the analytical approach and approximate approach. The analytical approaches are usually only applicable for some specific types of stochastic processes [11–14], thus are infeasible for general engineering problems with the above-mentioned features/challenges. Alternatively, two major types of approximate approaches are available to capture the EVD: parametric approach and non-parametric approach. The basic idea of parametric approach is to assume that the extreme value samples coming from a population can be adequately modeled by a probability distribution with a fixed set of parameters. Based on the extreme value theory, the Gumbel distribution is used to model the main body and tail behavior of the EVD [15, 16], followed by the generalized extreme value distribution (GEVD) and generalized Pareto distribution (GPD) [17]. Among them, the GEVD could be more flexible, but still restricted by its limitations. Besides, some four-parameter distributions are also developed to recover the EVD from the knowledge of its first-four moments, e.g., the fourth-moment normal transformation (FMNT) [18] and shifted generalized lognormal distribution (SGLD) [19]. Although these two models are generally versatile, the requirement of estimating the response moments with high accuracy and efficiency is still a challenging task. With the emergence of fractional moments, the maximum entropy method [20–22], kernel density maximum entropy method [23] and mixture distribution [24] are proposed. Despite these, other non-standard parametric methods can also be found in the literature [25–27]. For the non-parametric approach, assumptions on the distribution type of EVD are not required. The probability density evolution method (PDEM) can be employed to evaluate the EVD by constructing a virtual stochastic process associated to the extreme value of the studied stochastic process [28, 29]. Recently, the direct probability integral method (DPIM) [30] is proposed for stochastic response analysis of static and dynamic structural systems based on the principle of probability conservation, which could also be used to capture the EVD of stochastic response process. The PDEM and DPIM are generally applicable to strongly nonlinear systems, which, however, are not easy to extend for problems with high-dimensional random inputs and small failure probabilities. In summary, the success of the parametric approach relies on assumptions of the distributional type and methods for parameter estimation. In other words, a good approximation can yield desirable results with less computational efforts. On the contrary, the non-parametric

approach is distribution-free, but still suffers from several drawbacks.

The main objective of the present study is to develop an efficient approach for assessment of the EVD and small first-passage failure probabilities of nonlinear structures with uncertain parameters under stochastic seismic excitations. For this purpose, the moment-generating function (MGF) is firstly introduced to characterize the EVD, instead of the commonly-used integer moments and fractional moments. To calculate the MGF that is defined by a high-dimensional expectation integral, the latinized partially stratified sampling (LPSS) is employed with a reduced number of samples due to its simultaneous variance reduction associated with both main effects and variable interactions [31]. Then, a flexible mixture of two generalized inverse Gaussian distributions (MTGIGD) is proposed to recover the EVD from the knowledge of its estimated MGF, where a MGF-matching technique is developed for parameter estimation. Once the EVD is reconstructed in the entire distribution domain with reasonable accuracy, the first-passage failure probabilities under different thresholds can be obtained without difficulty.

The rest of this paper is arranged as follows. Section 2.2 outlines the EVD-based method for first-passage failure probabilities evaluation of stochastic dynamic systems. The MGF is introduced in Section 2.3 to characterize the EVD, the advantages and numerical approximation of which are also elaborated. In Section 2.4, a mixture distribution is developed to recovered the EVD from its estimated MGF. Three numerical examples are investigated to validate the proposed approach in Section 2.5. Some concluding remarks are available in Section 2.6.

2.2 First-passage probability: an EVD perspective

Without loss of generality, consider a generic stochastic dynamic structural system governed by the following state equation and initial condition:

$$\dot{\boldsymbol{X}} = \boldsymbol{A}(\boldsymbol{X}, \boldsymbol{\Theta}, t),\ \boldsymbol{X}(t)|_{t=0} = \boldsymbol{x}_0 \tag{2.1}$$

where $\boldsymbol{X} = (X_1, X_2, \ldots, X_d)^{\mathrm{T}}$ is a d-dimensional state vector; $\boldsymbol{A} = (A_1, A_1, \ldots, A_d)^{\mathrm{T}}$ is the state mapping; $\boldsymbol{\Theta} = (\Theta_1, \Theta_2, \ldots, \Theta_n)^{\mathrm{T}}$ denotes a n-dimensional random vector with known joint PDF $p_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$, which might arise from both the system properties and external excitations. In general, the system properties might originally occur as stochastic fields or directly as some random parameters.

Besides, it is widely recognized that the random seismic excitations should be modelled as non-stationary stochastic processes or fields [32]. The stochastic fields or processes can usually be further decomposed to a set of standard random variables, e.g., by K-L expansion [33] and spectral representation method [34]. Therefore, a large number of random variables could be involved in a dynamic structural system with uncertain parameters subjected to non-stationary stochastic seismic excitations, leading to a high-dimensional system to be handled.

For a well-posed dynamical system, it is reasonable to assume that the solution to Eq. (2.1) exists uniquely as a function of $\boldsymbol{\Theta}$, which is expressed as:

$$\boldsymbol{X}(t) = \boldsymbol{H}(\boldsymbol{\Theta}, t), \ \dot{\boldsymbol{X}}(t) = \boldsymbol{h}(\boldsymbol{\Theta}, t) \tag{2.2}$$

where $\boldsymbol{H}$ and $\boldsymbol{h} = \partial \boldsymbol{H}/\partial t$ are the deterministic operators.

If there are a set of system response quantities of interest $\boldsymbol{Q}(t) = (Q_1(t), Q_2(t), \ldots, Q_m(t))^{\mathrm{T}}$ for reliability analysis, such as the inter-story drifts and strains or stresses at some points, then $\boldsymbol{Q}$ can usually be calculated from its connection with the state vectors. It is convenient to assume that the relationship takes the form:

$$\boldsymbol{Q}(t) = \boldsymbol{\Psi}\left[\boldsymbol{X}(t), \dot{\boldsymbol{X}}(t)\right] = \boldsymbol{W}(\boldsymbol{\Theta}, t) \tag{2.3}$$

where $\boldsymbol{\Psi}$ is a transfer operator; $\boldsymbol{W}$ denotes a mapping from $\boldsymbol{\Theta}$ to $\boldsymbol{Q}$. Alternatively, Eq. (2.3) can be expressed as its component form:

$$Q_l(t) = W_l(\boldsymbol{\Theta}, t), \ l = 1, 2, \ldots, m \tag{2.4}$$

For notational simplicity, the subscript l will be omitted hereafter without inducing confusions.

In general, the first-passage reliability is defined as:

$$P_r = \Pr\{Q(t) \in \Omega_s, t \in [0, T]\} \tag{2.5}$$

where $\Pr\{\cdot\}$ is the probability operator; Ω_s is the safe domain; T is the duration of time. For the case of symmetric double-boundary safe domain concerned in this study, the first-passage reliability

can be further written as:

$$P_r = \Pr\{|Q(t)| < b, t \in [0, T]\} \tag{2.6}$$

where b is the prescribed threshold. Since $Q(t)$ can be a general stochastic response process, evaluation of P_r remains a non-trivial task, especially for a expensive-to-evaluate model.

Let Z denote the maximum value of $|Q(t)|$, i.e.

$$Z = \max_{t\in[0,T]} |Q(t)| = \max_{t\in[0,T]} |W(\boldsymbol{\Theta}, t)| = G(\boldsymbol{\Theta}) \tag{2.7}$$

Note that Z is conveniently assumed to be as a function of $\boldsymbol{\Theta}$ and should be a positive random variable in this form. Then, Eq. (2.6) is equivalent to:

$$P_r = \Pr\{Z = G(\boldsymbol{\Theta}) < b\} \tag{2.8}$$

That is to say, once the probability distribution of Z is available, which is refer to as the extreme value distribution (EVD), P_r can be straightforwardly and conveniently obtained. Denote the probability density function (PDF) and cumulative distribution function (CDF) of Z as $f_Z(z)$ and $F_Z(z)$, respectively, and then the first-passage reliability is:

$$P_r = \int_{-\infty}^{b} f_Z(z)\mathrm{d}z = F_Z(b) \tag{2.9}$$

and the corresponding failure probability is

$$P_f = \int_{b}^{+\infty} f_Z(z)\mathrm{d}z = 1 - F_Z(b) \tag{2.10}$$

The problem of first-passage reliability is now equivalently transformed to evaluation of the EVD. Despite of the conceptual elegance, it is still not easy to directly derive the analytical PDF or CDF of the EVD of a general stochastic response process. To address this challenge, an efficient approach will be developed in the following sections to capture the EVD.

2.3 Moment-generating function of the EVD

To characterize the EVD, the statistical moments, such as the mean value and variance, are commonly employed in the literature. However, it is known that the probabilistic information from the low-order moments is not sufficient to accurately capture an unknown distribution. Additionally, the high-order moments, e.g., the skewness and kurstosis, containing more information regarding the tail behavior are considered for a more accurate characterization [18]. Nevertheless, high-order moments are invariably complicated to analyze and sampling variability of power moments increases with the order of moment. Recently, the fractional moments with low-order exponents are introduced primarily due to the fact that even a single low-order fractional moment can contain a large amount of information on numerous integer moments [24]. Even so, it is worth mentioning that either a finite sequence of integer moments or fractional moments cannot uniquely determine a underlying probability distribution. Based on those considerations, the moment-generating function will be introduced in this work to characterize the EVD.

2.3.1 Definition of the moment-generating function

For the positive random variable Z defined in Eq. (2.7), its moment-generating function (MGF) can by expressed as:

$$M_Z(\tau) = \mathbb{E}\left[e^{\tau Z}\right] = \int_0^{+\infty} e^{\tau z} f_Z(z)\mathrm{d}z, \tau \in \mathbb{R} \tag{2.11}$$

where $\mathbb{E}[\cdot]$ denotes the expectation operator.

Consider the series expansion of $e^{\tau Z}$

$$e^{\tau Z} = 1 + \tau Z + \frac{\tau^2 Z^2}{2!} + \frac{\tau^3 Z^3}{3!} + \cdots + \frac{\tau^r Z^r}{r!} + \cdots \tag{2.12}$$

and hence

$$M_Z(\tau) = 1 + \tau\mathbb{E}[Z] + \frac{\tau^2\mathbb{E}[Z^2]}{2!} + \frac{\tau^3\mathbb{E}[Z^3]}{3!} + \cdots + \frac{\tau^r\mathbb{E}[Z^r]}{r!} + \cdots \tag{2.13}$$

where $\mathbb{E}[Z^r]$ is the r-th moment of Z. Eq. (2.13) implies that the MGF at any non-zero τ includes a bulk of probabilistic information on infinite number of integer moments. This feature is similar to that of fractional moments, while fractional moments only make sense with positive random variables.

Further, the r-th moment of Z can be obtained from its MGF such that:

$$\mathbb{E}\left[Z^r\right] = M_Z^{(r)}(0) = \left.\frac{\mathrm{d}^r M_Z}{\mathrm{d}\tau^r}\right|_{\tau=0} \tag{2.14}$$

Eq. (2.14) implies that only the MGF near the origin already contains all the information for any order of integer moment. The MGF is so named because all moments of Z can be computed from its derivative evaluated at $\tau = 0$. However, it is impossible to obtain any order integer moment from a single fractional moment (not the integer moment itself), even though it includes the information conceptually.

Besides, the MGF (if it exists) can uniquely determine the underlying distribution (see Section 30 of [35]). In other words, if two random variables have the same MGF, then they must follow the same distribution. Thus, the MGF characterization of the EVD is more suitable than traditional integer moments or fractional moments. For more properties concerning the MGF, please refer to, e.g., [36].

2.3.2 Numerical approximation of the moment-generating function

For the EVD of a general stochastic response process, derivation of closed-form solution of its MGF is not trivial. Therefore, numerical evaluation is necessary for general applications. For our purpose, the MGF defined in Eq. (2.11) can be further expressed as:

$$M_Z(\tau) = \int_{\mathcal{D}_{\boldsymbol{\Theta}}} e^{\tau G(\boldsymbol{\theta})} p_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} \tag{2.15}$$

where $\mathcal{D}_{\boldsymbol{\Theta}}$ is the distribution domain of $\boldsymbol{\Theta}$. As the dimension of $\boldsymbol{\Theta}$ could be very large, $M_Z(\tau)$ is generally a high-dimensional integral with a complicated integrand.

To evaluate such a integral, a recently developed low-discrepancy sampling technique, named Latinized partially stratified sampling (LPSS) [31], will be employed. Eq. (2.15) can be estimated by the following estimator:

$$\hat{M}_Z(\tau) = \frac{1}{N} \sum_{i=1}^{N} e^{\tau G(\boldsymbol{\theta}^{(i)})} \tag{2.16}$$

where N is the total number of samples; $\{\boldsymbol{\theta}^{(i)}\}_{i=1}^{N}$ are a set of samples of $\boldsymbol{\Theta}$. Since LPSS method is originally designed for generating uniformly distributed samples over the unit hypercube, i.e,

$\mathcal{D}_{\boldsymbol{U}} = [0,1]^d$, a transformation $\boldsymbol{\theta}^{(i)} = T(\boldsymbol{u}^{(i)})$ should be applied to obtain the sample $\boldsymbol{\theta}^{(i)}$ from $\boldsymbol{u}^{(i)}$ ($\{\boldsymbol{u}^{(i)}\}_{i=1}^N$ are a set of samples generated by LPSS). A novel feature of the LPSS is that it can reduce the variance related to the main effects and variable interactions simultaneously, and hence a reduced set of samples could be sufficient to accurately evaluate the MGF. Readers may refer to [31] for technical details and [24, 37] for implementation issues of the LPSS method. The basic algorithm is presented in Algorithm 1. The Matlab code is readily available at: `https://www.mathworks.com/matlabcentral/fileexchange/54847-latinized-partially-stratified-sampling`.

Algorithm 1 Latinized partially stratified sampling

1: **Input:** Specify the sample size N, number of dimensions d and orthogonal subspaces $\mathcal{S}_j (j = 1, 2, \cdots, L)$, where $\mathcal{S}_{j_1} \perp \mathcal{S}_{j_2} (j_1 \neq j_2)$ and $\mathcal{S}_1 \oplus \mathcal{S}_2 \oplus \cdots \oplus \mathcal{S}_L = \mathcal{D}_{\boldsymbol{U}}$.
2: For each subspace $\mathcal{S}_j$, generate N low-dimensional samples using Latinized stratified sampling (LSS) [31].
3: Randomly select a sample from each subspace (without replacement) and group the selected samples to produce a sample in d-dimensional space.
4: Repeat step 3 until N samples are constructed in d-dimensional space.
5: **Output:** N samples in d-dimensional unit hypercube.

Regardless of the distribution of Z, we have

$$\mathbb{E}\left[\hat{M}_Z(\tau)\right] = \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} e^{\tau G(\boldsymbol{\theta}^{(i)})}\right] = \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}\left[e^{\tau G(\boldsymbol{\theta}^{(i)})}\right] = \frac{1}{N}\sum_{i=1}^{N} M_Z(\tau) = M_Z(\tau) \tag{2.17}$$

Thus, the MGF estimator in Eq. (2.16) is proven to be unbiased. Further, the estimator will approximately follow the normal distribution with mean $\hat{\mu}(\tau) = \hat{M}_Z(\tau)$ and variance given by:

$$\hat{\sigma}^2(\tau) = \frac{1}{N(N-1)}\sum_{i=1}^{N}\left[e^{\tau G(\boldsymbol{\theta}^{(i)})} - \hat{M}_Z(\tau)\right]^2 \tag{2.18}$$

Confidence interval can then be obtained accordingly. For example, the 95% confidence interval of the estimator is as follows:

$$[\hat{\mu}(\tau) - 1.96\hat{\sigma}(\tau), \hat{\mu}(\tau) + 1.96\hat{\sigma}(\tau)] \tag{2.19}$$

Eq. (2.16) can be used to obtain a point estimate of the MGF. In order to inform the accuracy of the point estimate, Eq. (2.19) gives a interval estimate of the MGF, which actually reflects the sampling variability and can be narrowed by increasing the sample size N.

2.4 Recovering the EVD from its moment-generating function by a mixture distribution

In this section, the problem of how to find the underlying EVD from the knowledge of its estimated MGF is studied. Although the MGF can uniquely determine a probability distribution, it is not easy to deduce the underlying distribution directly. Alternatively, we focus on offering an efficient and accurate parametric approach to approximate the EVD. For this purpose, a flexible mixture distribution will be developed on the basis of the generalized inverse Gaussian distribution, and then the parameter estimation based on the MGF is elaborated.

2.4.1 Generalized inverse Gaussian distribution

The generalized inverse Gaussian distribution (GIGD) is a three-parameter family of continuous probability distributions, which has a wide variety of applications in many fields. A random variable has a GIGD if its PDF is given by [38]:

$$f_{\mathrm{GIGD}}(z;\alpha,\beta,\lambda)=\frac{(\alpha/\beta)^{\lambda/2}}{2K_{\lambda}(\sqrt{\alpha\beta})}z^{(\lambda-1)}e^{-(\alpha z+\beta/z)/2},z>0 \tag{2.20}$$

where $\alpha>0$, $\beta>0$ and $\lambda\in\mathbb{R}$ are the three parameters; K_{λ} is a modified Bessel function of the second kind with order λ.

The inverse Gaussian distribution (IGD) is a special case of GIGD when $\lambda=-1/2$. The special and limiting cases of the IGD also belong to the GIGD, such as the normal distribution and Lévy distribution. Besides, other special cases include the hyperbolic distribution for $\lambda=0$, Gamma distribution for $\beta\to 0$ and inverse-gamma distribution for $\alpha\to 0$.

The closed-form solution of the MGF of GIGD exists and reads:

$$M_{\mathrm{GIGD}}(\tau;\alpha,\beta,\lambda)=\left(\frac{\alpha}{\alpha-2\tau}\right)^{\frac{\lambda}{2}}\frac{K_{\lambda}(\sqrt{\beta(\alpha-2\tau)})}{K_{\lambda}(\sqrt{\alpha\beta})} \tag{2.21}$$

Based on the MGF, the mean and variance can be obtained as:

$$\mu_{\mathrm{GIGD}}=\frac{\sqrt{\beta}K_{\lambda+1}(\sqrt{\alpha\beta})}{\sqrt{\alpha}K_{\lambda}(\sqrt{\alpha\beta})} \tag{2.22}$$

$$\sigma^2_{\mathrm{GIGD}} = \left(\frac{\beta}{\alpha}\right)\left[\frac{K_{\lambda+2}(\sqrt{\alpha\beta})}{K_{\lambda}(\sqrt{\alpha\beta})} - \left(\frac{K_{\lambda+1}(\sqrt{\alpha\beta})}{K_{\lambda}(\sqrt{\alpha\beta})}\right)^2\right] \tag{2.23}$$

The reason why we specially select the GIGD as the basis of the proposed parametric distribution model is mostly based on the two following considerations. First of all, the GIGD is flexible in shapes and tail properties since several theoretical distributions with distinct features are included (as mentioned early). It is worthy mentioning that the IGD as a special case of the GIGD is the exact first-passage time distribution of the Brownian motion (Wiener process) [39]. Second, the MGF of the GIGD is readily available in a closed form (Eq. (2.21)), which enables the deviation of the MGF and parameter estimation of the proposed mixture distribution model (see the following two subsections).

2.4.2 A mixture of two generalized inverse Gaussian distributions

Although the GIGD is more versatile than the IGD by adding an extra free parameter, its flexibility should be further enhanced to capture the EVD since more emphasis is placed on the tail, in addition to the main body of the distribution, especially for evaluation of small first-passage failure probabilities. Finite mixture models allow us to create new distributions with increased number of free parameters and hence flexibility in a simple, but efficient way [40]. However, identification of free parameters is an intractable task as the number of mixture components increases, since they usually cannot be derived explicitly. To this end, a mixture of two generalized inverse Gaussian distributions (MTGIGD) will be specially developed to capture the EVD in the present work. The PDF of the MTGIGD is given by:

$$\begin{aligned} f_{\mathrm{MTGIGD}}(z;\Sigma) =& \varpi f_{\mathrm{GIGD}}(z;\alpha^{(1)},\beta^{(1)},\lambda^{(1)}) + (1-\varpi) f_{\mathrm{GIGD}}(z;\alpha^{(2)},\beta^{(2)},\lambda^{(2)}) \\ =& \varpi \frac{(\alpha^{(1)}/\beta^{(1)})^{\lambda^{(1)}/2}}{2K_{\lambda^{(1)}}(\sqrt{\alpha^{(1)}\beta^{(1)}})} z^{(\lambda^{(1)}-1)} e^{-(\alpha^{(1)}x+\beta^{(1)}/z)/2} \\ &+ (1-\varpi)\frac{(\alpha^{(2)}/\beta^{(2)})^{\lambda^{(2)}/2}}{2K_{\lambda^{(2)}}(\sqrt{\alpha^{(2)}\beta^{(2)}})} z^{(\lambda^{(2)}-1)} e^{-(\alpha^{(2)}x+\beta^{(2)}/z)/2} \end{aligned} \tag{2.24}$$

where $0 < \varpi < 1$ and $1-\varpi$ are the mixture weights of the first and second mixture components, respectively; $\Sigma = [\varpi, \alpha^{(1)}, \beta^{(1)}, \lambda^{(1)}, \alpha^{(2)}, \beta^{(2)}, \lambda^{(2)}]^{\mathrm{T}}$ is a set of seven free parameters to be determined; The superscript distinguishes those parameters from different mixture components.

The MGF of the MTGIGD can be derived as:

$$
\begin{aligned}
M_{\mathrm{MTGIGD}}(\tau;\Sigma) =& \int_0^{+\infty} e^{\tau z} f_{\mathrm{MTGIGD}}(z;\Sigma)\mathrm{d}z \\
=& \int_0^{+\infty} e^{\tau z}\left(\varpi f_{\mathrm{GIGD}}(z;\alpha^{(1)},\beta^{(1)},\lambda^{(1)}) + (1-\varpi) f_{\mathrm{GIGD}}(z;\alpha^{(2)},\beta^{(2)},\lambda^{(2)})\right)\mathrm{d}z \\
=& \varpi \int_0^{+\infty} e^{\tau z} f_{\mathrm{GIGD}}(z;\alpha^{(1)},\beta^{(1)},\lambda^{(1)})\mathrm{d}z + (1-\varpi)\int_0^{+\infty} e^{\tau z} f_{\mathrm{GIGD}}(z;\alpha^{(2)},\beta^{(2)},\lambda^{(2)})\mathrm{d}z \\
=& \varpi M_{\mathrm{GIGD}}(\tau;\alpha^{(1)},\beta^{(1)},\lambda^{(1)}) + (1-\varpi) M_{\mathrm{GIGD}}(\tau;\alpha^{(2)},\beta^{(2)},\lambda^{(2)}) \\
=& \varpi \left(\frac{\alpha^{(1)}}{\alpha^{(1)}-2\tau}\right)^{\frac{\lambda^{(1)}}{2}} \frac{K_{\lambda^{(1)}}(\sqrt{\beta^{(1)}(\alpha^{(1)}-2\tau)})}{K_{\lambda^{(1)}}(\sqrt{\alpha^{(1)}\beta^{(1)}})} \\
&+ (1-\varpi)\left(\frac{\alpha^{(2)}}{\alpha^{(2)}-2\tau}\right)^{\frac{\lambda^{(2)}}{2}} \frac{K_{\lambda^{(2)}}(\sqrt{\beta^{(2)}(\alpha^{(2)}-2\tau)})}{K_{\lambda^{(2)}}(\sqrt{\alpha^{(2)}\beta^{(2)}})}
\end{aligned}
\tag{2.25}
$$

If we assume that the EVD follows the MTGIGD, i.e., $\hat{f}_Z(z) = f_{\mathrm{MTGIGD}}(z;\Sigma)$, then how to estimate the unknown parameters collecting in Σ could be another critical task.

2.4.3 Moment-generating function based parameter estimation

The MGF of the EVD is obtained in Section 2.3 and the closed-form expression of MGF of the MTGIGD exists. A natural idea for estimating parameters is to match the EVD's MGF with that of the MTGIGD, and the following equations can be established accordingly:

$$
\begin{cases}
M_{\mathrm{MTGIGD}}(\tau_1;\Sigma) = \hat{M}_Z(\tau_1) \\
M_{\mathrm{MTGIGD}}(\tau_2;\Sigma) = \hat{M}_Z(\tau_2) \\
\qquad\qquad \cdots \\
M_{\mathrm{MTGIGD}}(\tau_7;\Sigma) = \hat{M}_Z(\tau_7)
\end{cases}
\tag{2.26}
$$

where $\{\tau_j\}_{j=1}^7$ are seven discrete values of τ. In fact, τ around zero is more concerned due to its high estimation accuracy and sufficient probabilistic information included. However, $\hat{M}_Z(\tau)$ could be sufficient large for $\tau > 0$, since Z is a positive random variable in the context of this article. For this reason, the discrete values of τ are adopted as $\{\tau_j\}_{j=1}^7 = \{-0.1, -0.2, -0.3, -0.4, -0.5, -0.6, -0.7\}$ in this study. One can use the built-in function "fsolve" in Matlab to solve the above system of equations.

As a suggestion, the following estimates can be served as the initial values of the parameters as a guess:

$$\varpi = 0.4 \tag{2.27}$$

$$\alpha^{(1)} = \alpha^{(2)} = \frac{\hat{\mu}_Z}{\hat{\sigma}_Z^2} \tag{2.28}$$

$$\beta^{(1)} = \beta^{(2)} = \frac{\hat{\mu}_Z^3}{\hat{\sigma}_Z^2} \tag{2.29}$$

$$\lambda^{(1)} = \lambda^{(2)} = -\frac{1}{2} \tag{2.30}$$

where the parameters except for ω are estimated by the method of moments in the case that each mixture component is reduced to the IGD; $\hat{\mu}_Z$ and $\hat{\sigma}_Z^2$ are the estimated mean and variance of the EVD, which can be evaluated by:

$$\hat{\mu}_Z = \frac{1}{N}\sum_{i=1}^{N} G(\boldsymbol{\theta}^{(i)}) \tag{2.31}$$

$$\hat{\sigma}_Z^2 = \frac{1}{N-1}\sum_{i=1}^{N}\left(G(\boldsymbol{\theta}^{(i)}) - \hat{\mu}_Z\right)^2 \tag{2.32}$$

Note that no extra dynamic structural analysis is actually required in this step.

Once the free parameters in the MTGIGD model are identified properly, one can obtain a point estimate of the EVD, and hence the first-passage failure probabilities under different prescribed thresholds. However, how to measure the accuracy of the results is a non-trivial task since both the sampling error containing in the MGF estimate and the model assumption error for functional form of the EVD should be addressed simultaneously. This is not attempted in this study, but we hope that our work will stimulate further studies.

The flowchart of the proposed approach for efficient assessment of EVD and first-passage failure probabilities of structures with uncertain parameters under stochastic seismic excitations is depicted in Fig. 2.1.

2.5 Numerical examples

To validate the proposed approach, three numerical examples will be presented in this section. In the first example, the EVD of random variables is investigated where the analytical solution of the

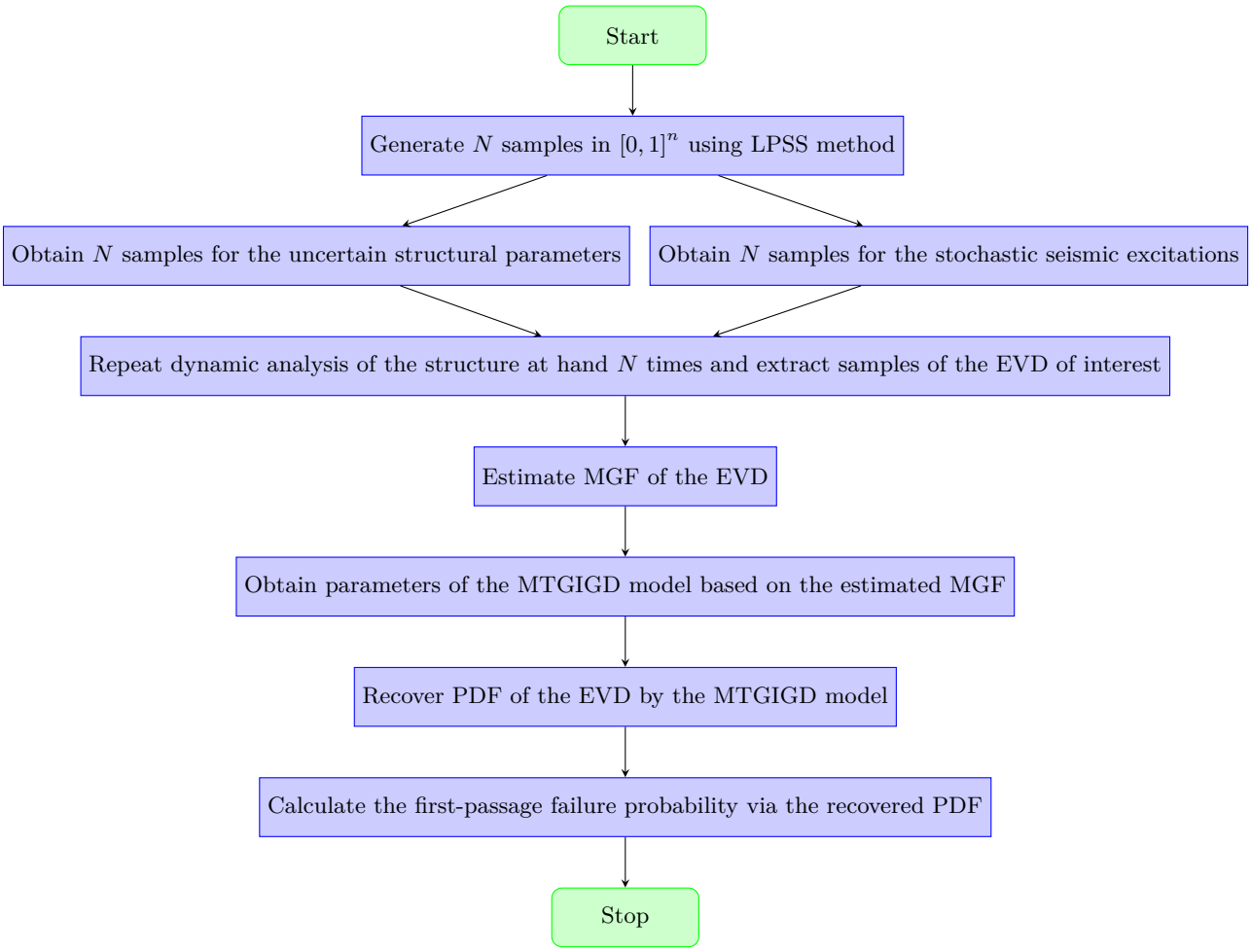


Figure 2.1: Flowchart of the proposed approach.

EVD is available. The EVD approximated by the proposed method is verified through comparison with the analytical solution, indicating the accuracy of the proposed method. The EVD of the response of two nonlinear structures with uncertain parameters subjected to fully non-stationary stochastic seismic excitations is computed through the proposed approach, and the first-passage failure probabilities under different prescribed thresholds are obtained in example 2 and 3. In comparison with Monte Carlo simulation and other state-of-the-art methods, advantages of the proposed method are demonstrated.

2.5.1 Example 1: EVD of random variables

Suppose that $\boldsymbol{X} = [X_1, X_2, \ldots, X_n]$ are i.i.d. standard normal random variables. Consider the maximum absolute value such that:

$$Z = \max(|X_1|, |X_2|, \ldots, |X_n|) \tag{2.33}$$

where $Y_i = |X_i|$ actually follows a folded standard normal distribution. Denote the PDF and CDF of Y_i as $f_Y(y)$ and $F_Y(y)$, respectively. According to the probability theory, the closed-form solutions of the PDF and CDF of Z are available, and given by:

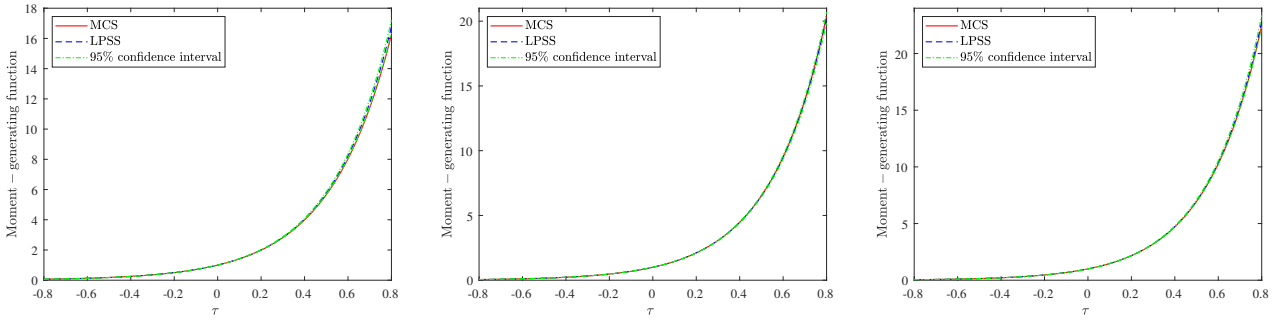
$$f_Z(z) = n f_Y(z) F_Y^{n-1}(z), F_Z(z) = F_Y^n(z) \ (z \geq 0) \tag{2.34}$$

The proposed approach can also be employed to approximate the PDF and CDF of Z. Three cases, i.e., $n = 1000,\ 3000,\ 5000$, will be studied to verify the proposed method. First, the MGFs of Z are estimated by the LPSS method with a small sample size of $N = 625$. The estimated MGFs are compared with those obtained by MCS (with 10^6 runs), as shown in Fig. 2.2. It is observed that the confidence intervals are all very narrow, and contain the results given by MCS, indicating the accuracy of LPSS method. The MTGIGD is then used to recover the PDF and CDF of Z by fitting seven values of the estimated MGF. The recovered PDF and CDF of those three cases are compared with the analytical solutions in Figs. 2.3 - 2.5, respectively. Note that the two mixture components of the proposed MIGIND multiplied by their corresponding mixture weights are also given. As seen, the recovered PDF and CDF agree well with those analytical solutions. This example demonstrates the accuracy of the proposed method, as well as the efficiency for such three high-dimensional cases.

2.5.2 Example 2: EVD and first-passage failure probabilities of a nonlinear shear frame structure

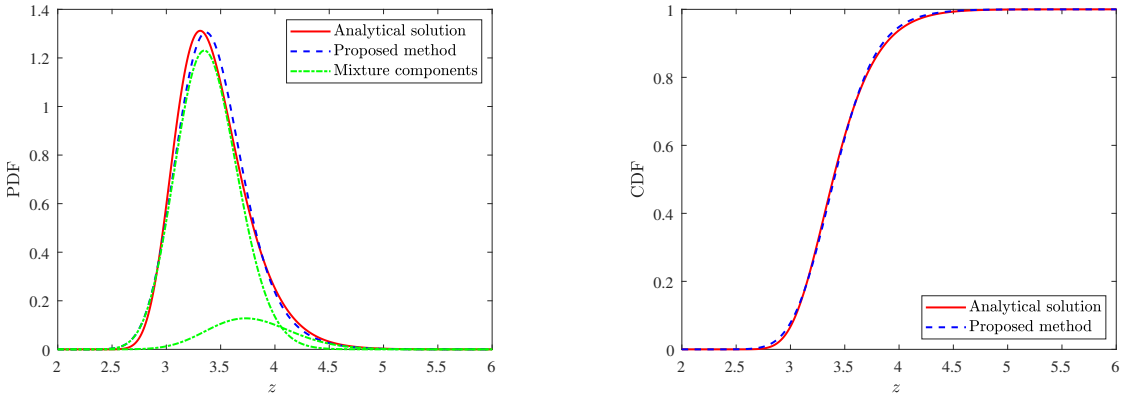
Consider a two-span ten-story nonlinear shear frame structure with uncertain parameters under stochastic seismic excitations, as shown in Fig. 2.6. The equation of motion of this structure is given by:

$$\boldsymbol{M}(\boldsymbol{\Theta})\ddot{\boldsymbol{X}}(t) + \boldsymbol{C}(\boldsymbol{\Theta})\dot{\boldsymbol{X}}(t) + \boldsymbol{K}(\boldsymbol{\Theta})\left[\varepsilon \boldsymbol{X}(t) + (1-\varepsilon)\boldsymbol{Y}(t)\right] = -\boldsymbol{M}(\boldsymbol{\Theta}) I \ddot{U}_g(\boldsymbol{\Theta}, t) \tag{2.35}$$



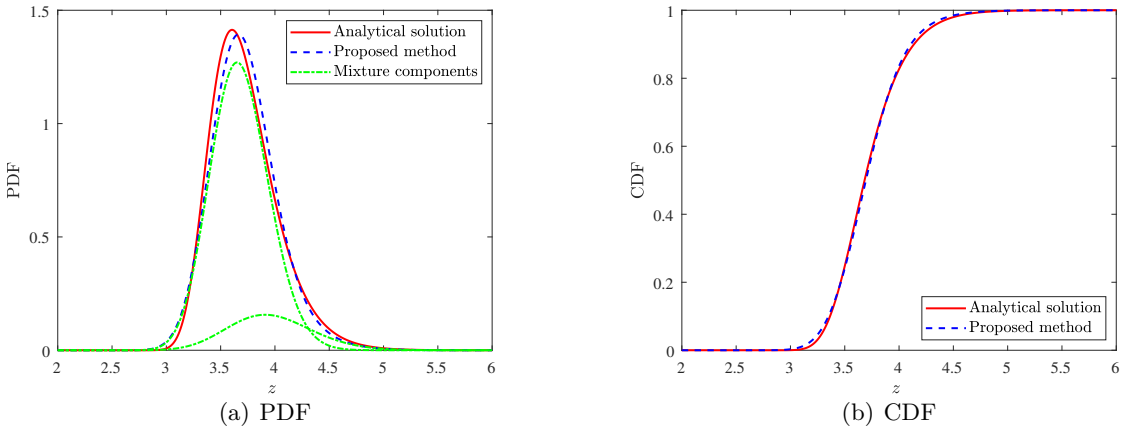
(a) $n = 1000$ (b) $n = 3000$ (c) $n = 5000$

Figure 2.2: Moment-generating function of the EVD in Example 1.



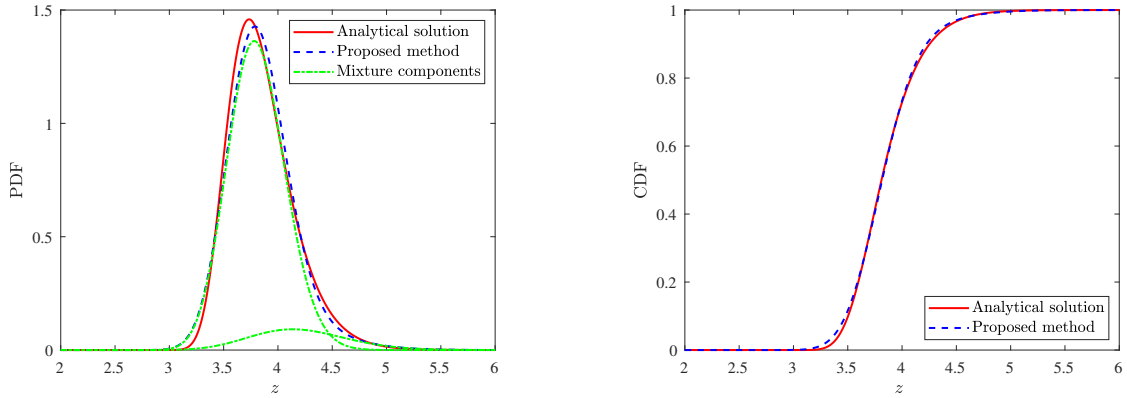
(a) PDF (b) CDF

Figure 2.3: Comparison of PDF and CDF of the EVD in Example 1 ($n = 1000$).



(a) PDF (b) CDF

Figure 2.4: Comparison of PDF and CDF of the EVD in Example 1 ($n = 3000$).



(a) PDF (b) CDF

Figure 2.5: Comparison of PDF and CDF of the EVD in Example 1 ($n = 5000$).

where $\boldsymbol{M}$, $\boldsymbol{C}$ and $\boldsymbol{K}$ are the mass, damping and initial stiffness matrices. Rayleigh damping is adopted such that $\boldsymbol{C} = a_1\boldsymbol{M} + a_2\boldsymbol{K}$, where a_1 and a_2 are obtained by assuming the damping ratio $\xi = 5\%$ for the first two modes. The lumped mass and inter-story stiffness of each story are assumed to be independent random variables, the probabilistic information of which is summarized in Tabs. 2.1 and 2.2, respectively. That is, $d_1 = 20$ random variables are involved in the uncertain structural properties. $\ddot{\boldsymbol{X}}(t)$, $\dot{\boldsymbol{X}}(t)$ and $\boldsymbol{X}(t)$ are the acceleration, velocity and displacement vectors. The parameter ε, that controls the degree of hysteresis, is set to be 0.20. The term $\boldsymbol{Y}(t)$ is characterized by the Bouc–Wen model [41, 42]:

$$\dot{\boldsymbol{Y}}(t) = \rho\dot{\boldsymbol{X}}(t) - \eta|\dot{\boldsymbol{X}}(t)||\boldsymbol{Y}(t)|^{n_0-1}\mathbf{Y}(t) - \gamma\dot{\boldsymbol{X}}(t)|\boldsymbol{Y}(t)|^{n_0} \tag{2.36}$$

where ρ, η, γ and n_0 are dimensionless quantities, which control the behaviour of the model. These parameters take the following values: $\rho = 1$, $n_0 = 3$ and $\eta = \gamma = 1/\left(2u_y^{n_0}\right)$, in which $u_y = 0.01$ m is the yielding displacement. I is the loading inference vector. The seismic excitation $\ddot{U}_g(t)$ is modeled by a fully non-stationary stochastic process via the spectral representation method [43]:

$$\ddot{U}_g(t) = \sum_{i=1}^{d_2} \sqrt{2S_{\ddot{u}_g}(\omega_i, t)\Delta\omega}\,(A_i\cos(\omega_i t) + B_i\sin(\omega_i t)) \tag{2.37}$$

where $S_{\ddot{u}_g}(\omega, t)$ is the evolutionary power spectral density (EPSD) function, which is defined as:

$$S_{\ddot{u}_g}(\omega, t) = |f(\omega, t)|^2 G(\omega) \tag{2.38}$$

in which $f(\omega, t)$ is the modulation function of time and frequency, given by:

$$f(\omega, t) = \exp\left(-\delta_0 \frac{\omega t}{\omega_a t_a}\right) \cdot \left[\frac{t}{c} \exp\left(1 - \frac{t}{c}\right)\right]^k \tag{2.39}$$

and $G(\omega)$ is the one-sided PSD, assumed to follow the Clough- Penzien spectrum:

$$G(\omega) = \frac{\omega_g^4 + 4\zeta_g^2 \omega_g^2 \omega^2}{\left(\omega_g^2 - \omega^2\right)^2 + \left(2\zeta_g \omega_g \omega\right)^2} \frac{\omega^4}{\left(\omega_f^2 - \omega^2\right)^2 + \left(2\zeta_f \omega_f \omega\right)^2} S_0 \tag{2.40}$$

$\{A_i, B_i\}_{i=1}^{d_2}$ in Eq. (2.37) are $2d_2$ independent standard normal random variables. In this example, $d_2 = 800$ is adopted, i.e., a total of $d = d_1 + 2d_2 = 1620$ random variables are included in $\boldsymbol{\Theta}$. Other involved parameters are specified as: $\Delta t = 0.02$ s, $T = 30$ s; $\Delta\omega = 0.15$ rad/s, $\omega_{\mathrm{l}} = 0.15$ rad/s, $\omega_{\mathrm{u}} = 240$ rad/s; $\delta_0 = 0.15$, $c = 9.0$, $k = 2.0$; $\omega_g = 5\pi$ rad/s, $\zeta_g = \zeta_f = 0.60$, $\omega_f = 0.1\omega_g$, $S_0 = 54.6296$ cm^2/s^3.

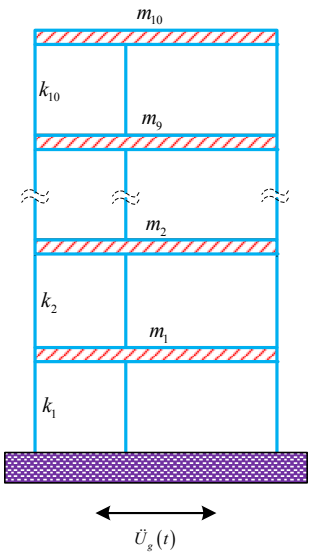


Figure 2.6: A two-bay ten-story shear frame structure subjected to stochastic seismic excitations.

The LPSS method is firstly employed to generate $N = 625$ sample points in $[0, 1]^d$ $(d = 1620)$, serving as a basic point set. Then, these samples are transformed to the random-variate space of the

Table 2.1: Probabilistic information of the lumped mass

Lumped mass	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Distribution	LN	LN	LN	LN	LN	LN	LN	LN	LN	LN
Mean ($\times 10^5$ kg)	6.0	5.9	5.8	5.7	5.6	5.5	5.4	5.3	5.2	5.1
C.O.V.	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10

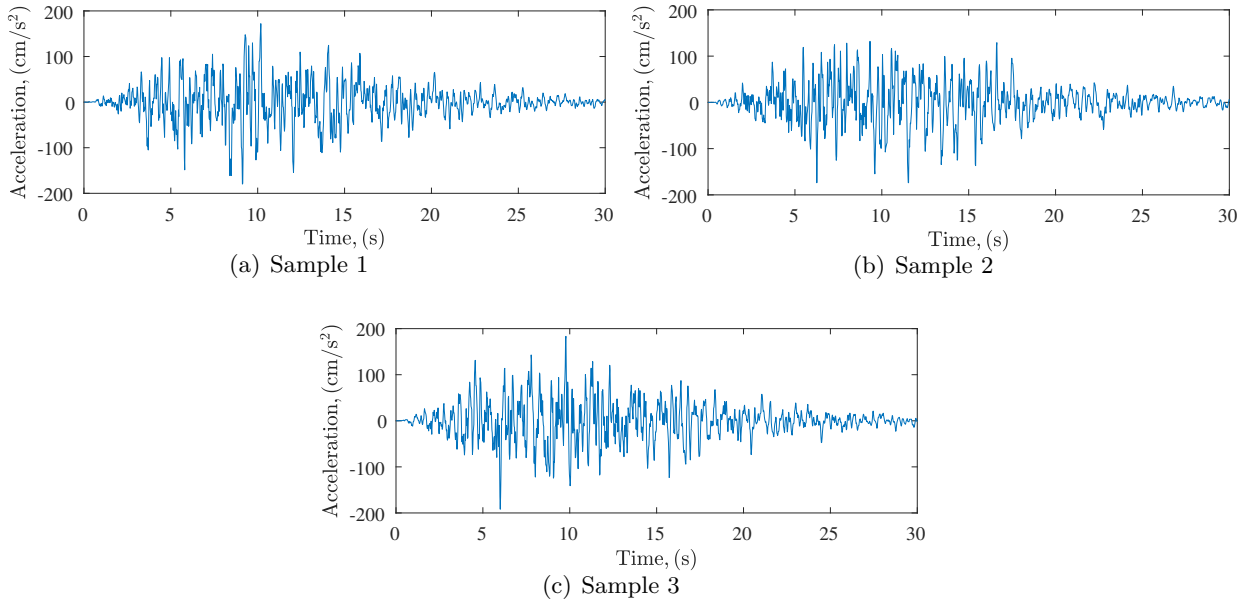
LN = Lognormal; C.O.V. = coefficient of variation

Table 2.2: Probabilistic information of the initial inter-story stiffness

Stiffness	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8	k_9	k_{10}
Distribution	G	G	G	G	G	G	G	G	G	G
Mean ($\times 10^7$ N/m)	2.50	2.45	2.40	2.35	2.30	2.25	2.20	2.15	2.10	2.05
C.O.V.	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15

G = Gaussian

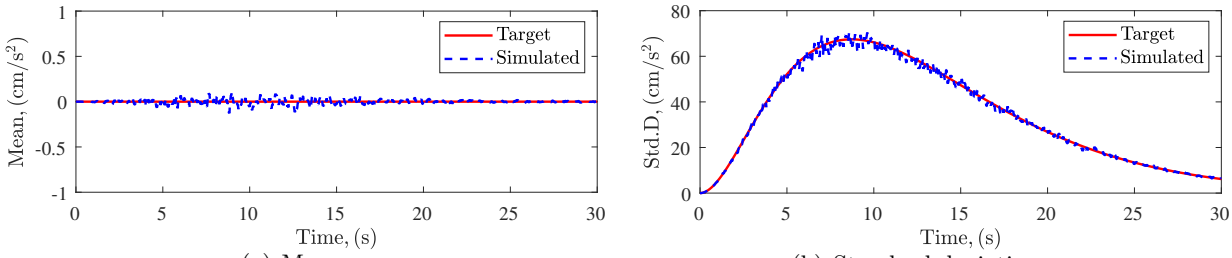
vector $\boldsymbol{\Theta}$. Integrating the samples of $\{A_i, B_i\}_{i=1}^{d_2}$ with the spectral representation, one can have a set of 625 representative samples of the fully non-stationary stochastic seismic accelerations. Three representative samples of the seismic acceleration time histories are shown in Fig. 2.7, where the non-stationary behavior in time domain can be clearly observed. Besides, the mean and standard deviation (Std.D) of 625 seismic acceleration samples against their targets are depicted in Fig. 2.8. As seen, the simulated results accord well with the targets, indicating the applicability of the LPSS based spectral representation.



(a) Sample 1 (b) Sample 2

(c) Sample 3

Figure 2.7: Representative samples of the fully non-stationary stochastic seismic acceleration.



(a) Mean (b) Standard deviation

Figure 2.8: Comparison of the simulated mean and standard deviation with their targets for the fully non-stationary stochastic seismic excitation.

Once the representative samples of the random structural properties and seismic excitation are produced, the deterministic dynamic structural analysis is performed repeatedly to extract samples of the EVD of concern. The ode45 function in Matlab is used to solve Eq. (2.35). Fig. 2.9 shows a typical sample of the restoring force v.s. inter-story drift, indicating a strong hysteretic behavior of the random structure under the fully non-stationary seismic excitations. In this study, we are especially interested in the EVDs of the 1-st, 5-th, and 10-th inter-story drift. The MGFs of interested EVDs computed by the LPSS method are depicted in Fig. 2.10, together with the reference results given by MCS (with 10^6 runs). It can be seen that all the three MGFs are accurately estimated since the 95% confidence intervals are very narrow and contain the corresponding reference results.

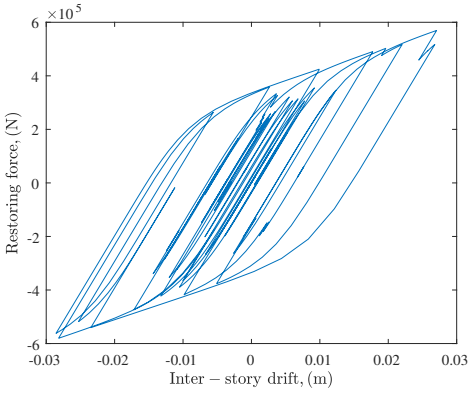
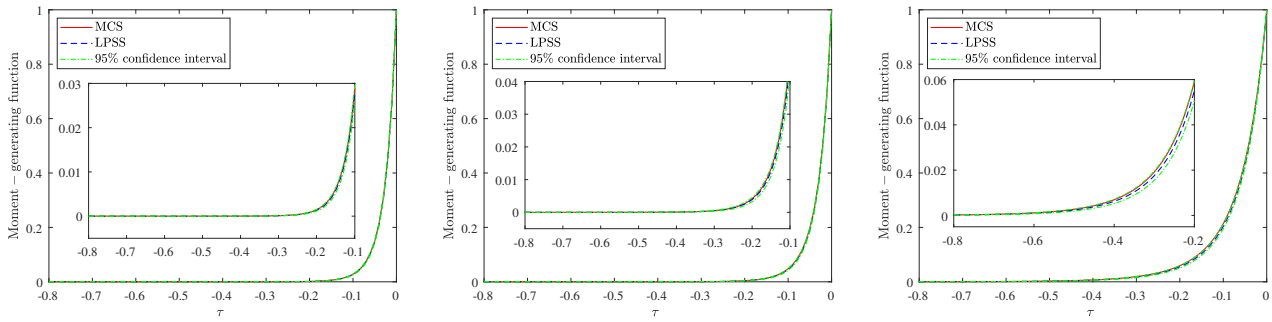


Figure 2.9: Typical hysteretic behavior for Example 3.

Based on the knowledge of the estimated MGFs, the EVDs can be recovered by the MTGIGD model. Figs. 2.11 - 2.13 show the recovered PDF, CDF and probability of exceedance (POE) curves, where the results given by MCS (with 10^6 runs) are also depicted for comparison. As seen, fairly



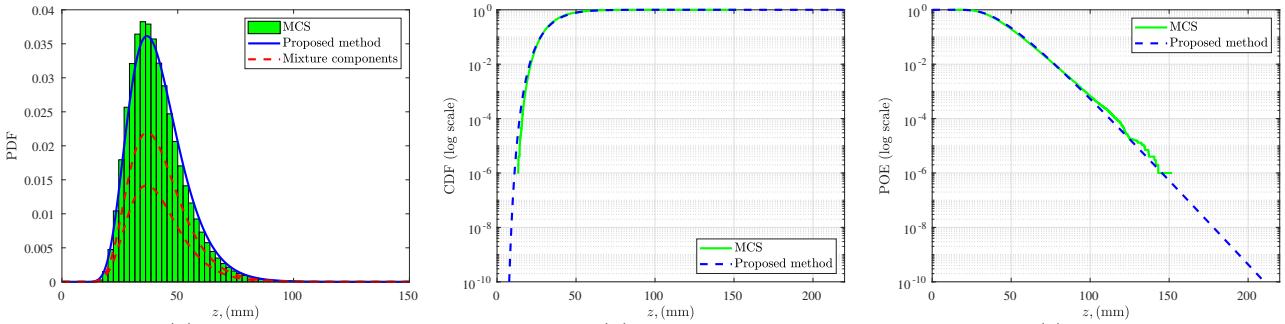
(a) 1-st inter-story drift (b) 5-th inter-story drift (c) 10-th inter-story drift

Figure 2.10: Moment-generating function of the EVD in Example 2.

good agreement can be observed between the results computed by the proposed method and those of MCS, indicating the accuracy of the proposed method for modeling not only the main body, but also the distribution tail of the EVD. From the POE curves, the first-passage failure probabilities under different thresholds can be simultaneously obtained. For example, the first-passage failure probabilities associated with the 1-st inter-story drift are listed in Tab. 2.3. In comparison with MCS, one can see that the proposed method can yield very accurate results, even for small failure probability levels less than 10^{-4}. Besides, the proposed method is much more efficient than MCS, since only $N = 625$ samples are required in this study. Finally, the EVD of the 10-th inter-story drift is also approximated by the generalized extreme value distribution (GEVD) [17] with maximum likelihood estimation, shifted generalized lognormal distribution (SGLD) [44] with the method of moments (MOM) and maximum entropy method with four sample moments as constraint. Note that the same set of samples with the proposed method is employed in all these three methods. As shown in Fig. 2.14, large approximation errors appear, especially for the CDF and POE curves, raveling the limited flexibility of those parametric distribution models and/or inadequate parameter estimation methods.

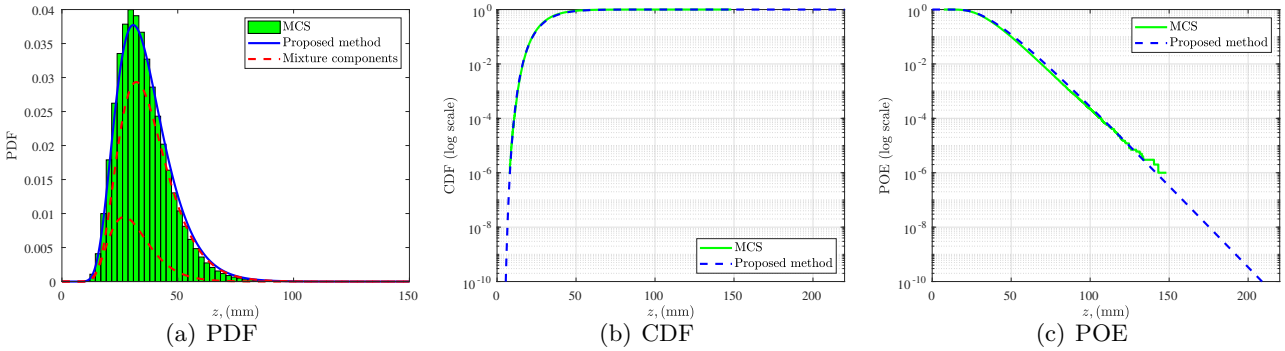
Table 2.3: Comparison of the first-passage failure probabilities under different thresholds for example 2.

Method	N	Threshold (mm)						
		72	80	88	96	104	112	120
MCS	10^6	0.0181	0.0073	0.0028	1.1020×10^{-3}	4.2000×10^{-4}	1.8000×10^{-4}	5.9000×10^{-5}
Proposed	625	0.0194	0.0072	0.0026	9.0353×10^{-4}	3.0960×10^{-4}	1.0450×10^{-4}	3.4841×10^{-5}



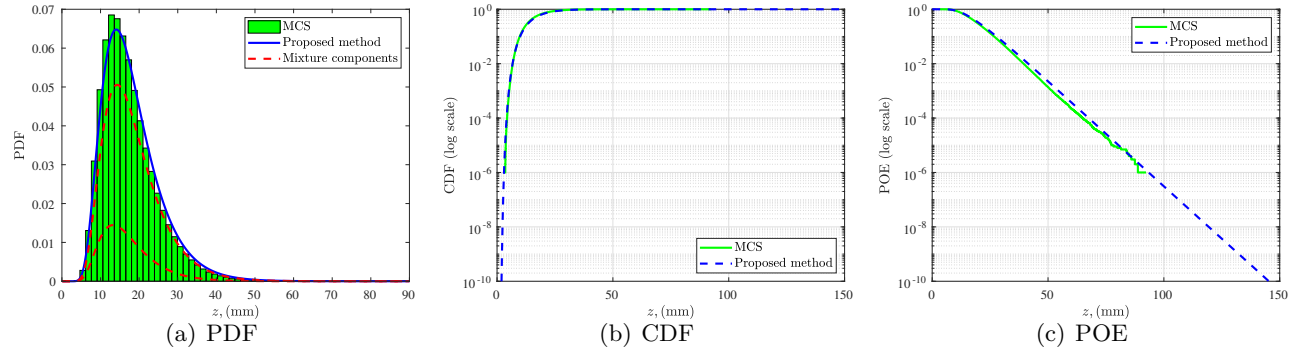
(a) PDF (b) CDF (c) POE

Figure 2.11: EVD of the 1-st inter-story drift.



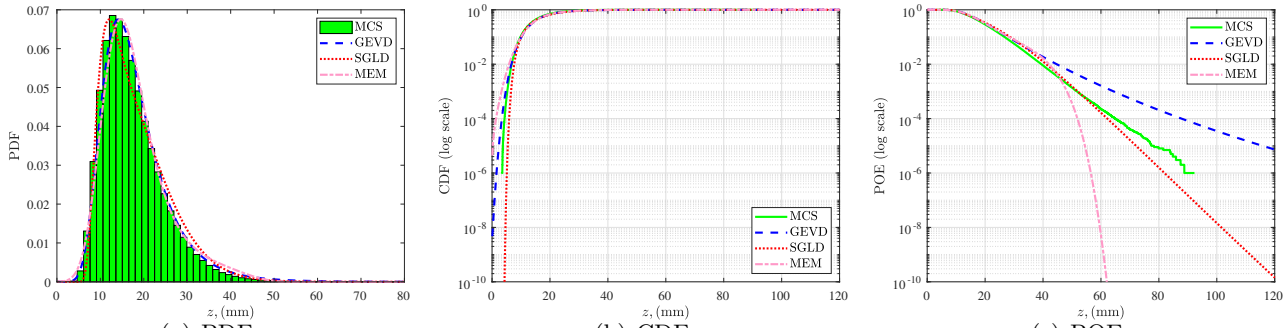
(a) PDF (b) CDF (c) POE

Figure 2.12: EVD of the 5-th inter-story drift.



(a) PDF (b) CDF (c) POE

Figure 2.13: EVD of the 10-th inter-story drift.



(a) PDF (b) CDF (c) POE

Figure 2.14: EVD of the 10-th inter-story drift by other methods.

2.5.3 Example 3: EVD and first-passage failure probabilities of a nonlinear bridge structure

A simplified two-span bridge structure is taken as the last example, as shown in Fig. 2.15. The bridge structure is modelled in OpenSees, an open software framework for earthquake engineering simulation. Since the beam is expected to remain linear elastic under seismic excitation, six elastic beam-column elements are used. The pier is modelled by four nonlinear beam-column elements, the cross section of which is discretized into 225 fibers. Constitutive law of the material adopts the bilinear model, as depicted in the figure. The Rayleigh damping is employed with assuming the damping ratio of 3% for the first two modes. Eight random variables are considered in the structural properties, whose information is listed in Tab. 2.4. For convenience, the same stochastic seismic excitations with example 2 are applied to this model. Therefore, a total of $d = 1608$ random variables are actually contained in this example. The EVD of displacement at the pile top is of particular concern for seismic reliability analysis.

Table 2.4: Random variables for Example 3

Variable	Description	Distribution	Mean	C.O.V.
m_1	Lumped mass	LN	2.00×10^5 kg	0.10
m_2	Lumped mass	LN	2.05×10^5 kg	0.10
m_3	Lumped mass	LN	1.00×10^4 kg	0.10
A	Cross-sectional area of the beam	G	5.00×10^5 mm^2	0.05
I_Z	Moment of inertia of the beam	G	5.00×10^{12} mm^4	0.05
E	Young's modulus	G	2.05×10^5 Mpa	0.15
F_y	Yield stress	G	400 Mpa	0.15
b	Strain-hardening ratio	G	0.02	0.10

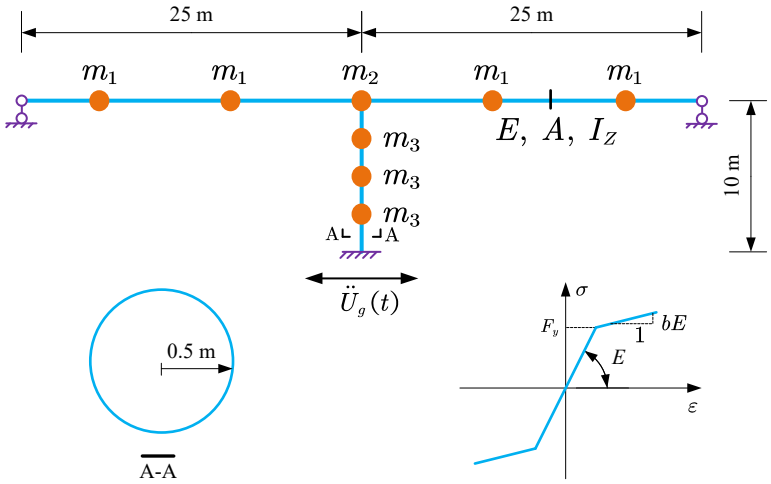


Figure 2.15: A two-span bridge structure subjected to stochastic seismic excitations.

Using the LPSS method with 625 samples, the MGF of the EVD is computed by repeatedly performing 625 deterministic dynamic structural analyses. For comparison purposes, the crude MCS is also carried out to obtain the MGF (with 10^6 runs). As depicted in Fig. 2.16, it can be seen that the 95% confidence interval of the MGF is quite narrow and contains the result by MCS, indicating the accuracy of the estimated MGF.

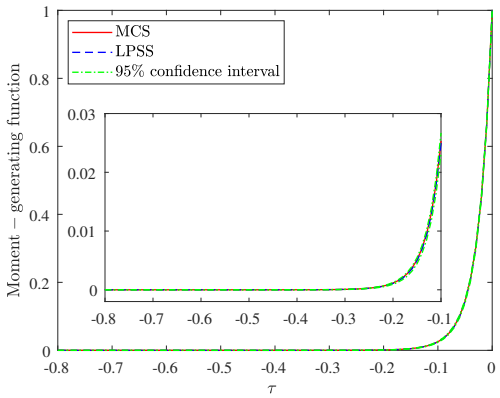
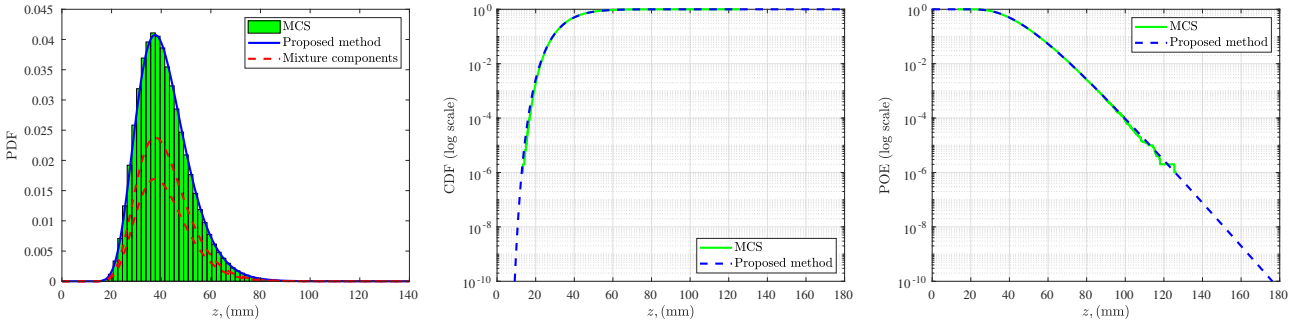


Figure 2.16: Moment-generating function of the EVD in Example 3.

The EVD can be recovered by the proposed mixture distribution, from the knowledge of the estimated MGF. In Fig. 2.17, one can observe that the PDF, CDF and POE in log scale are in good agreement with those reference results by crude MCS (with 10^6 runs). Note that we only perform 625 finite-element based structural dynamic analyses in this study. Therefore, the performance of

the proposed method for recovering the EVD is demonstrated once again in terms of accuracy and efficiency. From the recovered EVD, the first-passage failure probabilities under different prescribed thresholds can be obtained. As listed in Tab. 2.5, the results obtained from the proposed method accord well with those by MCS, even for a low level of failure probability. Further, two recently developed methods are also implemented in this example, i.e., MIGLD with fractional moments [24] and mixture of two skew normal distributions (MTSND) with Laplace transform [37]. For fair comparison, the same EVD samples used for the proposed method are adopted to calculate the fractional moments and Laplace transform. Different from the original selection, note that the Laplace transform is evaluated at a new set of points, i.e., $\{s_i\}_{i=1}^{7} = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7\}$, so as to avoid very large numbers. The results by these two methods are compared with those by MCS in Fig. 2.18. Even though there are slight differences between the PDF curve and histogram, the MIGLD model can yield fairly good results concerning the CDF and PDF curves in log scale. On contrary, the MTSND model is unable to produce a comparative POE curve probably due to its limited capability for modelling an EVD. However, it would be different to identify which one is better among the MTGIGD and MIGLD for a general problem, and further research is required.

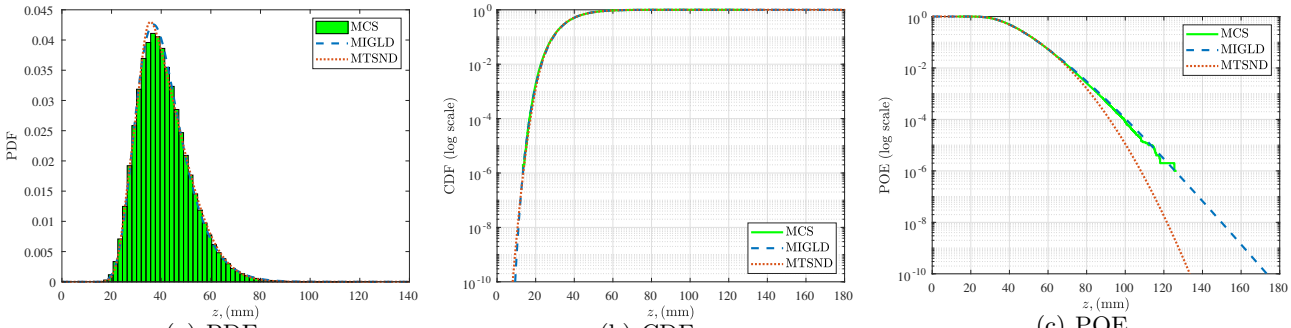


(a) PDF (b) CDF (c) POE

Figure 2.17: EVD of displacement at the pile top.

Table 2.5: Comparison of the first-passage failure probabilities under different thresholds for Example 3.

Method	N	Threshold (mm)						
		60	68	76	84	92	100	108
MCS	10^6	0.0535	0.0171	0.0049	0.0014	3.5200×10^{-4}	8.1000×10^{-5}	1.9000×10^{-5}
Proposed	625	0.0531	0.0170	0.0050	0.0014	3.6637×10^{-4}	9.3851×10^{-5}	2.3420×10^{-5}



(a) PDF (b) CDF (c) POE

Figure 2.18: EVD of displacement at the pile top by other methods.

2.6 Conclusions and remarks

In this paper, an efficient approach is proposed for evaluation of the EVD and small first-passage failure probabilities of nonlinear structures with random parameters under stochastic seismic excitations. Unlike the traditional partial characterization, i.e., integer moments or fractional moments, the MGF is first introduced to characterize the EVD of a general stochastic response process, which is usually defined as a high-dimensional expectation integral. The advantages of MGF characterization are also demonstrated and discussed. Then, a recently developed variance-reduction sampling technique, named LPSS, is employed to estimate the MGF with a small sample size. Besides, the unbiasedness of the estimator is proven and the confidence interval is given. To recover the EVD from the knowledge of its estimated MGF, a versatile MTGIGD model is proposed. The closed-form solution of MGF of the MTGIGD is derived, and a novel parameter estimation technique based on the MGF is developed accordingly. Once the EVD is reconstructed, the first-passage failure probabilities with different thresholds can be simultaneously obtained. Three numerical examples are investigated to validate the proposed approach. The first example considers the EVD of random variables, where the close-form solution of the EVD is available. A nonlinear shear frame structure and a nonlinear bridge structure with random parameters subjected to fully non-stationary stochastic seismic excitations are studied as another two examples. Results indicate that the proposed method: (1) is able to take both the randomness from structural properties and external seismic excitations into consideration; (2) can be applied to very high-dimensional stochastic dynamic systems; (3) is capable of capturing the EVD with high efficiency, not only the main body, but also the distribution tail; (4) can yield very accurate estimation of the first-passage failure prob-

abilities under different prescribed thresholds, even for a small level less than 10^{-4}; (5) is applicable for evaluation of multiple EVDs in one single run.

Two main types of errors are included in the proposed approach. The first type is caused by the assumption on parametric form for the EVD, which results in biased estimates and cannot be eliminated. Another type of error relates to the fact that the MGF of EVD is evaluated by a sampling method. Since the estimator for the MGF is proven to be unbiased, the latter error can be reduced by increasing the sample size.

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

The authors would like to appreciate the National Natural Science Foundation of China (NSFC 51905430) for financially supporting the research. The first author also appreciates the support by the China Scholarship Council (CSC).

Bibliography

[1] Junho Song and Armen Der Kiureghian. Joint first-passage probability and reliability of systems under stochastic excitation. *Journal of Engineering Mechanics*, 132(1):65–77, 2006.

[2] Jie Li and Jian-Bing Chen. *Stochastic dynamics of structures*. John Wiley & Sons, 2009.

[3] RS Langley. A first passage approximation for normal stationary random processes. *Journal of Sound and Vibration*, 122(2):261–275, 1988.

[4] Jun He. Numerical calculation for first excursion probabilities of linear systems. *Probabilistic Engineering Mechanics*, 24(3):418–425, 2009.

[5] JB Roberts and PD Spanos. Stochastic averaging: an approximate method of solving random vibration problems. *International Journal of Non-Linear Mechanics*, 21(2):111–134, 1986.

[6] Yaping Zhao. An improved energy envelope stochastic averaging method and its application to a nonlinear oscillator. *Journal of Vibration and Control*, 23(1):119–130, 2017.

[7] A Naess and V Moe. Efficient path integration methods for nonlinear dynamic systems. *Probabilistic Engineering Mechanics*, 15(2):221–231, 2000.

[8] A Di Matteo, M Di Paola, and A Pirrotta. Path integral solution for nonlinear systems under parametric poissonian white noise input. *Probabilistic Engineering Mechanics*, 44:89–98, 2016.

[9] Siu-Kui Au and James L Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, 16(4):263–277, 2001.

[10] Hong-Shuang Li, Yuan-Zhuo Ma, and Zijun Cao. A generalized subset simulation approach for estimating small failure probabilities of multiple stochastic responses. *Computers & Structures*, 153:239–251, 2015.

[11] Jie Li, Jian-Bing Chen, and Wen-liang Fan. The equivalent extreme-value event and evaluation of the structural system reliability. *Structural Safety*, 29(2):112–131, 2007.

[12] B Goller, HJ Pradlwarter, and GI Schuëller. Reliability assessment in structural dynamics. *Journal of Sound and Vibration*, 332(10):2488–2499, 2013.

[13] Meng-Ze Lyu, Jian-Bing Chen, and Antonina Pirrotta. A novel method based on augmented markov vector process for the time-variant extreme value distribution of stochastic dynamical systems enforced by poisson white noise. *Communications in Nonlinear Science and Numerical Simulation*, 80:104974, 2020.

[14] Jian-Bing Chen and Meng-Ze Lyu. A new approach for time-variant probability density function of the maximal value of stochastic dynamical systems. *Journal of Computational Physics*, page 109525, 2020.

[15] Arvid Næss and Oleg Gaidai. Estimation of extreme values from sampled time series. *Structural Safety*, 31(4):325–334, 2009.

[16] A Naess and O Gaidai. Monte carlo methods for estimating the extreme response of dynamical systems. *Journal of Engineering Mechanics*, 134(8):628–636, 2008.

[17] M Grigoriu and G Samorodnitsky. Reliability of dynamic systems in random environment by extreme value theory. *Probabilistic Engineering Mechanics*, 38:54–69, 2014.

[18] Yan-Gang Zhao, Long-Wen Zhang, Zhao-Hui Lu, and Jun He. First passage probability assessment of stationary non-gaussian process using the third-order polynomial transformation. *Advances in Structural Engineering*, 22(1):187–201, 2019.

[19] Tong Zhou and Yongbo Peng. Adaptive bayesian quadrature based statistical moments estimation for structural reliability analysis. *Reliability Engineering & System Safety*, page 106902, 2020.

[20] Jun Xu and Shengyang Zhu. An efficient approach for high-dimensional structural reliability analysis. *Mechanical Systems and Signal Processing*, 122:152–170, 2019.

[21] Jun Xu and Chao Dang. A novel fractional moments-based maximum entropy method for high-dimensional reliability analysis. *Applied Mathematical Modelling*, 75:749–768, 2019.

[22] Zhi-Qiang Chen, Shi-Xiong Zheng, Jin Zhang, Chao Dang, Kai Wei, and Xi Li. Efficient seismic reliability analysis of non-linear structures under non-stationary ground motions. *Soil Dynamics and Earthquake Engineering*, 139:106385, 2020.

[23] Umberto Alibrandi and Khalid M Mosalam. Kernel density maximum entropy method with generalized moments for evaluating probability distributions, including tails, from a small sample of data. *International Journal for Numerical Methods in Engineering*, 113(13):1904–1928, 2018.

[24] Chao Dang and Jun Xu. A mixture distribution with fractional moments for efficient seismic reliability analysis of nonlinear structures. *Engineering Structures*, 208:109912, 2020.

[25] Jun He and Jinghai Gong. Estimate of small first passage probabilities of nonlinear random vibration systems by using tail approximation of extreme distributions. *Structural Safety*, 60:28–36, 2016.

[26] Jun Xu, Zhaodong Ding, and Jia Wang. Extreme value distribution and small failure probabilities estimation of structures subjected to non-stationary stochastic seismic excitations. *Structural Safety*, 70:93–103, 2018.

[27] Jun Xu, Jia Wang, and Ding Wang. Evaluation of the probability distribution of the extreme value of the response of nonlinear structures subjected to fully nonstationary stochastic seismic excitations. *Journal of Engineering Mechanics*, 146(2):06019006, 2020.

[28] Jianbing Chen and Jie Li. Extreme value distribution and reliability of nonlinear stochastic structures. *Earthquake Engineering and Engineering Vibration*, 4(2):275–286, 2005.

[29] Jian-Bing Chen and Jie Li. The extreme value distribution and dynamic reliability analysis of nonlinear structures with uncertain parameters. *Structural Safety*, 29(2):77–93, 2007.

[30] Guohai Chen and Dixiong Yang. Direct probability integral method for stochastic response analysis of static and dynamic structural systems. *Computer Methods in Applied Mechanics and Engineering*, 357:112612, 2019.

[31] Michael D Shields and Jiaxin Zhang. The generalization of latin hypercube sampling. *Reliability Engineering & System Safety*, 148:96–108, 2016.

[32] Zixin Liu, Zhangjun Liu, Xinxin Ruan, and Qi Zhang. Spectral representation-based dimension reduction for simulating multivariate non-stationary ground motions. *Soil Dynamics and Earthquake Engineering*, 114:313–325, 2018.

[33] KK Phoon, HW Huang, and ST Quek. Simulation of strongly non-gaussian processes using karhunen–loeve expansion. *Probabilistic Engineering Mechanics*, 20(2):188–198, 2005.

[34] Jianwen Liang, Samit Ray Chaudhuri, and Masanobu Shinozuka. Simulation of nonstationary stochastic processes by spectral representation. *Journal of Engineering Mechanics*, 133(6):616–627, 2007.

[35] Patrick Billingsley. *Probability and Measure: Anniversary Edition*. Wiley, 2012.

[36] Di Zhou, Xufang Zhang, and Yimin Zhang. Dynamic reliability analysis for planetary gear system in shearer mechanisms. *Mechanism and Machine Theory*, 105:244–259, 2016.

[37] Chao Dang and Jun Xu. Unified reliability assessment for problems with low- to high-dimensional random inputs using the laplace transform and a mixture distribution. *Reliability Engineering & System Safety*, page 107124, 2020.

[38] Bent Jorgensen. *Statistical properties of the generalized inverse Gaussian distribution*, volume 9. Springer Science & Business Media, 2012.

[39] J Leroy Folks and Raj S Chhikara. The inverse gaussian distribution and its statistical application—a review. *Journal of the Royal Statistical Society: Series B (Methodological)*, 40(3):263–275, 1978.

[40] Geoffrey J McLachlan and David Peel. *Finite mixture models*. John Wiley & Sons, 2004.

[41] Yi-Kwei Wen. Method for random vibration of hysteretic systems. *Journal of the Engineering Mechanics Division*, 102(2):249–263, 1976.

[42] Y. K. Wen. Equivalent Linearization for Hysteretic Systems Under Random Excitation. *Journal of Applied Mechanics*, 47:150–154, 1980.

[43] Zhangjun Liu, Wei Liu, and Yongbo Peng. Random function based spectral representation of stationary and non-stationary stochastic processes. *Probabilistic Engineering Mechanics*, 45:115–126, 2016.

[44] Ying M Low. A new distribution for fitting four moments and its applications to reliability analysis. *Structural Safety*, 42:12–25, 2013.

Chapter 3

Interval uncertainty propagation by Bayesian global optimization

Interval uncertainty propagation by a parallel Bayesian global optimization method

Chao Dang[a], Pengfei Wei[b,⋆],Matthias G.R. Faes[c],Marcos A. Valdebenito[d], Michael Beer[a,e,f]

[a]*Institute for Risk and Reliability, Leibniz University Hannover, Callinstr. 34, Hannover 30167, Germany*

[b]*School of Power and Energy, Northwestern Polytechnical University, Xi'an 710072, PR China*

[c]*Chair for Reliability Engineering, TU Dortmund University, Leonhard-Euler-Str. 5, Dortmund 44227, Germany*

[d]*Faculty of Engineering and Sciences, Universidad Adolfo Ibáñez, Av. Padre Hurtado 750, Viña del Mar 2562340, Chile*

[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 20 0 092, PR China*

Published in Applied Mathematical Modelling in August 2022

Abstract: This paper is concerned with approximating the scalar response of a complex computational model subjected to multiple input interval variables. Such task is formulated as finding both the global minimum and maximum of a computationally expensive black-box function over a prescribed hyper-rectangle. On this basis, a novel non-intrusive method, called 'triple-engine parallel Bayesian global optimization', is proposed. The method begins by assuming a Gaussian process prior (which can also be interpreted as a surrogate model) over the response function. The main contribution lies in developing a novel infill sampling criterion, i.e., triple-engine pseudo expected improvement strategy, to identify multiple promising points for minimization and/or maximization based on the past obser- vations at each iteration. By doing so, these identified points can be evaluated on the real response function in parallel. Besides, another potential benefit is that both the lower and upper bounds of the model response can be obtained with a single run of the developed method. Four numerical examples with varying complexity are investigated to demonstrate the proposed method against some existing techniques, and results indicate that significant computational savings can be achieved by making full use of prior knowledge and parallel computing.

*Corresponding Author
E-mail address: pengfeiwei@nwpu.edu.cn (P. Wei)

Keywords: Interval uncertainty propagation, Bayesian global optimization, Gaussian process, Infill sampling criterion, Parallel computing

3.1 Introduction

Along with the rapid development of computation techniques, deterministic numerical analysis has made great progress in various fields over the past several decades [1]. In this context, all parameters of a computational model designed to describe underlying structures or systems are typically treated as precise (crisp) numbers. This kind of numerical analysis, however, is essentially not suitable for situations where non-determinism has to be properly considered, which is the common case for a broad range of modern science and engineering disciplines. A typical example of such situations is the design and analysis of engineering systems at an early stage where many aspects could be only imprecisely known. Alternatively, non-deterministic numerical analysis is emerging as an exciting research frontier with new opportunities and also challenges. Such opportunities and challenges arise throughout the whole analysis, e.g., non-determinism characterization on the input side and response uncertainty quantification on the output side.

In general, three types of approaches are available for modelling non-determinism: probabilistic approach, imprecise probabilistic approach and non-probabilistic approach [2]. On the basis of classical probability theory and statistical techniques, the probabilistic approach is most widely used. Herein, an uncertain parameter is modelled as a random variable with a precisely known probability distribution. Thus, it is often challenging to apply the probabilistic approach in reality since a large amount of high-quality data is required to infer an accurate probability distribution. Against this background, by generalizing traditional probability and statistics concepts, the imprecise probabilistic approach has evolved as a powerful and elegant framework for quantifying uncertainty from incomplete information [3, 4]. Within this approach, one needs to assign a pair of lower and upper probabilities to an event, rather than a single probability. On the other hand, the non-probabilistic approach, such as interval models and fuzzy sets [5], is also gaining increasing interest for non-determinism modelling, especially when the available information is limited. With the interval concept, a non-deterministic parameter is treated as an interval variable specified by a pair of numbers, i.e., the lower and upper bounds, and potentially a function modelling the auto-dependencies among multiple interval parameters [6]. Thus, instead of a full probability distribution

the analyst only needs to determine the bounds and auto-dependency functions, which can be easily and objectively acquired from a small number of samples. The present study limits its scope to interval uncertainty.

There have been plenty of methods developed to propagate interval uncertainty via a computational model, which can be roughly classified into four kinds. The first kind of methods is based on using the interval arithmetic of Moore, e.g., refer to [7]. Despite its efficiency, the interval calculus cannot trace parameter dependency by definition (the so-called dependency problem), which therefore can lead to a severe overestimation of the size of a response interval. Recent developments are focused on limiting the overestimation by, e.g., accounting for dependency among interval variables [8–12] or using interval fields [13, 14], parameterizing intervals via trigonometric functions [15, 16] and representing intervals by affine arithmetic [17, 18], etc. Although these methods are able to provide sharp bounds within reasonable computational cost, their applicability is still limited due to the intrusive nature of interval arithmetic. The approximate analytical methods that rely on constructing a simplified approximation of true response function falls in the second group. Typical examples of such methods include, Taylor series expansion methods [19–23] and Chebyshev series expansion methods [24, 25], which are intrusive and non-intrusive, respectively. However, these Taylor methods tend to lose accuracy when the considered problem involves large uncertainty (i.e.,the widths of interval variables being large) and/or highly nonlinear behavior. For these Chebyshev methods, the required number of response function evaluations grows exponentially with the number of dimensions. As for the third type, the vertex method [26, 27] and interval multilevel quasi-Monte Carlo (I-MLQMC)) [28, 29] are non-intrusive and can produce accurate response bounds under certain conditions. The classical vertex method is exact on the premise that the response function is monotonic with respect to d interval parameters, while at the cost of 2^d model evaluations. More strictly, the I-MLQMC method requires a linearity assumption on the response function. As such, these two methods suffer from non-linearity and/or dimensionality.

In the last group, global optimization methods are naturally applicable to the topic of interval numerical analysis. In this context, several studies have been conducted by directly using, e.g., genetic algorithm [30, 31]. Generally, global optimization algorithms require a large number of model evaluations to find the minimum/maximum, and hence can be computationally demanding especially when each such evaluation is expensive. To alleviate the computation burden, a cheap-to-evaluate surrogate model can be adopted to substitute the original computational model based on

some observations. Along this line, Kriging-assisted global optimization (formally called Bayesian global optimization (BGO)) algorithms are attracting increasing attention due to their high efficiency for optimizing expensive black-box functions. A typical BGO method starts by building an initial Kriging model for the objective function based on a small number of observations, and then refines the initial model by sequentially selecting more updating points according to a infill sampling criterion [32]. Existing studies then focus more on developing efficient infill sampling criteria so as to reduce the total number of function evaluations. On this aspect, representative works in the context of interval uncertainty propagation include the maximum improvement criterion [33], expected improvement criterion [34, 35] and a comparison study of several criteria [36]. It is shown that these methods exhibit encouraging features regarding the computational efficiency and accuracy for computationally expensive black-box problems over other existing methods. Despite these advantages, one of the major limitations of the existing BGO methods is that they are sequential in nature and hence unsuitable for parallelization, or at least high-level parallelization, hindering the potential benefits from parallel distributed processing.

In this paper, a parallel Bayesian global optimization (PBGO) method is proposed for estimating the response bounds of a computational model in the presence of interval variables. Our main objective is to further reduce the computational time of existing BGO methods by making use of parallelism. For this purpose, a novel infill sampling criterion is developed to select multiple points at each iteration, and hence corresponding model evaluations can be distributed on multiple processing cores simultaneously. Such parallelisation is relevant when the model at hand is computationally intensive and parallel computing facilities are available. Besides, in contrast to the traditional way of searching the lower and upper bounds of a scalar response quantity via two separate optimization problems, we consider it only as one problem. Following the developed scheme, the lower and upper bounds can be obtained simultaneously with a single run. Last but not least, a Matlab implementation of the developed algorithm is also readily available to the public [1].

The remainder of the paper is organized as follows. Section 3.2 describes the interval analysis problem to be solved in this study. The proposed PBGO method is introduced in Section 3.3, with its relationship to other PBGO methods also being discussed. Four numerical examples are studied in Section 3.4 to demonstrate the performance of the developed method. In Section 3.5, some concluding remarks and perspectives are given to end the paper.

[1] `https://github.com/Chao-Dang/Triple-engine-Bayesian-global-optimization`

3.2 Problem formulation

Let us consider a computational model represented by a deterministic, continuous, and real-valued function $y = g(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}$. Here the model response y is a scalar quantity of interest, the g-function is assumed to be an expensive-to-evaluate black box, and the model input vector $\boldsymbol{x}$ consists of d variables, i.e., $\boldsymbol{x} = [x_1, x_2, \cdots, x_d]$.

Under the assumption that available information on the model inputs is poor or incomplete, we proceed to treat them with interval models. For identifying intervals from real observations, one can refer to, e.g., [37, 38]. An interval vector $\boldsymbol{x}^I = [x_1^I, x_2^I, \cdots, x_d^I] \in \mathbb{IR}^d$ can be defined as:

$$\boldsymbol{x}^I = [\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}] = \left\{ \boldsymbol{x} \in \mathbb{R}^d | \underline{\boldsymbol{x}} \leq \boldsymbol{x} \leq \overline{\boldsymbol{x}} \right\}, \tag{3.1}$$

and its component x_i^I satisfies

$$x_i^I = [\underline{x}_i, \overline{x}_i] = \{ x \in \mathbb{R} | \underline{x}_i \leq x \leq \overline{x}_i \}, i = 1, 2, \cdots, d, \tag{3.2}$$

where $\underline{\boldsymbol{x}} = [\underline{x}_1, \underline{x}_2, \cdots, \underline{x}_d]$ and $\overline{\boldsymbol{x}} = [\overline{x}_1, \overline{x}_2, \cdots, \overline{x}_d]$ represent the lower and upper bounds of $\boldsymbol{x}^I$, respectively. Further, the midpoint $\boldsymbol{x}^C$ and radius $\boldsymbol{x}^R$ of $\boldsymbol{x}^I$ can be defined as:

$$\boldsymbol{x}^C = \frac{\underline{\boldsymbol{x}} + \overline{\boldsymbol{x}}}{2}, \tag{3.3}$$

$$\boldsymbol{x}^R = \frac{\overline{\boldsymbol{x}} - \underline{\boldsymbol{x}}}{2}. \tag{3.4}$$

It follows that the interval vector defined in Eq. (3.1) can also be rewritten in terms of $\boldsymbol{x}^C$ and $\boldsymbol{x}^R$ as:

$$\boldsymbol{x}^I = \boldsymbol{x}^C + \delta\boldsymbol{x}, \tag{3.5}$$

where $\delta\boldsymbol{x} \in [-1, 1]\boldsymbol{x}^R$. For convenience, the interval variables are assumed to be independent. In fact, for dependent interval variables one can transform them into independent ones by applying a suitable transformation, e.g., [39].

With the interval vector $\boldsymbol{x}^I$ as input, the g-function will also give rise to a interval output y^I in our context, i.e., $y^I = \left\{ y \in \mathbb{R} | y = g(\boldsymbol{x}), \boldsymbol{x} \in \boldsymbol{x}^I \right\}$. The resulting interval can be fully characterized by its lower and upper bounds, which correspond to the worst or best case of y^I that we might

be interested in. Therefore, the main objective is to determine the lower and upper bounds of y^I, which are naturally defined as the solutions of the following two optimization problems:

$$\underline{y} = \min_{\boldsymbol{x} \in \boldsymbol{x}^I} \{y | y = g(\boldsymbol{x})\}, \tag{3.6}$$

$$\overline{y} = \max_{\boldsymbol{x} \in \boldsymbol{x}^I} \{y | y = g(\boldsymbol{x})\}, \tag{3.7}$$

where $\underline{y}$ and $\overline{y}$ can be interpreted as the global minimum and maximum of $y = g(\boldsymbol{x})$ subject to $\boldsymbol{x} \in \boldsymbol{x}^I$, respectively.

Although their definitions are rather simple, the analytical solutions to $\underline{y}$ and $\overline{y}$ are unavailable for a general black-box problem. Thus, numerical approximation techniques are necessary and useful tools for practical applications. Existing numerical methods, however, still suffer from their respective limitations as discussed in the introduction section. This motivates us to develop a PBGO method for propagating interval uncertainty in the following section.

3.3 Triple-engine parallel Bayesian global optimization

In this section, the propagation of interval uncertainty via an expensive black-box computational model is treated by a kind of Bayesian numerical method, i.e., the so-called Bayesian global optimization (BGO) [32]. Specifically, an efficient method, termed "Triple-engine parallel Bayesian global optimization" (T-PBGO), is proposed to approximate the lower and upper bounds of the model output y^I (defined in Eqs. (3.6) and (3.7)) when the model input is characterized by a interval vector $\boldsymbol{x}^I$ (defined in Eq. (3.1)). The proposed method makes use of the Gaussian process model and a newly developed infill sampling criterion, as will be introduced in what follows. For notational simplicity, the superscripts of $\boldsymbol{x}^I$ and y^I are omitted when there is no confusion.

3.3.1 Gaussian process model

Under the black-box assumption, no additional knowledge on the inner structure of the g-function is available and the only possibility for us is to evaluate it at some points. That is, we know nothing about the behavior of the g-function (e.g., concavity and linearity) before seeing any observations, let along its minimum and maximum. The lack of knowledge on $g(\cdot)$ is referred to as a kind of epistemic uncertainty simply because it is numerically unknown until we actually evaluate

it, and hence reduceable. Following a Bayesian approach, our prior beliefs on the g-function can be modeled by assigning a Bayesian prior distribution. In this study, we adopt a Gaussian process (GP) prior over g. In the following, we only give a brief introduction to the GP model, and for further details the reader can refer to [40]. The GP prior assumes that the g-function is a realization of a GP indexed by $\boldsymbol{x}$. To formalize this, we write the GP prior as:

$$\hat{g}_0(\boldsymbol{x}) \sim \mathcal{GP}(m_0(\boldsymbol{x}), k_0(\boldsymbol{x}, \boldsymbol{x}')) = m_0(\boldsymbol{x}) + Z(\boldsymbol{x}), \tag{3.8}$$

where $\hat{g}_0$ denotes the prior distribution of g; $m_0(\boldsymbol{x})$ is the mean function of the GP prior; $Z(\boldsymbol{x})$ is a stationary GP with zero-mean and covariance function $k_0(\boldsymbol{x}, \boldsymbol{x}')$. The GP prior is completely characterized by its prior mean function $m_0(\boldsymbol{x})$ and covariance function $k_0(\boldsymbol{x}, \boldsymbol{x}')$. The prior mean function reflects the general trend of the GP model, while the prior covariance function encodes the key features of the g-function, e.g., stationarity, isotropy, smoothness and periodicity. There are many kinds of specific functional forms available in literature for the prior mean and covariance functions [40]. In this paper, without loss of generality, the prior mean function is assumed to be a constant (i.e., $m_0(\boldsymbol{x}) = \beta$) and the prior covariance function is of squared exponential form expressed as:

$$k_0(\boldsymbol{x}, \boldsymbol{x}') = \sigma_g^2 \exp\left[-\frac{1}{2}\left(\boldsymbol{x} - \boldsymbol{x}'\right)\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{x} - \boldsymbol{x}'\right)^{\mathrm{T}}\right], \tag{3.9}$$

where σ_g^2 is the overall variance with $\sigma_g > 0$; $\boldsymbol{\Sigma} = \mathrm{diag}\left(l_1^2, l_2^2, \cdots, l_d^2\right)$ with $l_i > 0$ being the characteristic length-scale in i-th dimension; and $\mathrm{diag}(\cdot)$ denotes a diagonal matrix whose entries are equal to the argument values. The $d + 2$ free parameters β, σ_g^2 and $\{l_i\}_{i=1}^d$ are referred to hyper-parameters whose values need to be determined, denoted by $\boldsymbol{\theta} = \left\{\beta, \sigma_g^2, l_1, l_2, \cdots, l_d\right\}$.

Now assume that we have evaluated the g-function at several (e.g., $n \in \mathbb{Z}^+$) points. We aggregate the sampled points in an $n \times d$ matrix $\boldsymbol{X}$ with its j-th row being the j-th point $\boldsymbol{x}^{(j)}$, and the corresponding g-function values in an $n \times 1$ vector $\boldsymbol{y}$ with its j-th element being $y^{(j)}$, where $y^{(j)} = g(\boldsymbol{x}^{(j)})$. The set of hyper-parameters can then be estimated by minimizing the negative log marginal likelihood (NLML) [40]:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \left(-\log\left[p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})\right]\right), \tag{3.10}$$

with

$$-\log\left[p(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta})\right] = \frac{1}{2}(\boldsymbol{y}-\beta)^{\mathrm{T}}\boldsymbol{K}_0^{-1}(\boldsymbol{y}-\beta) + \frac{1}{2}\log\left[|\boldsymbol{K}_0|\right] + \frac{n}{2}\log\left[2\pi\right], \tag{3.11}$$

where $\boldsymbol{K}_0$ is an $n \times n$ covariance matrix with its (i,j)-th entry being $[\boldsymbol{K}_0]_{ij} = k_0(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$. Eq. (3.10) can be solved by gradient-based optimization schemes since the derivatives of NLML in Eq. (3.11) with respect to $\boldsymbol{\theta}$ are analytically tractable.

Conditioning on the observations $(\boldsymbol{X}, \boldsymbol{y})$ and GP prior will give rise to a posterior distribution $\hat{g}_n$ of g. This distribution still follows a GP $\hat{g}_n(\boldsymbol{x}) \sim \mathcal{GP}(m_n(\boldsymbol{x}), k_n(\boldsymbol{x}, \boldsymbol{x}'))$, with the posterior mean and covariance functions as follows:

$$m_n(\boldsymbol{x}) = m_0(\boldsymbol{x}) + \boldsymbol{k}_0(\boldsymbol{x}, \boldsymbol{X})\boldsymbol{K}_0^{-1}(\boldsymbol{y} - \boldsymbol{m}_0(\boldsymbol{X})), \tag{3.12}$$

$$k_n(\boldsymbol{x}, \boldsymbol{x}') = k_0(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{k}_0(\boldsymbol{x}, \boldsymbol{X})\boldsymbol{K}_0^{-1}\boldsymbol{k}_0(\boldsymbol{x}', \boldsymbol{X})^{\mathrm{T}}, \tag{3.13}$$

where $\boldsymbol{k}_0(\boldsymbol{x}, \boldsymbol{X})$ is a $1 \times n$ covariance vector between $\boldsymbol{x}$ and $\boldsymbol{X}$, whose j-th element is $k_0(\boldsymbol{x}, \boldsymbol{x}^{(j)})$; $\boldsymbol{k}_0(\boldsymbol{x}', \boldsymbol{X})$ is similarly defined; $\boldsymbol{m}_0(\boldsymbol{X})$ is a $n \times 1$ mean vector, whose j-th element is $m_0(\boldsymbol{x}^{(j)})$. It is seen that via a Bayesian treatment a full predictive distribution $\hat{g}(\boldsymbol{x}) \sim \mathcal{N}(m_n(\boldsymbol{x}), \sigma_n^2(\boldsymbol{x}))$ is now available, where the posterior mean function $m_n(\boldsymbol{x})$ can be used as a predictor, while the posterior variance function $\sigma_n^2(\boldsymbol{x}) = k_n(\boldsymbol{x}, \boldsymbol{x})$ can measure the prediction uncertainty.

3.3.2 Proposed triple-engine pseudo expected improvement criterion

In order to make inference about the minimum and maximum of the g-function using as few function evaluations as possible, our main concern is to design an efficient infill sampling criterion that can effectively suggest future evaluation points based on the posterior GP (implicitly the past observations). In particular, we seek to identify a batch of informative and diverse points at each iteration. Hence, multiple evaluations of the g-function can be distributed on several cores simultaneously so as to reduce the overall wall-clock time. For convenience of illustration, we assume that the number of points we would like to select at each iteration is a even number q in sequel, though it should not to be. Our purposes are achieved by generalizing the pseudo expected improvement (PEI) criterion [41], which has been recently developed in the field of global optimization, to an enhanced version, termed 'triple-engine pseudo expected improvement' (T-PEI) criterion. The T-PEI criterion actually involves a set of three infill sampling criteria that we call

them 'engines', as discussed below.

3.3.2.1 Engine 1: PEI for minimum

The first engine is the PEI criterion originally developed in [41] for global minimization problems (denoted by PEI-MIN for convenience). In the present study, this criterion will be directly used to select q promising points for the propose of minimizing the g-function wherever applicable.

Let $y_{\min} = \min_{1 \leq j \leq n} y^{(j)}$ indicate the minimum value of y observed so far. The improvement at point $\boldsymbol{x}$ over the current best solution $y_{\min}$ can be defined as [32]:

$$I_{\min}(\boldsymbol{x}) = \max\left(y_{\min} - \hat{g}_n(\boldsymbol{x}), 0\right) = \begin{cases} y_{\min} - \hat{g}_n(\boldsymbol{x}), \text{if } \hat{g}_n(\boldsymbol{x}) < y_{\min} \\ 0, \text{ otherwise} \end{cases}, \tag{3.14}$$

which is a random variable at site $\boldsymbol{x}$. The so-called expected improvement (EI) over the current minimum $y_{\min}$ consists of taking expectation of $I_{\min}(\boldsymbol{x})$, and can be derived in a closed-form expression as [32]:

$$EI_{\min}(\boldsymbol{x}) = \mathbb{E}\left[I_{\min}(\boldsymbol{x})\right] = (y_{\min} - m_n(\boldsymbol{x}))\Phi\left(\frac{y_{\min} - m_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right) + \sigma_n(\boldsymbol{x})\varphi\left(\frac{y_{\min} - m_n(\boldsymbol{x})}{\sigma_n(\boldsymbol{x})}\right), \tag{3.15}$$

where $\varphi(\cdot)$ and $\Phi(\cdot)$ are the probability density function and cumulative distribution function of the standard normal variable, respectively. The next best point be acquired within the minimization process can be selected by maximizing $EI_{\min}(\boldsymbol{x})$, i.e.,

$$\boldsymbol{x}_{\min}^{(n+1)} = \arg\max_{\boldsymbol{x} \in \boldsymbol{x}^I} EI_{\min}(\boldsymbol{x}). \tag{3.16}$$

This criterion is referred to as EI-MIN for the sake of convenience. Note that the first term of $EI_{\min}(\boldsymbol{x})$ (see Eq. (3.15)) prefers the point whose prediction $m_n(\boldsymbol{x})$ is small, whereas the second term prefers the point whose variance $\sigma_n^2(\boldsymbol{x})$ is large.Thus, the EI-MIN criterion gives an elegant balance between exploitation (i.e., local search) and exploration (i.e., global search). Despite this, the EI-MIN criterion can only produce one single point at each iteration, and hence not suitable for parallelization.

To overcome the limitation, the basic idea of the PEI-MIN criterion is to modify the initial EI function (Eq. (3.15)) sequentially, by multiplying it by an influence function. That is, the first

updating point $\boldsymbol{x}_{\min}^{(n+1)}$ is still generated by using the initial EI-MIN criterion (Eq. (3.16)). Then, the second one $\boldsymbol{x}_{\min}^{(n+2)}$ can be identified by maximizing a modified EI function that considers the possible impact of the first updated point bringing to the EI function. In such a sequential way, a desired number of points can be obtained at each iteration without evaluating the g-function at any newly selected points. Thus, a good influence function should capture the real influence of the newly identified points on the initial EI function as much as possible, while remaining computationally tractable. The influence function proposed in [41] is motivated by the fact that the EI function (Eq. (3.15)) is zero at the sampled points, and positive in between. After $q-1$ points have been identified, the synthesized influence function for the q-th point is formulated as [41]:

$$\begin{aligned} IF(\boldsymbol{x}; \boldsymbol{x}_{\min}^{(n+1)}, \boldsymbol{x}_{\min}^{(n+2)}, \cdots, \boldsymbol{x}_{\min}^{(n+q-1)}) &= \prod_{j=1}^{q-1} \left[1 - \rho\left(\boldsymbol{x}, \boldsymbol{x}_{\min}^{(n+j)}\right)\right] \\ &= \prod_{j=1}^{q-1} \left[1 - \exp\left[-\frac{1}{2}\left(\boldsymbol{x} - \boldsymbol{x}_{\min}^{(n+j)}\right) \boldsymbol{\Sigma}^{-1} \left(\boldsymbol{x} - \boldsymbol{x}_{\min}^{(n+j)}\right)^{\mathrm{T}}\right]\right], \end{aligned} \tag{3.17}$$

where $\rho\left(\boldsymbol{x}, \boldsymbol{x}_{\min}^{(n+j)}\right)$ is the correlation function between two points $\boldsymbol{x}$ and $\boldsymbol{x}_{\min}^{(n+j)}$. It should be noted that the influence function is zero at the $q-1$ newly selected points $\boldsymbol{x}_{\min}^{(n+1)}, \boldsymbol{x}_{\min}^{(n+2)}, \cdots, \boldsymbol{x}_{\min}^{(n+q-1)}$, and approaches to one when far away from these points. The PEI function for the q-th point can be defined as [41]:

$$PEI_{\min}(\boldsymbol{x}; \boldsymbol{x}_{\min}^{(n+1)}, \boldsymbol{x}_{\min}^{(n+2)}, \cdots, \boldsymbol{x}_{\min}^{(n+q-1)}) = EI_{\min}(\boldsymbol{x}) \times IF(\boldsymbol{x}; \boldsymbol{x}_{\min}^{(n+1)}, \boldsymbol{x}_{\min}^{(n+2)}, \cdots, \boldsymbol{x}_{\min}^{(n+q-1)}). \tag{3.18}$$

The $PEI_{\min}$ function can be interpreted as an approximation of the 'real' $EI_{\min}$ function because it is constructed without evaluating the g-function at these $q-1$ points and updating the GP model (i.e., re-evaluating the hyper-parameters). Besides, it reduces to the standard $EI_{\min}$ function when $q = 1$, and hence the standard $EI_{\min}$ function can be seen as a special case of the PEI function. The q-th point can be selected by maximizing the $PEI_{\min}$ function such that:

$$\boldsymbol{x}_{\min}^{(n+q)} = \arg\max_{\boldsymbol{x} \in \boldsymbol{x}^I} PEI_{\min}(\boldsymbol{x}; \boldsymbol{x}_{\min}^{(n+1)}, \boldsymbol{x}_{\min}^{(n+2)}, \cdots, \boldsymbol{x}_{\min}^{(n+q-1)}).$$

3.3.2.2 Engine 2: PEI for maximum

Inspired by the PEI-MIN criterion, we can also define a similar criterion to select q promising points for maximizing the g-function if needed. The resulting criterion is called PEI-MAX, which is regarded as the second engine.

Let $y_{\max} = \max_{1 \leq j \leq n} y^{(j)}$ denote the maximum value of y among the past n observations. In analogy to Eq. (3.14), the improvement at point $\boldsymbol{x}$ beyond the current best solution $y_{\max}$ can be defined as:

$$I_{\max}(\boldsymbol{x}) = \max\left(\hat{g}_n(\boldsymbol{x}) - y_{\max}, 0\right) = \begin{cases} \hat{g}_n(\boldsymbol{x}) - y_{\max}, \text{if } \hat{g}_n(\boldsymbol{x}) > y_{\max} \\ 0, \text{ otherwise} \end{cases}. \tag{3.19}$$

The EI for the maximum is analytically derived in closed form as follows:

$$EI_{\max}(\boldsymbol{x}) = \mathbb{E}\left[I_{\max}(\boldsymbol{x})\right] = (m_n(\boldsymbol{x}) - y_{\max})\Phi\left(\frac{m_n(\boldsymbol{x}) - y_{\max}}{\sigma_n(\boldsymbol{x})}\right) + \sigma_n(\boldsymbol{x})\varphi\left(\frac{m_n(\boldsymbol{x}) - y_{\max}}{\sigma_n(\boldsymbol{x})}\right). \tag{3.20}$$

However, by maximizing the $EI_{\max}$ function (the EI-MAX criterion), only one point for maximization is produced. In order to obtain a batch of q points, the the first point $\boldsymbol{x}_{\max}^{(n+1)}$ can be identified by $\boldsymbol{x}_{\max}^{(n+1)} = \arg\max_{\boldsymbol{x} \in \boldsymbol{x}^I} EI_{\max}(\boldsymbol{x})$. The following $q-1$ points should be sequentially selected by using a modified $EI_{\max}$ function. In analogy to the $PEI_{\min}$ function (Eq. (3.18)), we can define the $PEI_{\max}$ function for the q-th point such that:

$$PEI_{\max}(\boldsymbol{x}; \boldsymbol{x}_{\max}^{(n+1)}, \boldsymbol{x}_{\max}^{(n+2)}, \cdots, \boldsymbol{x}_{\max}^{(n+q-1)}) = EI_{\max}(\boldsymbol{x}) \times IF(\boldsymbol{x}; \boldsymbol{x}_{\max}^{(n+1)}, \boldsymbol{x}_{\max}^{(n+2)}, \cdots, \boldsymbol{x}_{\max}^{(n+q-1)}), \tag{3.21}$$

where the $IF(\cdot)$ function is defined in Eq. (3.17). The q-th point $\boldsymbol{x}_{\max}^{(n+q)}$ is obtained by:

$$\boldsymbol{x}_{\max}^{(n+q)} = \arg\max_{\boldsymbol{x} \in \boldsymbol{x}^I} PEI_{\max}(\boldsymbol{x}; \boldsymbol{x}_{\max}^{(n+1)}, \boldsymbol{x}_{\max}^{(n+2)}, \cdots, \boldsymbol{x}_{\max}^{(n+q-1)}).$$

3.3.2.3 Engine 3: PEI for both minimum and maximum

As we would like to infer both the minimum and maximum simultaneously, rather than in a sequential order, promising points for both extrema should be identified within one iteration until some predefined criteria are satisfied. Based on the PEI-MIN and PEI-MAX criteria, a infill sampling criterion for both minimizing and maximizing the g-function can be developed. This

criterion is denoted by PEI-MIN-MAX, and it is served as the third engine.

The proposed PEI-MIN-MAX criterion proceeds as follows. The first updating point is identified by $\boldsymbol{x}_{\min}^{(n+1)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} EI_{\min}(\boldsymbol{x})$, which is used for minimization. Likewise, the second one (the first point for maximization) is computed by maximizing the $PEI_{\max}(\boldsymbol{x})$ function, i.e., $\boldsymbol{x}_{\max}^{(n+2)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} PEI_{\max}(\boldsymbol{x}; \boldsymbol{x}_{\min}^{(n+1)})$. The third point (the second for minimization) is produced by maximizing the $PEI_{\min}$ function, i.e., $\boldsymbol{x}_{\min}^{(n+3)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} PEI_{\min}(\boldsymbol{x}; \boldsymbol{x}_{\min}^{(n+1)}, \boldsymbol{x}_{\max}^{(n+2)})$, and the fourth one (the second point for maximization) is determined by maximizing the $PEI_{\max}$ function, i.e., $\boldsymbol{x}_{\max}^{(n+4)} = PEI_{\max}(\boldsymbol{x}; \boldsymbol{x}_{\min}^{(n+1)}, \boldsymbol{x}_{\max}^{(n+2)}, \boldsymbol{x}_{\min}^{(n+3)})$. As the process goes on, a desired q (≥ 2) updating points can be obtained sequentially ahead of observing their g-function values. Note that one can also start the first point with $\boldsymbol{x}_{\max}^{(n+1)}$, and then generate a set of q points $(\boldsymbol{x}_{\max}^{(n+1)}, \boldsymbol{x}_{\min}^{(n+2)}, \boldsymbol{x}_{\max}^{(n+3)}, \cdots)$ following a similar procedure.

3.3.3 Proposed T-PBGO algorithm

Based on the GP model and T-PEI infill sampling criterion, we propose a T-PBGO algorithm for interval analysis. The numerical implementation procedure of the proposed T-PBGO algorithm, which is also illustrated in Fig. 3.1, includes the following main steps:

Step 1: Define the problem and initialize the optimization

Define the minimization and maximization problem to be solved in terms of its ojective function $g(\boldsymbol{x})$ and feasible region $\boldsymbol{x}^I$, as in Eqs. (3.6) and (3.7). Initialize the parameters of the proposed T-PBQO method, namely, the initial sample size n_0, and two thresholds $\varepsilon_{\min}$ and $\varepsilon_{\max}$. Details about these parameters and possible numerical values for them are discussed below.

Step 2: Generate an initial training dataset

Generate an initial set of n_0 samples using Latin hypercube sampling (LHS) over $\boldsymbol{x}^I$, denoted by a $n_0 \times d$ matrix $\boldsymbol{X} = \left\{\boldsymbol{x}^{(j)}\right\}_{j=1}^{n_0}$. Observations of the g-function at these points can be computed in parallel, which are denoted by a $n_0 \times 1$ vector $\boldsymbol{y} = \left\{y^{(j)}\right\}_{j=1}^{n_0}$ with $\boldsymbol{y}^{(j)} = g(\boldsymbol{x}^{(j)})$. The initial training dataset is defined as $\boldsymbol{\mathcal{D}} = \{\boldsymbol{X}, \boldsymbol{y}\}$. Set $n = n_0$.

As we seek to enlarge the training dataset sequentially, the initial size n_0 should not be chosen too large and it is usually set as 5-10.

Step 3: Construct a GP model for the g-function

Construct a GP model $\mathcal{GP}(m_n(\boldsymbol{x}), k_n(\boldsymbol{x}, \boldsymbol{x}'))$ for $y = g(\boldsymbol{x})$ based on the training dataset $\boldsymbol{\mathcal{D}}$. This step mainly consists of specifying the hyper-parameters by using the maximum likelihood estimation. All the numerical examples in this study are performed with the *fitrgp* function in Matlab Statistics and Machine Learning Toolbox.

Step 4: Check the predefined criteria and select the engine

In this stage, we first need to check whether the GP has achieved reasonable accuracy at both the minimum and maximum. If not, the GP should be then improved further, and this improvement means computing additional points. Thus, it should be clear what kind of additional points is still required, for minimization, maximization or both. Let $y_{\min} = \min_{1\leq j\leq n} y^{(j)}$ and $y_{\max} = \max_{1\leq j\leq n} y^{(j)}$ denote the minimum and maximum values of y observed so far, respectively. Compute the maxima of $EI_{\min}(\boldsymbol{x})$ and $EI_{\max}(\boldsymbol{x})$ by: $\delta y_1 = \max_{\boldsymbol{x}\in\boldsymbol{x}^I} EI_{\min}(\boldsymbol{x})$ and $\delta y_2 = \max_{\boldsymbol{x}\in\boldsymbol{x}^I} EI_{\max}(\boldsymbol{x})$. In this study, five criteria consisting in judging the ratios of the maximum expected improvements (i.e., δy_1 and δy_2) to the best current minimum and maximum (i.e., $y_{\min}$ and $y_{\max}$) respectively, are given as follows:

• ***Criterion 1 (Stopping criterion)***. If both $\frac{\delta y_1}{|y_{\min}|+\delta} < \varepsilon_{\min}$ and $\frac{\delta y_2}{|y_{\max}|+\delta} < \varepsilon_{\max}$ are satisfied for two successive iterations, go to **Step 7**; Else, check Criterion 2.

• ***Criterion 2***. If $\frac{\delta y_1}{|y_{\min}|+\delta} \geq \varepsilon_{\min}$ and $\frac{\delta y_2}{|y_{\max}|+\delta} \geq \varepsilon_{\max}$, this indicates that the GP could be still not accurate enough for estimating both the minimum and maximum and one should go to **Step 5c**; Else, check Criterion 3.

• ***Criterion 3***. If $\frac{\delta y_1}{|y_{\min}|+\delta} < \varepsilon_{\min}$ and $\frac{\delta y_2}{|y_{\max}|+\delta} < \varepsilon_{\max}$, this indicates that the GP could be still not accurate enough for both estimating the minimum and maximum (due to potential fake convergence) and one should go to **Step 5c**; Else, check Criterion 4.

• ***Criterion 4***. If $\frac{\delta y_1}{|y_{\min}|+\delta} \geq \varepsilon_{\min}$ and $\frac{\delta y_2}{|y_{\max}|+\delta} < \varepsilon_{\max}$, this indicates that the GP could be still not accurate for estimating the minimum one should go to **Step 5a**; Else, check Criterion 5.

• ***Criterion 5***. If $\frac{\delta y_1}{|y_{\min}|+\delta} < \varepsilon_{\min}$ and $\frac{\delta y_2}{|y_{\max}|+\delta} \geq \varepsilon_{\max}$, this indicates that the GP could be still not accurate enough for estimating the maximum one should go to **Step 5b**.

In Criteria 1-5, δ is a small number to ensure that the denominators are always greater than zero, which is specified as 10^{-6} in this study. It should be noted that these two quantities $\frac{\delta y_1}{|y_{\min}|+\delta}$ and $\frac{\delta y_2}{|y_{\max}|+\delta}$ play a pivotal role for our decision-making. The first one represents the ratio of maximum expected improvement for the minimum to the current absolute minimum, while the second one is the ratio of maximum expected improvement for the maximum to the current absolute maximum,

if δ is treated as zero. When the current GP model is relatively accurate for both the minimum and maximum, it is expected that these two ratios should be very small. Thus, it is appropriate to judge the convergence of the proposed method by monitoring these two ratios. According to our experience, $\varepsilon_{\min}$ and $\varepsilon_{\max}$ can be set in the order of 0.001.

Step 5a: Identify q updating points for minimization (Engine 1)

Identify q updating points for minimization by using the PEI-MIN criterion. The first point is selected by $\boldsymbol{x}_{\min}^{(n+1)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} EI_{\min}(\boldsymbol{x})$, the second one $\boldsymbol{x}_{\min}^{(n+2)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} PEI_{\min}(\boldsymbol{x}; \boldsymbol{x}_{\min}^{(n+1)})$, and the third one $\boldsymbol{x}_{\min}^{(n+3)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} PEI_{\min}(\boldsymbol{x}; \boldsymbol{x}_{\min}^{(n+1)}, \boldsymbol{x}_{\min}^{(n+2)})$, etc. The q updating points can be denoted by $\boldsymbol{X}_{\text{add}} = \left\{\boldsymbol{x}_{\min}^{(n+1)}, \boldsymbol{x}_{\min}^{(n+2)}, \cdots, \boldsymbol{x}_{\min}^{(n+q)}\right\}$. Then, go to **Step 6**.

Step 5b: Identify q updating points for maximization (Engine 2)

Identify q updating points for maximization by using the PEI-MAX criterion. The first point is selected by $\boldsymbol{x}_{\max}^{(n+1)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} EI_{\max}(\boldsymbol{x})$, the second one $\boldsymbol{x}_{\max}^{(n+2)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} PEI_{\max}(\boldsymbol{x}; \boldsymbol{x}_{\max}^{(n+1)})$, and the third one $\boldsymbol{x}_{\max}^{(n+3)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} PEI_{\max}(\boldsymbol{x}; \boldsymbol{x}_{\max}^{(n+1)}, \boldsymbol{x}_{\max}^{(n+2)})$, etc. The q updating points can be denoted by $\boldsymbol{X}_{\text{add}} = \left\{\boldsymbol{x}_{\max}^{(n+1)}, \boldsymbol{x}_{\max}^{(n+2)}, \cdots, \boldsymbol{x}_{\max}^{(n+q)}\right\}$. Then, go to **Step 6**.

Step 5c: Identify q updating points for both minimization and maximization (Engine 3)

Identify q updating points for both minimization and maximization by using the PEI-MIN-MAX criterion. The first point is selected by $\boldsymbol{x}_{\min}^{(n+1)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} EI_{\min}(\boldsymbol{x})$, the second one $\boldsymbol{x}_{\max}^{(n+2)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} PEI_{\max}(\boldsymbol{x}; \boldsymbol{x}_{\min}^{(n+1)})$, and the third one $\boldsymbol{x}_{\min}^{(n+3)} = \arg\max_{\boldsymbol{x}\in\boldsymbol{x}^I} PEI_{\min}(\boldsymbol{x}; \boldsymbol{x}_{\min}^{(n+1)}, \boldsymbol{x}_{\max}^{(n+2)})$, etc. The q updating points can be denoted by $\boldsymbol{X}_{\text{add}} = \left\{\boldsymbol{x}_{\min}^{(n+1)}, \boldsymbol{x}_{\max}^{(n+2)}, \cdots, \boldsymbol{x}_{\max}^{(n+q)}\right\}$. Then, go to **Step 6**.

Step 6: Enrich the training dataset

The q updating points $\boldsymbol{X}_{\text{add}}$ are evaluated on the g-function in parallel, and the corresponding observations are denoted by $\boldsymbol{y}_{\text{add}} = \left\{y^{(n+1)}, y^{(n+2)}, \cdots, y^{(n+q)}\right\}$. The training dataset $\boldsymbol{\mathcal{D}}$ is enriched by $\boldsymbol{\mathcal{D}}_{\text{add}} = \{\boldsymbol{X}_{\text{add}}, \boldsymbol{y}_{\text{add}}\}$, i.e., $\boldsymbol{\mathcal{D}} = \boldsymbol{\mathcal{D}} \cup \boldsymbol{\mathcal{D}}_{\text{add}}$. Set $n = n + q$ and then go to **Step 3**.

Step 7: Record results and end the algorithm

Record $y_{\min} = \min_{1\le j\le n} y^{(j)}$ and $y_{\max} = \max_{1\le j\le n} y^{(j)}$ as approximate solutions to the lower and upper bounds of y^I respectively, and end the algorithm.

In Steps 4 and 5a-5c, the involved optimization problems are solved by a nature-inspired global optimizer, called Teaching–learning-based optimization (TLBO) [42], as they are usually much more

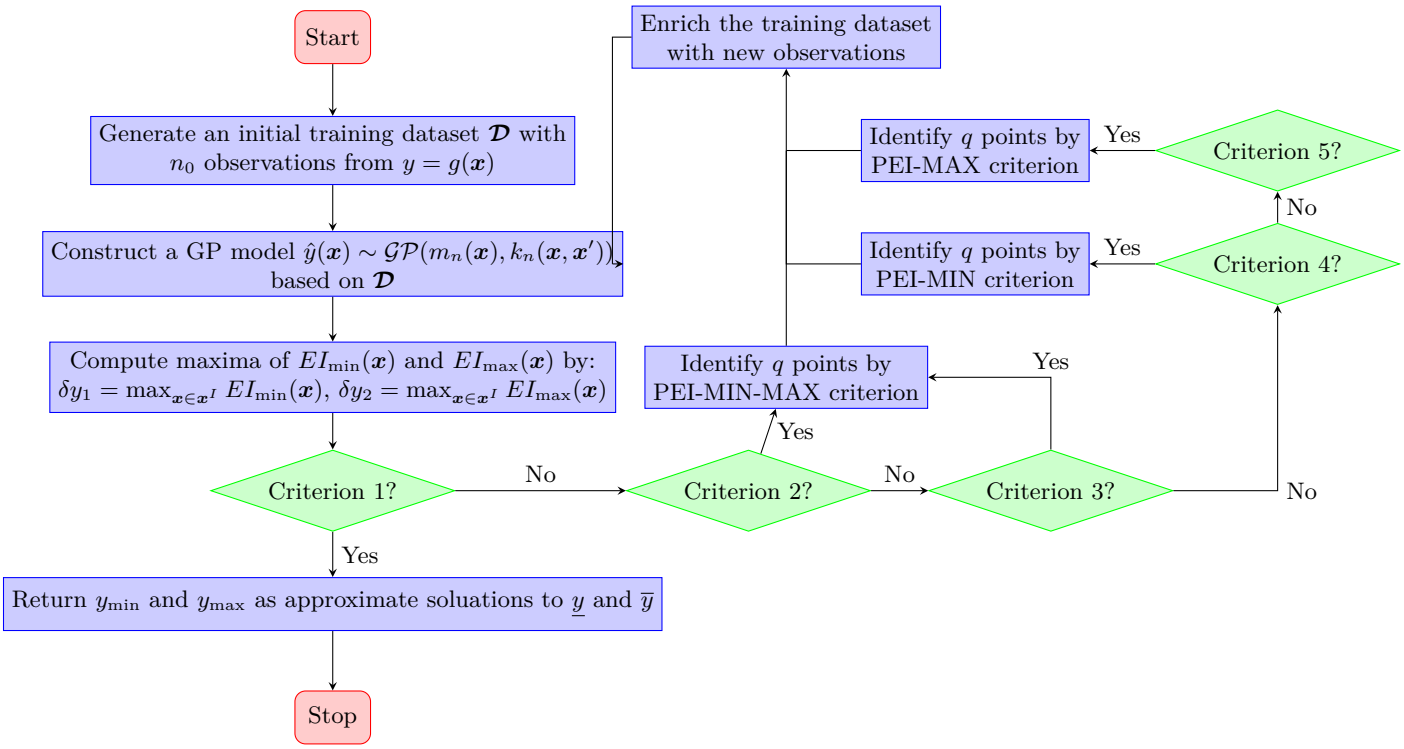


Figure 3.1: Flowchart of the proposed T-PBGO method.

cheaper compared to one call of the computational model. As the proposed method is rooted in the classical BGO method, its theoretical analysis may refer to, e,g., [43], which, however, is beyond the scope of the present study.

The proposed method has four major advantages. First, the technique often requires relatively few g-function evaluations. This is possible because one can incorporate prior knowledge to explore the design space. Second, our method allows a high-level parallelization as the proposed T-PEI criterion is computationally tractable for selecting multiple informative and diverse points. This feature further makes the method time-saving when parallel computing is available. Third, the proposed method is derivative-free and directly works with black-boxes, and thus is easy to implement and widely applicable (e.g., no matter the g-function is linear or non-linear and how large the supports of the input intervals are). Fourth, accurate approximate solutions to both lower and upper bounds of model response can be obtained with only one single run of the proposed algorithm.

3.3.4 Relationship to existing PBGO approaches

With the emergence of the classical BGO (originally called efficient global optimization) [32], there has been an growing interest to enable its capability of parallel processing. Representative works of PBGO include the q-EI criterion [44–46], multi-modal EI criterion [47, 48], PEI [41], Kriging Believer or Constant Liar strategy [45] and multiple surrogate assisted approach [49, 50], etc. The T-PEI criterion in the proposed T-PBGO method can be regarded as an improved PEI. The difference between the proposed method and the other PBGO methods is significant. The objective of the proposed method is to obtain both the minimum and maximum in one single run, while the other methods are only designed for minimum or maximum, not both.

3.4 Numerical examples

In order to illustrate and verify the proposed method, four numerical examples are studied in this section. These examples cover a wide range of types, from simple test problems to real-world applications. In all numerical examples, the proposed method is compared with several existing methods in terms of efficiency and accuracy. Besides, we propose a non-parallel BGO (N-PBGO) (given in Appendix 3.5) as a potential competitor for the proposed method, which is also conducted for comparison.

3.4.1 Example 1: A one-dimensional test function

The first example consists of a test function with one interval:

$$y = g(x) = (2x-1)^2 \sin\left(4\pi x - \frac{\pi}{8}\right),$$

where $x \in [0, 1]$. As can also be seen in Fig. 3.2, the g-function is multi-modal and has multiple maxima and minima.

The lower and upper bounds of y are computed by the analytical method, vertex method, genetic algorithm, N-PBGO and proposed T-PBGO method ($n_0 = 5$ and $\varepsilon_{\min} = \varepsilon_{\max} = 0.002$). The results are summarized in Table 3.1 together with the total number of function evaluations N, and the number of iterations $N^\star$. Although the vertex method outperforms the other numerical methods in terms of both N and $N^\star$, it produces totally wrong estimates for the response bounds.

The inaccuracy of the interval method is caused by its underlying assumption that y should be monotonic with respect to $\boldsymbol{x}$. As a representative of nature-inspired optimization algorithms, the genetic algorithm is able to yield accurate results, but at the expense of large computation cost. The N-PBGO method requires a relatively small number of function evaluations ($N = 16$), while still providing good results for both the lower and upper bounds. The N-PBGO method, however, is limited by its non-parallelism. On the contrary, the proposed T-PBGO method can overcome this limitation by taking advantage of the developed infill sampling criterion (i.e., T-PEI). Compared to N-PBGO, T-PBGO can significantly reduce the function evaluations in terms of $N^\star$, while still maintaining high accuracy. In addition, it also can be found that $N^\star$ gradually decreases with the increase of q, and remains the same when $q = 8, 10$, though N also increases non-monotonously.

Table 3.1: Interval analysis for Example 1 by different methods.

Method		Lower bound	Upper bound	N	$N^\star$	Reference
Exact solution		-0.7081	0.5197	-	-	-
Vertex method ($q = 2$)		-0.3827	-0.3827	2	1	[26]
Genetic algorithm ($q = 10$)		-0.7081	0.5197	$520 + 520 = 1040$	104	[51]
N-PBGO ($q = 1$)		-0.7081	0.5197	$5 + 6 + 5 = 16$	16	Appendix 3.5
	$q = 2$	-0.7081	0.5197	$5 + 8 = 13$	7	-
	$q = 4$	-0.7081	0.5197	$5 + 16 = 21$	6	-
Proposed T-PBGO	$q = 6$	-0.7081	0.5197	$5 + 24 = 29$	5	-
	$q = 8$	-0.7081	0.5197	$5 + 24 = 29$	4	-
	$q = 10$	-0.7081	0.5197	$5 + 30 = 35$	4	-

Note: N = the total number of g-function evaluations, and $N^\star$ = the number of iterations

To visually illustrate the proposed method, one special case is considered here (i.e., $q = 4$). It can be observed from Fig. 3.2 that the proposed method gradually approaches to the exact bounds as the iterative process goes on. Besides, these added points are more densely distributed around the global minimum and maximum, and thereby informative for our purposes.

3.4.2 Example 2: A two-dimensional test function

The second example takes a test function with two intervals [21]:

$$y = g(\boldsymbol{x}) = (1.5x_1 - 2)^2 - (x_2 - 3)^2 + x_1 x_2 + 10\sin(2\pi x_1) + 10\sin(2\pi x_2),$$

where $x_1, x_2 \in [2, 5]$. As shown in Fig. 3.3, the test function is highly nonlinear and has several local optima over the prescribed region.

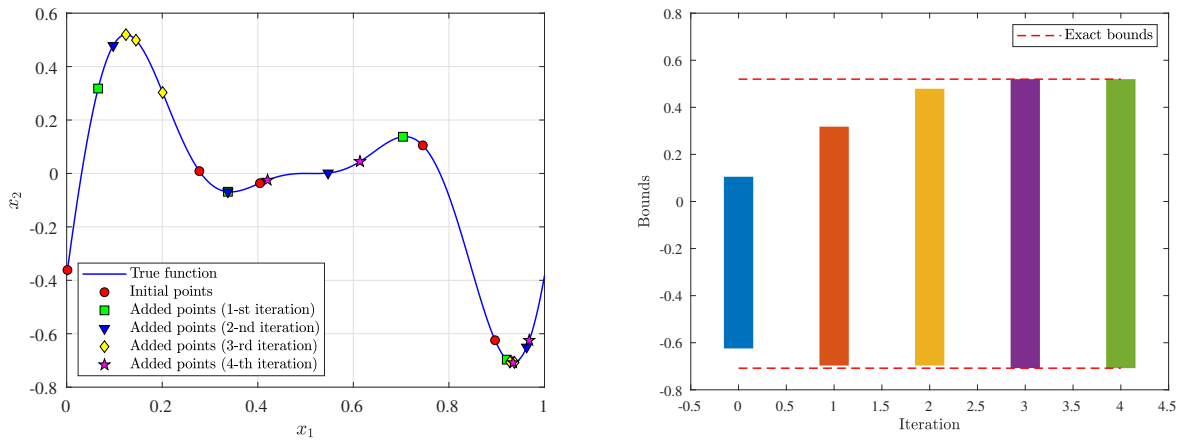


Figure 3.2: Illustration of the proposed method ($q = 4$) in Example 1: (a) True function, initial points and added points identified by T-PEI criterion; (b) Exact bounds and approximate bounds after each iteration.

The lower and upper bounds of y are computed by several methods, as listed in Table 3.2. The exact response bounds of y are obtained as -8.10 and 59.95. The genetic algorithm can yield accurate results, but at the expense of 4000 g-function evaluations. Although the classical vertex method requires the minimum number of g-function evaluations among all the numerical methods, it gives completely wrong estimates for the lower and upper bounds. At the cost of 6912 g-function calls (the largest among all the numerical methods), the subinterval method is able to produce acceptable results. The subinterval decomposition analysis method yields close results to these of the subinterval method, while requires significantly less g-function evaluations. For the N-PGBO method, fairly good results can be produced using a total number of 74 g-function evaluations, and 65 iterations. The proposed T-PBGO method ($n_0 = 10$, $\varepsilon_{\min} = 0.002$ and $\varepsilon_{\max} = 0.001$) is capable of generating quite accurate lower and upper bounds, while reducing the number of iterations down to 9 when $q = 8$.

3.4.3 Example 3: A transmission tower subjected to wind loads

This example consists of a transmission tower subjected to wind loads (shown in Fig. 3.4), which is modified from Ref. [52]. The tower is modelled as a three-dimensional (3D) truss structure with 24 joints and 80 elements in OpenSees. Three kinds of members, i.e., columns, diagonal members and horizontal members, are included in the model, the cross-sectional area of which are denoted

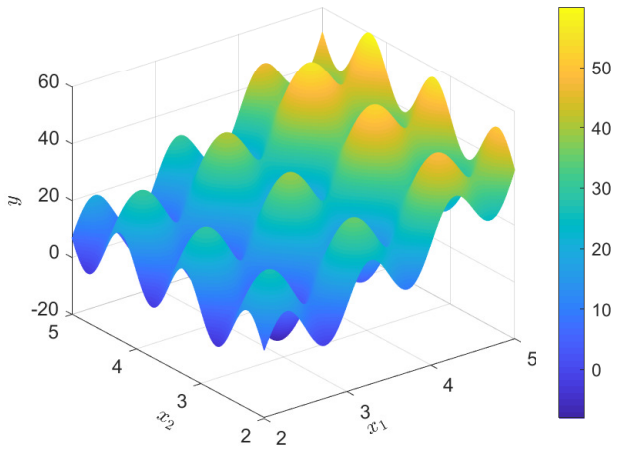


Figure 3.3: Plot of the two-dimensional test function in Example 2.

Table 3.2: Interval analysis results for Example 2 by different methods.

Method		Lower bound	Upper bound	N	$N^\star$	Reference
Exact solution		-8.10	59.95	-	-	-
Genetic algorithm		-8.10	59.95	4000	-	Tab. 7 in [21]
Vertex method ($q = 4$)		4.00	51.25	4	1	[26]
Subinterval method		-8.70	60.39	6912	-	Tab. 7 in [21]
Subinterval decomposition analysis		-8.55	58.81	97	-	Tab. 7 in [21]
N-PBGO ($q = 1$)		-8.01	59.92	$10 + 42 + 22 = 74$	65	Appendix 3.5
Proposed T-PBGO	$q = 2$	-8.08	59.94	$10 + 58 = 68$	30	-
	$q = 4$	-8.08	59.94	$10 + 72 = 82$	19	-
	$q = 6$	-8.09	59.93	$10 + 72 = 82$	13	-
	$q = 8$	-8.10	59.94	$10 + 80 = 90$	9	-
	$q = 10$	-8.10	59.93	$10 + 90 = 100$	10	-

as A_1, A_2 and A_3, respectively. The geometry of the model is shown in Fig. 3.4(a). The wind effect acting on the tower is simplified to four equivalent static loads at the top four nodes, and inclined by θ° relative to the x-axis (Fig. 3.4(b)). The constitutive law of the steel material adopts the bi-linear model, as depicted in Fig. 3.4(c). Eight interval variables are included in the 3D truss model, which are descried in Table 3.3. The response of interest is defined as the horizontal displacement of node A, i.e.,

$$y = g(P, \theta, F_y, E, b, A_1, A_2, A_3) = \sqrt{u_{\mathrm{A},x}^2 + u_{\mathrm{A},y}^2},$$

where $u_{\mathrm{A},x}$ and $u_{\mathrm{A},y}$ denote the displacements of node A in x and y directions, respectively.

The bounds of y are solved by several methods, and the results are summarized in Table 3.4. The particle swarm optimization (PSO) ($q = 10$) is used to provide reference results for the

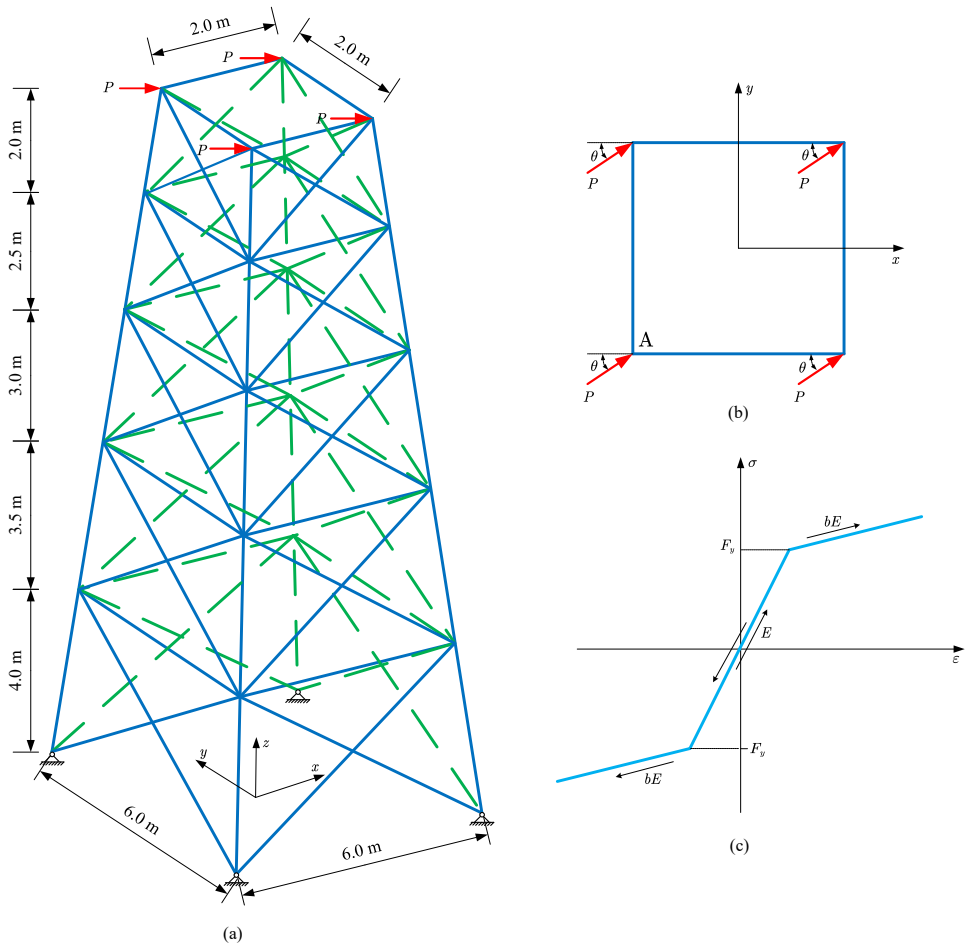


Figure 3.4: A transmission tower subject to wind loads: (a) 3D truss model; (b) loading at the top of tower; (c) bi-linear constitutive model.

Table 3.3: Interval variables for Example 3.

Variable	Description	Interval	Unit
P	Wind load	$[100, 200]$	kN
θ	Angle between the load direction and the x-axis	$[-45, 45]$	$^\circ$
F_y	Yield strength of steel	$[300, 400]$	MPa
E	Young's modulus of steel	$[1.8, 2.4] \times 10^5$	MPa
b	Strain hardening ratio	$[0.015, 0.025]$	-
A_1	Cross-sectional area of the column members	$[4000, 5000]$	mm^2
A_2	Cross-sectional area of the diagonal members	$[3000, 4000]$	mm^2
A_3	Cross-sectional area of the horizontal members	$[2000, 3000]$	mm^2

bounds. For the proposed T-PBGO method, we set the user-specified parameters as: $n_0 = 10$, $\varepsilon_{\min} = 0.002$ and $\varepsilon_{\max} = 0.001$. The vertex method requires 256 g-function calls, which, however, greatly underestimates the upper bound. Both N-PBGO and T-PBGO can give close results to

these of particle swarm optimization. The N-PBGO method is computationally advantageous in terms of N among all methods, while the proposed T-PBGO can further reduce $N^\star$ by taking advantage of its parallelism.

Table 3.4: Interval analysis results for Example 3 by different methods.

Method		Lower bound (mm)	Upper bound (mm)	N	$N^\star$	Reference
PSO ($q = 10$)		11.9592	57.2421	1920 + 3840 = 5760	576	[51]
Vertex method ($q = 10$)		11.9592	44.3887	256	25.60	[26]
N-PBGO ($q = 1$)		11.9592	57.2421	10 + 9 + 5 = 24	24	Appendix 3.5
	$q = 2$	11.9592	57.2403	10 + 22 = 32	16	-
	$q = 4$	11.9592	57.2421	10 + 28 = 38	10	-
Proposed T-PBGO	$q = 6$	11.9592	57.2372	10 + 36 = 46	8	-
	$q = 8$	11.9592	57.2421	10 + 40 = 50	7	-
	$q = 10$	11.9760	57.2388	10 + 60 = 70	7	-

3.4.4 Example 4: A spatial frame with viscous dampers subjected to earthquake

The last example considers a spatial frame with viscous dampers subjected to earthquake, as shown in Fig. 3.5. The 3-D finite element model is developed in OpenSees, the geometry of which can be found in Fig. 3.5(a). Each beam/column member is modelled with an elastic beam-column element with cross section IPE270/IPB300 (Fig. 3.5(b)/(c)). For each viscous damper (see Fig. 3.5(d)), a two-node link element is used with the viscous damper material. We only consider the self weight as the mass source for the columns, while for beams the mass source is determined based on "self weight + dead load + 0.2 live load". The structure is subjected to a base acceleration corresponding to the N-S component of the El-Centro 1940 earthquake, as shown in 3.5(e). The ground motion is applied along the direction with a rotation angle θ° with respect to the y-axis (Fig. 3.5(a)). As summarized in Table 3.5, eleven interval variables are involved in this example. Of interest is the maximum horizontal displacement of node A, i.e.,

$$y = g(\theta, AF, DL, LL, K_{\mathrm{D}}, C_{\mathrm{D}}, \alpha, \rho, E, v, \zeta) = \max_t \sqrt{u_{\mathrm{A},x}^2(t) + u_{\mathrm{A},y}^2(t)},$$

where $u_{\mathrm{A},x}(t)$ and $u_{\mathrm{A},y}(t)$ denote the displacements of node A in x and y directions, respectively.

The bounds of the model response y are computed by the particle swarm optimization, vertex method, N-PBGO and T-PBGO ($n_0 = 10$, $\varepsilon_{\min} = 0.002$ and $\varepsilon_{\max} = 0.001$), and the results are summarized in Table 3.6. The reference solution is taken from the particle swarm optimization

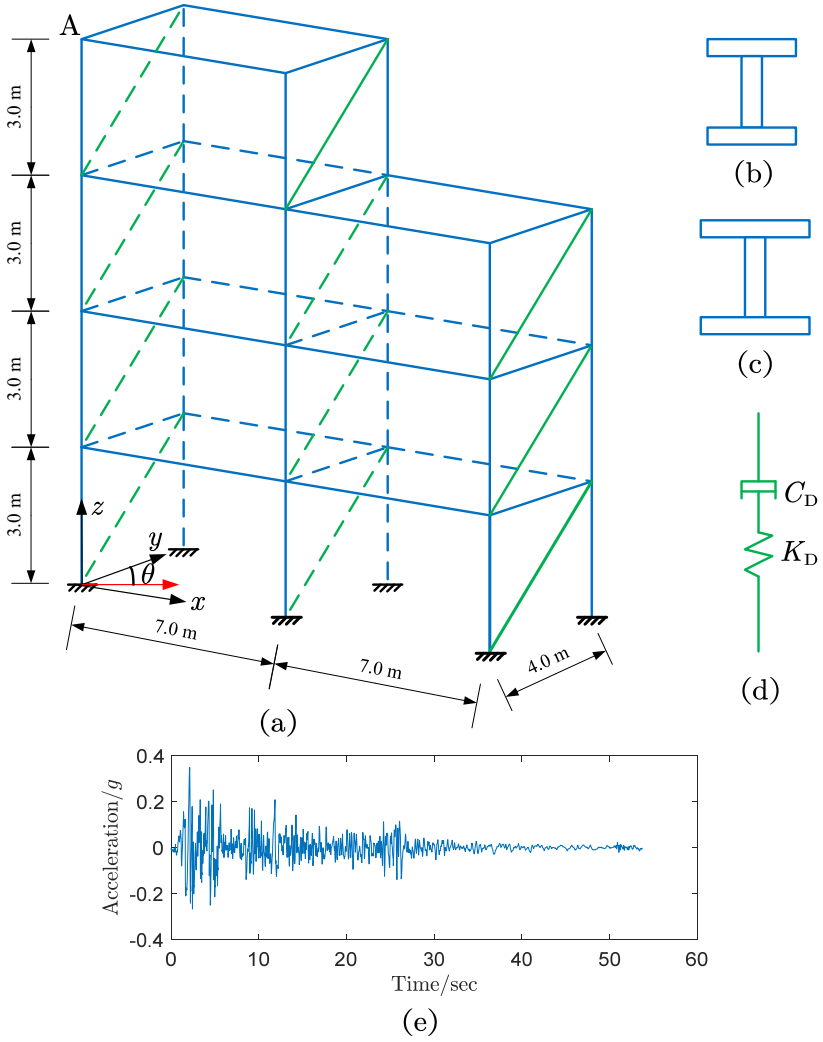


Figure 3.5: A spatial frame with viscous dampers subject to earthquake: (a) 3D frame model; (b) IPE270 for beams; (c) IPB300 for columns; (4) Viscous Damper; (e) N-S component of El Centro earthquake (1940)

method. The vertex method is able to produce good estimates, but requires a large number of g-function evaluations ($N = 2048$ and $N^\star = 204.8$) in this example. Compared to the N-PBGO method and vertex method, the proposed T-PBGO method can significantly reduce the number of g-function calls per core, though the total number of g-function calls may increase (e.g., $q = 4, 8$) relative to the N-PBGO method. Besides, the proposed method still gives desirable results for the response bounds. It should be emphasized that $N^\star$ does not decrease monotonically as q increases. This means that there may be an optimal parallelization level q that minimizes $N^\star$, e.g., $q = 6$ in the example.

Table 3.5: Interval variables for Example 4.

Variable	Description	Interval	Unit
θ	Angle between the earthquake direction and the y-axis	$[-45, 45]$	$^\circ$
AF	Amplification factor of the earthquake ground motion	$[0.5, 1.5]$	-
DL	Floor dead load	$[4, 5]$	kN/m^2
LL	Floor live load	$[2, 3]$	kN/m^2
K_{D}	Axial Stiffness of the viscous damper	$[3, 4] \times 10^4$	kN/m
C_{D}	Damping coefficient of the viscous damper	$[20, 30]$	kN(s/m)$^{\mathrm{ff}}$
α	Velocity exponent	$[0.2, 0.4]$	-
ρ	Density of steel	$[7.8, 7.9] \times 10^3$	kg/m^3
E	Young's modulus of steel	$[1.8, 2.2] \times 10^5$	MPa
v	Poisson's ratio	$[0.25, 0.30]$	-
ζ	Damping ratio	$[0.02, 0.04]$	-

Table 3.6: Interval analysis results for Example 4 by different methods.

Method		Lower bound (mm)	Upper bound (mm)	N	$N^\star$	Reference
PSO ($q = 10$)		11.9762	137.4651	3000 + 2400 = 5400	540	[51]
Vertex method ($q = 10$)		12.0084	137.4651	2048	204.8	[26]
N-PBGO ($q = 1$)		12.0929	137.3746	10 + 14 + 4 = 28	28	Appendix 3.5
Proposed T-PBGO	$q = 2$	12.0483	137.4651	10 + 14 = 24	12	-
	$q = 4$	12.0084	137.2062	10 + 24 = 34	9	-
	$q = 6$	12.0489	137.4651	10 + 18 = 28	5	-
	$q = 8$	12.0063	137.4651	10 + 32 = 42	6	-

3.4.5 Final remarks

In practical applications, the g-function can be rather expensive-to-evaluate and the computational budget is limited. In such cases, one may need to prespecify optimal values for the parameters n_0, q, $\varepsilon_{\min}$ and $\varepsilon_{\max}$ before running the proposed method in order to save the computational time, while remaining a desired level of accuracy. As a rule of thumb, the initial sample size n_0 can be set as 10. As observed in the four numerical examples, the number of iterations $N^\star$ does not decrease monotonically with q and takes its minimum value when $q = 8$ in most cases. Therefore, $q = 8$ is recommended in case that at least 8 cores are available. The two thresholds $\varepsilon_{\min}$ and $\varepsilon_{\max}$ not only influence the efficiency of the proposed method, but also the accuracy, The smaller $\varepsilon_{\min}$ and $\varepsilon_{\max}$ are, the proposed method usually requires more iterations and more accurate results can be obtained. According to our experience, $\varepsilon_{\min} = 0.002$ and $\varepsilon_{\max} = 0.002$ can be adopted.

3.5 Conclusions

In this study, a triple-engine parallel Bayesian global optimization (T-PBGO) method is proposed for efficient interval numerical analysis, especially when the computational model is a expensive-to-evaluate black box. The advancement of the proposed method lies in utilizing the Gaussian process (GP, also known as Kriging) prior for the expensive black-box g-function and an acquisition function (or infill sampling criterion) that can suggest promising points to be evaluated next. In order to make full use of prior knowledge and parallel computing, the main contribution of this paper is the development of a multi-points selection strategy, called 'triple-engine pseudo expected improvement' (T-PEI), which can select a batch of informative and diversity points for minimization and/or maximization at each iteration. Four numerical examples are investigated to demonstrate the proposed method. The main advantages of T-PBGO can be summarized as follows:

(i) The proposed method usually requires less g-function evaluations to achieve the same accuracy compared to non-Bayesian methods, due to its ability to exploit prior knowledge;

(ii) Compared to N-PBGO, T-PBGO allows for identifying multiple points at each iteration, and hence could be more efficient when parallel computing is available;

(iii) The developed method is non-intrusive in nature (directly works with black-box problems), and therefore easy-to-implement and broadly applicable;

(iv) Both lower and upper bounds can be obtained with one single run of the proposed method.

However, the proposed method still has several major limitations. First, T-PBGO works only well in low dimensions (typically, $d < 20$), and for high-dimensional problems new developments are needed. Second, as the parallelization level q and the size of training dataset increase, optimizing the T-PEI criterion can be time-consuming. Third, only the bounds of a single model response can be captured by the proposed method in its current form. Future works can be done along these directions.

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). Marcos Valdebenito acknowledges the support by ANID (National Agency for Research and Development, Chile) under its program FONDECYT, grant number 1180271. Chao Dang, Pengfei Wei and Michael Beer also would like to appreciate the support of Sino-German Mobility Program under grant number M-0175.

Appendix A: Non-parallel Bayesian global optimization

The traditional Bayesian global optimization is sequential in nature, which means that only one update point is identified at each iteration. Therefore, it cannot take advantage of parallelism. Besides, finding the minimum and maximum of a function is typically treated as two separate optimization problems. However, this is not advisable when computational efficiency is of great concern. That is because that the observations obtained when searching the minimum can be reused to speed up searching the maximum, and vice versa. This strategy is adopted in this study as a potential competitor to the proposed method, and we simply call it non-parallel Bayesian global optimization (N-PBGO). The main procedure of N-PBGO is summarized as follows:

Step A.1: Generate an initial training dataset

Generate an initial set of n_0 samples using LHS over $\boldsymbol{x}^I$, denoted by a $n_0 \times d$ matrix $\boldsymbol{X} = \left\{\boldsymbol{x}^{(j)}\right\}_{j=1}^{n_0}$. Observations of the g-function at these points can be computed in parallel, which are denoted by a $n_0 \times 1$ vector $\boldsymbol{y} = \left\{y^{(j)}\right\}_{j=1}^{n_0}$ with $\boldsymbol{y}^{(j)} = g(\boldsymbol{x}^{(j)})$. The initial training dataset can be written as $\boldsymbol{\mathcal{D}} = \{\boldsymbol{X}, \boldsymbol{y}\}$. Set $n = n_0$.

Step A.2: Construct a GP model

Construct a GP model $\mathcal{GP}(m_n(\boldsymbol{x}), k_n(\boldsymbol{x}, \boldsymbol{x}'))$ based on the initial training dataset $\boldsymbol{\mathcal{D}}$. This step mainly consists of choosing the hyper-parameters by using the maximum likelihood estimation. All the numerical examples in this study are performed with the *fitrgp* function in Matlab Statistics and Machine Learning Toolbox.

Step A.3: Compute maximum of $EI_{\mathbf{min}}(\boldsymbol{x})$

Let $y_{\min} = \min_{1 \le j \le n} y^{(j)}$ denote the minimum value of y observed so far, respectively. Compute the maximum of $EI_{\min}(\boldsymbol{x})$ by $\delta y_1 = \max_{\boldsymbol{x} \in \boldsymbol{x}^I} EI_{\min}(\boldsymbol{x})$.

Step A.4: Check stopping criterion for minimization

if $\frac{\delta y_1}{|y_{\min}|+\delta} < \varepsilon_{\min}$ is satisfied for two successive times, go to **Step A.7**; Otherwise, go to **Step A.5**.

Step A.5: Identify one point by EI-MIN criterion

Identify the next point to evaluate by $\boldsymbol{x}_{\min}^{(n+1)} = \arg\max_{\boldsymbol{x} \in \boldsymbol{x}^I} EI_{\min}(\boldsymbol{x})$.

Step A.6: Enrich the training dataset

Compute the corresponding g-function value at the identified point at $\boldsymbol{x}_{\min}^{(n+1)}$, i.e., $y^{(n+1)} = g(\boldsymbol{x}_{\min}^{(n+1)})$. Enrich the training dataset $\boldsymbol{\mathcal{D}}$ with $(\boldsymbol{x}_{\min}^{(n+1)}, y^{(n+1)})$. Set $n = n+1$, and go to **Step A.2**.

Step A.7: Compute maximum of $EI_{\max}(\boldsymbol{x})$

Let $y_{\max} = \max_{1 \le j \le n} y^{(j)}$ denote the maximum value of y observed so far, respectively. Compute the maxima of $EI_{\max}(\boldsymbol{x})$ by $\delta y_2 = \max_{\boldsymbol{x} \in \boldsymbol{x}^I} EI_{\max}(\boldsymbol{x})$.

Step A.8: Check stopping criterion for maximization

if $\frac{\mu_{\max}^{\max}}{|y_{\max}|+\delta} < \varepsilon_{\max}$ is satisfied for two successive times, go to **Step A.12**; Otherwise, go to **Step A.9**.

Step A.9: Identify one point by EI-MAX criterion

Identify the next point to evaluate by $\boldsymbol{x}_{\max}^{(n+1)} = \arg\max_{\boldsymbol{x} \in \boldsymbol{x}^I} EI_{\max}(\boldsymbol{x})$.

Step A.10: Enrich the training dataset

Compute the corresponding g-function value at the identified point at $\boldsymbol{x}_{\max}^{(n+1)}$, i.e., $y^{(n+1)} = g(\boldsymbol{x}_{\max}^{(n+1)})$. Enrich the training dataset $\boldsymbol{\mathcal{D}}$ with $(\boldsymbol{x}_{\max}^{(n+1)}, y^{(n+1)})$. Set $n = n+1$.

Step A.11: Construct a GP model

Construct a GP model $\mathcal{GP}(m_n(\boldsymbol{x}), k_n(\boldsymbol{x}, \boldsymbol{x}'))$ based on the initial training dataset $\boldsymbol{\mathcal{D}}$, and go to **Step A.7**.

Step A.12: End the algorithm

Take $y_{\min} = \min_{1 \le j \le n} y^{(j)}$ and $y_{\max} = \max_{1 \le j \le n} y^{(j)}$ as approximate solutions to the lower and upper bounds of y respectively, and end the algorithm.

In the above steps, TLBO is used for all optimization problems. Besides, for fair comparison the user-specified parameters (n_0, δ, $\varepsilon_{\min}$ and $\varepsilon_{\max}$) are set according to the proposed method in all numerical examples.

Bibliography

[1] Josef Stoer and Roland Bulirsch. *Introduction to numerical analysis*, volume 12. Springer Science & Business Media, 2013.

[2] Jon C Helton, Jay D Johnson, and William L Oberkampf. An exploration of alternative approaches to the representation of uncertainty in model predictions. *Reliability Engineering & System Safety*, 85(1-3):39–71, 2004.

[3] Michael Beer, Scott Ferson, and Vladik Kreinovich. Imprecise probabilities in engineering analyses. *Mechanical Systems and Signal Processing*, 37(1-2):4–29, 2013.

[4] Thomas Augustin, Frank PA Coolen, Gert De Cooman, and Matthias CM Troffaes. *Introduction to imprecise probabilities*. John Wiley & Sons, 2014.

[5] Matthias Faes and David Moens. Recent trends in the modeling and quantification of non-probabilistic uncertainty. *Archives of Computational Methods in Engineering*, 27(3):633–671, 2020.

[6] Chao Jiang, Xu Han, GY Lu, J Liu, Z Zhang, and YC Bai. Correlation analysis of non-probabilistic convex model and corresponding structural reliability technique. *Computer Methods in Applied Mechanics and Engineering*, 200(33-36):2528–2546, 2011.

[7] Ramon E Moore. *Methods and applications of interval analysis*. SIAM, 1979.

[8] C Jiang, QF Zhang, X Han, J Liu, and DA Hu. Multidimensional parallelepiped model—a new type of non-probabilistic convex model for structural uncertainty analysis. *International Journal for Numerical Methods in Engineering*, 103(1):31–59, 2015.

[9] BY Ni, C Jiang, and X Han. An improved multidimensional parallelepiped non-probabilistic model for structural uncertainty analysis. *Applied Mathematical Modelling*, 40(7-8):4727–4745, 2016.

[10] Chao Jiang, Chun-Ming Fu, Bing-Yu Ni, and Xu Han. Interval arithmetic operations for uncertainty analysis with correlated interval variables. *Acta Mechanica Sinica*, 32(4):743–752, 2016.

[11] Matthias Faes and David Moens. Multivariate dependent interval finite element analysis via convex hull pair constructions and the extended transformation method. *Computer Methods in Applied Mechanics and Engineering*, 347:85–102, 2019.

[12] Matthias Faes and David Moens. On auto-and cross-interdependence in interval field finite element analysis. *International Journal for Numerical Methods in Engineering*, 121(9):2033–2050, 2020.

[13] Di Wu and Wei Gao. Hybrid uncertain static analysis with random and interval fields. *Computer Methods in Applied Mechanics and Engineering*, 315:222–246, 2017.

[14] Alba Sofi, Eugenia Romeo, Olga Barrera, and Alan Cocks. An interval finite element method for the analysis of structures with spatially varying uncertainties. *Advances in Engineering Software*, 128:1–19, 2019.

[15] Isaac Elishakoff and Yohann Miglis. Novel parameterized intervals may lead to sharp bounds. *Mechanics Research Communications*, 44:1–8, 2012.

[16] I Elishakoff and K Thakkar. Overcoming overestimation characteristic to classical interval analysis. *AIAA Journal*, 52(9):2093–2097, 2014.

[17] G Manson. Calculating frequency response functions for uncertain systems using complex affine analysis. *Journal of Sound and Vibration*, 288(3):487–521, 2005.

[18] Giuseppe Muscolino and Alba Sofi. Stochastic analysis of structures with uncertain-but-bounded parameters via improved interval analysis. *Probabilistic Engineering Mechanics*, 28:152–163, 2012.

[19] Zhiping Qiu, Lihong Ma, and Xiaojun Wang. Non-probabilistic interval analysis method for dynamic response analysis of nonlinear systems with uncertainty. *Journal of Sound and Vibration*, 319(1-2):531–540, 2009.

[20] Zhongmin Deng, Zhaopu Guo, and Xuede Zhang. Non-probabilistic set-theoretic models for transient heat conduction of thermal protection systems with uncertain parameters. *Applied Thermal Engineering*, 95:10–17, 2016.

[21] CM Fu, LX Cao, JC Tang, and XY Long. A subinterval decomposition analysis method for uncertain structures with large uncertainty parameters. *Computers & Structures*, 197:58–69, 2018.

[22] MA Valdebenito, CA Pérez, HA Jensen, and M Beer. Approximate fuzzy analysis of linear structural systems applying intervening variables. *Computers & Structures*, 162:116–129, 2016.

[23] Marcos A Valdebenito, Héctor A Jensen, Pengfei Wei, Michael Beer, and André T Beck. Application of a reduced order model for fuzzy analysis of linear static systems. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering*, 7(2):020904, 2021.

[24] Jinglai Wu, Zhen Luo, Yunqing Zhang, Nong Zhang, and Liping Chen. Interval uncertain method for multibody mechanical systems using Chebyshev inclusion functions. *International Journal for Numerical Methods in Engineering*, 95(7):608–630, 2013.

[25] Jinglai Wu, Yunqing Zhang, Liping Chen, and Zhen Luo. A Chebyshev interval method for nonlinear dynamic systems under uncertainty. *Applied Mathematical Modelling*, 37(6):4578–4591, 2013.

[26] Weimin Dong and Haresh C Shah. Vertex method for computing functions of fuzzy variables. *Fuzzy sets and Systems*, 24(1):65–78, 1987.

[27] Zhiping Qiu, Yuying Xia, and Jialing Yang. The static displacement and the stress analysis of structures with bounded uncertainties using the vertex solution theorem. *Computer Methods in Applied Mechanics and Engineering*, 196(49-52):4965–4984, 2007.

[28] Robin RP Callens, Matthias GR Faes, and David Moens. Interval analysis using multilevel quasi-monte carlo. In *International Workshop on Reliable Engineering Computing (REC2021)*, volume 9, pages 53–67. International Workshop on Reliable Engineering Computing, 2021.

[29] Robin R.P. Callens, Matthias G.R. Faes, and David Moens. Multilevel quasi-monte carlo for interval analysis. *International Journal for Uncertainty Quantification*, Accepted for Publication.

[30] Fabio Biondini, Franco Bontempi, and Pier Giorgio Malerba. Fuzzy reliability analysis of concrete structures. *Computers & structures*, 82(13-14):1033–1052, 2004.

[31] Luciano Catallo. Genetic anti-optimization for reliability structural assessment of precast concrete structures. *Computers & Structures*, 82(13-14):1053–1065, 2004.

[32] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.

[33] Maarten De Munck, David Moens, Wim Desmet, and Dirk Vandepitte. An efficient response surface based optimisation method for non-deterministic harmonic and transient dynamic analysis. *Computer Modeling in Engineering & Sciences*, 47(2):119–166, 2009.

[34] Yisi Liu, Xiaojun Wang, Lei Wang, and Zheng Lv. A bayesian collocation method for static analysis of structures with unknown-but-bounded uncertainties. *Computer Methods in Applied Mechanics and Engineering*, 346:727–745, 2019.

[35] Hua-Ping Wan and Yi-Qing Ni. A new approach for interval dynamic analysis of train-bridge system based on bayesian optimization. *Journal of Engineering Mechanics*, 146(5):04020029, 2020.

[36] Alice Cicirello and Filippo Giunta. Machine learning based optimization for interval uncertainty propagation. *Mechanical Systems and Signal Processing*, 170:108619, 2022.

[37] Zhongmin Deng and Zhaopu Guo. Interval identification of structural parameters using interval overlap ratio and monte carlo simulation. *Advances in Engineering Software*, 121:120–130, 2018.

[38] Maurice Imholz, Matthias Faes, Dirk Vandepitte, and David Moens. Robust uncertainty quantification in structural dynamics under scarse experimental modal data: A bayesian-interval approach. *Journal of Sound and Vibration*, 467:114983, 2020.

[39] C Jiang, ZG Zhang, QF Zhang, Xue Han, HC Xie, and John Liu. A new nonlinear interval programming method for uncertain problems with dependent interval variables. *European Journal of Operational Research*, 238(1):245–253, 2014.

[40] Christopher K Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

[41] Dawei Zhan, Jiachang Qian, and Yuansheng Cheng. Pseudo expected improvement criterion for parallel EGO algorithm. *Journal of Global Optimization*, 68(3):641–662, 2017.

[42] R Venkata Rao, Vimal J Savsani, and DP Vakharia. Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. *Computer-Aided Design*, 43(3):303–315, 2011.

[43] A Scotto Di Perrotolo. A theoretical framework for bayesian optimization convergence. Master's thesis, KTH Royal Institute of Technology, 2018.

[44] Matthias Schonlau. *Computer experiments and global optimization*. PhD thesis, University of Waterloo, 1997.

[45] David Ginsbourger, Rodolphe Le Riche, and Laurent Carraro. Kriging is well-suited to parallelize optimization. In *Computational intelligence in expensive optimization problems*, pages 131–162. Springer, 2010.

[46] Clément Chevalier and David Ginsbourger. Fast computation of the multi-points expected improvement with applications in batch selection. In *International Conference on Learning and Intelligent Optimization*, pages 59–69. Springer, 2013.

[47] András Sóbester, Stephen J Leary, and Andy J Keane. A parallel updating scheme for approximating and optimizing high fidelity computer simulations. *Structural and multidisciplinary optimization*, 27(5):371–383, 2004.

[48] Dawei Zhan, Jiachang Qian, and Yuansheng Cheng. Balancing global and local search in parallel efficient global optimization algorithms. *Journal of Global Optimization*, 67(4):873–892, 2017.

[49] Felipe AC Viana, Raphael T Haftka, and Layne T Watson. Efficient global optimization algorithm assisted by multiple surrogate techniques. *Journal of Global Optimization*, 56(2):669–689, 2013.

[50] José Carlos García-García, Ricardo García-Ródenas, and Esteve Codina. A surrogate-based cooperative optimization framework for computationally expensive black-box problems. *Optimization and Engineering*, 21(3):1053–1093, 2020.

[51] *MATLAB and Global Optimization Toolbox Release 2018a*. The MathWorks Inc., Natick, Massachusetts, United States, 2018.

[52] Quan Gu, Michele Barbato, Joel P Conte, Philip E Gill, and Frank McKenna. OpenSees-SNOPT framework for finite-element-based optimization of structural and geotechnical systems. *Journal of Structural Engineering*, 138(6):822–834, 2012.

Chapter 4

Hybrid uncertainty propagation by Bayesian active learning

Bayesian probabilistic propagation of hybrid uncertainties: Estimation of response expectation function, its variable importance and bounds

Chao Dang[a], Pengfei Wei[b,⋆],Matthias G.R. Faes[c], Michael Beer[a,d,e]

[a]*Institute for Risk and Reliability, Leibniz University Hannover, Callinstr. 34, Hannover 30167, Germany*

[b]*School of Power and Energy, Northwestern Polytechnical University, Xi'an 710072, PR China*

[c]*Chair for Reliability Engineering, TU Dortmund University, Leonhard-Euler-Str. 5, Dortmund 44227, Germany*

[d]*Institute for Risk and Uncertainty, University of Liverpool, Liverpool L69 7ZF, United Kingdom*

[e]*International Joint Research Center for Resilient Infrastructure & International Joint Research Center for Engineering Reliability and Stochastic Mechanics, Tongji University, Shanghai 20 0 092, PR China*

Published in Computers & Structures in October 2022

Abstract: Uncertainties existing in physical and engineering systems can be characterized by different kinds of mathematical models according to their respective features. However, efficient propagation of hybrid uncertainties via an expensive-to-evaluate computer simulator is still a computationally challenging task. In this contribution, estimation of response expectation function (REF), its variable importance and bounds under hybrid uncertainties in the form of precise probability models, parameterized probability-box models and interval models is investigated through a Bayesian approach. Specifically, a new method, termed "Parallel Bayesian Quadrature Optimization" (PBQO), is developed. The method starts by treating the REF estimation as a Bayesian probabilistic integration (BPI) problem with a Gaussian process (GP) prior, which in turn implies a GP posterior for the REF. Then, one acquisition function originally developed in BPI and other two in Bayesian global optimization are introduced for Bayesian experimental designs. Besides, an innovative strategy is also proposed to realize multi-point selection at each iteration. Overall, a novel advantage of PBQO is that it is capable of yielding the REF, its variable importance and bounds simultaneously via a pure single-loop procedure allowing for parallel computing. Three numerical examples are studied to demonstrate the performance of the proposed method over some

*Corresponding Author
E-mail address: pengfeiwei@nwpu.edu.cn (P. Wei)

existing methods.

Keywords: Hybrid uncertainties, Response expectation function, Bayesian probabilistic integration, Bayesian global optimization, Bayesian experimental design, Parallel computing

4.1 Introduction

Uncertainty quantification (UQ) is a hot topic and even research frontier in a broad range of modern science and engineering fields. UQ is primarily aimed at the quantitative characterization and consequent reduction of uncertainties in both physical and engineering systems. Uncertainties occur when all or some aspects of the system under consideration are not exactly known. Examples of such aspects include, e.g., system parameters and operating conditions. These uncertainties generally originate from a variety of sources such as inherent variation, manufacturing error, modelling assumptions or a combination hereof. In terms of the origin of uncertainties, they are typically classified into either aleatory or epistemic types [1, 2]. Aleatory uncertainty refers to the uncertainty due to the intrinsic randomness or variability, and thus is irreducible in nature. As such, aleatory uncertainty is an inherent property of the system under consideration. Epistemic uncertainty, on the other hand, is associated with a lack of knowledge (or information) on the side of the analysts, and hence can be potentially reduced or even eliminated by acquiring more knowledge. Commonly, these two types of uncertainties occur together in both science and engineering, and many different uncertainty models might appear simultaneously in just one single problem. In addition to characterizing these uncertainties with appropriate mathematical models, uncertainty propagation through a computational model has also been of central interest from both academia and industry.

Many approaches have been indeed developed to quantitatively describe uncertain phenomena, which can be broadly categorized into three major groups: probabilistic approach, non-probabilistic approach and imprecise probability approach. The probabilistic approach is rooted in classical probability theory, and is the most traditional way to quantify uncertainties. Following this approach, non-determinism is modelled by a precise probability distribution on the basis of a set of probability axioms [3]. Despite its rigor in theory and popularity in practical applications, it is often criticized that the probabilistic approach indispensably relies on very fine information, e.g., a large amount of high-quality data, which is not always available. Alternatively, the non-probabilistic approach, including interval models [4], fuzzy sets [5] and convex models [6], is emerging for characterizing

uncertainty with limited information, where the variation bounds need to be specified, instead of a precise probability distribution. However, it is argued that these methods are mostly suitable to deal with epistemic uncertainty. In recent years, the imprecise probability approach has gained increasing attention as a promising framework to quantify complex uncertainties, particularly when the available information or data is not sufficient to identify a unique probability distribution [7]. In essence, it is an extension to classical probability theory where the uncertainty is characterised by a set of probability measures, rather than a single one. Therefore, it allows for modelling both aleatory uncertainty and epistemic uncertainty separately within a uniform framework. Typically, the aleatory uncertainty is characterized by the traditional probabilistic models, and the epistemic uncertainty is handled by the non-probabilistic models. Representative techniques include the probability box (p-box) [8], evidence theory [9] and fuzzy probability [10] among others.

As for uncertainty propagation, great efforts have been made along each line of uncertainty characterization over the past several decades. The existing approaches for propagating precise probabilistic uncertainty can be roughly divided into five categories: stochastic simulation methods [11–13], approximate analytical methods [14, 15], surrogate-assisted methods [16–18], numerical integration methods [19–23] and probability conservation-based methods [24, 25]. Differently, the propagation of non-probabilistic uncertainty follows another district philosophy, more relaying on, e.g., interval arithmetic [26], optimization methods [27, 28], perturbation methods [29, 30] and etc. Also advanced sampling approaches for interval analysis have been introduced [31, 32]. One can refer to [5] for a good review on recent trends in propagation of non-probabilistic uncertainty. For imprecise probability propagation, however, the above two kinds of methods are not suitable, and hence new developments are necessary. The most common way to address the problem involves a double-loop procedure that uses the aforementioned two types of methods in a nested way, such as optimized parameter sampling [33] and interval Monte Carlo simulation [34], which often suffers from a heavy computational burden. To improve the computational efficiency, decoupled strategies have recently attracted increasing attention, and representative works include the augmented subset simulation [35], non-intrusive imprecise stochastic simulation [36, 37], operator norm theory [38], active learning augmented probabilistic integration [39], non-intrusive imprecise probabilistic integration (NIPI) [40], and collaborative and adaptive Bayesian optimization (CABO) [41]. For a review of the computation methods for propagating p-boxes, the reader is referred to [42]. Besides, some progress has also been made in the context of hybrid uncertainty propagation, e.g., surrogate

modelling-based methods [43–48], stochastic simulation-based methods [49–51] and others [52, 53]. For propagating probabilistic-interval hybrid uncertainty, one can refer to the review [54]. Overall, propagation of hybrid uncertainties poses a more significant computational challenge in UQ community, and the existing methodologies are far from desirable for general practical applications.

In this paper, a novel method is presented to propagate hybrid uncertainties in the form of precise probabilistic models, parameterized p-box models and interval models, where the response expectation function (REF), its variable importance and bounds are of concern. The method belongs to the class of Bayesian probabilistic numerical methods [55], and can also be seen as an important extension to the NIPI [40] and CABO [41] methods originally developed for propagating parameterized p-box models. The main contributions of the present work can be summarized as follows:

- A general Bayesian framework is presented for propagating hybrid uncertainties, which is non-intrusive and fully decoupled in nature;
- Posterior means and variances of the REF and its random-sampling high-dimensional model representation (RS-HDMR) decomposition are analytically derived in closed form;
- Parallelized Bayesian experiment design is realized so as to take advantage of parallel computing at each iteration;
- A Matlab implementation of our methodology is freely available to the public [1].

The remaining of this paper is organized as follows. We start by stating the problem to be solved in this study in Section 4.2. Section 4.3 presents the theoretical basis and numerical implementation procedure of the proposed method, with the relationship to the existing NIPI and CABO methods being discussed. How to extend the proposed method to a relatively more general case of hybrid uncertainties is briefly explained in Section 4.4. In Section 4.5, three numerical examples are studied to demonstrate the proposed method. The paper ends with some concluding remarks and perspectives in Section 4.6.

[1] `https://github.com/Chao-Dang/Hybrid-Uncertainty-Propagation-by-Parallel-Bayesian-Quadrature-Optimization`

4.2 Problem statement

In this work, three kinds of uncertainty characterization models are considered to model non-deterministic inputs of a computer simulator, i.e., precise probability models, parameterized p-box models and interval models. The precise probability models that are deeply rooted in probability theory are assumed to be used for describing pure aleatory uncertainty. As a representative of imprecise probabilities, the parameterized p-box models are able to account for both aleatory uncertainty and epistemic uncertainty simultaneously. The interval models serve as a representative of non-probabilistic models and are useful to model the constant-but-unknown epistemic uncertainty. As such, the developed method is expected to work in the following four cases:

Case I: Precise probabilistic models and parameterized p-box models coexist in the model inputs;

Case II: Only parameterized p-box models exist in the model inputs;

Case III: Precise probabilistic models and interval models coexist in the model inputs;

Case IV: Precise probabilistic models, parameterized p-box models and interval models coexist in the model inputs.

Among the four cases, **Case IV** constitutes a more general situation of hybrid uncertainties. For notational clarity, however, we only take **Case III** as an example to illustrate the proposed method in the following, and when it comes to the general case (i.e., **Case IV**) one can refer to Section 4.4. Let $\boldsymbol{X} = [X_1, X_2, \ldots, X_{d_1}] \in \mathscr{X} \subseteq \mathbb{R}^{d_1}$ and $\boldsymbol{A} = [A_1, A_2, \ldots, A_{d_2}] \in \mathscr{A} \subseteq \mathbb{R}^{d_2}$ denote a d_1-dimensional vector of precise random variables and a d_2-dimensional vector of interval variables, respectively. The random variables are said to be 'precise' when their distribution types and distribution parameters are exactly known, and we assume that the joint probability density function (PDF) of $\boldsymbol{X}$ exists, denoted as $f_{\boldsymbol{X}}(\boldsymbol{x})$. The interval variables refer to the uncertain parameters with limited information, and can only be specified by their lower and upper bounds, i.e., $\boldsymbol{A} = [\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}]$, where $\underline{\boldsymbol{\alpha}} = [\underline{\alpha}_1, \underline{\alpha}_2, \ldots, \underline{\alpha}_{d_2}]$ and $\overline{\boldsymbol{\alpha}} = [\overline{\alpha}_1, \overline{\alpha}_2, \cdots, \overline{\alpha}_{d_2}]$. As such, $\boldsymbol{A}$ represents a d_2-dimensional hyper-rectangle. In this study, these $d_1 + d_2$ variables are assumed to be independent just for the convenience of describing our method. The computer simulator is represented by a deterministic, continuous and real-valued function $g : \mathbb{R}^{d_1+d_2} \mapsto \mathbb{R}, \{x, \alpha\} \to z$, with $Z = g(\boldsymbol{X}, \boldsymbol{A})$ being a scalar quantity of interest. Due to the existence of interval variables, Z is no longer a random variable unless $\boldsymbol{A}$ is fixed at a value $\boldsymbol{\alpha} \in \boldsymbol{A}$. Thus, the expectation of Z, is not a deterministic values

anymore, but function of the interval variables. More precisely, it only assume a crisp value for a realisation of the input intervals. To formalize, the definition of the so-called REF is given as follows:

$$m(\boldsymbol{\alpha}) = \int_{\mathscr{X}} g(\boldsymbol{x}, \boldsymbol{\alpha}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{4.1}$$

The lower and upper bounds of $m(\boldsymbol{\alpha})$ can be defined as:

$$m_l = \min_{\boldsymbol{\alpha} \in [\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}]} m(\boldsymbol{\alpha}) = \min_{\boldsymbol{\alpha} \in [\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}]} \int_{\mathscr{X}} g(\boldsymbol{x}, \boldsymbol{\alpha}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{4.2}$$

$$m_u = \max_{\boldsymbol{\alpha} \in [\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}]} m(\boldsymbol{\alpha}) = \max_{\boldsymbol{\alpha} \in [\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}]} \int_{\mathscr{X}} g(\boldsymbol{x}, \boldsymbol{\alpha}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}. \tag{4.3}$$

The REF can provide complete information about how the response expectation changes with its argument $\boldsymbol{\alpha}$, whereas the interval $[m_l, m_u]$ measures the amount of epistemic uncertainty present in the response expectation. Besides, the analyst may also concern the variable importance of the REF. Intuitively, the bounds and variable importance analysis of the REF can be proceeded straightforwardly once the REF is available. However, it is still a non-trivial task to compute the REF in an efficient manner since each evaluation of the response function $g(\boldsymbol{x}, \boldsymbol{\alpha})$ can be prohibitively expensive for a real-world problem.

4.3 Parallel Bayesian quadrature optimization

As the REF defined in Eq. (4.1) is given in the form of an integral, the Bayesian probabilistic integration (BPI) [23] can be applied to efficiently obtain an estimate for the REF. If we assign a Gaussian process (GP) prior for the integrand $g(\boldsymbol{x}, \boldsymbol{\alpha})$, the induced posterior of the REF is also a GP. Following this, the lower and upper bounds defined in Eqs. (4.2) and (4.3) may be further solved by the Bayesian global optimization (BGO) [56]. In this section, a novel Bayesian approach combining the BPI and BGO, called Parallel Bayesian Quadrature Optimization (PBQO), is presented to produce the REF, its variable importance and bounds simultaneously in an efficient manner.

4.3.1 Variable transformation

Before introducing our method, a pre-processing step should be performed to transform the original input variable vector $\{\boldsymbol{X}, \boldsymbol{A}\}$ to a new one so as to make the proposed method analytically

tractable. In this study, the random variable vector $\boldsymbol{X}$ is transformed to be a standard normal one by a certain transformation (e.g., isoprobabilistic transformation), which is denoted as $\boldsymbol{U} = T_1(\boldsymbol{X})$. In contrast, we consider transforming the interval vector $\boldsymbol{A}$ to be a standard one (i.e., $[0,1]^{d_2}$) by a simple linear transformation such that $\boldsymbol{V} = T_2(\boldsymbol{A})$. For convenience, the two transformations can be written in a uniform form $\boldsymbol{W} = T(\boldsymbol{X}, \boldsymbol{A})$, where $\boldsymbol{W} = \{\boldsymbol{U}, \boldsymbol{V}\}$. The REF with respect to $\boldsymbol{v}$ is defined as:

$$\mathcal{M}(\boldsymbol{v}) = \int_{\mathscr{U}} \mathcal{G}(\boldsymbol{w}) f_{\boldsymbol{U}}(\boldsymbol{u}) \mathrm{d}\boldsymbol{u}, \tag{4.4}$$

where $\mathcal{G}(\boldsymbol{w}) = g(T(\boldsymbol{x}, \boldsymbol{\alpha}))$, $f_{\boldsymbol{U}}(\boldsymbol{u})$ is the joint PDF of $\boldsymbol{U}$. Once $\mathcal{M}(\boldsymbol{v})$ is available, $m(\boldsymbol{\alpha})$ can be easily obtained as $m(\boldsymbol{\alpha}) = \mathcal{M}(T_2(\boldsymbol{\alpha}))$. Note that the T_1 transformation is necessary for the analytical tractability of the proposed method, while T_2 transformation is not. However, we introduce the T_2 transformation only for the purpose of producing concise analytical expressions.

4.3.2 Prior Gaussian process

In the proposed PBQO method, we first place a GP prior over the space $\mathscr{G}$ of functions: $\mathcal{G} : \mathcal{W} \to \mathbb{R}$, denoted as $\hat{\mathcal{G}}(\boldsymbol{w}) \sim \mathcal{GP}(\mu_0(\boldsymbol{w}), k_0(\boldsymbol{w}, \boldsymbol{w}'))$, where $\mu_0(\boldsymbol{w})$ and $k_0(\boldsymbol{w}, \boldsymbol{w}')$ are the prior mean and covariance functions, respectively. The prior mean function reflects the general trend of the GP, and can be assumed to be, e.g., zero, constant or a linear polynomial. The covariance function is a more crucial ingredient of the GP since it encodes our basic assumptions about the function to be inferred, e.g., smoothness and periodicity. In this study, the prior mean function adopts a constant, i.e., $\mu_0(\boldsymbol{w}) = \beta$, and the prior covariance function takes the squared exponential kernel:

$$\begin{aligned} k_0(\boldsymbol{w}, \boldsymbol{w}') =& s_0^2 \exp\left[-\frac{1}{2}\left(\boldsymbol{w} - \boldsymbol{w}'\right)\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{w} - \boldsymbol{w}'\right)^{\mathrm{T}}\right] \\ =& s_0^2 \exp\left[-\frac{1}{2}\left(\boldsymbol{u} - \boldsymbol{u}'\right)\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1}\left(\boldsymbol{u} - \boldsymbol{u}'\right)^{\mathrm{T}}\right] \exp\left[-\frac{1}{2}\left(\boldsymbol{v} - \boldsymbol{v}'\right)\boldsymbol{\Sigma}_{\boldsymbol{v}}^{-1}\left(\boldsymbol{v} - \boldsymbol{v}'\right)^{\mathrm{T}}\right], \end{aligned} \tag{4.5}$$

where s_0^2 is the process variance, $\boldsymbol{\Sigma} = \mathrm{diag}\left\{l_1^2, l_2^2, \cdots, l_{d_1+d_2}^2\right\}$ with l_i being the characteristic lengthscale in i-th dimension, $\boldsymbol{\Sigma}_{\boldsymbol{u}} = \mathrm{diag}\left\{l_1^2, l_2^2, \cdots, l_{d_1}^2\right\}$ and $\boldsymbol{\Sigma}_{\boldsymbol{v}} = \mathrm{diag}\left\{l_{d_1+1}^2, l_{d_1+2}^2, \cdots, l_{d_1+d_2}^2\right\}$; Throughout the paper, the symbol $\mathrm{diag}\{\cdot\}$ means to create a square diagonal matrix with the elements of its argument when its argument is a vector or to get a column vector of the diagonal elements of its argument when its argument is a matrix. The parameters β, s_0, $l_1, l_2, \cdots, l_{d_1+d_2}$

are called hyperparameters. Note that the analytical tractability of the proposed method relies on using the squared exponential kernel.

4.3.3 Bayesian posterior inference

Suppose that we have evaluated the $\mathcal{G}$-function at n points. Let an $n \times (d_1 + d_2)$ matrix $\boldsymbol{\mathcal{W}} = (\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) = \left\{\boldsymbol{w}^{(j)}\right\}_{j=1}^{n}$ denote the n points at which the $\mathcal{G}$-function are evaluated, and an $n \times 1$ vector $\boldsymbol{\mathcal{Z}} = \left\{z^{(j)}\right\}_{j=1}^{n}$ denote the corresponding $\mathcal{G}$-function values at $\boldsymbol{\mathcal{W}}$. Given $\boldsymbol{\mathcal{D}} = \{\boldsymbol{\mathcal{W}}, \boldsymbol{\mathcal{Z}}\}$, the hyperparameters involved in the prior mean and covariance functions can be determined by, e.g., maximum likelihood estimation [57]. Besides, conditioning on the data $\boldsymbol{\mathcal{D}}$, we can arrive at a posterior GP over functions $\mathcal{G} \in \mathscr{G}$, which is denoted as $\mathcal{GP}(\mu_n(\boldsymbol{w}), k_n(\boldsymbol{w}, \boldsymbol{w}'))$. According to [57], the posterior mean $\mu_n(\boldsymbol{w})$ and posterior covariance function $k_n(\boldsymbol{w}, \boldsymbol{w}')$ can be given by:

$$\mu_n(\boldsymbol{w}) = \mu_0(\boldsymbol{w}) + \boldsymbol{k}_0(\boldsymbol{w}, \boldsymbol{\mathcal{W}})^{\mathrm{T}} \boldsymbol{K}_0^{-1} (\boldsymbol{\mathcal{Z}} - \boldsymbol{\mu}_0(\boldsymbol{\mathcal{W}})), \tag{4.6}$$

$$k_n(\boldsymbol{w}, \boldsymbol{w}') = k_0(\boldsymbol{w}, \boldsymbol{w}') - \boldsymbol{k}_0(\boldsymbol{w}, \boldsymbol{\mathcal{W}})^{\mathrm{T}} \boldsymbol{K}_0^{-1} \boldsymbol{k}_0(\boldsymbol{w}', \boldsymbol{\mathcal{W}}), \tag{4.7}$$

where $\boldsymbol{\mu}_0(\boldsymbol{\mathcal{W}}) = [\mu_0(\boldsymbol{w}^{(1)}), \mu_0(\boldsymbol{w}^{(2)}), \cdots, \mu_0(\boldsymbol{w}^{(n)})]^{\mathrm{T}}$ is the mean vector at $\boldsymbol{\mathcal{W}}$; $\boldsymbol{k}_0(\boldsymbol{w}, \boldsymbol{\mathcal{W}}) = [k_0\left(\boldsymbol{w}, \boldsymbol{w}^{(1)}\right), k_0\left(\boldsymbol{w}, \boldsymbol{w}^{(2)}\right), \cdots, k_0\left(\boldsymbol{w}, \boldsymbol{w}^{(n)}\right)]^{\mathrm{T}}$ is the covariance vector between $\boldsymbol{w}$ and $\boldsymbol{\mathcal{W}}$; $\boldsymbol{k}_0(\boldsymbol{w}', \boldsymbol{\mathcal{W}}) = [k_0\left(\boldsymbol{w}', \boldsymbol{w}^{(1)}\right), k_0\left(\boldsymbol{w}', \boldsymbol{w}^{(2)}\right), \cdots, k_0\left(\boldsymbol{w}', \boldsymbol{w}^{(n)}\right)]^{\mathrm{T}}$ is the covariance vector between $\boldsymbol{w}'$ and $\boldsymbol{\mathcal{W}}$; $\boldsymbol{K}_0$ is the covariance matrix of $\boldsymbol{\mathcal{W}}$ with entry $[\boldsymbol{K}_0]_{ij} = k_0(\boldsymbol{w}^{(i)}, \boldsymbol{w}^{(j)})$.

4.3.3.1 Bayesian inference of REF

As an extended result of BPI [58], the posterior distribution of REF (denoted as $\hat{\mathcal{M}}(\boldsymbol{v})$), i.e., integrating $\hat{\mathcal{G}}(\boldsymbol{w})$ with respect to $\boldsymbol{u}$ under the Gaussian weight $f_{\boldsymbol{U}}(\boldsymbol{u})$, still follows a GP. By repeated application of Fubini's theorem, one can derive the analytical expressions of the posterior mean function $\mu_{\hat{\mathcal{M}}}(\boldsymbol{v})$ and posterior variance function $\sigma^2_{\hat{\mathcal{M}}}(\boldsymbol{v})$ such that:

$$\mu_{\hat{\mathcal{M}}}(\boldsymbol{v}) = \mathbb{E}_{\boldsymbol{\mathcal{D}}}\left[\hat{\mathcal{M}}(\boldsymbol{v})\right] = \Pi_{\boldsymbol{u}}\left[\mu_0(\boldsymbol{w})\right] + \Pi_{\boldsymbol{u}}\left[\boldsymbol{k}_0(\boldsymbol{w}, \boldsymbol{\mathcal{W}})^{\mathrm{T}}\right] \boldsymbol{K}_0^{-1} (\boldsymbol{\mathcal{Z}} - \boldsymbol{\mu}_0(\boldsymbol{\mathcal{W}})), \tag{4.8}$$

$$\sigma^2_{\hat{\mathcal{M}}}(\boldsymbol{v}) = \mathbb{V}_{\boldsymbol{\mathcal{D}}}\left[\hat{\mathcal{M}}(\boldsymbol{v})\right] = \Pi_{\boldsymbol{u}}\Pi_{\boldsymbol{u}'}\left[k_0(\boldsymbol{w}, (\boldsymbol{u}', \boldsymbol{v}))\right] - \Pi_{\boldsymbol{u}}\left[\boldsymbol{k}_0(\boldsymbol{w}, \boldsymbol{\mathcal{W}})^{\mathrm{T}}\right] \boldsymbol{K}_0^{-1} \Pi_{\boldsymbol{u}}\left[\boldsymbol{k}_0(\boldsymbol{w}, \boldsymbol{\mathcal{W}})\right], \tag{4.9}$$

where $\mathbb{E}_{\boldsymbol{\mathcal{D}}}[\cdot]$ and $\mathbb{V}_{\boldsymbol{\mathcal{D}}}[\cdot]$ refer to the expectation and variance operators taken with respect to the posterior distributions of their arguments given data $\boldsymbol{\mathcal{D}}$; $\Pi_{\boldsymbol{u}}[\cdot]$ denotes the integral operator taken with respect to $\boldsymbol{u}$ under Gaussian weight $f_{\boldsymbol{U}}(\boldsymbol{u})$; $\Pi_{\boldsymbol{u}'}[\cdot]$ is similarly defined; $\Pi_{\boldsymbol{u}}\Pi_{\boldsymbol{u}'}[\cdot]$ is the integral operator taken respect to both $\boldsymbol{u}$ and $\boldsymbol{u}'$ under Gaussian weights $f_{\boldsymbol{U}}(\boldsymbol{u})$ and $f_{\boldsymbol{U}}(\boldsymbol{u}')$; The term $\Pi_{\boldsymbol{u}}[\mu_0(\boldsymbol{w})]$ can be easily obtained as $\Pi_{\boldsymbol{u}}[\mu_0(\boldsymbol{w})] = \beta$; The other terms can be derived as:

$$\Pi_{\boldsymbol{u}}[\boldsymbol{k}_0(\boldsymbol{w}, \boldsymbol{\mathcal{W}})] = s_0^2 \left|\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1} + I\right|^{-1/2} \exp\left[-\frac{1}{2}\mathrm{diag}\left\{\boldsymbol{\mathcal{U}}(\boldsymbol{\Sigma}_{\boldsymbol{u}} + \boldsymbol{I})^{-1}\boldsymbol{\mathcal{U}}^{\mathrm{T}} - (\boldsymbol{v} - \boldsymbol{\mathcal{V}})\boldsymbol{\Sigma}_{\boldsymbol{v}}^{-1}(\boldsymbol{v} - \boldsymbol{\mathcal{V}})^{\mathrm{T}}\right\}\right], \tag{4.10}$$

$$\Pi_{\boldsymbol{u}}\Pi_{\boldsymbol{u}'}[k_0(\boldsymbol{w}, \boldsymbol{w}')] = s_0^2 \left|2\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1} + \boldsymbol{I}\right|^{-1/2}, \tag{4.11}$$

where $|\cdot|$ means the determinant of its argument; $\boldsymbol{I}$ is a identity matrix of size d_1.

Note that the expressions for $\mu_{\hat{\mathcal{M}}}(\boldsymbol{v})$ and $\sigma^2_{\hat{\mathcal{M}}}(\boldsymbol{v})$ are similar in form to those of NIPI and CABO, but essentially different due to the fact that the proposed method is established on the basis of the joint space of standard normal variables and standard interval variables, while both NIPI and CABO are cast in the standard normal space. The posterior mean function $\mu_{\mathcal{M},n}(\boldsymbol{v})$ can be used as an estimate of $\mathcal{M}(\boldsymbol{v})$ and the posterior variance function $\sigma^2_{\hat{\mathcal{M}}}(\boldsymbol{v})$ measures our uncertainty of the estimate after n observations have been available. By using the linear transformation, one can easily obtain the posterior mean function $\mu_{\hat{m}}(\boldsymbol{\alpha}) = \mu_{\hat{\mathcal{M}}}(T_2(\boldsymbol{\alpha}))$ and posterior variance function $\sigma^2_{\hat{m}}(\boldsymbol{\alpha}) = \sigma^2_{\hat{\mathcal{M}}}(T_2(\boldsymbol{\alpha}))$ for $\hat{m}(\boldsymbol{\alpha})$.

4.3.3.2 Bayesian inference of RS-HDMR component functions of REF

In addition to the REF $\hat{m}(\boldsymbol{\alpha})$, the analyst may also be concerned about, e.g., identifying key variables among $\boldsymbol{A}$ that are more important for $m(\boldsymbol{\alpha})$. For this propose, the RS-HDMR is first employed to express $\mathcal{M}(\boldsymbol{v})$ as the summation of a set of component functions with increasing dimensions [59]:

$$\mathcal{M}(\boldsymbol{v}) = \mathcal{M}_0 + \sum_{i=1}^{d_2}\mathcal{M}_i(v_i) + \sum_{1\le i<j\le d_2}\mathcal{M}_{ij}(v_i, v_j) + \cdots + \mathcal{M}_{ij\ldots d_2}(v_1, v_2, \cdots, v_{d_2}), \tag{4.12}$$

where the zeroth-order component function $\mathcal{M}_0$ is a constant representing the average value of $\mathcal{M}(\boldsymbol{v})$ over the entire domain $\mathcal{V}$, the first-order component function $\mathcal{M}_i(v_i)$ represents the independent contribution of v_i acting alone to $\mathcal{M}(\boldsymbol{v})$, the second-order component function $\mathcal{M}_{ij}(v_i, v_j)$ denotes

the cooperative effects of v_i and v_j upon $\mathcal{M}(\boldsymbol{v})$, etc. The last term $\mathcal{M}_{ij\ldots d_2}(v_1, v_2, \cdots, v_{d_2})$ describes any residual cooperative effects of all input variables acting together to influence $\mathcal{M}(\boldsymbol{v})$. The component functions up to the second-order can be defined as:

$$\mathcal{M}_0 = \int_{\mathcal{V}} \mathcal{M}(\boldsymbol{v}) \mathrm{d}\boldsymbol{v} = \int_{\mathcal{V}} \int_{\mathcal{U}} \mathcal{G}(\boldsymbol{w}) f_{\boldsymbol{U}}(\boldsymbol{u}) \mathrm{d}\boldsymbol{u} \mathrm{d}\boldsymbol{v}, \tag{4.13}$$

$$\mathcal{M}_i(v_i) = \int_{\mathcal{V}_{-i}} \mathcal{M}(\boldsymbol{v}) \mathrm{d}\boldsymbol{v}_{-i} - \mathcal{M}_0 = \int_{\mathcal{V}_{-i}} \int_{\mathcal{U}} \mathcal{G}(\boldsymbol{w}) f_{\boldsymbol{U}}(\boldsymbol{u}) \mathrm{d}\boldsymbol{u} \mathrm{d}\boldsymbol{v}_{-i} - \mathcal{M}_0, \tag{4.14}$$

$$\mathcal{M}_{ij}(v_i, v_j) = \int_{\mathcal{V}_{-ij}} \mathcal{M}(\boldsymbol{v}) \mathrm{d}\boldsymbol{v}_{-ij} - \mathcal{M}_i(v_i) - \mathcal{M}_j(v_j) - \mathcal{M}_0 = \int_{\mathcal{V}_{-ij}} \int_{\mathcal{U}} \mathcal{G}(\boldsymbol{w}) f_{\boldsymbol{U}}(\boldsymbol{u}) \mathrm{d}\boldsymbol{u} \mathrm{d}\boldsymbol{v}_{-ij} - \mathcal{M}_i(v_i) - \mathcal{M}_j(v_j) - \mathcal{M}_0, \tag{4.15}$$

where $\mathcal{V}_{-i}$ and $\boldsymbol{v}_{-i}$ denote the space $\mathcal{V}$ and the vector $\boldsymbol{v}$ excluding the i-th dimension, respectively; $\mathcal{V}_{-ij}$ and $\boldsymbol{v}_{-ij}$ are similarly defined.

As high-order component functions have small contributions for many realistic systems, the second-order truncated RS-HDMR expansion is often considered [36, 40]. For this reason, only the component functions up to the second-order are provided in the following via Bayesian inference. If necessary, high-order component functions can also be derived similarly.

Zeroth-order RS-HDMR component As defined in Eq. (4.13), the zeroth-order RS-HDMR component $\mathcal{M}_0$ is actually an integral of $\mathcal{G}(\boldsymbol{w})$ with respect to $\boldsymbol{w}$. From a Bayesian quadrature perspective, the posterior distribution of $\mathcal{M}_0$ (denoted as $\hat{\mathcal{M}}_0$) is Gaussian with posterior mean $\mu_{\hat{\mathcal{M}}_0}$ and posterior variance $\sigma^2_{\hat{\mathcal{M}}_0}$ being:

$$\mu_{\hat{\mathcal{M}}_0} = \mathbb{E}_{\boldsymbol{\mathcal{D}}}\left[\hat{\mathcal{M}}_0\right] = \Pi_{\boldsymbol{w}}\left[\mu_0(\boldsymbol{w})\right] + \Pi_{\boldsymbol{w}}\left[\boldsymbol{k}_0\left(\boldsymbol{w}, \boldsymbol{\mathcal{W}}\right)^{\mathrm{T}}\right] \boldsymbol{K}_0^{-1}\left(\boldsymbol{\mathcal{Z}} - \boldsymbol{\mu}_0\left(\boldsymbol{\mathcal{W}}\right)\right), \tag{4.16}$$

$$\sigma^2_{\hat{\mathcal{M}}_0} = \mathbb{V}_{\boldsymbol{\mathcal{D}}}\left[\hat{\mathcal{M}}_0\right] = \Pi_{\boldsymbol{w}}\Pi_{\boldsymbol{w}'}\left[k_0(\boldsymbol{w}, \boldsymbol{w}')\right] - \Pi_{\boldsymbol{w}}\left[\boldsymbol{k}_0\left(\boldsymbol{w}, \boldsymbol{\mathcal{W}}\right)^{\mathrm{T}}\right] \boldsymbol{K}_0^{-1}\Pi_{\boldsymbol{w}'}\left[\boldsymbol{k}_0\left(\boldsymbol{w}', \boldsymbol{\mathcal{W}}\right)\right], \tag{4.17}$$

where $\Pi_{\boldsymbol{w}}\left[\mu_0(\boldsymbol{w})\right] = \beta$, and other terms can be derived as:

$$\begin{aligned}
\Pi_{\boldsymbol{w}}\left[\boldsymbol{k}_0\left(\boldsymbol{w}, \boldsymbol{\mathcal{W}}\right)\right] =& \Pi_{\boldsymbol{w}'}\left[\boldsymbol{k}_0\left(\boldsymbol{w}', \boldsymbol{\mathcal{W}}\right)\right] \\
=& s_0^2 \left|\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1} + \boldsymbol{I}\right|^{-1/2} \exp\left[-\frac{1}{2}\mathrm{diag}\left\{\boldsymbol{\mathcal{U}}\left(\boldsymbol{\Sigma}_{\boldsymbol{u}} + \boldsymbol{I}\right)^{-1}\boldsymbol{\mathcal{U}}^{\mathrm{T}}\right\}\right] \\
& \cdot \left(\frac{\pi}{2}\right)^{d_2/2} \mathrm{prod}_2\left\{\left[\mathrm{erf}\left(\left(1 - \boldsymbol{\mathcal{V}}\right)\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}}\right)^{-1/2}\right) - \mathrm{erf}\left(-\boldsymbol{\mathcal{V}}\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}}\right)^{-1/2}\right)\right] \boldsymbol{\Sigma}_{\boldsymbol{v}}^{1/2}\right\},
\end{aligned} \tag{4.18}$$

$$\begin{aligned}\Pi_{\boldsymbol{w}}\Pi_{\boldsymbol{w}'}\left[k_0(\boldsymbol{w},\boldsymbol{w}')\right] =& s_0^2\left|2\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1}+\boldsymbol{I}\right|^{-1/2} \\ & \cdot 2^{d_2}\mathrm{prod}_1\left\{\mathrm{diag}\left\{\boldsymbol{\Sigma}_{\boldsymbol{v}}\left[-1+\exp\left[-(2\boldsymbol{\Sigma}_{\boldsymbol{v}})^{-1}\right]+(2\pi^{-1}\boldsymbol{\Sigma}_{\boldsymbol{v}})^{-1/2}\mathrm{erf}\left((2\boldsymbol{\Sigma}_{\boldsymbol{v}})^{-1/2}\right)\right]\right\}\right\},\end{aligned} \tag{4.19}$$

where $\mathrm{prod}_1\{\cdot\}$ means to return the product of the elements of its argument; $\mathrm{prod}_2\{\cdot\}$ is to get a column vector containing the products of each row of its argument; $\mathrm{erf}(\cdot)$ stands for the error function. Note that in Eq. (4.18) the argument in $\mathrm{prod}_2\{\cdot\}$ is an n-by-d_2 matrix, while in Eq. (4.19) the argument in $\mathrm{prod}_1\{\cdot\}$ is a d_2-by-1 vector.

First-order RS-HDMR component The first-order RS-HDMR component function $\mathcal{M}_i(v_i)$ defined in Eq. (4.14) is an integral (i.e, integrating $\mathcal{G}(\boldsymbol{w})$ with respect to $\boldsymbol{w}$ excluding v_i) minus $\mathcal{M}_0$, and thus its posterior distribution $\hat{\mathcal{M}}_i(v_i)$ should follow a one-dimensional GP.

The posterior mean function $\mu_{\hat{\mathcal{M}}_i}(v_i)$ of the first-order RS-HDMR component function $\hat{\mathcal{M}}_i(v_i)$ can be expressed as:

$$\mu_{\hat{\mathcal{M}}_i}(v_i)=\mathbb{E}_{\boldsymbol{\mathcal{D}}}\left[\hat{\mathcal{M}}_i(v_i)\right]=\Pi_{-v_i}\left[\mu_0(\boldsymbol{w})\right]+\Pi_{-v_i}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)^{\mathrm{T}}\right]\boldsymbol{K}_0^{-1}\left(\boldsymbol{\mathcal{Z}}-\boldsymbol{\mu}_0\left(\boldsymbol{\mathcal{W}}\right)\right)-\mu_{\hat{\mathcal{M}}_0}, \tag{4.20}$$

where $\Pi_{-v_i}[\cdot]$ denotes the integration of its argument taken over $\boldsymbol{w}$ except v_i; it is obvious that $\Pi_{-v_i}\left[\mu_0(\boldsymbol{w})\right]=\beta$; the term $\Pi_{-v_i}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right]$ can be derived as:

$$\begin{aligned}\Pi_{-v_i}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right] =& s_0^2\left|\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1}+\boldsymbol{I}\right|^{-1/2}\exp\left[-\frac{1}{2}\mathrm{diag}\left\{\boldsymbol{\mathcal{U}}\left(\boldsymbol{\Sigma}_{\boldsymbol{u}}+\boldsymbol{I}\right)^{-1}\boldsymbol{\mathcal{U}}^{\mathrm{T}}\right\}\right] \\ & \cdot\left(\frac{\pi}{2}\right)^{(d_2-1)/2}\mathrm{prod}_2\left\{\left[\mathrm{erf}\left(\left(1-\boldsymbol{\mathcal{V}}_{,-i}\right)\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}}\right)^{-1/2}\right)-\mathrm{erf}\left(-\boldsymbol{\mathcal{V}}_{,-i}\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}}\right)^{-1/2}\right)\right]\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}}^{1/2}\right\} \\ & \cdot\exp\left[-\frac{1}{2}\mathrm{diag}\left\{-\left(v_i-\boldsymbol{\mathcal{V}}_{,i}\right)\boldsymbol{\Sigma}_{\boldsymbol{v}_i}^{-1}\left(v_i-\boldsymbol{\mathcal{V}}_{,i}\right)^{\mathrm{T}}\right\}\right],\end{aligned} \tag{4.21}$$

in which $\boldsymbol{\mathcal{V}}_{,i}$ is the i-th column of $\boldsymbol{\mathcal{V}}$, $\boldsymbol{\mathcal{V}}_{,-i}$ represents the matrix generated by removing $\boldsymbol{\mathcal{V}}_{,i}$ from $\boldsymbol{\mathcal{V}}$, $\boldsymbol{\Sigma}_{\boldsymbol{v}_i}$ denotes the (i,i)-th element of $\boldsymbol{\Sigma}_{\boldsymbol{v}}$, and $\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}}$ stands for the matrix generated by removing the i-th column and i-th row of $\boldsymbol{\Sigma}_{\boldsymbol{v}}$.

For the posterior variance function $\sigma^2_{\hat{\mathcal{M}}_i}(v_i)$ of the first-order RS-HDMR component function $\hat{\mathcal{M}}_i(v_i)$, one can refer to 4.6.

Second-order RS-HDMR component Similarly, the second-order RS-HDMR component function $\mathcal{M}_{ij}(v_i,v_j)$ defined in Eq. (4.15) is an integral (i.e., integrating $\mathcal{G}(\boldsymbol{w})$ with respect to $\boldsymbol{w}$ exclud-

ing v_i and v_j) diminished by $\mathcal{M}_i(v_i)$, $\mathcal{M}_j(v_j)$ and $\mathcal{M}_0$, and thus its posterior distribution $\hat{\mathcal{M}}_{ij}(v_i, v_j)$ should follow a two-dimensional GP.

The posterior mean function $\mu_{\hat{\mathcal{M}}_{ij}}(v_i, v_j)$ of the first-order RS-HDMR component function $\hat{\mathcal{M}}_{ij}(v_i, v_j)$ can be given by:

$$\begin{aligned}\mu_{\hat{\mathcal{M}}_{ij}}(v_i, v_j) =& \mathbb{E}_{\boldsymbol{\mathcal{D}}}\left[\hat{\mathcal{M}}_{ij}(v_i, v_j)\right] \\ =& \Pi_{-\boldsymbol{v}_{ij}}\left[\mu_0(\boldsymbol{w})\right] + \Pi_{-\boldsymbol{v}_{ij}}\left[\boldsymbol{k}_0\left(\boldsymbol{w}, \boldsymbol{\mathcal{W}}\right)^{\mathrm{T}}\right] \boldsymbol{K}_0^{-1}\left(\boldsymbol{\mathcal{Z}} - \boldsymbol{\mu}_0\left(\boldsymbol{\mathcal{W}}\right)\right) - \mu_{\hat{\mathcal{M}}_i}(v_i) - \mu_{\hat{\mathcal{M}}_j}(v_j) - \mu_{\hat{\mathcal{M}}_0},\end{aligned} \tag{4.22}$$

where the term $\Pi_{-\boldsymbol{v}_{ij}}\left[\mu_0(\boldsymbol{w})\right]$ is equal to β, and the term $\Pi_{-\boldsymbol{v}_{ij}}\left[\boldsymbol{k}_0\left(\boldsymbol{w}, \boldsymbol{\mathcal{W}}\right)\right]$ is derived as:

$$\begin{aligned}\Pi_{-\boldsymbol{v}_{ij}}\left[\boldsymbol{k}_0\left(\boldsymbol{w}, \boldsymbol{\mathcal{W}}\right)\right] =& s_0^2\left|\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1} + \boldsymbol{I}\right|^{-1/2} \exp\left[-\frac{1}{2}\mathrm{diag}\left\{\boldsymbol{\mathcal{U}}\left(\boldsymbol{\Sigma}_{\boldsymbol{u}} + \boldsymbol{I}\right)^{-1}\boldsymbol{\mathcal{U}}^{\mathrm{T}}\right\}\right] \\ & \cdot \left(\frac{\pi}{2}\right)^{(d_2-2)/2} \mathrm{prod}_2\left\{\left[\mathrm{erf}\left(\left(1 - \boldsymbol{\mathcal{V}}_{,-ij}\right)\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}}\right)^{-1/2}\right) - \mathrm{erf}\left(-\boldsymbol{\mathcal{V}}_{,-ij}\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}}\right)^{-1/2}\right)\right]\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}}^{1/2}\right\} \\ & \cdot \exp\left[-\frac{1}{2}\mathrm{diag}\left\{-\left(v_{ij} - \boldsymbol{\mathcal{V}}_{,ij}\right)\boldsymbol{\Sigma}_{\boldsymbol{v}_{ij}}^{-1}\left(v_{ij} - \boldsymbol{\mathcal{V}}_{,ij}\right)^{\mathrm{T}}\right\}\right].\end{aligned} \tag{4.23}$$

For the posterior variance function $\sigma^2_{\hat{\mathcal{M}}_{ij}}(v_i, v_j)$ of the second-order RS-HDMR component function $\hat{\mathcal{M}}_{ij}(v_i, v_j)$, one can refer to 4.6. ■

One should note that the above results are essentially different from those in NIPI. Once these RS-HDMR component functions of $\hat{\mathcal{M}}(\boldsymbol{v})$ are properly inferred, they can be transformed by a linear transformation to yield the RS-HDMR component functions for $\hat{m}(\boldsymbol{\alpha})$.

4.3.3.3 Bayesian inference of extrema of REF

If we stop after obtaining n observations of the $\mathcal{G}$-function, a risk-neutral choice for the minimum or maximum of the REF would be the minimum or maximum of the posterior mean function $\mu_{\hat{m}}(\boldsymbol{\alpha})$. As $\mu_{\hat{m}}(\boldsymbol{\alpha})$ has been derived in a closed-form, the extrema of the REF can be inferred from $\mu_{\hat{m}}(\boldsymbol{\alpha})$ by simply applying a global optimization algorithm such that:

$$\hat{m}_l = \min_{\boldsymbol{\alpha}\in[\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}]} \mu_{\hat{m}}(\boldsymbol{\alpha}), \tag{4.24}$$

$$\hat{m}_u = \max_{\boldsymbol{\alpha}\in[\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}]} \mu_{\hat{m}}(\boldsymbol{\alpha}). \tag{4.25}$$

Besides, since the posterior variance function $\sigma^2_{\hat{m}}(\boldsymbol{\alpha})$ is also available, the prediction errors regarding the minimum and maximum estimators in Eqs. (4.24) and (4.25) can be measured by the posterior variances:

$$\mathrm{Var}\left[\hat{m}_l\right] = \sigma^2_{\hat{m}}(\boldsymbol{\alpha}^-), \tag{4.26}$$

$$\mathrm{Var}\left[\hat{m}_u\right] = \sigma^2_{\hat{m}}(\boldsymbol{\alpha}^+), \tag{4.27}$$

where $\boldsymbol{\alpha}^- = \arg\min_{\boldsymbol{\alpha}\in[\underline{\boldsymbol{\alpha}},\overline{\boldsymbol{\alpha}}]} \mu_{\hat{m}}(\boldsymbol{\alpha})$ and $\boldsymbol{\alpha}^+ = \arg\max_{\boldsymbol{\alpha}\in[\underline{\boldsymbol{\alpha}},\overline{\boldsymbol{\alpha}}]} \mu_{\hat{m}}(\boldsymbol{\alpha})$ are the minimum point and maximum point, respectively.

4.3.4 Parallel Bayesian experimental design

Another significant advantage of the above framework is that it offers the possibility for incorporating our prior knowledge and developing a Bayesian experimental design strategy. This advantage is also realized in both NIPI and CABO. These two methods, however, are in a pure sequential manner to acquire the $\mathcal{G}$-function. That is, at each iteration only one point is allowed to be selected and a single $\mathcal{G}$-function evaluation is subsequently performed. The sequential experimental strategies would be less efficient and flexible when parallel computing architectures are available. Besides, the one for NIPI is specifically designed for inferring RS-HDMR component functions, whereas the one for CABO is only developed for inferring the extrema of the REF. Based on these considerations, a novel contribution here is to present a multi-point selection criterion that can support parallel evaluations of the $\mathcal{G}$-function and also enable us to estimate the REF, its RS-HDMR component functions and bounds at the same time. In this study, the preferred number of CPU cores or workers in a parallel pool is assumed to be an even number, denoted by c.

Stage 1: Global improvement Supposing that we have only obtained a small set of initial observations, the first stage of our strategy aims to improve the global accuracy of the REF. The key lies in three main aspects: (1) how can we measure the global accuracy of the REF? (2) how to select c points at each iteration that are expected to improve the global accuracy of the REF? (3) when to stop the iteration at this stage?

As the zero-th order RS-HDMR component $\mathcal{M}_0$ is defined as an integral of the REF $\mathcal{M}(\boldsymbol{v})$ with respect to $\boldsymbol{v}$ (called augmented expectation), its accuracy may reflect the global accuracy of the REF to some extent. Therefore, the accuracy of $\hat{\mathcal{M}}_0$ is taken as a global accuracy measure of $\hat{\mathcal{M}}(\boldsymbol{v})$

in this study, which can be quantified by the posterior variance $\sigma^2_{\hat{\mathcal{M}}_0}$. Inspired by [23, 40, 41], a new acquisition function, called posterior variance contribution to the augmented expectation (denoted as $\mathrm{PVC}^{\mathrm{A}}$), is given by:

$$\mathrm{PVC}^{\mathrm{A}}(\boldsymbol{w}) = \Pi_{\boldsymbol{w}'}\left[k_n(\boldsymbol{w},\boldsymbol{w}')\right]\times f_{\boldsymbol{W}}(\boldsymbol{w}) = \left\{\Pi_{\boldsymbol{w}'}\left[k_0(\boldsymbol{w},\boldsymbol{w}')\right] - \boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)^{\mathrm{T}}\boldsymbol{K}_0^{-1}\Pi_{\boldsymbol{w}'}\left[\boldsymbol{k}_0\left(\boldsymbol{w}',\boldsymbol{\mathcal{W}}\right)\right]\right\}\times f_{\boldsymbol{W}}(\boldsymbol{w}), \tag{4.28}$$

where the closed-form expression of $\Pi_{\boldsymbol{w}'}\left[\boldsymbol{k}_0\left(\boldsymbol{w}',\boldsymbol{\mathcal{W}}\right)\right]$ has been given in Eq. (4.18); Similarly, the term $\Pi_{\boldsymbol{w}'}\left[k_0(\boldsymbol{w},\boldsymbol{w}')\right]$ can be derived as:

$$\begin{aligned}\Pi_{\boldsymbol{w}'}\left[k_0(\boldsymbol{w},\boldsymbol{w}')\right] =& s_0^2\left|\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1}+\boldsymbol{I}\right|^{-1/2}\exp\left[-\frac{1}{2}\mathrm{diag}\left\{\boldsymbol{u}\left(\boldsymbol{\Sigma}_{\boldsymbol{u}}+\boldsymbol{I}\right)^{-1}\boldsymbol{u}^{\mathrm{T}}\right\}\right]\\ &\cdot\left(\frac{\pi}{2}\right)^{d_2/2}\mathrm{prod}_2\left\{\left[\mathrm{erf}\left((1-\boldsymbol{v})\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}}\right)^{-1/2}\right)-\mathrm{erf}\left(-\boldsymbol{v}\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}}\right)^{-1/2}\right)\right]\boldsymbol{\Sigma}_{\boldsymbol{v}}^{1/2}\right\}.\end{aligned} \tag{4.29}$$

The acquisition function in Eq. (4.28) is said to be 'new' because it is essentially not the same as those in the cited references. It should be noted that $\sigma^2_{\hat{\mathcal{M}}_0} = \int_{\mathcal{W}}\mathrm{PVC}^{\mathrm{A}}(\boldsymbol{w})\mathrm{d}\boldsymbol{w}$ holds, which implies that the $\mathrm{PVC}^{\mathrm{A}}$ function can measure the contribution of our epistemic uncertainty at $\boldsymbol{w}$ to $\sigma^2_{\hat{\mathcal{M}}_0}$. For this reason, by selecting $\boldsymbol{w}^{(n+1)} = \arg\max_{\boldsymbol{w}\in\mathcal{W}}\mathrm{PVC}^{\mathrm{A}}(\boldsymbol{w})$ as the best next point to evaluate the $\mathcal{G}$-function, it is expected that the posterior variance of the augmented expectation will decrease the most, and hence the accuracy of the posterior mean of the augmented expectation will be improved the most. However, adding one single point at a time may waste other useful information and cannot allow to make use of parallelization, and hence it could be inefficient especially when parallel evaluations are possible.

In this study, we propose a novel strategy to parallelize the developed PBQO method by providing c points at each iteration. This strategy is motivated by the fact that the $\mathrm{PVC}^{\mathrm{A}}$ function (defined in Eq. (4.28)) only explicitly depends on the sampled locations, not on function values at these points. For this reason, we can rewrite the $\mathrm{PVC}^{\mathrm{A}}$ function as $\mathrm{PVC}^{\mathrm{A}}(\boldsymbol{w},\boldsymbol{\mathcal{W}})$. Therefore, it is possible to select c points ahead of observing their $\mathcal{G}$-function values based on the $\mathrm{PVC}^{\mathrm{A}}(\boldsymbol{w},\boldsymbol{\mathcal{W}})$ function. Specifically, each point can be selected sequentially, with the $\mathrm{PVC}^{\mathrm{A}}$ function modified by considering the newly selected points at the current iteration. The assumption behind this strategy is that the hyper-parameters will not change, and hence the $\mathrm{PVC}^{\mathrm{A}}$ function remains the same during the process of identifying the next $c-1$ points. In fact, the hyper-parameters do change if we update immediately the GP after each point is chosen and its $\mathcal{G}$-function value is computed,

which, however, corresponds to the single-point selection strategy. Our idea is expected to work since the hyper-parameters may not vary too much within the next few steps. The pseudocode of the proposed multi-point selection strategy is given in Algorithm 2. Until c points are obtained, evaluating the $\mathcal{G}$-function at these points can be run in parallel, and the GP model can be updated subsequently. This iteration process is repeated until a stopping criterion is reached, which is defined as the posterior coefficient of variation (COV) of the augmented expectation less than a pre-specified tolerance $\varepsilon^{\mathrm{BPI}}$, i.e., $\frac{\sigma_{\hat{\mathcal{M}}_0}}{|\mu_{\hat{\mathcal{M}}_0}|} < \varepsilon^{\mathrm{BPI}}$. To avoid possible premature convergence, the stopping criterion is required to be satisfied several (e.g., two) times in successive iterations. It should be noted that the proposed multi-point selection strategy is computationally inexpensive and can usually produce a batch of c diverse points according to our computational experience, which are thus effective and informative for parallelization.

Algorithm 2 Proposed multi-point selection strategy based on the $\mathrm{PVC}^{\mathrm{A}}(\boldsymbol{w}, \boldsymbol{\mathcal{W}})$ function

1: **Input:** c and $\mathrm{PVC}^{\mathrm{A}}(\boldsymbol{w}, \boldsymbol{\mathcal{W}})$
2: **for** $i = 1 \rightarrow c$ **do**
3: $\quad \boldsymbol{w}^{(n+i)} = \arg\max_{\boldsymbol{w} \in \mathcal{W}} \mathrm{PVC}^{\mathrm{A}}(\boldsymbol{w}, \boldsymbol{\mathcal{W}})$
4: $\quad \boldsymbol{\mathcal{W}} = \boldsymbol{\mathcal{W}} \cup \boldsymbol{w}^{(n+i)}$
5: **end for**
6: **Output:** $\boldsymbol{w}^{(n+1)}, \boldsymbol{w}^{(n+2)}, \cdots, \boldsymbol{w}^{(n+c)}$

Stage 2: Local improvement After stage 1, it is expected that the general trend of the REF has been captured. However, the local features of the REF, e.g., minimum and maximum, may still be inaccurate. In this regard, the second stage of our strategy attempts to further improve the accuracy of the resulting REF from stage 1, with special emphasis on its extrema.

As the posterior distribution of the REF follows a GP, the expected improvement criterion originally introduced in BGO [56] could be adopted for our purposes. Let $\hat{\mathcal{M}}_l(\boldsymbol{v}^-)$ denote the current minimum, and $\boldsymbol{v}^-$ the minimum point, i.e., $\boldsymbol{v}^- = \arg\min_{\boldsymbol{v} \in \mathcal{V}} \mu_{\hat{\mathcal{M}}}(\boldsymbol{v})$. The improvement for the current minimum at the point $\boldsymbol{v}$ can be defined as $I(\boldsymbol{v}) = \max\left(\hat{\mathcal{M}}_l(\boldsymbol{v}^-) - \mu_{\hat{\mathcal{M}}}(\boldsymbol{v}), 0\right)$. The acquisition function, called expected improvement for the minimization (abbreviated as $\mathrm{EI}^{\mathrm{MIN}}$), is to simple take the expected value of $I(\boldsymbol{v})$, i.e., $\mathrm{EI}^{\mathrm{MIN}}(\boldsymbol{v}) = \mathbb{E}\left[I(\boldsymbol{v})\right]$. The closed-form expression of

$\mathrm{EI}^{\mathrm{MIN}}$ can be written as [56]:

$$\mathrm{EI}^{\mathrm{MIN}}(\boldsymbol{v}) = \left(\hat{\mathcal{M}}_l(\boldsymbol{v}^-) - \mu_{\hat{\mathcal{M}}}(\boldsymbol{v})\right) \Phi\left(\frac{\hat{\mathcal{M}}_l(\boldsymbol{v}^-) - \mu_{\hat{\mathcal{M}}}(\boldsymbol{v})}{\sigma_{\hat{\mathcal{M}}}(\boldsymbol{v})}\right) + \sigma_{\hat{\mathcal{M}}}(\boldsymbol{v})\varphi\left(\frac{\hat{\mathcal{M}}_l(\boldsymbol{v}^-) - \mu_{\hat{\mathcal{M}}}(\boldsymbol{v})}{\sigma_{\hat{\mathcal{M}}}(\boldsymbol{v})}\right), \quad (4.30)$$

where $\varphi(\cdot)$ and $\Phi(\cdot)$ are the PDF and and cumulative distribution function (CDF) of the standard normal distribution, respectively. The $\mathrm{EI}^{\mathrm{MIN}}$ function actually measures how much improvement for the minimum is expected to achieve by sampling at $\boldsymbol{v}$. Thus, the next best point for $\boldsymbol{v}$ can be selected by maximizing the $\mathrm{EI}^{\mathrm{MIN}}$ function, i.e., $\underline{\boldsymbol{v}}^\star = \arg\max_{\boldsymbol{v}\in\mathcal{V}} \mathrm{EI}^{\mathrm{MIN}}(\boldsymbol{v})$. The first summation term in Eq. (4.30) is the exploitation term encouraging to sample where $\mu_{\hat{\mathcal{M}}}(\boldsymbol{v})$ is small, whereas the second summation term is the exploration term encouraging to sample where $\sigma_{\hat{\mathcal{M}}}(\boldsymbol{v})$ is large. At this stage, the associated stopping criterion can be given as [60]:

$$\frac{|\max_{\boldsymbol{v}\in\mathcal{V}} \mathrm{EI}^{\mathrm{MIN}}(\boldsymbol{v})|}{\max \boldsymbol{\mathcal{Z}} - \min \boldsymbol{\mathcal{Z}}} < \varepsilon^{\mathrm{BGO}}, \quad (4.31)$$

where $\varepsilon^{\mathrm{BGO}}$ is a user-defined tolerance. Similarly, the stopping criterion also needs to be met for two times in succession. Once $\underline{\boldsymbol{v}}^\star$ is identified, the best next point for $\boldsymbol{u}$ can also be specified. In order to improve the accuracy of $\mu_{\hat{\mathcal{M}}}(\underline{\boldsymbol{v}}^\star)$, an acquisition function measuring the posterior variance contribution to $\sigma^2_{\hat{\mathcal{M}}}(\underline{\boldsymbol{v}}^\star)$ (abbreviated as $\mathrm{PVC}^{\mathrm{MIN}}$), can be defined:

$$\begin{aligned}\mathrm{PVC}^{\mathrm{MIN}}(\boldsymbol{u}) &= \Pi_{\boldsymbol{u}'}\left[k_n((\boldsymbol{u},\underline{\boldsymbol{v}}^\star),(\boldsymbol{u}',\underline{\boldsymbol{v}}^\star))\right] \times f_U(\boldsymbol{u}) \\ &= \left\{\Pi_{\boldsymbol{u}'}\left[k_0((\boldsymbol{u},\underline{\boldsymbol{v}}^\star),(\boldsymbol{u}',\underline{\boldsymbol{v}}^\star))\right] - \boldsymbol{k}_0\left((\boldsymbol{u},\underline{\boldsymbol{v}}^\star),\boldsymbol{\mathcal{W}}\right)^{\mathrm{T}} \boldsymbol{K}_0^{-1} \Pi_{\boldsymbol{u}'}\left[\boldsymbol{k}_0\left((\boldsymbol{u}',\underline{\boldsymbol{v}}^\star),\boldsymbol{\mathcal{W}}\right)\right]\right\} \times f_U(\boldsymbol{u}),\end{aligned} \quad (4.32)$$

where the term $\Pi_{\boldsymbol{u}'}\left[\boldsymbol{k}_0\left((\boldsymbol{u}',\underline{\boldsymbol{v}}^\star),\boldsymbol{\mathcal{W}}\right)\right]$ can be generated as Eq. (4.10) by replacing $\boldsymbol{v}$ by $\underline{\boldsymbol{v}}^\star$, and the term $\Pi_{\boldsymbol{u}'}\left[k_0((\boldsymbol{u},\underline{\boldsymbol{v}}^\star),(\boldsymbol{u}',\underline{\boldsymbol{v}}^\star))\right]$ can be derived as:

$$\Pi_{\boldsymbol{u}'}\left[k_0((\boldsymbol{u},\underline{\boldsymbol{v}}^\star),(\boldsymbol{u}',\underline{\boldsymbol{v}}^\star))\right] = s_0^2 \left|\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1} + \boldsymbol{I}\right|^{-1/2} \exp\left[-\frac{1}{2}\mathrm{diag}\left\{\boldsymbol{u}\left(\boldsymbol{\Sigma}_{\boldsymbol{u}} + \boldsymbol{I}\right)^{-1}\boldsymbol{u}^{\mathrm{T}}\right\}\right]. \quad (4.33)$$

In analogy to $\mathrm{PVC}^{\mathrm{A}}$ criterion (see Algorithm 2), $c/2$ points for $\boldsymbol{u}$ can be selected sequentially by maximizing the $\mathrm{PVC}^{\mathrm{MIN}}$ function, denoted as $\underline{\boldsymbol{u}}^{(n+i)}$ $(i = 1, 2, \ldots, c/2)$. The stopping criterion is defined as $\frac{\sigma_{\hat{\mathcal{M}}}(\underline{\boldsymbol{v}}^\star)}{|\mu_{\hat{\mathcal{M}}}(\underline{\boldsymbol{v}}^\star)|} < \varepsilon^{\mathrm{BPI}}$, which should be sissified two times in succession. The identified points for $\boldsymbol{w}$ can be simply formed as: $(\underline{\boldsymbol{u}}^{(n+1)},\underline{\boldsymbol{v}}^\star)$, $(\underline{\boldsymbol{u}}^{(n+2)},\underline{\boldsymbol{v}}^\star)$, $\cdots$, $(\underline{\boldsymbol{u}}^{(n+c/2)},\underline{\boldsymbol{v}}^\star)$.

Similar to Eqs. (4.30) and (4.32), the expected improvement and posterior variance contribution

for maximization can also be defined, which are denoted as $\mathrm{EI}^{\mathrm{MAX}}$ and $\mathrm{PVC}^{\mathrm{MAX}}$, respectively. To limit the paper length, however, we will not give them in detail. The next point for $\boldsymbol{v}$ can be determined by maximizing the $\mathrm{EI}^{\mathrm{MAX}}$ function, i.e., $\overline{\boldsymbol{v}}^{\star} = \arg\max_{\boldsymbol{v}\in\mathcal{V}} \mathrm{EI}^{\mathrm{MAX}}(\boldsymbol{v})$. Then, based on the $\mathrm{PVC}^{\mathrm{MAX}}$ function, one can sequentially identify $c/2$ points for $\boldsymbol{u}$, denoted as $\overline{\boldsymbol{u}}^{(n+i)}$ ($i = 1, 2, \ldots, c/2$). The remaining $c/2$ points for $\boldsymbol{w}$ can be generated as: $(\overline{\boldsymbol{u}}^{(n+1)}, \overline{\boldsymbol{v}}^{\star})$, $(\overline{\boldsymbol{u}}^{(n+2)}, \overline{\boldsymbol{v}}^{\star})$, $\cdots$, $(\overline{\boldsymbol{u}}^{(n+c/2)}, \overline{\boldsymbol{v}}^{\star})$

As a result, a total number of c points for $\boldsymbol{w}$ can be obtained, and the corresponding $\mathcal{G}$-function values can be computed at the same time by running on c cores simultaneously. After that, the GP model can be updated based on the past observations. Once pre-defined stopping criteria are reached, these quantities of interest can be extracted from the finial GP model.

4.3.5 Numerical implementation of PBQO

For numerical implementation of the proposed PBQO method, the basic procedures are summarized as follows, which are also illustrated by Fig. 4.1.

Step 1: Get initial observations

The first step consists of generating a small set of n_0 initial samples using Latin hypercube sampling (LHS), denoted as $\boldsymbol{\mathcal{W}} = (\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) = \left\{\boldsymbol{w}^{(j)}\right\}_{i=1}^{n_0}$. The real $\mathcal{G}$-function is then evaluated at these points to obtain corresponding observations, i.e., $\boldsymbol{\mathcal{Z}} = \left\{z^{(i)} = \mathcal{G}\left(\boldsymbol{w}^{(i)}\right)\right\}_{j=1}^{n_0}$, which can be parallelized straightforwardly. The initial training dataset can be constructed: $\boldsymbol{\mathcal{D}} = \{\boldsymbol{\mathcal{W}}, \boldsymbol{\mathcal{Z}}\}$. Let $n = n_0$;

Step 2: Train a GP model

Based on data $\boldsymbol{\mathcal{D}}$, train a new GP model $\mathcal{GP}(\mu_n(\boldsymbol{w}), k_n(\boldsymbol{w}, \boldsymbol{w}'))$ for the $\mathcal{G}$-function. In this study, the *fitrgp* function in Matlab Statistics and Machine Learning Toolbox is used. The prior mean function and covariance function are specified as constant and squared exponential kernel, respectively.

Step 3: Check the stopping criterion

From the trained GP model, one can compute the posterior mean $\mu_{\hat{\mathcal{M}}_0}$ and posterior variance $\sigma^2_{\hat{\mathcal{M}}_0}$ of the augmented expectation by Eqs. (4.16) and (4.17), respectively. If the stopping criterion $\frac{\sigma_{\hat{\mathcal{M}}_0}}{|\mu_{\hat{\mathcal{M}}_0}|} < \varepsilon^{\mathrm{BPI}}$ is satisfied two times in succession, go to **Step 5**; else, go to **Step 4**;

Step 4: Identify new observations by the PVC$^{\mathrm{A}}$ criterion

At this stage, one can identify c points for $\boldsymbol{W}$ by sequentially maximizing the PVC$^{\mathrm{A}}$ function (Eq. (4.28)), denoted as $\boldsymbol{\mathcal{W}}^{\star} = \{\boldsymbol{w}^{\star}\}_{j=1}^{c}$. Then, these points are evaluated on the real $\mathcal{G}$-function in parallel to obtain corresponding observations, which are denoted as $\boldsymbol{\mathcal{Z}}^{\star} = \{z^{\star}\}_{j=1}^{c}$. At last, the training dataset $\boldsymbol{\mathcal{D}}$ can be enriched with $\boldsymbol{\mathcal{D}}^{\star} = \{\boldsymbol{\mathcal{W}}^{\star}, \boldsymbol{\mathcal{Z}}^{\star}\}$. Let $n = n + c$ and go to **Step 2**;

Step 5: Select new points by the quadruplet criteria

The next best points $\underline{\boldsymbol{v}}^{\star}$ and $\overline{\boldsymbol{v}}^{\star}$ can be selected by $\underline{\boldsymbol{v}}^{\star} = \arg\max_{\boldsymbol{v}\in\mathcal{V}} \mathrm{EI}^{\mathrm{MIN}}(\boldsymbol{v})$ and $\overline{\boldsymbol{v}}^{\star} = \arg\max_{\boldsymbol{v}\in\mathcal{V}} \mathrm{EI}^{\mathrm{MAX}}(\boldsymbol{v})$, respectively. Then, one can select $c/2$ points ($\underline{\boldsymbol{u}}^{(i)}$ $(i = 1, 2, \ldots, c/2)$) and ($\overline{\boldsymbol{u}}^{(i)}$ $(i = 1, 2, \ldots, c/2)$) by sequentially maximizing the PVC$^{\mathrm{MIN}}$ function and PVC$^{\mathrm{MAX}}$ function, respectively. For convenience, we denote the $c/2$ points for minimization by $\underline{\boldsymbol{\mathcal{W}}} = \left\{\left(\underline{\boldsymbol{v}}^{\star}, \underline{\boldsymbol{u}}^{(1)}\right), \left(\underline{\boldsymbol{v}}^{\star}, \underline{\boldsymbol{u}}^{(2)}\right), \cdots, \left(\underline{\boldsymbol{v}}^{\star}, \underline{\boldsymbol{u}}^{(c/2)}\right)\right\}$, $c/2$ points for maximization by $\overline{\boldsymbol{\mathcal{W}}} = \{\left(\overline{\boldsymbol{v}}^{\star}, \overline{\boldsymbol{u}}^{(1)}\right), \left(\overline{\boldsymbol{v}}^{\star}, \overline{\boldsymbol{u}}^{(2)}\right), \cdots, \left(\overline{\boldsymbol{v}}^{\star}, \overline{\boldsymbol{u}}^{(c/2)}\right)\}$, and $\boldsymbol{\mathcal{W}}^{\star} = \left\{\underline{\boldsymbol{\mathcal{W}}}, \overline{\boldsymbol{\mathcal{W}}}\right\}$;

Step 6: Judge the stopping criteria

In this step, four stopping criteria should be judged, i.e., $\frac{|\max_{\boldsymbol{v}\in\mathcal{V}} \mathrm{EI}^{\mathrm{MIN}}(\boldsymbol{v})|}{\max \boldsymbol{\mathcal{Z}} - \min \boldsymbol{\mathcal{Z}}} < \varepsilon^{\mathrm{BGO}}$, $\frac{\sigma_{\hat{\mathcal{M}}}(\underline{\boldsymbol{v}}^{\star})}{|\mu_{\hat{\mathcal{M}}}(\underline{\boldsymbol{v}}^{\star})|} < \varepsilon^{\mathrm{BPI}}$, $\frac{|\max_{\boldsymbol{v}\in\mathcal{V}} \mathrm{EI}^{\mathrm{MAX}}(\boldsymbol{v})|}{\max \boldsymbol{\mathcal{Z}} - \min \boldsymbol{\mathcal{Z}}} < \varepsilon^{\mathrm{BGO}}$ and $\frac{\sigma_{\hat{\mathcal{M}}}(\overline{\boldsymbol{v}}^{\star})}{|\mu_{\hat{\mathcal{M}}}(\overline{\boldsymbol{v}}^{\star})|} < \varepsilon^{\mathrm{BPI}}$. If all these stopping criteria are met two times in succession, go to **Step 9**; else, go to **Step 7**;

Step 7: Obtain new observations by parallel computing

Evaluation of the real $\mathcal{G}$-function at these c points $\boldsymbol{\mathcal{W}}^{\star}$ from **Step 5** can be performed in parallel, and c observations are obtained $\boldsymbol{\mathcal{Z}}^{\star} = \{z^{\star}\}_{j=1}^{c}$. Finally, the training dataset $\boldsymbol{\mathcal{D}}$ is updated with the new data $\boldsymbol{\mathcal{D}}^{\star} = \{\boldsymbol{\mathcal{W}}^{\star}, \boldsymbol{\mathcal{Z}}^{\star}\}$. Let $n = n + c$;

Step 8: Train a GP model

Train a new GP model $\mathcal{GP}(\mu_n(\boldsymbol{w}), k_n(\boldsymbol{w}, \boldsymbol{w}'))$ for the $\mathcal{G}$-function with data $\boldsymbol{\mathcal{D}}$, and go to **Step 5:**.

Step 9: Return quantities of interest

The posterior means and variances of these quantities of interest, such as REF, its RS-HDMR component functions and bounds, can be extracted form the trained GP model. The posterior means can serve as estimates for these quantities, and the posterior variances measure the epistemic uncertainties (numerical errors) about our estimates.

To initialize the algorithm, there parameters n_0, $\varepsilon^{\mathrm{BPI}}$ and $\varepsilon^{\mathrm{BGO}}$ need to be specified. The initial sample size n_0 should not choose too large as we wish to enlarge the sample size sequentially. For the

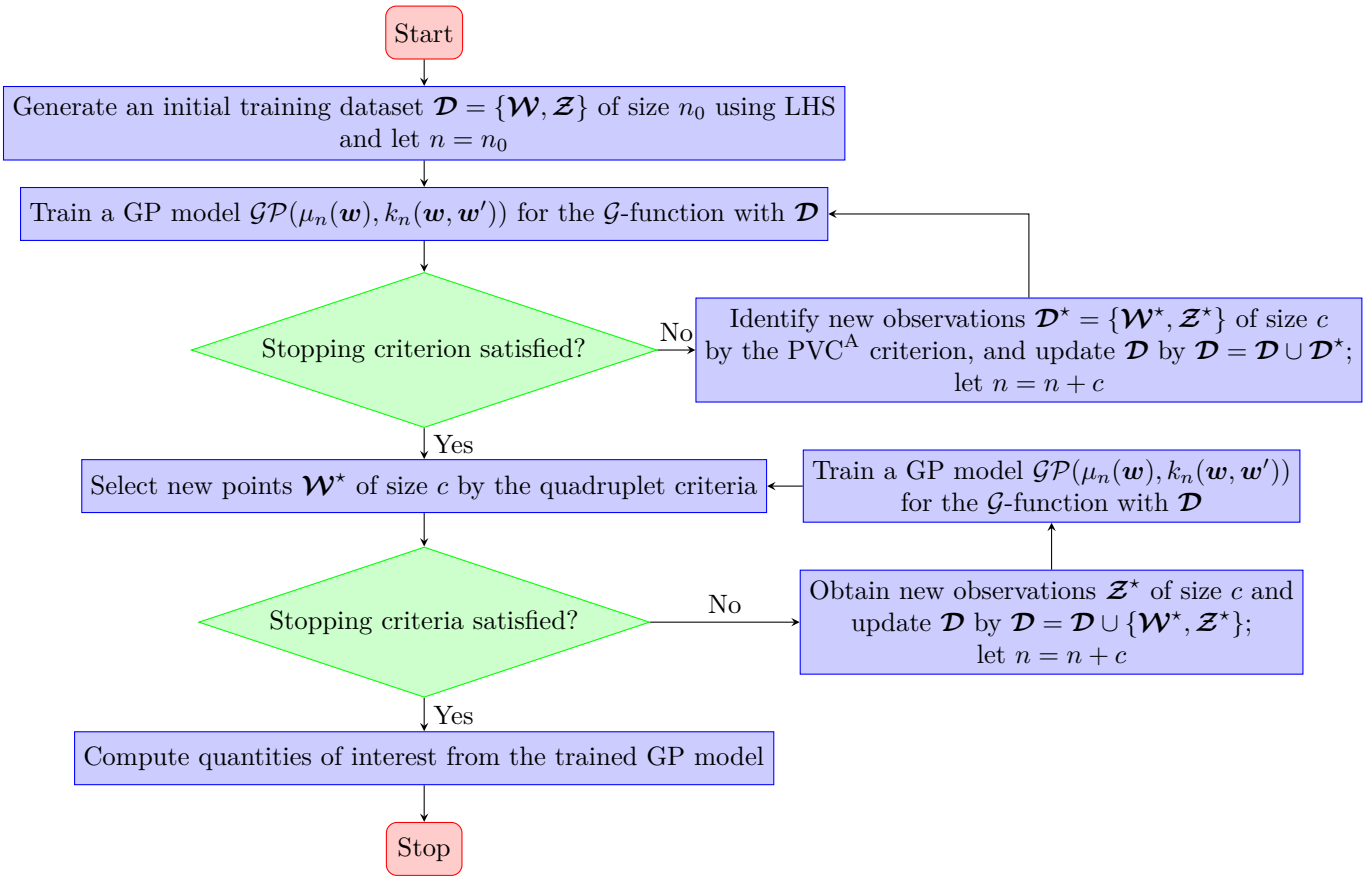


Figure 4.1: Flowchart of the proposed PBQO method.

two thresholds $\varepsilon^{\mathrm{BPI}}$ and $\varepsilon^{\mathrm{BGO}}$, proper values are also important as they influence the accuracy and efficiency of the proposed method. According to our experience, n_0 can take values between 5 and 20 depending on the complexity of the problem at hand, and $\varepsilon^{\mathrm{BPI}}$ and $\varepsilon^{\mathrm{BGO}}$ can be set in the orders of 0.01 and 0.001 respectively. Several optimization problems are involved in the implementation procedures, one can simply use the global optimization algorithms (e.g., genetic algorithm) as the objective functions are all in closed form.

4.3.6 Relationship to existing NIPI and CABO methods

The proposed PBQO method does share some similarities with the NIPI method and CABO method. For example, they all rely on the use of the GP model in a Bayesian fashion, and can avoid nested loops. However, the differences among the three methods are also significant on several main aspects:

a) The proposed PBQO method transforms the interval variables (including the interval variables in p-boxes) into standard interval ones by a linear transformation. On the contrary, by assuming auxiliary uniform distributions for the interval variables, the NIPI and CABO methods convert the interval variables to standard normal ones by a nonlinear transformation. In conjunction with the squared exponential kernel, both of those two strategies can result in analytically tractable results for the REF and its HS-HDMR. However, the NIPI and CABO methods introduce an additional assumption and artificially added nonlinearity. More importantly, the transformation strategy for NIPI and CABO is the cause of poor performance near the bounds of the interval variables. To mitigate this problem, one needs to relax the support of the interval variables when applying NIPI and CABO;

b) Due to the differences in a), the posterior means and variances of the REF and its RS-HDMR component functions are re-derived in the proposed PBQO method, along with some of the acquisition functions;

c) The proposed PBQO method is able to support parallel distributed processing owing to the proposed multi-point selection strategy, while both NIPI and CABO cannot. This advantage is desired when each evaluation of the $\mathcal{G}$-function is costly and parallel computing facilities are available;

d) The proposed PBQO method is capable of yielding the REF, its variable importance and bounds simultaneously with a single run. However, the NIPI method and CABO method are only designed for evaluating the variable importance and bounds, respectively.

4.4 Extending the proposed method to Case IV

The proposed PBQO method is mainly illustrated in case that hybrid uncertainties present as both random variables and interval variables. When parameterized p-boxes are involved, the proposed method is also applicable, but needs slight adaptations. In this section, we will show how to extend the proposed PBQO method established in Section 4.3 to **Case IV**.

Let $\boldsymbol{Y} = [Y_1, Y_2, \cdots, Y_{d_3}]$ denote an imprecise random vector containing d_3 variables. These variables are assumed to be characterized by parameterized p-boxes, and their joint PDF is denoted as $f_{\boldsymbol{Y}|\boldsymbol{\Theta}}(\boldsymbol{y}|\boldsymbol{\theta})$, which depends on a set of d_4 interval variables $\boldsymbol{\Theta} = [\Theta_1, \Theta_2, \cdots, \Theta_{d_4}]$ with lower and

upper bounds $\underline{\boldsymbol{\theta}} = [\underline{\theta}_1, \underline{\theta}_2, \cdots, \underline{\theta}_{d_4}]$ and $\overline{\boldsymbol{\theta}} = [\overline{\theta}_1, \overline{\theta}_2, \cdots, \overline{\theta}_{d_4}]$, respectively. In **Case IV**, the response function is represented by $Z = g(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{A})$. In analogy to **Case III**, an augmented response function $Z = g(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{A}, \boldsymbol{\Theta})$ needs to be artificially constructed to account for $\boldsymbol{\Theta}$ like $\boldsymbol{A}$. Then, we map the random vector $\{\boldsymbol{X}, \boldsymbol{Y}\}$ to a standard normal one $\boldsymbol{U}$, while the interval vector $\{\boldsymbol{A}, \boldsymbol{\Theta}\}$ to a standard interval one $\boldsymbol{V}$. Accordingly, the response function is changed to be $Z = \mathcal{G}(\boldsymbol{W})$, where $\boldsymbol{W} = \{\boldsymbol{U}, \boldsymbol{V}\}$. See, e.g., [39–41], for the details of how to use an augmented response function when parameterized p-boxes are involved. Note that this does not mean that the original g-function has to be modified, but only for numerical implementation. By doing so, the remaining procedures are similar to those given in Section 4.3.

4.5 Numerical examples

In this section, three numerical examples are investigated to demonstrate the proposed method. For comparison purposes, the NIPI and CABO methods are mainly implemented in all examples. These methods are used in a similar way as the proposed PBQO method even though they are originally developed for only propagating parameterized p-boxes. Besides, in both methods the support of interval variables has been increased by 10% and the stopping tolerances are specified in accordance with the proposed method.

4.5.1 Example 1: A test function

Consider a test function of the form:

$$Z = g(X, A_1, A_2) = X^2 + A_1 + A_2^3, \tag{4.34}$$

where X, A_1 and A_2 are three uncertain input variables, as listed in Tab. 4.1.

Table 4.1: Uncertainty characterization of input variables for Example 1.

Notation	Type	Mathematical model
X	Random variable	$\mathcal{N}(0, 1^2)$
A_1	Interval variable	[1 2]
A_2	Interval variable	[1 2]

Note: $\mathcal{N}$ stands for normal distribution.

We first consider the REF $m(\alpha_1, \alpha_2)$, the closed-form expression of which is obtained as

$m(\alpha_1, \alpha_2) = 1 + \alpha_1 + \alpha_2^3$. The proposed PBQO method can be implemented to yield a numerical estimate of $m(\alpha_1, \alpha_2)$. In this example, we set $c = 2$, $n_0 = 5$, $\varepsilon^{\text{BPI}} = 0.02$ and $\varepsilon^{\text{BGO}} = 0.002$. Fig. 4.2(a) depicts the REF estimated by PBQO v.s. its analytical solution, which coincide almost perfectly. Besides, as shown in Fig. 4.2(b) the coefficient of variation (COV) of the PBQO estimate is quite small, indicting that the estimate is highly reliable. In order to compare with other existing methods, we also employ the NIPI and CABO methods in this example. It can be seen from Figs. 4.2(c) and 4.2(e) that both NIPI and CABO methods give poor estimates for the ERF, especially in the boundary area. In addition, Figs. 4.2(d) and 4.2(f) show that the results by these two methods also process relatively large variability.

Second, the RS-HDMR component functions of the REF are of concern. For limiting the paper length, we just show the first-order RS-HDMR component functions as an illustration. The analytical expressions of $m_1(\alpha_1)$ and $m_1(\alpha_2)$ can be derived as: $m_1(\alpha_1) = -\frac{3}{2} + \alpha_1$ and $m_2(\alpha_2) = -\frac{15}{4} + \alpha_2^3$. From Fig. 4.3, one can observe that for both component functions: (1) the proposed PBQO method is able to yield very close estimates to analytical solutions; (2) the 99% confidence intervals (CIs) of PBQO estimates are very narrow; (3) the NIPI and CABO methods are shown to be less accurate than the proposed method; (4) the 99% CIs of both NIPI and CABO estimates are obviously wider than these by the proposed method. These observations demonstrate the accuracy of the proposed method against both NIPI and CABO methods. Besides, through the first-order RS-HDMR component functions it is easy to know that α_2 has significantly larger influence on the REF than α_1. Therefore, if one would like to reduce the epistemic uncertainty in the REF (i.e., narrow the interval), a more rational way is to shrink A_2 by collecting more data of it.

Third, we discuss the results of the response expectation bounds. The analytical lower and upper bounds of the REF are 3 and 11, respectively. Tab. 4.2 compares the numerical estimates given by the PBQO, NIPI and CABO methods to the analytical solutions. It can be seen that for both lower and upper bounds: (1) PBQO and CABO methods are capable of producing close estimates to the analytical solutions, and restively small posterior COVs; (2) NIPI method gives poor estimates with large posterior COV.

At last, the efficiency and accuracy of these three methods should be emphasized. As listed in Tab. 4.2, the number of response function evaluations for the PBQO, NIPI and CABO is 13, 8 and 22, respectively. However, the PBQO method can support for parallel computing, and hence its number of calls to the response function for each CPU core is only 6.5 on average. To this end,

the number of effective response function evaluations required by the proposed PBQO method is close to that of the NIPI method, but less than the CABO method. Besides, the proposed PBQO method is able to produce the REF, its RS-HDMR component functions and bound simultaneously with reasonable accuracy, while the NIPI method may perform worse in all these three aspects and the CABO method could be reliable only in capturing the REF bounds.

Table 4.2: Response expectation bound for Example 1.

Method	$\hat{m}_l$	COV$[\hat{m}_l]/\%$	$\hat{m}_u$	COV$[\hat{m}_u]$ /%	N	$\frac{N}{c}$
Analytical	3	-	11	-	-	-
PBQO ($c=2$)	2.9820	0.22	11.0027	0.00	5+8=13	6.5
NIPI ($c=1$)	2.6795	8.11	10.0148	2.26	5+3=8	8
CABO ($c=1$)	3.0033	0.08	10.9966	0.00	5+17=22	22

Note: N is the total number of response function evaluations; c is the number of points selected at each iteration, and hence the number of CPU cores used in parallel; and N/c is referred to as the number of effective respone function evaluations.

4.5.2 Example 2: A non-linear oscillator

The second example considers a nonlinear undamped single degree-of-freedom (SDOF) oscillator subjected to a rectangular pulse load (as shown in Fig. 4.4), which was extensively studied in context of reliability analysis (see, e.g., [39, 61, 62]). The response function is defined as the maximum displacement of the oscillator:

$$Z = g(c_1, c_2, m, F_1, t_1) = \left| \frac{2F_1}{c_1 + c_2} \sin\left(\frac{t_1}{2} \sqrt{\frac{c_1 + c_2}{m}} \right) \right|, \tag{4.35}$$

where c_1, c_2, m, F_1, t_1 are five uncertain input variables, detailed description of which can be found in Tab. 4.3. For notational clarity, we denote the three intervals as $A_1 = [1\ 2]$, $A_2 = [0.1\ 0.3]$ and $A_3 = [0.5\ 1.5]$ in what follows.

In this example, the REF, its RS-HDMR component functions and bounds are also of our interest. Due to the complexity of the response function, the corresponding analytical solutions are not available, and thus we use Monte Carlo simulation (MCS) or double-loop MCS (DL-MCS) [63] to provide reference results. The initial parameters of the proposed PBQO method are specified as: $c = 4$, $n_0 = 15$, $\varepsilon^{\mathrm{BPI}} = 0.01$ and $\varepsilon^{\mathrm{BGO}} = 0.001$. It should be noted that the REF is three-dimensional, and hence we simply set $\alpha_3 = 1$ in order to visualize the results. As can be seen

Table 4.3: Uncertainty characterization of input variables for Example 2.

Notation	Type	Mathematical model
c_1	Random variable	$\mathcal{N}(1, 0.1^2)$
c_2	Random variable	$\mathcal{N}(0.1, 0.01^2)$
m	Random variable	$\mathcal{N}(1, 0.1^2)$
F_1	P-box variable	$\mathcal{LN}([1\ 2], [0.1\ 0.3]^2)$
t_1	Interval variable	[0.5 1.5]

Note: $\mathcal{LN}$ stands for Lognormal distribution.

from Fig. 4.5(b), the COV of the MCS estimate is extremely small, indicting that we can take the MCS estimate as a reference result. From Figs. 4.5(a), 4.5(c) and 4.5(e), it is obvious that the proposed PBQO method can produce a much better REF estimate than the NIPI and CABO methods. Besides, the posterior COV of the PBQO estimate is also much smaller than those by NIPI and CABO methods, as shown in Figs. 4.5(b), 4.5(d) and 4.5(f). As for the RS-HDMR component functions of the REF, we only give three first-order RS-HDMR component functions $\hat{m}_1(\alpha_1)$, $\hat{m}_2(\alpha_2)$ and $\hat{m}_3(\alpha_3)$ as an illustration. It can be seen from Fig. 4.6 that for all the three component functions the proposed PBQO method can produce fairly good results, in comparison to these given by MCS. However, the NIPI and CABO methods perform much worse than PBQO, especially for $\hat{m}_2(\alpha_2)$. Tab. 4.4 compares the lower and upper bounds of the REF by different methods. As can be seen, the PBQO and CABO methods are able to yield desirable estimates with relatively small posterior COVs, while the NIPI method does not work well. It should be noted that the proposed method only requires 7.75 effective response function evaluations to produce the above results, which are less than those by NIPI and CABO.

Table 4.4: Response expectation bound for Example 2.

Method	$\hat{m}_l$	COV$[\hat{m}_l]$/%	$\hat{m}_u$	COV$[\hat{m}_u]$ /%	N	$\frac{N}{c}$
DL-MCS	0.4953	0.87	2.5766	0.37	10^6	-
PBQO ($c = 4$)	0.4583	0.35	2.5935	0.07	15+16=31	7.75
NIPI ($c = 1$)	0.4160	15.94	2.6343	3.04	15+3=18	18
CABO ($c = 1$)	0.4721	0.25	2.5866	0.08	15+24=39	38

4.5.3 Example 3: A 56-bar spatial truss structure

The third example consists of a 56-bar spatial truss structure, as shown in Fig. 4.7. Nine vertical loads are applied to the structure at joints 1-9, which are denoted as $P_1 \sim P_9$. The external

loads $P_1 - P_9$ are assumed to be uncertain, together with the elastic modulus E and cross-sectional area A. These uncertainties are characterized by three kinds of models, which are summarized in Tab. 4.5. It can be seen that four intervals are involved and we denote them as $A_1 = [20\ 30]$ kN, $A_2 = [30\ 40]$ kN, $A_3 = [200\ 220]$ Gpa and $A_4 = [150\ 250]$ mm^2. The response of concern is selected as the vertical displacement of joint 1, which can implicitly expressed as a function of $P_1 \sim P_9$, E and A, i.e., $Z = g(P_1 \sim P_9, E, A)$.

Table 4.5: Uncertainty characterization of input variables for Example 3.

Notation	Type	Unit	Mathematical model
$P_2 \sim P_9$	Random variable	kN	$\mathcal{LN}(20, 4^2)$
P_1	P-box variable	kN	$\mathcal{LN}([20\ 30], [30\ 40]^2)$
E	Interval variable	GPa	[200 220]
A	Interval variable	mm^2	[150 250]

The proposed PBQO method is initialized with $c = 4$, $n_0 = 20$, $\varepsilon^{\mathrm{BPI}} = 0.02$ and $\varepsilon^{\mathrm{BGO}} = 0.002$. Fig. 4.8 depicts the REF estimates by three methods and their corresponding posterior COVs, where we fix α_3 and α_4 at their midpoints, i.e., $\alpha_3 = 210$ Gpa and $\alpha_4 = 200$ mm^2. It is shown that the posterior COV of the PBQO estimate is much smaller that those by both NIPI and CABO methods, indicating that the proposed PBQO method is more reliable for capturing the REF. The results of four first-order HDMR component functions in Fig. 4.9 also imply that the proposed method has better accuracy than the NIPI and CABO methods. Besides, it is easy to know from Fig. 4.9 that the four intervals can be ranked as $A_4 > A_1 > A_3 > A_2$ in terms of their first-order importance to the REF. Through Tab. 4.6, one can find that for both lower and upper bounds of the REF the PBQO and CABO can yield better estimates than the NIPI, indicating by their posterior COVs. It should be emphasized that by taking advantage of parallel computing the effective response function calls required by the proposed PBQO method are much less than that of CABO.

Table 4.6: Response expectation bound for Example 3.

Method	$\hat{m}_l$	COV$[\hat{m}_l]/\%$	$\hat{m}_u$	COV$[\hat{m}_u]$ $/\%$	N	$\frac{N}{c}$
DL-MCS	11.1793	2.11	35.2535	1.64	10^4	-
PBQO ($c = 4$)	11.8785	0.23	35.4109	0.08	20+12=32	8
NIPI ($c = 1$)	12.5791	4.74	34.4007	2.36	20+2=22	22
CABO ($c = 1$)	11.5818	0.12	35.3302	0.07	20+23=43	43

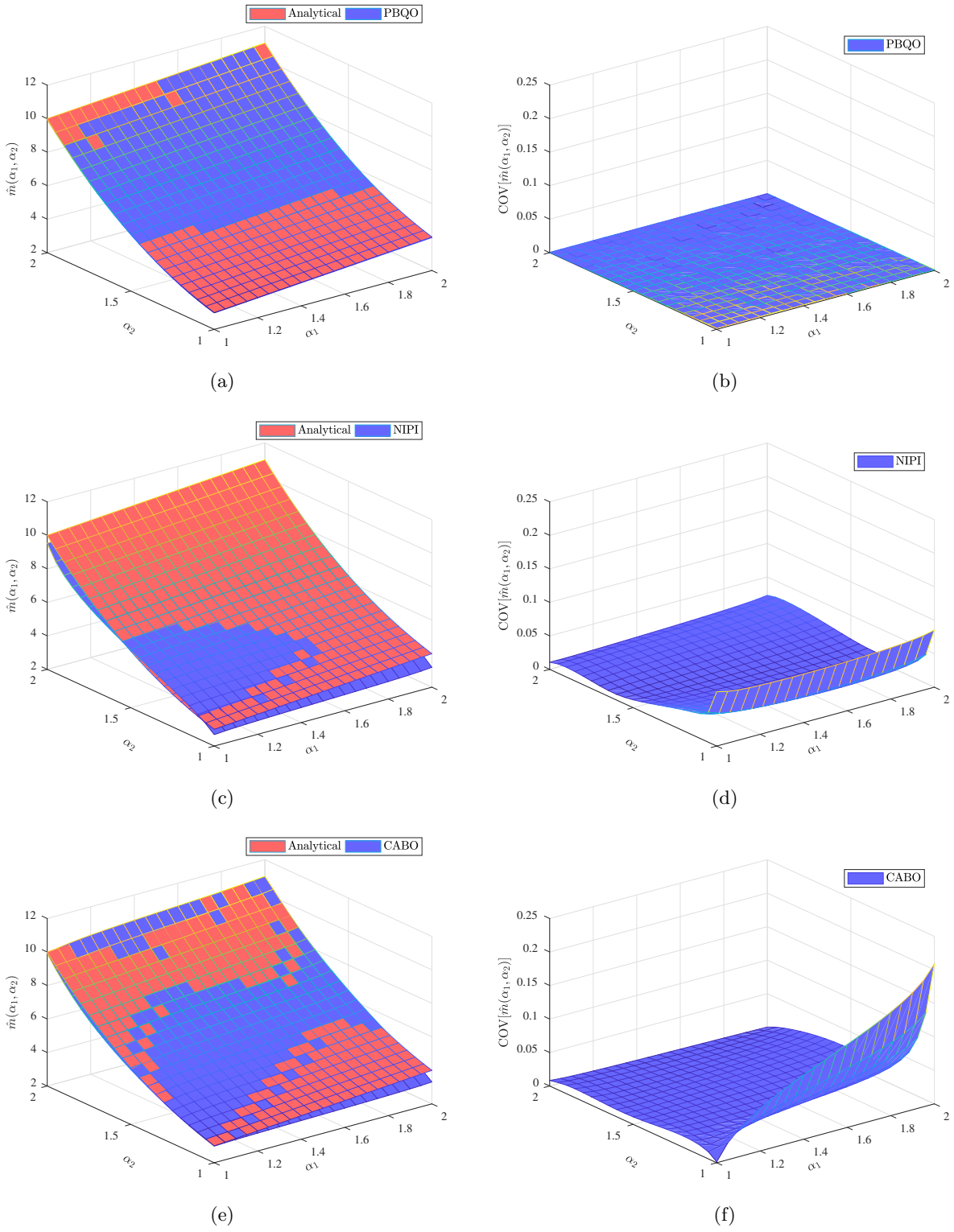
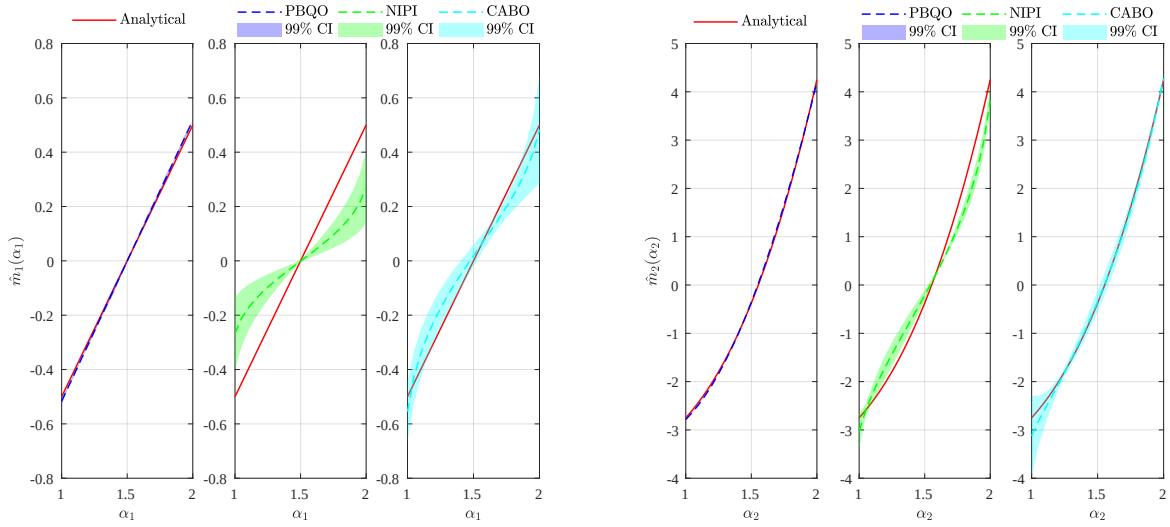


Figure 4.2: Response expectation function for Example 1 by different methods.



(a) $\hat{m}_1(\alpha_1)$ (b) $\hat{m}_2(\alpha_2)$

Figure 4.3: First-order RS-HDMR component functions for Example 1 by different methods.

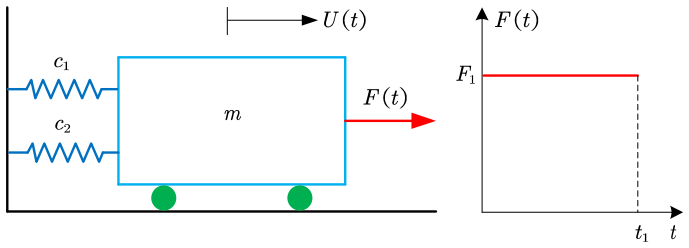


Figure 4.4: A nonlinear SDOF oscillator subjected to a rectangular pulse load.

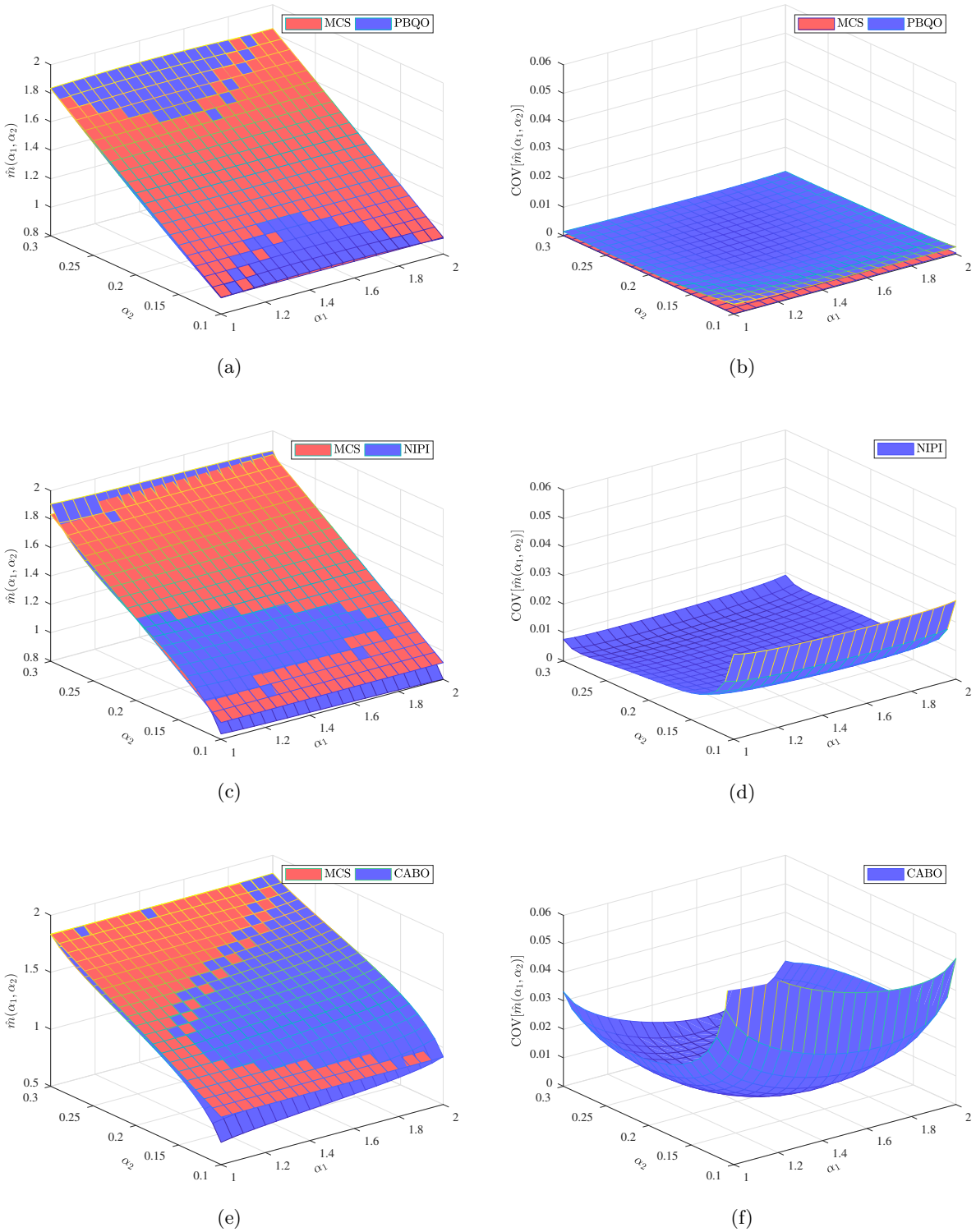
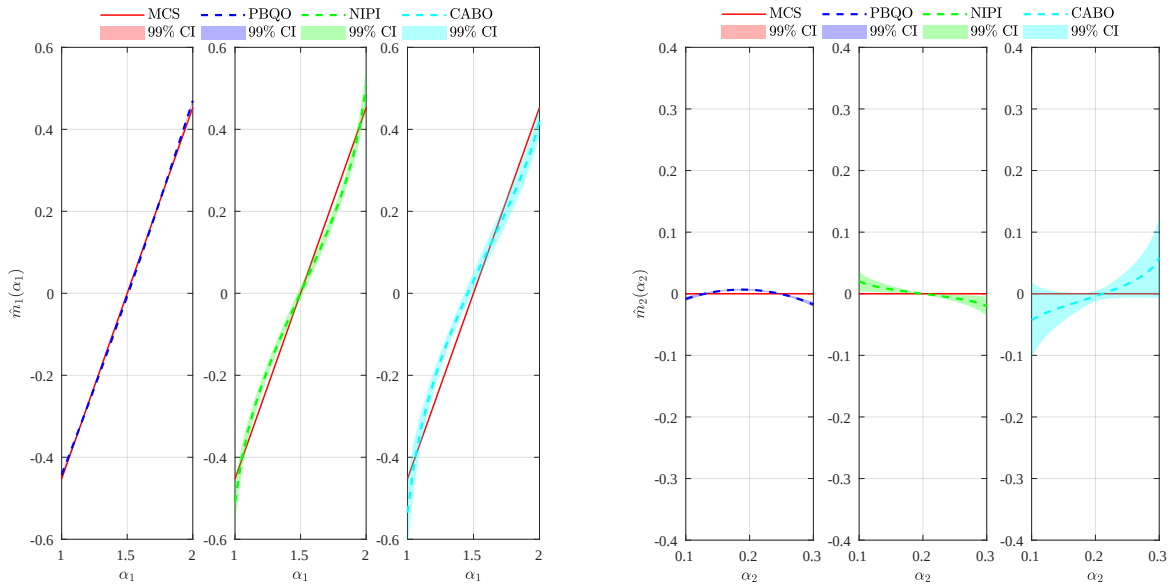
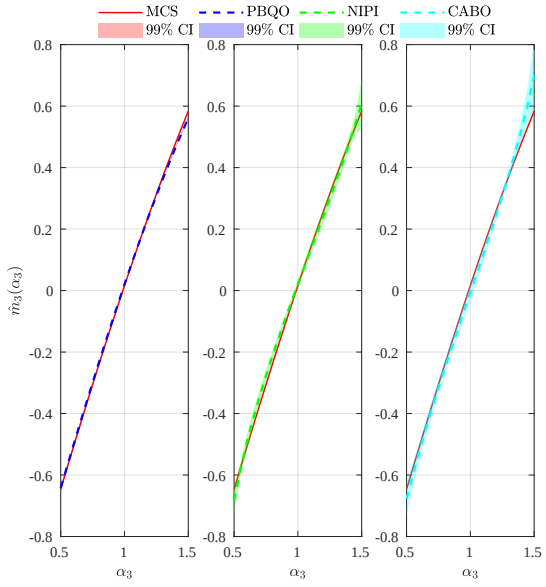


Figure 4.5: Response expectation function for Example 2 by different methods ($\alpha_3 = 1$).



(a) $\hat{m}_1(\alpha_1)$

(b) $\hat{m}_2(\alpha_2)$



(c) $\hat{m}_3(\alpha_3)$

Figure 4.6: First-order RS-HDMR component functions for Example 2 by different methods.

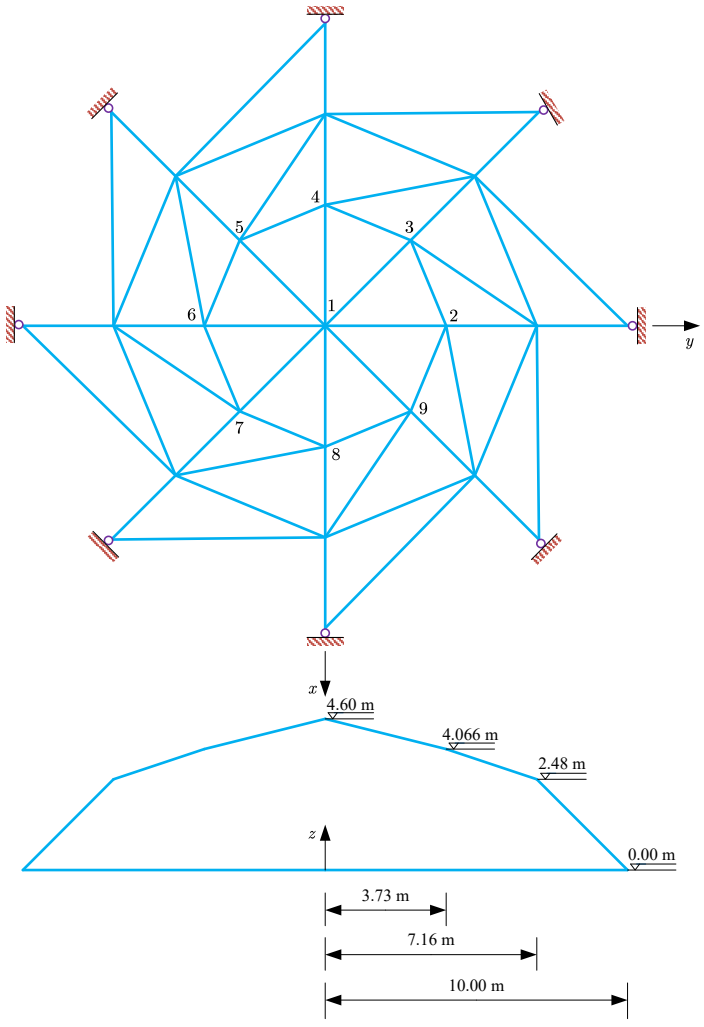


Figure 4.7: A 56-bar spatial truss structure.

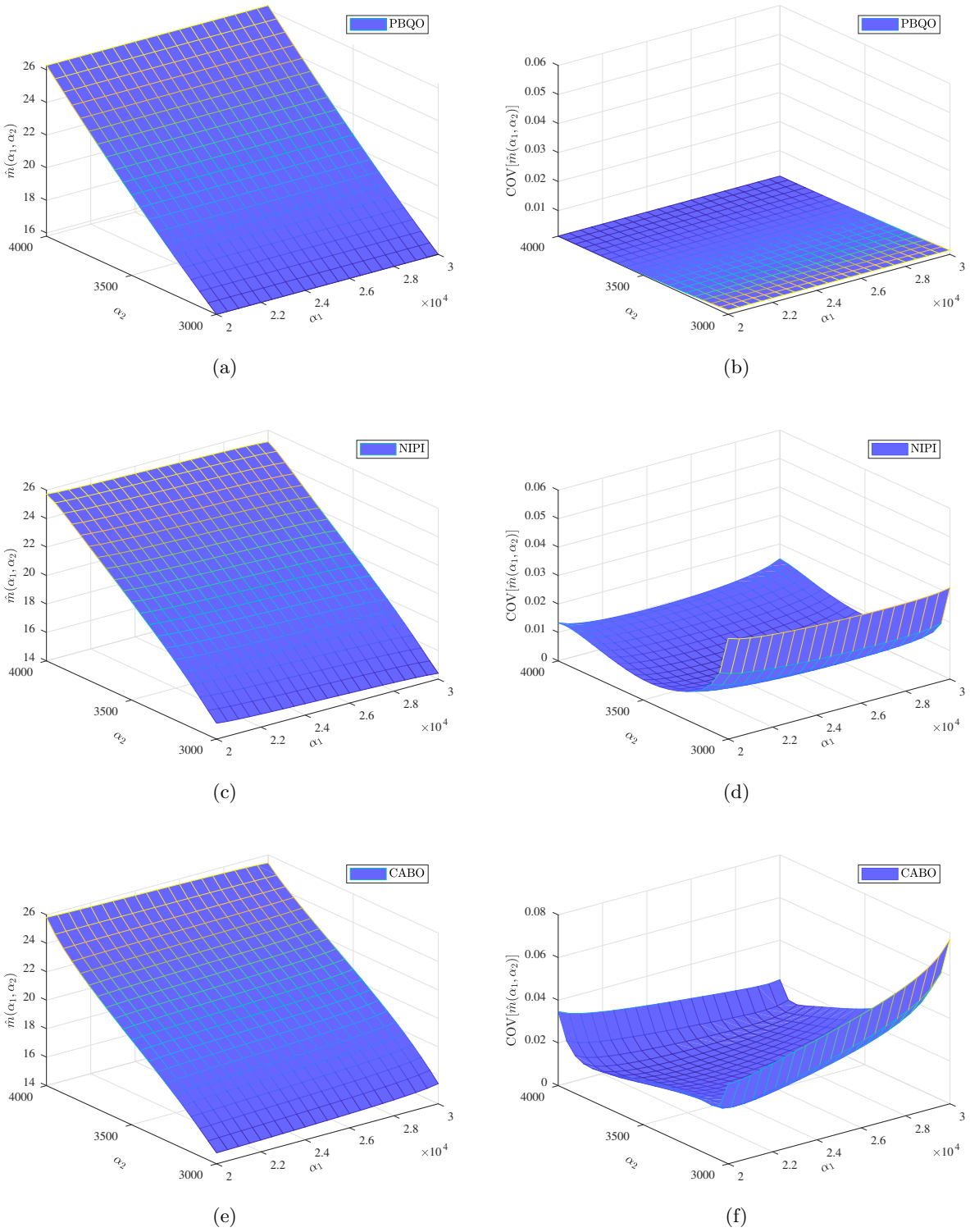
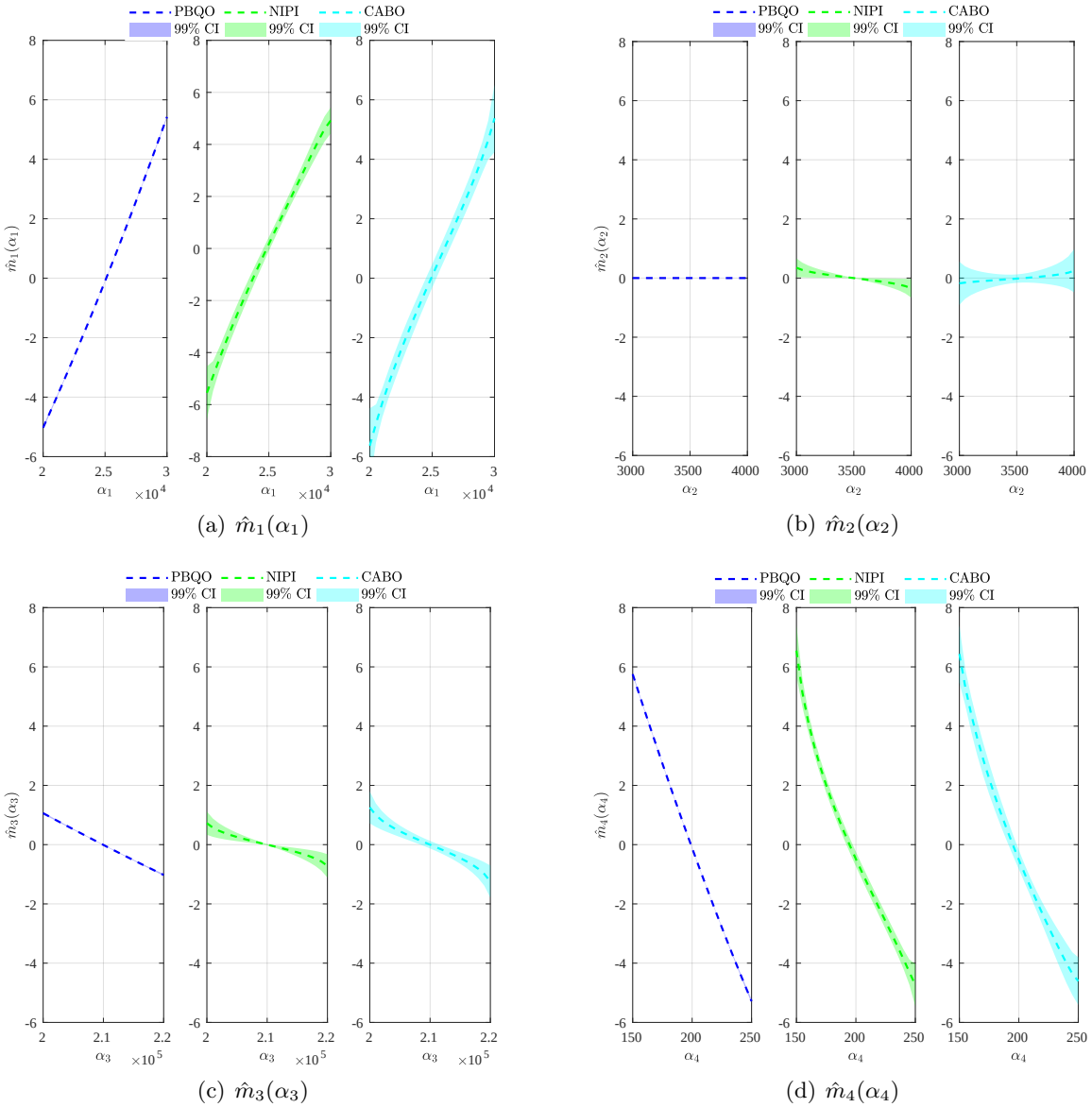


Figure 4.8: Response expectation function for Example 3 by different methods ($\alpha_3 = 210$ Gpa and $\alpha_4 = 200$ mm^2).



(a) $\hat{m}_1(\alpha_1)$ (b) $\hat{m}_2(\alpha_2)$

(c) $\hat{m}_3(\alpha_3)$ (d) $\hat{m}_4(\alpha_4)$

Figure 4.9: First-order RS-HDMR component functions for Example 3 by different methods.

4.6 Conclusions and perspectives

In this work, propagation of hybrid uncertainties in the form of precise random variables, parameterized p-boxes and interval variables is studied via Bayesian numerical analysis. The main contribution lies in the development of a novel method, termed 'Parallel Bayesian Quadrature Optimization', for estimation of response expectation function, its RS-HDMR component functions and bounds simultaneously. Compared to the state-of-the-art methods for propagating hybrid uncertainties, the proposed method has several significant advantages. First, the proposed method breaks the double-loop paradigm that typically propagates aleatory and epistemic uncertainty separately in a nested way. That is, it can propagate both types of uncertainties simultaneously, and is a fully-decoupled procedure in nature, yielding a major improvement in computational efficiency. Second, the proposed method is able to exploit prior knowledge thanks to its Bayesian nature, and it also supports parallel computing, further leading to much higher computational efficiency. Third, the estimators (i.e., posterior means) of the response moment function and its RS-HDMR component functions are analytically derived, together with their posterior variances for indicating numerical errors.

While these advantages are encouraging, there are still some issues that need further study. For example, one should note that the analytical tractability of the proposed method is based on using the squared exponential kernel that is appropriate for modelling smooth and moderately nonlinear functions. This, however, is not always justified for a general practical problem. Besides, the proposed method relies on a total number of five acquisition functions, which could be reduced by developing more efficient Bayesian experimental design strategies. The proposed method could be extended to evaluate the second-order raw moment function, while more research efforts may still be required.

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 and 72171194). Chao Dang, Pengfei Wei and Michael Beer also would like to appreciate the support of Sino-German Mobility Program under grant number M-0175.

Appendix A: Derivation of the posterior variance for the first-order RS-HDMR component function

The posterior variance function $\sigma^2_{\hat{\mathcal{M}}_i}(v_i)$ for the first-order RS-HDMR function $\hat{\mathcal{M}}_i(v_i)$ can be given by:

$$\sigma^2_{\hat{\mathcal{M}}_i}(v_i) = \mathbb{V}_{\boldsymbol{\mathcal{D}}}\left[\hat{\mathcal{M}}_i(v_i)\right] = \mathbb{V}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_i}\left[\mathcal{G}(\boldsymbol{w})\right]\right] + \sigma^2_{\hat{\mathcal{M}}_0} - 2\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_i}\left[\hat{\mathcal{G}}(\boldsymbol{w})\right], \hat{\mathcal{M}}_0\right], \tag{4.36}$$

where $\mathbb{COV}_{\boldsymbol{\mathcal{D}}}[\cdot,\cdot]$ refers to the covariance taken with respect to the posterior distributions of its arguments given data $\boldsymbol{\mathcal{D}}$; the term $\sigma^2_{\hat{\mathcal{M}}_0}$ has been given in Eq. (4.17).

The term $\mathbb{V}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_i}\left[\mathcal{G}(\boldsymbol{w})\right]\right]$ in Eq. (4.36) can be further deduced by applying Fubini's theorem such that:

$$\mathbb{V}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_i}\left[\mathcal{G}(\boldsymbol{w})\right]\right] = \Pi_{-v_i}\Pi_{-v_i}\left[k_0\left(\boldsymbol{w}, \left(\boldsymbol{w}'_{-v'_i}, v_i\right)\right)\right] - \Pi_{-v_i}\left[\boldsymbol{k}_0\left(\boldsymbol{w}, \boldsymbol{\mathcal{W}}\right)^{\mathrm{T}}\right]\boldsymbol{K}_0^{-1}\Pi_{-v_i}\left[\boldsymbol{k}_0\left(\boldsymbol{w}, \boldsymbol{\mathcal{W}}\right)\right], \tag{4.37}$$

where the term $\Pi_{-v_i}\left[\boldsymbol{k}_0\left(\boldsymbol{w}, \boldsymbol{\mathcal{W}}\right)\right]$ has been given in Eq. (4.21); the term $\Pi_{-v_i}\Pi_{-v_i}\left[k_0\left(\boldsymbol{w}, \left(\boldsymbol{w}'_{-(d_1+i)}, v_i\right)\right)\right]$ can be derived as:

$$\begin{aligned}
&\Pi_{-v_i}\Pi_{-v_i}\left[k_0\left(\boldsymbol{w}, \left(\boldsymbol{w}'_{-(d_1+i)}, v_i\right)\right)\right] \\
=&s_0^2\left|2\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1} + \boldsymbol{I}\right|^{-1/2} \\
&\cdot 2^{(d_2-1)}\mathrm{prod}_1\left\{\mathrm{diag}\left\{\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}}\left[-1 + \exp\left[-(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}})^{-1}\right] + (2\pi^{-1}\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}})^{-1/2}\mathrm{erf}\left((2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}})^{-1/2}\right)\right]\right\}\right\}.
\end{aligned} \tag{4.38}$$

Likewise, the term $\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_i}\left[\hat{\mathcal{G}}(\boldsymbol{w})\right], \hat{\mathcal{M}}_0\right]$ in Eq. (4.36) can be formulated as:

$$\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_i}\left[\hat{\mathcal{G}}(\boldsymbol{w})\right], \hat{\mathcal{M}}_0\right] = \Pi_{-v_i}\Pi\left[k_0\left(\boldsymbol{w}, \boldsymbol{w}'\right)\right] - \Pi_{-v_i}\left[\boldsymbol{k}_0\left(\boldsymbol{w}, \boldsymbol{\mathcal{W}}\right)^{\mathrm{T}}\right]\boldsymbol{K}_0^{-1}\Pi\left[\boldsymbol{k}_0\left(\boldsymbol{w}, \boldsymbol{\mathcal{W}}\right)\right], \tag{4.39}$$

where

$$\begin{aligned}
&\Pi_{-v_i}\Pi\left[k_0\left(\boldsymbol{w},\boldsymbol{w}'\right)\right]\\
=&s_0^2\left|2\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1}+\boldsymbol{I}\right|^{-1/2}\\
&\cdot 2^{(d_2-1)}\mathrm{prod}_1\left\{\mathrm{diag}\left\{\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}}\left[-1+\exp\left[-(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}})^{-1}\right]+(2\pi^{-1}\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}})^{-1/2}\mathrm{erf}\left((2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-i}})^{-1/2}\right)\right]\right\}\right\}\\
&\cdot\left(\frac{\pi}{2}\right)^{1/2}\mathrm{prod}_2\left\{\left[\mathrm{erf}\left((1-v_i)\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_i}\right)^{-1/2}\right)-\mathrm{erf}\left(-v_i\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_i}\right)^{-1/2}\right)\right]\boldsymbol{\Sigma}_{\boldsymbol{v}_i}^{1/2}\right\}.
\end{aligned}\tag{4.40}$$

Appendix B: Derivation of the posterior variance for the second-order RS-HDMR component function

The posterior variance function $\sigma^2_{\hat{\mathcal{M}}_{ij}}(v_i,v_j)$ for the second-order RS-HDMR component function $\hat{\mathcal{M}}_{ij}(v_i,v_j)$ can be formulated as:

$$\begin{aligned}
\sigma^2_{\hat{\mathcal{M}}_{ij}}(v_i,v_j)=&\mathbb{V}_{\boldsymbol{\mathcal{D}}}\left[\hat{\mathcal{M}}_{ij}(v_i,v_j)\right]\\
=&\mathbb{V}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-\boldsymbol{v}_{ij}}\left[\mathcal{G}(\boldsymbol{w})\right]\right]+\sigma^2_{\hat{\mathcal{M}}_i}(v_i)+\sigma^2_{\hat{\mathcal{M}}_j}(v_j)+\sigma^2_{\hat{\mathcal{M}}_0}\\
&-2\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-\boldsymbol{v}_{ij}}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi_{-v_i}\left[\mathcal{G}(\boldsymbol{w})\right]\right]-2\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-\boldsymbol{v}_{ij}}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi_{-v_j}\left[\mathcal{G}(\boldsymbol{w})\right]\right]\\
&+2\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-\boldsymbol{v}_{ij}}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi\left[\mathcal{G}(\boldsymbol{w})\right]\right]+2\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_i}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi_{-v_j}\left[\mathcal{G}(\boldsymbol{w})\right]\right]\\
&-2\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_i}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi\left[\mathcal{G}(\boldsymbol{w})\right]\right]-2\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_j}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi\left[\mathcal{G}(\boldsymbol{w})\right]\right],
\end{aligned}\tag{4.41}$$

where the terms $\sigma^2_{\hat{\mathcal{M}}_i}(v_i)$ and $\sigma^2_{\hat{\mathcal{M}}_j}(v_j)$ can refer to Eq. (4.36); the term $\sigma^2_{\hat{\mathcal{M}}_0}$ has been derived in Eq. (4.17); the last two covariance terms $\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_i}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi\left[\mathcal{G}(\boldsymbol{w})\right]\right]$ and $\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_j}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi\left[\mathcal{G}(\boldsymbol{w})\right]\right]$ has been given in Eq. (4.39).

The term $\mathbb{V}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-\boldsymbol{v}_{ij}}\left[\mathcal{G}(\boldsymbol{w})\right]\right]$ in Eq. (4.41) can be derived as:

$$\mathbb{V}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-\boldsymbol{v}_{ij}}\left[\mathcal{G}(\boldsymbol{w})\right]\right]=\Pi_{-\boldsymbol{v}_{ij}}\Pi_{-\boldsymbol{v}_{ij}}\left[k_0\left(\boldsymbol{w},\left(\boldsymbol{w}'_{-\boldsymbol{v}'_{ij}},\boldsymbol{v}_{ij}\right)\right)\right]-\Pi_{-\boldsymbol{v}_{ij}}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)^{\mathrm{T}}\right]\boldsymbol{K}_0^{-1}\Pi_{-\boldsymbol{v}_{ij}}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right],\tag{4.42}$$

where the term $\Pi_{-\boldsymbol{v}_{ij}}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right]$ has been given in Eq. (4.23); the term $\Pi_{-\boldsymbol{v}_{ij}}\Pi_{-\boldsymbol{v}_{ij}}\left[k_0\left(\boldsymbol{w},\left(\boldsymbol{w}'_{-\boldsymbol{v}'_{ij}},\boldsymbol{v}_{ij}\right)\right)\right]$

can be derived as:

$$\begin{aligned}
&\Pi_{-\boldsymbol{v}_{ij}}\Pi_{-\boldsymbol{v}_{ij}}\left[k_0\left(\boldsymbol{w},\left(\boldsymbol{w}'_{-\boldsymbol{v}'_{ij}},\boldsymbol{v}_{ij}\right)\right)\right]\\
=&s_0^2\left|2\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1}+\boldsymbol{I}\right|^{-1/2}\\
&\cdot 2^{(d_2-2)}\mathrm{prod}_1\left\{\mathrm{diag}\left\{\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}}\left[-1+\exp\left[-(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}})^{-1}\right]+(2\pi^{-1}\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}})^{-1/2}\mathrm{erf}\left(\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}}\right)^{-1/2}\right)\right]\right\}\right\}.
\end{aligned} \tag{4.43}$$

The term $\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-\boldsymbol{v}_{ij}}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi_{-v_i}\left[\mathcal{G}(\boldsymbol{w})\right]\right]$ in Eq. (4.41) is formulated as:

$$\begin{aligned}
\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-\boldsymbol{v}_{ij}}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi_{-v_i}\left[\mathcal{G}(\boldsymbol{w})\right]\right]=&\Pi_{-\boldsymbol{v}_{ij}}\Pi_{-v_i}\left[k_0\left(\boldsymbol{w},\left(\boldsymbol{w}'_{-v'_i},v_i\right)\right)\right]\\
&-\Pi_{-\boldsymbol{v}_{ij}}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)^{\mathrm{T}}\right]\boldsymbol{K}_0^{-1}\Pi_{-v_i}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right],
\end{aligned} \tag{4.44}$$

where the terms $\Pi_{-\boldsymbol{v}_{ij}}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right]$ and $\Pi_{-v_i}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right]$ have been given in Eq. (4.23) and Eq. (4.21) respectively; the term $\Pi_{-\boldsymbol{v}_{ij}}\Pi_{-v_i}\left[k_0\left(\boldsymbol{w},\left(\boldsymbol{w}'_{-v'_i},v_i\right)\right)\right]$ can be derived as:

$$\begin{aligned}
&\Pi_{-\boldsymbol{v}_{ij}}\Pi_{-v_i}\left[k_0\left(\boldsymbol{w},\left(\boldsymbol{w}'_{-v'_i},v_i\right)\right)\right]\\
=&s_0^2\left|2\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1}+\boldsymbol{I}\right|^{-1/2}\\
&\cdot 2^{(d_2-2)}\mathrm{prod}_1\left\{\mathrm{diag}\left\{\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}}\left[-1+\exp\left[-(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}})^{-1}\right]+(2\pi^{-1}\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}})^{-1/2}\mathrm{erf}\left(\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}}\right)^{-1/2}\right)\right]\right\}\right\}\\
&\cdot\left(\frac{\pi}{2}\right)^{1/2}\mathrm{prod}_2\left\{\left[\mathrm{erf}\left((1-v_j)\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_j}\right)^{-1/2}\right)-\mathrm{erf}\left(-v_j\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_j}\right)^{-1/2}\right)\right]\boldsymbol{\Sigma}_{\boldsymbol{v}_j}^{1/2}\right\}.
\end{aligned} \tag{4.45}$$

Note that the term $\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-\boldsymbol{v}_{ij}}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi_{-v_j}\left[\mathcal{G}(\boldsymbol{w})\right]\right]$ in Eq. (4.41) can be similarly derived as the term $\Pi_{-\boldsymbol{v}_{ij}}\Pi_{-v_i}\left[k_0\left(\boldsymbol{w},\left(\boldsymbol{w}'_{-(d_1+i)},v_i\right)\right)\right]$ given in Eq. (4.44).

The covariance term $\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-\boldsymbol{v}_{ij}}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi\left[\mathcal{G}(\boldsymbol{w})\right]\right]$ in Eq. (4.41) can be formulated as:

$$\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-\boldsymbol{v}_{ij}}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi\left[\mathcal{G}(\boldsymbol{w})\right]\right]=\Pi_{-\boldsymbol{v}_{ij}}\Pi\left[k_0\left(\boldsymbol{w},\boldsymbol{w}'\right)\right]-\Pi_{-\boldsymbol{v}_{ij}}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)^{\mathrm{T}}\right]\boldsymbol{K}_0^{-1}\Pi\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right], \tag{4.46}$$

where the terms $\Pi\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right]$ and $\Pi_{-\boldsymbol{v}_{ij}}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right]$ have been given in Eq. (4.18) and Eq. 4.23

respectively; the term $\Pi_{-\boldsymbol{v}_{ij}}\Pi\left[k_0\left(\boldsymbol{w},\boldsymbol{w}'\right)\right]$ can be derived as:

$$\begin{aligned}
&\Pi_{-\boldsymbol{v}_{ij}}\Pi\left[k_0\left(\boldsymbol{w},\boldsymbol{w}'\right)\right]\\
=&s_0^2\left|2\boldsymbol{\Sigma}_{\boldsymbol{u}}^{-1}+\boldsymbol{I}\right|^{-1/2}\\
&\cdot 2^{(d_2-2)}\mathrm{prod}_1\left\{\mathrm{diag}\left\{\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}}\left[-1+\exp\left[-(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}})^{-1}\right]+(2\pi^{-1}\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}})^{-1/2}\mathrm{erf}\left(\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{-ij}}\right)^{-1/2}\right)\right]\right\}\right\}\\
&\cdot\left(\frac{\pi}{2}\right)^{2/2}\mathrm{prod}_2\left\{\left[\mathrm{erf}\left((1-\boldsymbol{v}_{ij})\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{ij}}\right)^{-1/2}\right)-\mathrm{erf}\left(-\boldsymbol{v}_{ij}\left(2\boldsymbol{\Sigma}_{\boldsymbol{v}_{ij}}\right)^{-1/2}\right)\right]\boldsymbol{\Sigma}_{\boldsymbol{v}_{ij}}^{1/2}\right\}.
\end{aligned}\tag{4.47}$$

The covariance term $\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_i}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi_{-v_j}\left[\mathcal{G}(\boldsymbol{w})\right]\right]$ in Eq. (4.41) can be formulated as:

$$\begin{aligned}
\mathbb{COV}_{\boldsymbol{\mathcal{D}}}\left[\Pi_{-v_i}\left[\mathcal{G}(\boldsymbol{w})\right],\Pi_{-v_j}\left[\mathcal{G}(\boldsymbol{w})\right]\right]=&\Pi_{-v_i}\Pi_{-v_j}\left[k_0\left(\boldsymbol{w},\left(\boldsymbol{w}'_{-v'_j},v_j\right)\right)\right]\\
&-\Pi_{-v_i}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)^{\mathrm{T}}\right]\boldsymbol{K}_0^{-1}\Pi_{-v_j}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right],
\end{aligned}\tag{4.48}$$

where the terms $\Pi_{-v_i}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right]$ and $\Pi_{-v_j}\left[\boldsymbol{k}_0\left(\boldsymbol{w},\boldsymbol{\mathcal{W}}\right)\right]$ have been given in Eq. (4.21); the term $\Pi_{-v_i}\Pi_{-v_j}\left[k_0\left(\boldsymbol{w},\left(\boldsymbol{w}'_{-v'_j},v_j\right)\right)\right]$ is actually equal to $\Pi_{-\boldsymbol{v}_{ij}}\Pi\left[k_0\left(\boldsymbol{w},\boldsymbol{w}'\right)\right]$ as given in Eq. (4.47).

Bibliography

[1] Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural Safety*, 31(2):105–112, 2009.

[2] Michael Beer, Scott Ferson, and Vladik Kreinovich. Imprecise probabilities in engineering analyses. *Mechanical Systems and Signal Processing*, 37(1-2):4–29, 2013.

[3] Andreĭ Nikolaevich Kolmogorov and Albert T Bharucha-Reid. *Foundations of the theory of probability: Second English Edition*. Courier Dover Publications, 2018.

[4] David Moens and Dirk Vandepitte. A survey of non-probabilistic uncertainty treatment in finite element analysis. *Computer Methods in Applied Mechanics and Engineering*, 194(12-16):1527–1555, 2005.

[5] Matthias Faes and David Moens. Recent trends in the modeling and quantification of non-probabilistic uncertainty. *Archives of Computational Methods in Engineering*, 27(3):633–671, 2020.

[6] Chao Jiang, RG Bi, GY Lu, and X Han. Structural reliability analysis using non-probabilistic convex model. *Computer Methods in Applied Mechanics and Engineering*, 254:83–98, 2013.

[7] Thomas Augustin, Frank PA Coolen, Gert De Cooman, and Matthias CM Troffaes. *Introduction to imprecise probabilities*. John Wiley & Sons, 2014.

[8] Scott Ferson, Vladik Kreinovich, Lev Grinzburg, Davis Myers, and Kari Sentz. Constructing probability boxes and dempster-shafer structures. Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2015.

[9] Z Zhang and C Jiang. Evidence-theory-based structural reliability analysis with epistemic uncertainty: a review. *Structural and Multidisciplinary Optimization*, pages 1–19, 2021.

[10] Bernd Möller and Michael Beer. *Fuzzy randomness: uncertainty in civil engineering and computational mechanics*. Springer Science & Business Media, 2004.

[11] Siu-Kui Au and James L Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, 16(4):263–277, 2001.

[12] Michael D Shields and Jiaxin Zhang. The generalization of latin hypercube sampling. *Reliability Engineering & System Safety*, 148:96–108, 2016.

[13] Sebastian Geyer, Iason Papaioannou, and Daniel Straub. Cross entropy-based importance sampling using gaussian densities revisited. *Structural Safety*, 76:15–27, 2019.

[14] Behrooz Keshtegar and Zeng Meng. A hybrid relaxed first-order reliability method for efficient structural reliability analysis. *Structural Safety*, 66:84–93, 2017.

[15] Xianzhen Huang, Yuxiong Li, Yimin Zhang, and Xufang Zhang. A new direct second-order reliability analysis method. *Applied Mathematical Modelling*, 55:68–80, 2018.

[16] Benjamin Echard, Nicolas Gayton, and Maurice Lemaire. Ak-mcs: an active learning reliability method combining kriging and monte carlo simulation. *Structural Safety*, 33(2):145–154, 2011.

[17] Géraud Blatman and Bruno Sudret. Adaptive sparse polynomial chaos expansion based on least angle regression. *Journal of Computational Physics*, 230(6):2345–2367, 2011.

[18] Rui Teixeira, Maria Nogal, and Alan O'Connor. Adaptive approaches in metamodel-based reliability analysis: A review. *Structural Safety*, 89:102019, 2021.

[19] Yan-Gang Zhao and Tetsuro Ono. New point estimates for probability moments. *Journal of Engineering Mechanics*, 126(4):433–436, 2000.

[20] Sinan Xiao and Zhenzhou Lu. Reliability analysis by combining higher-order unscented transformation and fourth-moment method. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, 4(1):04017034, 2018.

[21] Jun Xu and Chao Dang. A new bivariate dimension reduction method for efficient structural reliability analysis. *Mechanical Systems and Signal Processing*, 115:281–300, 2019.

[22] Runyu Liu, Wenliang Fan, Yule Wang, Alfredo H-S Ang, and Zhengliang Li. Adaptive estimation for statistical moments of response based on the exact dimension reduction method in terms of vector. *Mechanical Systems and Signal Processing*, 126:609–625, 2019.

[23] Pengfei Wei, Xing Zhang, and Michael Beer. Adaptive experiment design for probabilistic integration. *Computer Methods in Applied Mechanics and Engineering*, 365:113035, 2020.

[24] Jie Li and JB Chen. Probability density evolution method for dynamic response analysis of structures with uncertain parameters. *Computational Mechanics*, 34(5):400–409, 2004.

[25] Guohai Chen and Dixiong Yang. A unified analysis framework of static and dynamic structural reliabilities based on direct probability integral method. *Mechanical Systems and Signal Processing*, 158:107783, 2021.

[26] David Moens and Michael Hanss. Non-probabilistic finite element analysis for parametric uncertainty treatment in applied mechanics: Recent advances. *Finite Elements in Analysis and Design*, 47(1):4–16, 2011.

[27] B Möller, W Graf, and M Beer. Fuzzy structural analysis using α-level optimization. *Computational Mechanics*, 26(6):547–565, 2000.

[28] Chao Dang, Pengfei Wei, Matthias GR Faes, Marcos A Valdebenito, and Michael Beer. Interval uncertainty propagation by a parallel bayesian global optimization method. *Applied Mathematical Modelling*, 108:220–235, 2022.

[29] Zhiping Qiu and Isaac Elishakoff. Antioptimization of structures with large uncertain-but-non-random parameters via interval analysis. *Computer Methods in Applied Mechanics and Engineering*, 152(3-4):361–372, 1998.

[30] Suhuan Chen, Huadong Lian, and Xiaowei Yang. Interval static displacement analysis for structures with interval parameters. *International Journal for Numerical Methods in Engineering*, 53(2):393–407, 2002.

[31] Robin RP Callens, Matthias GR Faess, and David Moens. Multilevel quasi-monte carlo for interval analysis. *International Journal for Uncertainty Quantification*, 12(4), 2022.

[32] BY Ni, C Jiang, PG Wu, ZH Wang, and WY Tian. A sequential simulation strategy for response bounds analysis of structures with interval uncertainties. *Computers & Structures*, 266:106785, 2022.

[33] Morgan Chase Bruns. *Propagation of imprecise probabilities through black box models.* PhD thesis, Georgia Institute of Technology, 2006.

[34] Hao Zhang, Robert L Mullen, and Rafi L Muhanna. Interval monte carlo methods for structural reliability. *Structural Safety*, 32(3):183–190, 2010.

[35] SK Au. Reliability-based design sensitivity by efficient simulation. *Computers & Structures*, 83(14):1048–1061, 2005.

[36] Pengfei Wei, Jingwen Song, Sifeng Bi, Matteo Broggi, Michael Beer, Zhenzhou Lu, and Zhufeng Yue. Non-intrusive stochastic analysis with parameterized imprecise probability models: I. performance estimation. *Mechanical Systems and Signal Processing*, 124:349–368, 2019.

[37] Pengfei Wei, Jingwen Song, Sifeng Bi, Matteo Broggi, Michael Beer, Zhenzhou Lu, and Zhufeng Yue. Non-intrusive stochastic analysis with parameterized imprecise probability models: II. reliability and rare events analysis. *Mechanical Systems and Signal Processing*, 126:227–247, 2019.

[38] Matthias GR Faes, Marcos A Valdebenito, David Moens, and Michael Beer. Bounding the first excursion probability of linear structures subjected to imprecise stochastic loading. *Computers & Structures*, 239:106320, 2020.

[39] Chao Dang, Pengfei Wei, Jingwen Song, and Michael 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):04021054, 2021.

[40] Pengfei Wei, Fuchao Liu, Marcos Valdebenito, and Michael Beer. Bayesian probabilistic propagation of imprecise probabilities with large epistemic uncertainty. *Mechanical Systems and Signal Processing*, 149:107219, 2021.

[41] Pengfei Wei, Fangqi Hong, Kok-Kwang Phoon, and Michael Beer. Bounds optimization of model response moments: a twin-engine bayesian active learning method. *Computational Mechanics*, 67(5):1273–1292, 2021.

[42] Matthias G.R. Faes, Marco Daub, Stefano Marelli, Edoardo Patelli, and Michael Beer. Engineering analysis with probability boxes: A review on computational methods. *Structural Safety*, 93:102092, 2021.

[43] Xufeng Yang, Yongshou Liu, Yi Gao, Yishang Zhang, and Zongzhan Gao. An active learning kriging model for hybrid reliability analysis with both random and interval variables. *Structural and Multidisciplinary Optimization*, 51(5):1003–1016, 2015.

[44] Xufeng Yang, Yongshou Liu, Yishang Zhang, and Zhufeng Yue. Hybrid reliability analysis with both random and probability-box variables. *Acta Mechanica*, 226(5):1341–1357, 2015.

[45] Xufeng Yang, Yongshou Liu, Yishang Zhang, and Zhufeng Yue. Probability and convex set hybrid reliability analysis based on active learning kriging model. *Applied Mathematical Modelling*, 39(14):3954–3971, 2015.

[46] Jinhao Zhang, Mi Xiao, Liang Gao, and Junjian Fu. A novel projection outline based active learning method and its combination with kriging metamodel for hybrid reliability analysis with random and interval variables. *Computer Methods in Applied Mechanics and Engineering*, 341:32–52, 2018.

[47] Xiao Chen and Zhiping Qiu. A novel uncertainty analysis method for composite structures with mixed uncertainties including random and interval variables. *Composite Structures*, 184:400–410, 2018.

[48] Mi Xiao, Jinhao Zhang, Liang Gao, Soobum Lee, and Amin Toghi Eshghi. An efficient kriging-based subset simulation method for hybrid reliability analysis under random and interval variables with small failure probability. *Structural and Multidisciplinary Optimization*, 59(6):2077–2092, 2019.

[49] Jingwen Song, Pengfei Wei, Marcos Valdebenito, Sifeng Bi, Matteo Broggi, Michael Beer, and Zuxiang Lei. Generalization of non-intrusive imprecise stochastic simulation for mixed uncertain variables. *Mechanical Systems and Signal Processing*, 134:106316, 2019.

[50] Xiukai Yuan, Matthias GR Faes, Shaolong Liu, Marcos A Valdebenito, and Michael Beer. Efficient imprecise reliability analysis using the augmented space integral. *Reliability Engineering & System Safety*, 210:107477, 2021.

[51] Matthias GR Faes, Marcos A Valdebenito, Xiukai Yuan, Pengfei Wei, and Michael Beer. Augmented reliability analysis for estimating imprecise first excursion probabilities in stochastic linear dynamics. *Advances in Engineering Software*, 155:102993, 2021.

[52] C Jiang, GY Lu, X Han, and LX Liu. A new reliability analysis method for uncertain structures with random and interval variables. *International Journal of Mechanics and Materials in Design*, 8(2):169–182, 2012.

[53] Matthias GR Faes, Marcos A Valdebenito, David Moens, and Michael Beer. Operator norm theory as an efficient tool to propagate hybrid uncertainties and calculate imprecise probabilities. *Mechanical Systems and Signal Processing*, 152:107482, 2021.

[54] C Jiang, J Zheng, and X Han. Probability-interval hybrid uncertainty analysis for structures with both aleatory and epistemic uncertainties: a review. *Structural and Multidisciplinary Optimization*, 57(6):2485–2502, 2018.

[55] Jon Cockayne, Chris J Oates, Timothy John Sullivan, and Mark Girolami. Bayesian probabilistic numerical methods. *SIAM Review*, 61(4):756–789, 2019.

[56] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.

[57] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[58] Carl Edward Rasmussen and Zoubin Ghahramani. Bayesian monte carlo. *Advances in Neural Information Processing Systems*, pages 505–512, 2003.

[59] Genyuan Li, Sheng-Wei Wang, and Herschel Rabitz. High dimensional model representations (HDMR): Concepts and applications. In *Proceedings of the Institute of Mathematics and Its Applicatons Workshop on Atmospheric Modeling*, pages 15–19. Citeseer, 2000.

[60] Deng Huang, Theodore T Allen, William I Notz, and Ning Zeng. Global optimization of stochastic black-box systems via sequential kriging meta-models. *Journal of Global Optimization*, 34(3):441–466, 2006.

[61] Nicolas Gayton, Jean M Bourinet, and Maurice Lemaire. CQ2RS: a new statistical approach to the response surface method for reliability analysis. *Structural Safety*, 25(1):99–121, 2003.

[62] Roland Schöbi and Bruno Sudret. Structural reliability analysis for p-boxes using multi-level meta-models. *Probabilistic Engineering Mechanics*, 48:27–38, 2017.

[63] Durga Rao Karanki, Hari Shankar Kushwaha, Ajit Kumar Verma, and Srividya Ajit. Uncertainty analysis based on probability bounds (p-box) approach in probabilistic safety assessment. *Risk Analysis: An International Journal*, 29(5):662–675, 2009.

Chapter 5

Distributional p-boxes propagation by Bayesian active learning

Estimation of failure probability function under imprecise probabilities by active learning–augmented probabilistic integration

Chao Dang[a], Pengfei Wei[b,⋆],Jingwen Song[c], Michael Beer[a,d,e]

[a]*Institute for Risk and Reliability, Leibniz University Hannover, Callinstr. 34, Hannover 30167, Germany*

[b]*School of Mechanics, Civil Engineering and Architecture, Northwestern Polytechnical University, Xi'an 710072, PR China*

[c]*Advanced Research Laboratories, Tokyo City University, 1-28-1 Tamazutsumi Setagaya-ku, Tokyo 158-8557, Japan*

[d]*Institute for Risk and Uncertainty, University of Liverpool, Liverpool L69 7ZF, United Kingdom*

[e]*International Joint Research Center for Engineering Reliability and Stochastic Mechanics, Tongji University, Shanghai 200092, PR China*

Published in ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering in August 2021

Abstract: Imprecise probabilities have gained increasing popularity for quantitatively modelling uncertainty under incomplete information, which is usually encountered in engineering analysis. In this contribution, a non-intrusive method, termed as '*Active Learning Augmented Probabilistic Integration*' (ALAPI), is developed to efficiently estimate the failure probability function (FPF) in the presence of imprecise probabilities. Specially, the parameterized probability-box models are of specific concern. By interpreting the failure probability integral from a Bayesian probabilistic integration perspective, the discretization error can be regarded as a kind of epistemic uncertainty, allowing it to be properly quantified and propagated through computational pipelines. Accordingly, an active learning probabilistic integration (ALPI) method is developed for failure probability estimation, in which a new learning function and a new stopping criterion associated with the upper bound of the posterior variance are proposed. Based on the idea of constructing an augmented uncertainty space, an imprecise augmented stochastic simulation (IASS) method is devised by using the RS-HDMR (random sampling high-dimensional representation model) for estimating the failure

*Corresponding Author
E-mail address: pengfeiwei@nwpu.edu.cn (P. Wei)

probability function in a pointwise stochastic simulation manner. To further improve the efficiency of IASS, the ALAPI is formed by an elegant combination of the ALPI and IASS, allowing the RS-HDMR component functions of the FPF to be properly inferred. Three benchmark examples are investigated to demonstrate the accuracy and efficiency of the proposed method.

Keywords: Failure probability function, Imprecise probability, Probability box, Gaussian process regression, Active learning, Bayesian probabilistic integration

5.1 Introduction

Uncertainty quantification and propagation have been essentially important, but still face critical challenges in many fields of science and engineering. This is because that in the real world, uncertainty is almost inevitable, and generally arises from a variety of distinct sources, e.g., statistical variability, measurement errors, instrumental uncertainty, imperfect information, limited data, abstraction and assumptions among others. Typically, these uncertainties can be categorized as either aleatory or epistemic according to their intrinsic features and effects on analysis [1, 2]. Aleatory uncertainty is related to the inherent randomness of an event or a parameter, and hence cannot be reduced even when sufficient information of high quality is available. On the contrary, epistemic uncertainty is due to a lack of knowledge, which therefore can be reduced by gaining more knowledge. In real-world applications, both kinds of uncertainties tend to be jointly present and are often easily confused with each other. As has been concluded by Der Kiureghian and Ditlevsen [1], without properly distinguishing different types of uncertainties, the results on risk and reliability analysis can be misleading.

As for the uncertainty representation, a large number of mathematical models have long been developed for quantitative characterization of uncertain phenomena in engineering practices. Generally, the existing uncertainty characterization models can be classified under three major frameworks: precise probability framework, non-probabilistic framework, and imprecise probability framework. The precise probability framework is deeply rooted in the well-established probability theory, and hence it is an essential tool in the quantitative mathematical treatment of uncertainty, especially for modelling aleatory uncertainty. A common criticism, however, is that large amounts of high-quality data are often required for inferring the potential precise probability model with sufficient credibility, which, unfortunately, may be rarely available for most engineering applications

[1, 2]. Alternatively, some representative models within the non-probabilistic framework, such as interval model [3], convex model [4], fuzzy set theory [5] among others, have been extensively investigated to describe the non-probabilistic uncertainty, especially those resulted from limited data with poor quality. In spite of their popularity, it has been argued that non-probabilistic models commonly fail to distinguish between the aleatory and epistemic uncertainties [6]. To fill this gap, the imprecise probability framework, mathematically as a combination of the non-probabilistic and probability frameworks, and physically making a clear separation of the two types of uncertainties, has gained increasingly attraction. Typical imprecise probability models include the evidence theory [7], interval probabilities [8], probability-box (p-box) [9], fuzzy probabilities [10], etc. A novel character of imprecise probability framework is that it enables the aleatory uncertainty and epistemic uncertainty to be treated separately within a unified framework, thanks to the hierarchical model structure. Based on the aforementioned considerations, we are mainly focusing on propagating uncertainty in the form of imprecise probabilities in the present paper.

In the imprecise probability framework, uncertainty propagation through computer simulators is a computationally challenging task primarily due to the double-layer structure inherent in imprecise probability models. To address this challenge, there has been an increasing attention on developing efficient numerical methods in recent years, which can be divided into two categories according to whether the method is decoupled or not. Typical coupled method includes the interval (quasi-) Monte Carlo simulation [11, 12], interval importance sampling [13], subset simulation based method [14], method of moments [15, 16], etc. Very often these coupled methods involve interval finite element analysis or numerical optimization within a nested loop, which still leads to high computational cost and limited applicability. For this reason, decoupled methods have drawn increasingly attention for propagating imprecise probabilities, such as the augmented subset simulation (ASS) [17], extended Monte Carlo simulation [18], non-intrusive imprecise stochastic simulation (NISS) [6, 19–21], augmented line sampling [22], operator norm theory [23, 24], augmented space integral [25, 26]. The most attractive feature of these methods is that only one simulation run is usually required, and hence very computationally efficient. Despite this, there still exist some respective drawbacks for those methods. For example, the NISS may not work well for problems with relatively large epistemic uncertainty due to the increasing variations of the NISS estimators; the application of operator norm theory is still limited to linear models with imprecision presented only in excitations; the augmented space integral is suffered from dimensionality of the epistemic

parameters. To tackle the former issue, Wei and his co-workers [27] recently proposed a novel imprecise probability propagation framework, termed as non-intrusive imprecise probabilistic integration (NIPI). In this framework, the estimation of response moment function (RMF) is treated as a Bayesian inference problem in the augmented space, and estimators for the component functions of RMF are analytically derived in closed form. Remarkably, it has been shown that the NIPI can be applied to the problems with large epistemic uncertainty resulted from extreme lack of information. However, the current NIPI method is only capable of evaluating RMF, and for FPF estimation, further developments need to be presented as will be shown in this work.

The main objective of this paper is to develop a new non-intrusive method, called 'Active Learning Augmented Probabilistic Integration', for estimation of FPF under imprecise probabilities. The core of the methodology is to interpret the failure probability integral from the perspective of Bayesian probabilistic integration, and hence the discretization error can be regarded as a kind of epistemic uncertainty. Through this treatment, the discretization error is propagated via the computational pipelines simultaneously together with the aleatory uncertainty and epistemic uncertainty present in the imprecise probability models, which is useful and important for developing an active learning strategy, and also for facilitating error assessment of the computational results. Besides, the approach also relies on an augmented idea that artificially constructs an augmented uncertainty space, enabling the propagation of two kinds of uncertainties to be fully decoupled. At last, the RS-HDMR (random sampling high-dimensional model representation) is employed to study the functional form of the FPF by decomposing it as a summation of component functions of increased orders, through which, the failure probability bounds and sensitivity analysis can also be obtained as byproducts.

The rest of this paper is arranged as follows. The problem to be solved in this work is briefly stated in the "Problem Statement" section. The "Active Learning Augmented Probabilistic Integration" section provides the detailed theoretical background and numerical implementation procedure of the proposed method. In the "Numerical Examples" section, three numerical examples are studied to verify the proposed method. The "Conclusions" section gives the findings of the present study.

5.2 Problem statement

Let the limit state function (also termed as performance function) of a physical system under consideration be denoted by a deterministic mapping $y = g(\boldsymbol{x})$, which is referred to as g-function hereinafter. Under this setting, the uncertainty in y only results from the uncertainty in $\boldsymbol{x}$, where $\boldsymbol{x} = [x_1, x_2, \ldots, x_n]$ is the n-dimensional row vector of input random variables that reflects the aleatory uncertainty of model inputs. In this paper, we only consider the case that each input random variable is characterized by a parameterized probability-box (p-box). Let $f(\boldsymbol{x}|\boldsymbol{\theta})$ denote the joint probability density function (PDF) of $\boldsymbol{x}$, which is conditional on its distribution parameters $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_m]$. Due to the epistemic uncertainty, the distribution parameters cannot be precisely known, but also uncertain. For simplicity, the interval model is employed to characterize the uncertainty of $\boldsymbol{\theta}$, i.e., $\boldsymbol{\theta} \in \left[\underline{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}\right]$, where $\underline{\boldsymbol{\theta}} = [\underline{\theta}_1, \underline{\theta}_2, \ldots, \underline{\theta}_m]$ and $\bar{\boldsymbol{\theta}} = [\bar{\theta}_1, \bar{\theta}_2, \ldots, \bar{\theta}_m]$ are the lower bound and upper bound, respectively. Besides, it is assumed that all the random variables and the distribution parameters are mutually independent. The output y is a state variable with $y \leq 0$ indicating that the system is failed, and safe otherwise. The FPF is expressed as:

$$P_f(\boldsymbol{\theta}) = \int_{\mathcal{X}} I_F(\boldsymbol{x}) f(\boldsymbol{x}|\boldsymbol{\theta}) \mathrm{d}\boldsymbol{x}, \tag{5.1}$$

where F in the subscript denotes the failure domain defined as $F = \{\boldsymbol{x} : g(\boldsymbol{x}) \leq 0\}$; $I_F(\boldsymbol{x})$ is an indicator function of failure: if $\boldsymbol{x} \in F$, $I_F(\boldsymbol{x}) = 1$, and $I_F(\boldsymbol{x}) = 0$ otherwise.

The main objective of this work is to evaluate the FPF defined by a integral with $\boldsymbol{\theta}$ being its argument. This is a more general task than calculating the failure probability bounds, since, with it, the failure probability bounds can be easily obtained without extra g-function evaluations. Besides, FPF also provides a basis for sensitivity analysis [28] and reliability-based design optimization [29, 30]. In most practical cases, however, the closed-form solution of the integral is not available because of the underlying complexity of the problem at hand. Alternatively, numerical techniques are thus especially desirable for more general applications.

5.3 Active learning augmented probabilistic integration (ALAPI)

In this section, we propose a method, termed as "active learning augmented probabilistic integration" (ALAPI), for efficiently propagating the p-box models and evaluating the failure prob-

ability function. The method starts by interpreting the estimation of failure probability integral with Bayesian inference, instead of a purely frequentist view. This will enable to incorporate our prior knowledge about the g-function and the possibility of an adaptive experimental design so as to develop an active learning probabilistic integration (ALPI) framework. Based on the idea of augmented uncertainty space, an imprecise augmented stochastic simulation (IASS) method is proposed to estimate the FPF in a pointwise stochastic simulation manner by utilizing the RS-HDMR. At last, the ALAPI is developed by an elegant combination of the ALPI and IASS.

5.3.1 Bayesian failure probability estimation: Active learning probabilistic integration (ALPI)

For brevity and convenience, let us first consider the case that $\boldsymbol{\theta}$ is precisely known and takes a fixed value $\boldsymbol{\theta}^\star$. That is, $f(\boldsymbol{x}|\boldsymbol{\theta}^\star)$ is now reduced to be a precise probability model. Under this setting, the failure probability should be a constant value from a theoretical standpoint, and expressed as:

$$P_f^\star = \int_{\mathcal{X}} I_F(\boldsymbol{x}) f(\boldsymbol{x}|\boldsymbol{\theta}^\star) \mathrm{d}\boldsymbol{x}. \tag{5.2}$$

As mentioned earlier, in most cases analytical derivation of the exact value of $P_f^\star$ is computationally intractable and even impossible, and usually we have to resort to numerical integration techniques for a crude estimate. Therefore, the introduction of error is unavoidable because the discretisation of the integrand is numerically necessary. Different from the frequentist theory of inference, we seek to reinterpret the problem of evaluating the failure probability integral in Eq. (5.2) via Bayesian inference, which is commonly known as Bayesian Quadrature (or Bayesian Probabilistic Integration) [31–34]. A novel feature of this treatment is that the discretisation error can be characterized as a kind of epistemic uncertainty, and then propagated through computational pipelines. One should not be confused with two kinds of epistemic uncertainties mentioned so far. One is the epistemic uncertainty here in the probabilistic integration, which arises from the computation due to the discretisation error. This is in contrast to the epistemic uncertainty revealed in the distribution parameters of input random variables, which comes from the computation setup, rather than the computation itself. In the framework of probabilistic integration, the integrand $I_F(\boldsymbol{x})$ at any fixed $\boldsymbol{x}$ is seen as a random variable simply because it is numerically unknown until we actually evaluate it. This is usually the case since $I_F(\boldsymbol{x})$ is computationally expensive, and we cannot afford to

compute $I_F(\boldsymbol{x})$ (or equivalently $g(\boldsymbol{x})$) at every site. Following a standard Bayesian approach, one needs to first assign a prior probability measure over the integrand $I_F(\boldsymbol{x})$, which expresses the investigator's prior beliefs about the actual function value. Conditioning on the limited observations $\left\{\boldsymbol{x}^{(i)}, I_F^{(i)}(\boldsymbol{x}^{(i)})\right\}_{i=1}^{d}$, we can obtain a posterior over $I_F(\boldsymbol{x})$ via Bayes' rule. This in turn will imply a posterior distribution over $P_f^\star$, which reflects the epistemic uncertainty resulted from the fact that we can only evaluate the integrand at a finite number of inputs.

The Gaussian process (GP) could be the most popular choice for the prior model, due to its broad applicability and sound theoretical background. However, we argue that it is inappropriate to directly specify a GP prior over the failure indicator function $I_F(\boldsymbol{x})$, since we know that it is discontinuous and actually follows a Bernoulli distribution. Alternatively, we put a GP prior over the performance function $g(\boldsymbol{x})$, denoted by

$$\hat{g}(\boldsymbol{x}) \sim \mathcal{GP}(\mu(\boldsymbol{x}), c(\boldsymbol{x}, \boldsymbol{x}')), \tag{5.3}$$

where $\mu(\boldsymbol{x})$ is the prior expectation function and $c(\boldsymbol{x}, \boldsymbol{x}')$ is the prior covariance function (also called kernel function). Various kinds of explicit functions with several hyper-parameters to be determined are available for the expectation function and covariance function in the literature. For more details, one can refer to [35, 36].

Given the experimental design matrix $\boldsymbol{X} = \{\boldsymbol{x}^{(i)}\}_{i=1}^{d}$ of size $d \times n$ and the corresponding response vector $Y = \{y^{(i)} = g(\boldsymbol{x}^{(i)})\}_{i=1}^{d}$ of size $d \times 1$, the hyper-parameters involved in the prior mean function and covariance function can be specified, e.g., by using maximum likelihood estimation [35].

Conditional on the observed data set $\mathcal{D} = \{\boldsymbol{X}, Y\}$, the posterior prediction of $\hat{g}(\boldsymbol{x})$ at a new site $\boldsymbol{x}$ follows a Gaussian random variable with expectation and variance being

$$\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})] = \mu(\boldsymbol{x}) + \boldsymbol{c}(\boldsymbol{x}, \boldsymbol{X})^{\mathrm{T}} \boldsymbol{C}^{-1}(Y - \boldsymbol{\mu}(\boldsymbol{X})), \tag{5.4}$$

$$\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})] = c(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{c}(\boldsymbol{x}, \boldsymbol{X})^{\mathrm{T}} \boldsymbol{C}^{-1} \boldsymbol{c}(\boldsymbol{x}, \boldsymbol{X}), \tag{5.5}$$

where $\mathbb{E}_{\mathcal{D}}[\cdot]$ and $\mathbb{V}_{\mathcal{D}}[\cdot]$ denote the posterior expectation and variance operators (a subscript "$\mathcal{D}$" is used to indicate the posterior), receptively; $\boldsymbol{\mu}(\boldsymbol{X}) = [\mu(\boldsymbol{x}^{(1)}), \mu(\boldsymbol{x}^{(2)}), \ldots, \mu(\boldsymbol{x}^{(d)})]^{\mathrm{T}}$ is the mean vector; $\boldsymbol{c}(\boldsymbol{x}, \boldsymbol{X}) = [c(\boldsymbol{x}, \boldsymbol{x}^{(1)}), c(\boldsymbol{x}, \boldsymbol{x}^{(2)}), \ldots, c(\boldsymbol{x}, \boldsymbol{x}^{(d)})]^{\mathrm{T}}$ is the covariance vector between $\boldsymbol{x}$ and $\boldsymbol{X}$; $\boldsymbol{C}$ is the covariance matrix of $\boldsymbol{X}$ with entry $[\boldsymbol{C}]_{ij} = c(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$.

Based on the Gaussian posterior of $\hat{g}(\boldsymbol{x})$, it is easy to know that the posterior stochastic process $\hat{I}_F(\boldsymbol{x})$ at site $\boldsymbol{x}$ is a Bernoulli random variable with

$$\mathbb{P}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})=1]=\mathbb{P}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})\leq 0]=\Phi\left(\frac{-\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right), \tag{5.6}$$

$$\mathbb{P}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})=0]=\mathbb{P}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})>0]=1-\Phi\left(\frac{0-\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right)=\Phi\left(\frac{\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right), \tag{5.7}$$

where $\mathbb{P}_{\mathcal{D}}[\cdot]$ denotes the posterior probability operator; Φ is the cumulative distribution function (CDF) of the standard normal variable.

Accordingly, the posterior expectation and variance of $\hat{I}_F(\boldsymbol{x})$ at site $\boldsymbol{x}$ are formulated as:

$$\mathbb{E}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})]=\Phi\left(\frac{-\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right), \tag{5.8}$$

and

$$\mathbb{V}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})]=\Phi\left(\frac{-\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right)\Phi\left(\frac{\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right). \tag{5.9}$$

Rewrite the failure probability integral in Eq. (5.2) as:

$$\hat{P}_f^{\star}=\int_{\mathcal{X}}\hat{I}_F(\boldsymbol{x})f(\boldsymbol{x}|\boldsymbol{\theta}^{\star})\mathrm{d}\boldsymbol{x}. \tag{5.10}$$

Since the integral above is just a linear projection of $\hat{I}_F(\boldsymbol{x})$, the posterior of $\hat{P}_f^{\star}$ is also random with expectation and variance being:

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}[\hat{P}_f^{\star}]&=\mathbb{E}_{\mathcal{D}}\left[\int_{\mathcal{X}}\hat{I}_F(\boldsymbol{x})f(\boldsymbol{x}|\boldsymbol{\theta}^{\star})\mathrm{d}\boldsymbol{x}\right]\\&=\int_{\mathcal{X}}\mathbb{E}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})]f(\boldsymbol{x}|\boldsymbol{\theta}^{\star})\mathrm{d}\boldsymbol{x}\\&=\int_{\mathcal{X}}\Phi\left(\frac{-\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right)f(\boldsymbol{x}|\boldsymbol{\theta}^{\star})\mathrm{d}\boldsymbol{x}\\&=\mathbb{E}_{\mathcal{X}}\left[\Phi\left(\frac{-\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right)\right],\end{aligned} \tag{5.11}$$

and

$$\begin{aligned}
\mathbb{V}_{\mathcal{D}}[\hat{P}_f^{\star}] =& \mathbb{E}_{\mathcal{D}}\left[\left(\hat{P}_f^{\star} - \mathbb{E}_{\mathcal{D}}[\hat{P}_f^{\star}]\right)^2\right] \\
=& \mathbb{E}_{\mathcal{D}}\left[\left(\int_{\mathcal{X}} \hat{I}_F(\boldsymbol{x}) f(\boldsymbol{x}|\boldsymbol{\theta}^{\star})\mathrm{d}\boldsymbol{x} - \int_{\mathcal{X}} \mathbb{E}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})] f(\boldsymbol{x}|\boldsymbol{\theta}^{\star})\mathrm{d}\boldsymbol{x}\right)^2\right] \\
=& \mathbb{E}_{\mathcal{D}}\left[\left(\int_{\mathcal{X}} \left(\hat{I}_F(\boldsymbol{x}) - \mathbb{E}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})]\right) f(\boldsymbol{x}|\boldsymbol{\theta}^{\star})\mathrm{d}\boldsymbol{x}\right)^2\right] \\
=& \mathbb{E}_{\mathcal{D}}\left[\left(\int_{\mathcal{X}} \left(\hat{I}_F(\boldsymbol{x}) - \mathbb{E}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})]\right) f(\boldsymbol{x}|\boldsymbol{\theta}^{\star})\mathrm{d}\boldsymbol{x}\right)\left(\int_{\mathcal{X}} \left(\hat{I}_F(\boldsymbol{x}^{'}) - \mathbb{E}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x}^{'})]\right) f(\boldsymbol{x}^{'}|\boldsymbol{\theta}^{\star})\mathrm{d}\boldsymbol{x}^{'}\right)\right] \\
=& \int_{\mathcal{X}}\int_{\mathcal{X}} \mathbb{E}_{\mathcal{D}}\left[\left(\hat{I}_F(\boldsymbol{x}) - \mathbb{E}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})]\right)\left(\hat{I}_F(\boldsymbol{x}^{'}) - \mathbb{E}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x}^{'})]\right)\right] f\left(\boldsymbol{x}|\boldsymbol{\theta}^{\star}\right) f\left(\boldsymbol{x}^{'}|\boldsymbol{\theta}^{\star}\right)\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}^{'} \\
=& \int_{\mathcal{X}}\int_{\mathcal{X}} \mathbb{COV}_{\mathcal{D}}[\hat{I}_F\left(\boldsymbol{x}\right), \hat{I}_F\left(\boldsymbol{x}^{'}\right)] f\left(\boldsymbol{x}|\boldsymbol{\theta}^{\star}\right) f\left(\boldsymbol{x}^{'}|\boldsymbol{\theta}^{\star}\right)\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}^{'},
\end{aligned} \tag{5.12}$$

where $\mathbb{E}_{\mathcal{X}}[\cdot]$ is the expectation operator with respect to $\boldsymbol{x}$; the term $\mathbb{COV}_{\mathcal{D}}[\hat{I}_F\left(\boldsymbol{x}\right), \hat{I}_F\left(\boldsymbol{x}^{'}\right)]$ is the posterior covariance between $\hat{I}_F(\boldsymbol{x})$ and $\hat{I}_F(\boldsymbol{x}')$, whose closed-form solution is not available.

It is reasonable to assume that $\hat{I}_F\left(\boldsymbol{x}\right)$ and $\hat{I}_F\left(\boldsymbol{x}^{'}\right)$ have finite variances, and then the following inequality holds via the Cauchy-Schwarz inequality:

$$\mathbb{COV}_{\mathcal{D}}\left[\hat{I}_F\left(\boldsymbol{x}\right), \hat{I}_F\left(\boldsymbol{x}^{'}\right)\right] \leq \sqrt{\mathbb{V}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})]}\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x}^{'})]}. \tag{5.13}$$

Substituting Eq. (5.13) into Eq. (5.12), gives the upper bound of the posterior variance of $\hat{P}_f^{\star}$:

$$\begin{aligned}
\mathbb{V}_{\mathcal{D}}[\hat{P}_f^{\star}] &= \int_{\mathcal{X}}\int_{\mathcal{X}} \mathbb{COV}_{\mathcal{D}}[\hat{I}_F\left(\boldsymbol{x}\right), \hat{I}_F\left(\boldsymbol{x}^{'}\right)] f\left(\boldsymbol{x}|\boldsymbol{\theta}^{\star}\right) f\left(\boldsymbol{x}^{'}|\boldsymbol{\theta}^{\star}\right)\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}^{'} \\
&\leq \int_{\mathcal{X}}\int_{\mathcal{X}} \sqrt{\mathbb{V}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})]}\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x}^{'})]} f\left(\boldsymbol{x}|\boldsymbol{\theta}^{\star}\right) f\left(\boldsymbol{x}^{'}|\boldsymbol{\theta}^{\star}\right)\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}^{'} \\
&= \left(\int_{\mathcal{X}} \sqrt{\mathbb{V}_{\mathcal{D}}[\hat{I}_F(\boldsymbol{x})]} f\left(\boldsymbol{x}|\boldsymbol{\theta}^{\star}\right)\mathrm{d}\boldsymbol{x}\right)^2 \\
&= \left(\mathbb{E}_{\mathcal{X}}\left[\sqrt{\Phi\left(\frac{-\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right)\Phi\left(\frac{\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right)}\right]\right)^2.
\end{aligned} \tag{5.14}$$

Note that similar equations with Eq. (5.11) and (5.14) have been available in the literature (e.g., [37, 38]), but they are derived from other perspectives, rather than Bayesian probabilistic integration. The posterior expectation in Eq. (5.11) can be used as the estimator of the failure probability, and the upper bound of the posterior variance in Eq. (5.14) can measure the epistemic uncertainty of this estimator induced by the limited number of observations, but roughly since it

might be magnified to a certain extent.

5.3.2 Adaptive experimental design

In order to accelerate the convergence of GP training process and increase the accuracy of failure probability predictor, a careful experimental design is required. It has been shown in the previous studies, e.g., AK-MCS [39], AK-IS [40], AK-MCMC [41] and AGPR-LS [42], an adaptive experimental design strategy is very useful for building a accurate GP model at less computational expense. The key is to develop a suitable learning function (or called acquisition function) that can decide the next evaluation point based on the current GP model. Since the upper bound of the posterior variance of the failure probability integral has been derived in the previous subsection, it is hence possible for us to develop an adaptive experimental design so as to reduce the epistemic uncertainty of the failure probability predictor as much as possible.

For the above purposes, we will define a new learning function, called upper bound posterior variance contribution (UPVC), which is given as follows:

$$\mathrm{UPVC}(\boldsymbol{x}) = \Phi\left(\frac{-\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right)\Phi\left(\frac{\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right), \tag{5.15}$$

which actually reflects the contribution of epistemic uncertainty at any site $\boldsymbol{x}$ to the upper bound of posterior variance of the failure probability predictor. If the point processing the largest UPVC value (i.e., $\boldsymbol{x}^{\star} = \arg\max_{\boldsymbol{x}} \mathrm{UPVC}(\boldsymbol{x})$) is sequentially added to the training data set $\mathcal{D}$, the upper bound of posterior variance of failure probability integral is expected to decrease most fastest, and hence we will obtain a more accurate prediction of failure probability at lower computational cost. Therefore, the active learning criterion proposed in this work is to find the maximum point of UPVC function, which is used as the best next point to evaluate on the real g-function.

In addition to the active learning criterion, a stopping criterion for indicating the convergence of the algorithm should also be presented. In this study, we propose a new stopping criterion, which is based on the judgment of the posterior coefficient of variation (COV) of failure probability predictor. In terms of Eqs. (5.11) and (5.14), the upper bound of the posterior COV of failure

probability can be expressed as:

$$\kappa^{\star} = \overline{\mathbb{COV}}_{\mathcal{D}}[\hat{P}_f^{\star}] = \frac{\mathbb{E}_{\mathcal{X}}\left[\sqrt{\Phi\left(\frac{-\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right)\Phi\left(\frac{\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right)}\right]}{\mathbb{E}_{\mathcal{X}}\left[\Phi\left(\frac{-\mathbb{E}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}{\sqrt{\mathbb{V}_{\mathcal{D}}[\hat{g}(\boldsymbol{x})]}}\right)\right]}. \tag{5.16}$$

Once the GP model becomes enough accurate, $\kappa^{\star}$ should be very small. Herein, the stopping criterion is defined by $\kappa^{\star} < \varepsilon$, where ε is a user-specified threshold.

5.3.3 Failure probability function estimation by Imprecise Augmented Stochastic Simulation (IASS)

In this subsection, we will consider the case that $\boldsymbol{\theta}$ is no longer a fixed value, but a vector of intervals. Accordingly, $P(\boldsymbol{\theta})$, as defined in Eq. (5.1), is not a deterministic value any more, but a function of interval variables. For instrumental purposes, all the distribution parameters are treated as random variables in the following. That is, we assume an auxiliary probability distribution for each interval variable of $\boldsymbol{\theta}$. Note that this assumption does not imply that $\boldsymbol{\theta}$ must be a random vector in nature, but just serves as an instrumental tool for performing the proposed method. Let the auxiliary joint PDF and CDF of $\boldsymbol{\theta}$ be denoted as $\varphi(\boldsymbol{\theta}) = \prod_{j=1}^{d} \varphi_j(\theta_j)$ and $\Phi(\boldsymbol{\theta}) = \prod_{j=1}^{d} \Phi_j(\theta_j)$ respectively, where $\varphi_j(\theta_j)$ and $\Phi_j(\theta_j)$ are the marginal PDF and CDF of θ_j respectively.

The random vector $\boldsymbol{x}$ is called ***aleatory uncertainty vector*** as the aleatory uncertainty of model inputs is represented by means of its probability characterization, and the corresponding random-variate space $\mathcal{X}$ is termed as ***aleatory uncertainty space***. Under the previous assumption, we shall refer to the random vector $\boldsymbol{\theta}$ as ***epistemic uncertainty vector*** and the associated support Θ as ***epistemic uncertainty space***, respectively, since $\boldsymbol{\theta}$ characterizes the epistemic uncertainty of distribution parameters of $\boldsymbol{x}$ due to the lack of information. Consider an ***augmented uncertainty vector*** $\boldsymbol{v} = [\boldsymbol{x}, \boldsymbol{\theta}]$, i.e., a composition of aleatory uncertainty vector and epistemic uncertainty vector, whose joint PDF and ***augmented uncertainty space*** are denoted as $w(\boldsymbol{v}) = f(\boldsymbol{x}|\boldsymbol{\theta})\varphi(\boldsymbol{\theta})$ and $\mathcal{V} = \mathcal{X} \oplus \Theta$ respectively. Therefore, the failure probability function defined in Eq. (5.1) can be

rewritten as:

$$\begin{aligned} P_f(\boldsymbol{\theta}) &= \int_{\mathcal{V}} I_F(\boldsymbol{v}) w(\boldsymbol{v}') \mathrm{d}\boldsymbol{v}' \\ &= \int_{\Theta} \int_{\mathcal{X}} I_F(\boldsymbol{v}) f(\boldsymbol{x}|\boldsymbol{\theta}) \varphi(\boldsymbol{\theta}') \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{\theta}' \\ &= \int_{\mathcal{X}} I_F(\boldsymbol{v}) f(\boldsymbol{x}|\boldsymbol{\theta}) \mathrm{d}\boldsymbol{x}, \end{aligned} \tag{5.17}$$

where $\boldsymbol{v}' = [\boldsymbol{x}, \boldsymbol{\theta}']$; $\boldsymbol{\theta}'$ is i.i.d. with $\boldsymbol{\theta}$; $I_F(\boldsymbol{v})$ is the augmented failure indicator function corresponding to the augmented g-function $g(\boldsymbol{v})$. With Eq. (5.17), the failure indicator function is extended to the augmented uncertainty space, while it is noted that the integral is only with respect to $\boldsymbol{x}$. This treatment can bring several benefits, which will be discussed later. However, it is still tricky to evaluate the functional form of $P_f(\boldsymbol{\theta})$ with respect to the full vector $\boldsymbol{\theta}$ due to the underling complexity.

Alternatively, the random-sampling high-dimensional model representation (RS-HDMR) [43] is adopted to decompose the original FPF into a summation of component functions of increasing orders such that:

$$P_{f,\mathrm{RS}}(\boldsymbol{\theta}) = P_{f,\mathrm{RS},0} + \sum_{j=1}^{m} P_{f,\mathrm{RS},j}\left(\theta_j\right) + \sum_{j<k}^{m} P_{f,\mathrm{RS},jk}\left(\theta_j, \theta_k\right) + \cdots + P_{f,\mathrm{RS},1,\ldots,m}(\boldsymbol{\theta}), \tag{5.18}$$

in which $P_{f,\mathrm{RS},0}$ is a zeroth-order (constant) component, $P_{f,\mathrm{RS},j}\left(\theta_j\right)$ is a first-order component function of the distribution parameter θ_j, $P_{f,\mathrm{RS},jk}\left(\theta_j, \theta_k\right)$ is a second-order component function of the distribution parameters θ_j and θ_k, etc. According to Eq. (5.17), these RS-HDMR component functions can be further derived as:

$$\begin{aligned} P_{f,\mathrm{RS},0} &= \int_{\Theta} P_f(\boldsymbol{\theta}) \varphi(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} \\ &= \int_{\Theta} \int_{\mathcal{X}} I_F(\boldsymbol{v}) f(\boldsymbol{x}|\boldsymbol{\theta}) \varphi(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{\theta} \\ &= \int_{\mathcal{V}} I_F(\boldsymbol{v}) \omega(\boldsymbol{v}) \mathrm{d}\boldsymbol{v} \\ &= \mathbb{E}_{\mathcal{V}}[I_F(\boldsymbol{v})], \end{aligned} \tag{5.19}$$

$$
\begin{aligned}
P_{f,\mathrm{RS},j}\left(\theta_j\right) &= \int_{\Theta_{-j}} P_f(\boldsymbol{\theta})\varphi(\theta_j,\boldsymbol{\theta}_{-j})\mathrm{d}\boldsymbol{\theta}_{-j} - P_{f,\mathrm{RS},0} \\
&= \int_{\Theta_{-j}}\int_{\mathcal{X}} I_F(\boldsymbol{v}) f(\boldsymbol{x}|\boldsymbol{\theta})\varphi(\theta_j,\boldsymbol{\theta}_{-j})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{\theta}_{-j} - P_{f,\mathrm{RS},0} \\
&= \int_{\mathcal{V}_{-\Theta_j}} I_F(\boldsymbol{v})\omega(\theta_j,\boldsymbol{v}_{-\theta_j})\mathrm{d}\boldsymbol{v}_{-\theta_j} - P_{f,\mathrm{RS},0} \\
&= \mathbb{E}_{\mathcal{V}_{-\Theta_j}}[I_F(\boldsymbol{v}|\theta_j,\boldsymbol{v}_{-\theta_j})] - P_{f,\mathrm{RS},0},
\end{aligned}
\tag{5.20}
$$

$$
\begin{aligned}
P_{f,\mathrm{RS},jk}\left(\theta_j,\theta_k\right) &= \int_{\Theta_{-jk}} P_f(\boldsymbol{\theta})\varphi(\theta_j,\theta_k,\boldsymbol{\theta}_{-jk})\mathrm{d}\boldsymbol{\theta}_{-jk} - P_{f,\mathrm{RS},j}\left(\theta_j\right) - P_{f,\mathrm{RS},k}\left(\theta_k\right) - P_{f,\mathrm{RS},0} \\
&= \int_{\Theta_{-jk}}\int_{\mathcal{X}} I_F(\boldsymbol{v}) f(\boldsymbol{x}|\boldsymbol{\theta})\varphi(\theta_j,\theta_k,\boldsymbol{\theta}_{-jk})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{\theta}_{-jk} - P_{f,\mathrm{RS},j}\left(\theta_j\right) - P_{f,\mathrm{RS},k}\left(\theta_k\right) - P_{f,\mathrm{RS},0} \\
&= \int_{\mathcal{V}_{-\Theta_{jk}}} I_F(\boldsymbol{v})\omega(\theta_j,\theta_k,\boldsymbol{v}_{-(\theta_j,\theta_k)})\mathrm{d}\boldsymbol{v}_{-(\theta_j,\theta_k)} - P_{f,\mathrm{RS},j}\left(\theta_j\right) - P_{f,\mathrm{RS},k}\left(\theta_k\right) - P_{f,\mathrm{RS},0} \\
&= \mathbb{E}_{\mathcal{V}_{-\Theta_{jk}}}\left[I_F(\boldsymbol{v}|\theta_j,\theta_k,\boldsymbol{v}_{-(\theta_j,\theta_k)})\right] - P_{f,\mathrm{RS},j}\left(\theta_j\right) - P_{f,\mathrm{RS},k}\left(\theta_k\right) - P_{f,\mathrm{RS},0},
\end{aligned}
\tag{5.21}
$$

where $\boldsymbol{\theta}_{-j}$ denotes the epistemic uncertainty vector excluding θ_j, $\boldsymbol{v}_{-\theta_j}$ denotes the augmented uncertainty vector excluding θ_j, $\boldsymbol{\theta}_{-jk}$ denotes the epistemic uncertainty vector excluding θ_j and θ_k, $\boldsymbol{v}_{-(\theta_j,\theta_k)}$ denotes the augmented uncertainty vector excluding θ_j and θ_k. Previous studies indicate that the high-order terms in the expansion often are negligible for many realistic problems [6, 19], and only the truncation up to the second order is considered in this work, but any higher-order RS-HDMR component function can be similarly derived if necessary. Within the RS-HDMR framework, one can notice that the main task now is to evaluate the low-order component functions for approximating the FPF. By using Eq. (5.17), the RS-HDMR component functions are further converted to the integrals with respect to the augmented uncertainty vector of decreasing dimensions. This conversation is useful since the two-fold integrals are equivalently transformed to be one-fold ones, which will reduce the computational complexity substantially. Besides, the computational efficiency for inferring these component functions is also improved if we apply the proposed ALPI by making full use of the correlation information revealed in both aleatory and epistemic uncertainty spaces.

For convenience, we can reformulate the second-order truncated RS-HDMR decomposition as:

$$
P_{f,\mathrm{RS}}(\boldsymbol{\theta}) \approx \frac{(m-1)(m-2)}{2}\mathcal{P}_{f,\mathrm{RS},0} - (m-2)\sum_{j=1}^{m}\mathcal{P}_{f,\mathrm{RS},j}(\theta_j) + \sum_{j<k}^{m}\mathcal{P}_{f,\mathrm{RS},jk}(\theta_j,\theta_k), \tag{5.22}
$$

where $\mathcal{P}_{f,\mathrm{RS},0} = P_{f,\mathrm{RS},0} = \mathbb{E}_{\mathcal{V}}[I_F(\boldsymbol{v})]$, $\mathcal{P}_{f,\mathrm{RS},j}(\theta_j) = \mathbb{E}_{\mathcal{V}_{-\Theta_j}}[I_F(\boldsymbol{v}|\theta_j, \boldsymbol{v}_{-\theta_j})]$ and $\mathcal{P}_{f,\mathrm{RS},jk}(\theta_j, \theta_k) = \mathbb{E}_{\mathcal{V}_{-\Theta_{jk}}}\left[I_F(\boldsymbol{v}|\theta_j, \theta_k, \boldsymbol{v}_{-(\theta_j,\theta_k)})\right]$. The constant component $\mathcal{P}_{f,\mathrm{RS},0}$ or $P_{f,\mathrm{RS},0}$ is also referred to as ***augmented failure probability*** since it integrates over the augmented uncertainty vector (see Eq. (5.19)). This reformulation is useful since one can easily derive the upper bound variance of the first-order and second-order component functions when implementing the ALAPI method (see Eqs. (5.36)-(5.37)). In this setting, the main focus is to evaluate the component functions in Eq. (5.22), and one should not be confused with the component functions defined in Eq. (5.18). Obviously, the crude Monte Carlo simulation (MCS) can be directly used to estimate those RS-HDMR components both in Eqs. (5.18) and (5.22). For example, the estimators for those components in Eq. (5.22) can be given by:

$$\hat{\mathcal{P}}_{f,\mathrm{RS},0} = \frac{1}{N}\sum_{s=1}^{N} I_F(\boldsymbol{v}^{(\boldsymbol{s})}), \tag{5.23}$$

$$\hat{\mathcal{P}}_{f,\mathrm{RS},j}(\theta_j) = \frac{1}{N}\sum_{s=1}^{N} I_F((\boldsymbol{v}|\theta_j, \boldsymbol{\theta}_{-j})^{(s)}), \tag{5.24}$$

$$\hat{\mathcal{P}}_{f,\mathrm{RS},jk}(\theta_j, \theta_k) = \frac{1}{N}\sum_{s=1}^{N} I_F((\boldsymbol{v}|\theta_j, \theta_k, \boldsymbol{\theta}_{-jk})^{(s)}), \tag{5.25}$$

where $\{\boldsymbol{v}^{(\boldsymbol{s})}\}_{s=1}^{N}$, $\{(\boldsymbol{v}|\theta_j, \boldsymbol{\theta}_{-j})^{(s)}\}_{s=1}^{N}$ and $\{(\boldsymbol{v}|\theta_j, \theta_k, \boldsymbol{\theta}_{-jk})^{(s)}\}_{s=1}^{N}$ given fixed θ_j and θ_k are three sets of N simple random samples generated from $w(\boldsymbol{v})$, $w(\boldsymbol{v}|\theta_j, \boldsymbol{\theta}_{-j})$ and $w(\boldsymbol{v}|\theta_j, \theta_k, \boldsymbol{\theta}_{-jk})$, respectively. It is easy to prove that the above estimators are all unbiased, so we simply omit the proofs. Their variances can also be derived as:

$$\mathbb{V}_{\mathcal{V}}\left[\hat{\mathcal{P}}_{f,\mathrm{RS},0}\right] = \frac{1}{(N-1)N}\sum_{s=1}^{N}\left[I_F(\boldsymbol{v}^{(\boldsymbol{s})}) - \hat{\mathcal{P}}_{f,\mathrm{RS},0}\right]^2, \tag{5.26}$$

$$\mathbb{V}_{\mathcal{V}_{-\Theta_j}}\left[\hat{\mathcal{P}}_{f,\mathrm{RS},j}(\theta_j)\right] = \frac{1}{(N-1)N}\sum_{s=1}^{N}\left[I_F((\boldsymbol{v}|\theta_j, \boldsymbol{\theta}_{-j})^{(s)}) - \hat{\mathcal{P}}_{f,\mathrm{RS},j}(\theta_j)\right]^2, \tag{5.27}$$

$$\mathbb{V}_{\mathcal{V}_{-\Theta_{jk}}}\left[\hat{\mathcal{P}}_{f,\mathrm{RS},jk}(\theta_j, \theta_k)\right] = \frac{1}{(N-1)N}\sum_{s=1}^{N}\left[I_F((\boldsymbol{v}|\theta_j, \theta_k, \boldsymbol{\theta}_{-jk})^{(s)}) - \hat{\mathcal{P}}_{f,\mathrm{RS},jk}(\theta_j, \theta_k)\right]^2. \tag{5.28}$$

When the sample size is large, the central limit theorem indicates that the sampling distributions of $\hat{\mathcal{P}}_{f,\mathrm{RS},0}$, $\hat{\mathcal{P}}_{f,\mathrm{RS},j}(\theta_j)$ and $\hat{\mathcal{P}}_{f,\mathrm{RS},jk}(\theta_j, \theta_k)$ approximately follow normal distributions. Therefore, their confidence intervals (CIs) can be derived by using the t interval. For example, the $(1-\alpha)100\%$ CI

of $\hat{\mathcal{P}}_{f,\mathrm{RS},0}$ can be given by:

$$\left[\hat{\mathcal{P}}_{f,\mathrm{RS},0} - t_{N-1}(\alpha/2)\sqrt{\mathbb{V}_{\mathcal{V}}\left[\hat{\mathcal{P}}_{f,\mathrm{RS},0}\right]}, \hat{\mathcal{P}}_{f,\mathrm{RS},0} + t_{N-1}(\alpha/2)\sqrt{\mathbb{V}_{\mathcal{V}}\left[\hat{\mathcal{P}}_{f,\mathrm{RS},0}\right]}\right], \tag{5.29}$$

where $t_{N-1}(\alpha/2)$ denotes the $(1-\alpha/2)$-th percentile of a Student's t-distribution with $N-1$ degrees of freedom. It should be noted that the proposed RS-HDMR based technique for estimating the FPF is actually a double-loop procedure, which is termed as ***Imprecise Augmented Stochastic Simulation*** (IASS) in this work. The computational efficiency of the IASS still depends on the sample size N and the grid size of $\boldsymbol{\theta}$, and hence it can be merely used as a reference method for verifying other newly-developed methods. For further reducing the computational burden, the proposed ALPI method will be incorporated into the IASS framework in next subsection.

5.3.4 Numerical implementation procedure of ALAPI

By combining the ALPI with IASS, a novel method, namely ALAPI, is proposed to efficiently estimate the FPF. The basic procedure for numerical implementation of the proposed method includes the following steps, which is also illustrated in Fig. 5.1.

Step 1: Generate a set of N simple random samples $\boldsymbol{V} = \{\boldsymbol{v}\}_{s=1}^{N}$ according to the augmented PDF $w(\boldsymbol{v})$, which serves as a sample pool for training a GP model for the augmented g-function $g(\boldsymbol{v})$. For this purpose, the auxiliary PDF $\varphi(\boldsymbol{\theta})$ for $\boldsymbol{\theta}$ should be specified in advance. In order to enable those points within the intervals to have the same chance of being sampled, we assume a uniform auxiliary PDF over its support for each θ_j in this work;

Step 2: Randomly select N_0 (e.g., $N_0 = 12$) samples among $\boldsymbol{V}$ and compute the corresponding augmented g-function values. An initial training sample set is then constructed by the N_0 input-output pairs, which is denoted as $\mathcal{T}$;

Step 3: Train or update a GP model, denoted as $\hat{g}(\boldsymbol{v})$, for the augmented g-function $g(\boldsymbol{v})$ based on $\mathcal{T}$. The Gaussian Process Regression toolbox in Matlab is used, and the mean function and covariance function are specified as the linear function and squared exponential kernel function respectively in this study;

Step 4: Compute the upper bound of posterior COV of augmented failure probability based

on the trained GP model such that:

$$\kappa = \frac{\sum_{s=1}^{N} \sqrt{\Phi\left(\frac{-\mathbb{E}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}}\right)\Phi\left(\frac{\mathbb{E}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}}\right)}}{\sum_{s=1}^{N}\Phi\left(\frac{-\mathbb{E}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}}\right)}. \tag{5.30}$$

If the stopping condition $\kappa < \varepsilon$ is satisfied, go to **Step 5**; otherwise, identify the point possessing maximum UPVC value among the sample pool $\boldsymbol{V}$ by

$$\boldsymbol{v}^{\star} = \underset{\boldsymbol{v}\in\boldsymbol{V}}{\arg\max}\,\mathrm{UPVC}(\boldsymbol{v}) = \underset{\boldsymbol{v}\in\boldsymbol{V}}{\arg\max}\,\Phi\left(\frac{-\mathbb{E}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}}\right)\Phi\left(\frac{\mathbb{E}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}}\right), \tag{5.31}$$

evaluate the corresponding g-function value $y^{\star} = g(\boldsymbol{v}^{\star})$, add $\{\boldsymbol{v}^{\star}, y^{\star}\}$ to the training sample set $\mathcal{T}$, and go to **Step 3**;

Step 5: Based on the well-trained GP model $\hat{g}(\boldsymbol{v})$, perform the IASS method to obtain a estimate $\hat{P}_f(\boldsymbol{\theta})$ for the FPF. As defined in Eq. (5.22), each component function of RS-HDMR can also be inferred from the GP predictor. According to the ALPI method, the unbiased estimators for RS-HDMR component functions can be given by:

$$\hat{\mathcal{P}}_{f,\mathrm{RS},0} = \frac{1}{N}\sum_{s=1}^{N}\Phi\left(\frac{-\mathbb{E}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}}\right), \tag{5.32}$$

$$\hat{\mathcal{P}}_{f,\mathrm{RS},j}\left(\theta_j\right) = \frac{1}{N}\sum_{s=1}^{N}\Phi\left(\frac{-\mathbb{E}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j)^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j)^{(s)})]}}\right), \tag{5.33}$$

$$\hat{\mathcal{P}}_{f,\mathrm{RS},jk}\left(\theta_j,\theta_k\right) = \frac{1}{N}\sum_{s=1}^{N}\Phi\left(\frac{-\mathbb{E}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j,\theta_k)^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j,\theta_k)^{(s)})]}}\right). \tag{5.34}$$

The upper bound of posterior variances of the component functions, which reflects the upper bound of the epistemic uncertainty due to the discretization error by using the ALAPI, can also be estimated by:

$$\overline{\mathbb{V}}_{\mathcal{T}}\left[\hat{\mathcal{P}}_{f,\mathrm{RS},0}\right] = \left[\frac{1}{N}\sum_{s=1}^{N}\sqrt{\Phi\left(\frac{-\mathbb{E}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}}\right)\Phi\left(\frac{\mathbb{E}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}(\boldsymbol{v}^{(s)})]}}\right)}\right]^2, \tag{5.35}$$

$$\overline{\mathbb{V}}_{\mathcal{T}}\left[\hat{\mathcal{P}}_{f,\mathrm{RS},j}\left(\theta_j\right)\right]=\left[\frac{1}{N}\sum_{s=1}^{N}\sqrt{\Phi\left(\frac{-\mathbb{E}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j)^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j)^{(s)})]}}\right)\Phi\left(\frac{\mathbb{E}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j)^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j)^{(s)})]}}\right)}\right]^2, \tag{5.36}$$

$$\overline{\mathbb{V}}_{\mathcal{T}}\left[\hat{\mathcal{P}}_{f,\mathrm{RS},jk}\left(\theta_j,\theta_k\right)\right]=\left[\frac{1}{N}\sum_{s=1}^{N}\sqrt{\Phi\left(\frac{-\mathbb{E}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j,\theta_k)^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j,\theta_k)^{(s)})]}}\right)\Phi\left(\frac{\mathbb{E}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j,\theta_k)^{(s)})]}{\sqrt{\mathbb{V}_{\mathcal{T}}[\hat{g}((\boldsymbol{v}|\theta_j,\theta_k)^{(s)})]}}\right)}\right]^2. \tag{5.37}$$

Note that this step does not require to evaluate on the original g-function, and then the computational burden can be alleviated significantly, especially for an expensive-to-evaluate computer simulator involved.

In the above steps, it should be emphasized that the user-specified threshold ε can affect the accuracy of resultant GP model, as well as the efficiency of the active learning process. Besides, there is a possibility that the stopping condition is satisfied even though the GP model is indeed not accurate enough, e.g., at the early stage of training. To avoid this situation, one can simply use a delay judgment strategy, which means that the active learning process is stopped only when the stopping condition is satisfied for several times in succession (e.g., three). Besides, the estimators in Eqs. (5.32)-(5.34) are only unbiased for the GP model, but biased for the real g-function.

The proposed ALAPI method has three main attractive features, making it very efficient for estimating the FPF. First, by assuming an auxiliary PDF for the distribution parameter $\boldsymbol{\theta}$, the GP model is built in the joint aleatory and epistemic uncertainty space (i.e., the augmented uncertainty space). The spatial correlation information in the augmented uncertainty space is shown to be quite useful for the active learning process. Second, the discretization error is regarded as a kind of epistemic uncertainty via interpreting the failure probability integral from Bayesian inference, which enables to derive the upper bounds of posterior variances of the ALAPI estimators. Third, the proposed method is essentially a decoupled procedure through an elegant combination of the ALPI and IASS, yielding a major improvement in computational efficiency.

5.4 Numerical examples

In this section, three numerical examples are studied to verify the proposed method. Among the available state-of-the-art techniques for estimating the FPF, the active learning NISS developed

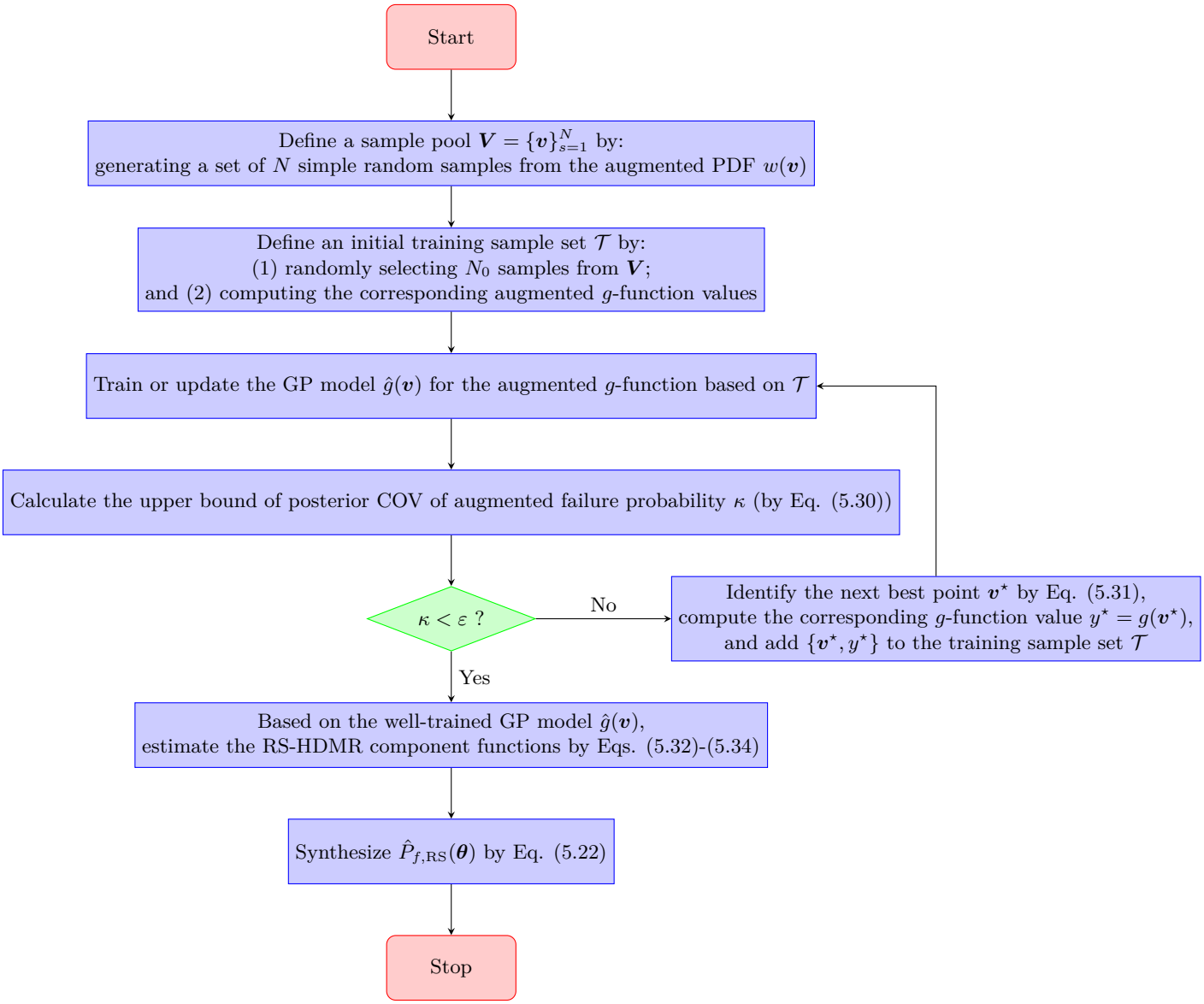


Figure 5.1: Flowchart of the proposed ALAPI method.

in [19] could be a potential competitor to the proposed method. Therefore, we mainly compare our method with this method by using the first numerical example. For notational clarity, we will denote this method simply as "NISS" below. One can refer to Appendix 5.7 for more detailed description of the NISS method used. In the third example, the ASS [17] is also implemented to evaluate the augmented failure probability (or constant RS-HDMR component). Besides, the developed IASS method is mainly adopted to provide reference results in all three numerical examples.

5.4.1 Example 1: a series system with four branches

The first example considers a series system with four branches, which has been extensively investigated in the context of precise probabilities [39, 44]. The performance function is given by:

$$y = g(x_1, x_2) = \min \begin{cases} 3 + \frac{(x_1 - x_2)^2}{10} - \frac{(x_1 + x_2)}{\sqrt{2}} \\ 3 + \frac{(x_1 - x_2)^2}{10} + \frac{(x_1 + x_2)}{\sqrt{2}} \\ (x_1 - x_2) + \frac{b}{\sqrt{2}} \\ (x_2 - x_1) + \frac{b}{\sqrt{2}} \end{cases}, \tag{5.38}$$

where b is a constant, specified as 4; The random variables x_1 and x_2 are normally distributed, denoted as $\mathcal{N}(\mu_1, \sigma_1^2)$ and $\mathcal{N}(\mu_2, \sigma_2^2)$ respectively. Due to the epistemic uncertainty, the distribution parameters (i.e., $\boldsymbol{\theta} = [\mu_1, \sigma_1, \mu_2, \sigma_2]$) are not deterministic, but uncertain. In this example, two cases by varying bounds of the distribution parameters are considered, as given in Tab. 5.1.

Table 5.1: Distribution parameters for Example 1.

Case	μ_1	σ_1	μ_2	σ_2
I	$[-0.5, 0.5]$	$[0.8, 1.2]$	$[-0.5, 0.5]$	$[0.8, 1.2]$
II	$[-0.8, 0.8]$	$[0.5, 1.5]$	$[-0.8, 0.8]$	$[0.5, 1.5]$

In the following, three methods, i.e., the proposed ALAPI, NISS and IASS, are employed to estimate the FPF. For both cases, the sample pool is constructed by 10^5 simple random samples for ALAPI and NISS, while the sample size for IASS is set to be 10^6. Besides, the threshold regarding the stopping condition is specified as $\varepsilon = 0.02$ for ALAPI.

5.4.1.1 Case I

For illustrating the active learning process of ALAPI, the upper bound of posterior COV of the augmented failure probability $\hat{\mathcal{P}}_{f,\mathrm{RS},0}$ (denoted as κ) against the number of adaptively added samples is plotted in Fig. 5.2(a). It can be seen that as more samples are sequentially added into the initial training data set, the general trend of κ tends to decrease. Until the initial training sample set is enriched by a total number of 81 samples, the stopping condition of the active learning procedure is satisfied. Thus, this implies that only 93 performance function evaluations are required by the proposed ALAPI method, which are much less than the NISS method, say 164. The constant RS-HDMR calculated by the three methods are listed in the second to fourth rows of Tab. 5.2. As seen, the estimate given by IASS has a relatively small COV, and hence we are highly confident that this reference result should be very close to the true value. Compared to the reference result, both ALAPI and NISS are capable of yielding very desirable estimates for the constant-HDMR component in this case. Note that the accuracy of the proposed method can also be revealed by the upper bound of posterior COV of $\hat{\mathcal{P}}_{f,\mathrm{RS},0}$ itself, given that the sampling variability for estimating $\overline{\mathbb{COV}}_{\mathcal{T}}[\hat{\mathcal{P}}_{f,\mathrm{RS},0}]$ is negligible. On the contrary, the COV (i.e., $\mathbb{COV}[\hat{\mathcal{P}}_{f,\mathrm{RS},0}]$) provided by the NISS method only accounts for the sampling variability. From Fig. 5.3(a), one can also conclude that all the three methods are able to produce very accurate estimates for the four first-order RS-HDMR component functions. For limited space, only one second-order RS-HDMR component function computed by ALAPI and IASS is depicted in Fig. 5.4(a). Remarkably, it is shown that the estimate by the proposed method accords well with that by the IASS, with the upper bound of posterior COV and COV being small.

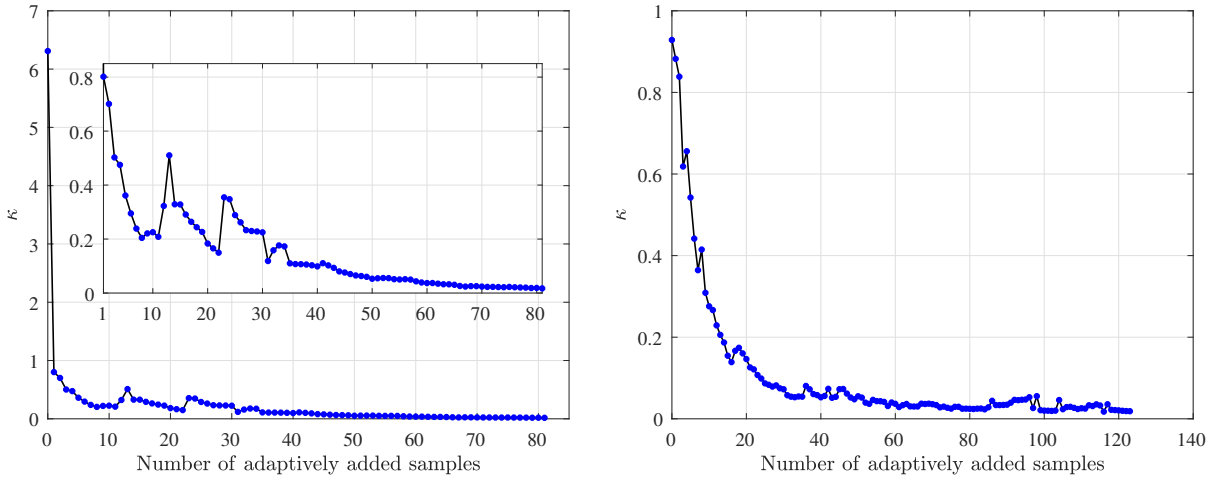
In short, the proposed ALAPI method can offer comparable results against the NISS method, but requires less g-function evaluations in such a case with smaller epistemic uncertainty presented in the distribution parameters compared with case II.

5.4.1.2 Case II

In this case, the intervals for those distribution parameters are enlarged a little bit compared to case I, as shown in Tab. 5.1. The active learning process of the proposed ALAPI method is illustrated by the upper bound of posterior COV of the augmented failure probability against the number of adaptively added samples, as depicted in Fig. 5.2(b). It is shown that the active learning

process is convergent after the initial training sample set is enriched with 123 samples. That is, the proposed ALAPI only requires 135 g-function evaluations. As a comparison, 352 g-function calls are needed by the NISS method, which is about 2.6 times more than the proposed method. The constant RS-HDMR component computed by the three methods is listed in the fifth to seventh rows of Tab. 5.2. The estimate from IASS method can be taken as the "exact" value because its COV is extremely small. Clearly, the proposed method can produce a more close estimate to the "exact" value than the NISS method in this case. For the first-order RS-HDMR component functions shown in Fig. 5.3(b), it can also be observed that the estimates $\hat{\mathcal{P}}_{\mathrm{RS},2}(\sigma_1)$ and $\hat{\mathcal{P}}_{\mathrm{RS},4}(\sigma_2)$ from the NISS method have larger errors than those by the proposed ALAPI method, by taking the results by IASS as reference. As shown in Fig. 5.4(b), the proposed method can still offer a very accurate estimate of $\hat{\mathcal{P}}_{f,\mathrm{RS},13}(\mu_1, \mu_2)$ with a small upper bound of posterior COV.

To sum up, the proposed method still requires far less g-function calls than the NISS method, but the accuracy of the NISS method becomes worse as the intervals of the distribution parameters are enlarged in this case. Such phenomenon is consistent with what is reported in Ref. [27].



(a) Case I

(b) Case II

Figure 5.2: Upper bound of the posterior COV of $\hat{\mathcal{P}}_{f,\mathrm{RS},0}$ against the number of adaptively added samples for Example 1.

Table 5.2: Constant RS-HDMR component by different methods for Example 1.

Case	Method	$\hat{\mathcal{P}}_{f,\mathrm{RS},0}$	$\overline{\mathbb{COV}}_{\mathcal{T}}[\hat{\mathcal{P}}_{f,\mathrm{RS},0}]$ or $\mathbb{COV}[\hat{\mathcal{P}}_{f,\mathrm{RS},0}]$
I	ALAPI	0.0594	0.0172
	NISS	0.0589	0.0126
	IASS	0.0593	0.0040
II	ALAPI	0.0860	0.0186
	NISS	0.0855	0.0103
	IASS	0.0861	0.0033

5.4.2 Example 2: a nonlinear oscillator

An undamped single-degree-of-freedom oscillator with nonlinear restoring force subject to rectangular pulse load [45] is adapted for the case of imprecise probability, which is shown in Fig. 5.5. The corresponding limit state function reads:

$$y = g\left(m, c_1, c_2, r, F_1, t_1\right) = 3r - \left|\frac{2F_1}{m(c_1+c_2)}\sin\left(\frac{t_1}{2}\sqrt{\frac{c_1+c_2}{m}}\right)\right|, \tag{5.39}$$

As listed in Tab. 5.3, six random variables are included in this example. Due to different levels of knowledge, the mean values are assumed to be deterministic, but the standard deviations are characterized by interval models.

Table 5.3: Statistical information of the random variables for Example 2.

Variable	Description	Distribution	Mean	Standard deviation
m	Mass	Normal	1.0	$\sigma_1 \in [0.02, 0.08]$
c_1	Stiffness of the first spring	Normal	1.0	$\sigma_2 \in [0.05, 0.15]$
c_2	Stiffness of the second spring	Normal	0.1	$\sigma_3 \in [0.005, 0.015]$
r	Yield displacement	Normal	0.5	$\sigma_4 \in [0.02, 0.08]$
F_1	Load amplitude	Lognormal	1.0	$\sigma_5 \in [0.10, 0.30]$
t_1	Load duration	Normal	1.0	$\sigma_6 \in [0.15, 0.25]$

For the ALAPI method, the sample pool is constructed with a set of 10^6 samples, and the threshold ε for the stopping condition is set to be 0.01. A number of 10^6 samples are used for IASS method. As shown in Fig. 5.6, the stopping condition indicates that the GP model is well-trained after a total number of 29 samples are adaptively added into the initial training data set. Therefore, the ALAPI method only requires 41 performance function evaluations in this example, even though the stopping criteria is somehow strict. Tab. 5.4 lists the constant RS-HDMR component estimated by ALAPI and IASS, where it is found that the results of both methods are in good agreement

with each other, and process a quite small upper bound of posterior COV or COV. Thus, we can conclude that both methods offer fairly good estimates for $\hat{\mathcal{P}}_{f,\mathrm{RS},0}$. As shown in Figs. 5.7 and 5.8, the first- and second-order RS-HDMR components are also computed with high accuracy by ALAPI and IASS. Note that the higher-order component functions can also be computed on the basis of the trained GP model if necessary.

Table 5.4: Constant RS-HDMR component by ALAPI and IASS for Example 2.

Method	$\hat{\mathcal{P}}_{f,\mathrm{RS},0}$	$\overline{\mathbb{COV}}_{\mathcal{T}}[\hat{\mathcal{P}}_{f,\mathrm{RS},0}]$ or $\mathbb{COV}[\hat{\mathcal{P}}_{f,\mathrm{RS},0}]$
ALAPI	0.0356	0.0100
IASS	0.0359	0.0052

5.4.3 Example 3: a 120-bar space truss structure

As shown in Fig. 5.9, the third example consists of a 120-bar space truss structure, which has been extensively used as a benchmark in the context of design optimization of structures. In this case study, we would like to estimate the failure probability function when the structure is subjected to some uncertainties characterized by probability boxes, i.e., the Young's modulus of the material E, cross-sectional area A and applied load P. The detailed description of these variables is summarized in Tab. 5.5. The limit state function is defined as:

$$y = g(E, A, P) = \Delta - V(E, A, P), \tag{5.40}$$

where Δ is a threshold, specified as 55 mm; $V(E, A, P)$ is the vertical displacement of the top node, which is solved by a finite-element software, OpenSees.

The proposed ALAPI method is implemented to obtain the failure probability function $\hat{P}_f(\boldsymbol{\theta})$. The number of samples used to construct the sample pool and the threshold of the stopping criterion are set as 10^5 and 0.01, respectively. From Fig. 5.10, it can be found that the stopping criterion is reached after a total of 21 samples are added in the initial training data set. Therefore, the proposed method only needs 33 limit state function evaluations to train a GP model. From the GP model, the RS-HDMR component functions of the FPF can be inferred. For the constant RS-HDMR component, the proposed method is compared to the ASS and IASS. As summarized in Tab. 5.6, the proposed method is computationally much more saving compared to the other two methods in terms of the number of calls to the limit state function, but can still yield fairly good estimate.

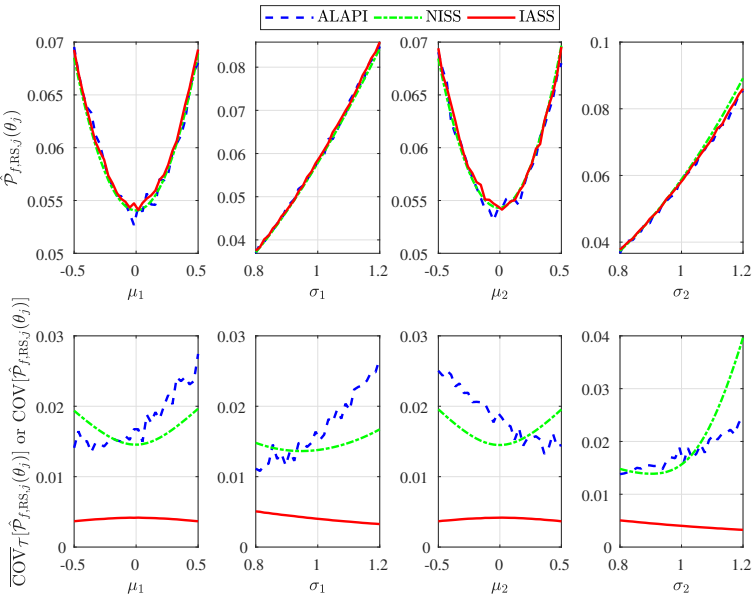
Fig. 5.11 shows the six first-order RS-HDMR component functions and their corresponding upper bound COVs. For limiting the length of our paper, only one second-order RS-HDMR component functions is given, as depicted in Fig. 5.12. From these RS-HDMR component functions, one can perform sensitivity analysis to determine the contribution of each single variable or variable pairs. These information is extremely useful for directing the future information collection so as to further reduce the epistemic uncertainty of the failure probability.

Table 5.5: Statistical information of the random variables for Example 3.

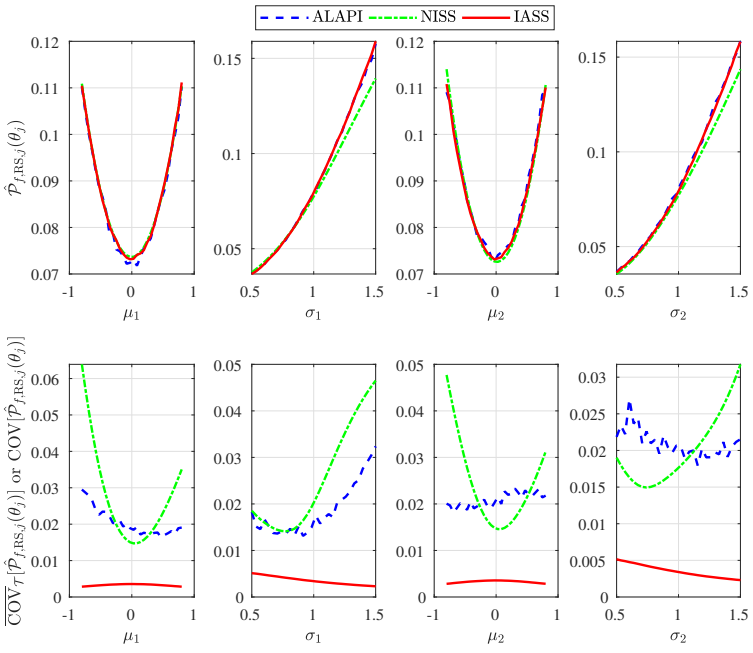
Variable	Distribution	Mean	Standard deviation
E/Mpa	Normal	$\mu_E \in [2.10 \times 10^5, 2.20 \times 10^5]$	$\sigma_E \in [2.10 \times 10^4, 2.20 \times 10^4]$
A/mm	Normal	$\mu_A \in [1000, 1100]$	$\sigma_A \in [100, 110]$
P/kN	Lognormal	$\mu_P \in [500, 600]$	$\sigma_P \in [50, 60]$

Table 5.6: Constant RS-HDMR component by ALAPI, ASS and IASS for Example 3.

Method	$\hat{\mathcal{P}}_{f,\mathrm{RS},0}$	$\overline{\mathbb{COV}}_{\mathcal{T}}[\hat{\mathcal{P}}_{f,\mathrm{RS},0}]$ or $\mathbb{COV}[\hat{\mathcal{P}}_{f,\mathrm{RS},0}]$	N
ALAPI	0.0782	0.0004	33
ASS	0.0803	0.0938	3800
IASS	0.0754	0.0111	10^5

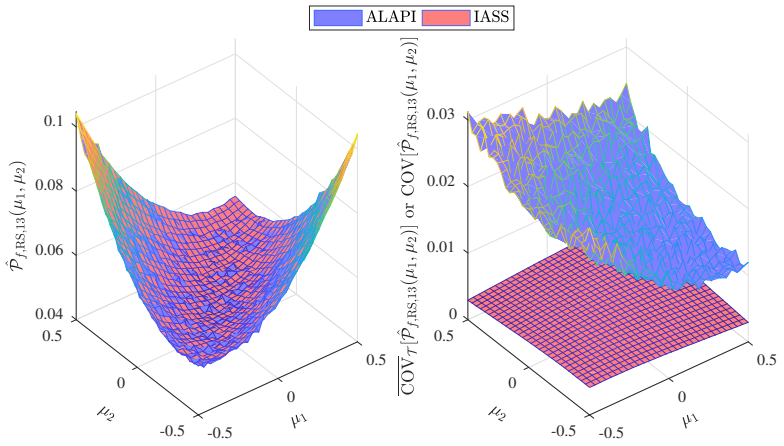


(a) Case I

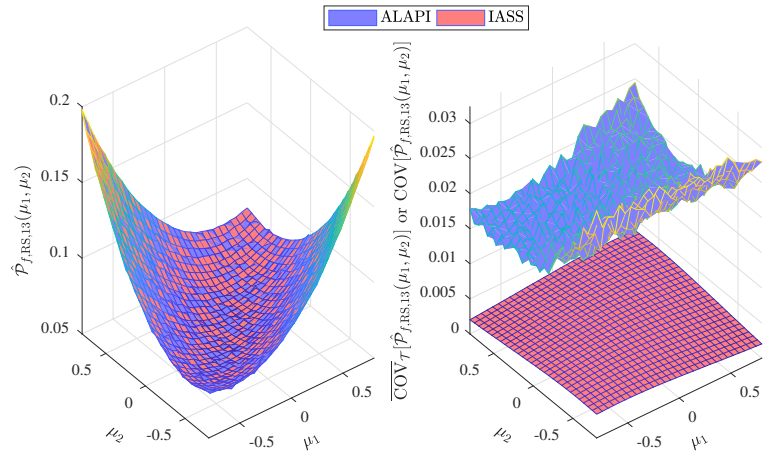


(b) Case II

Figure 5.3: Four first-order RS-HDMR component functions for Example 1.



(a) Case I



(b) Case II

Figure 5.4: A second-order RS-HDMR component function for Example 1.

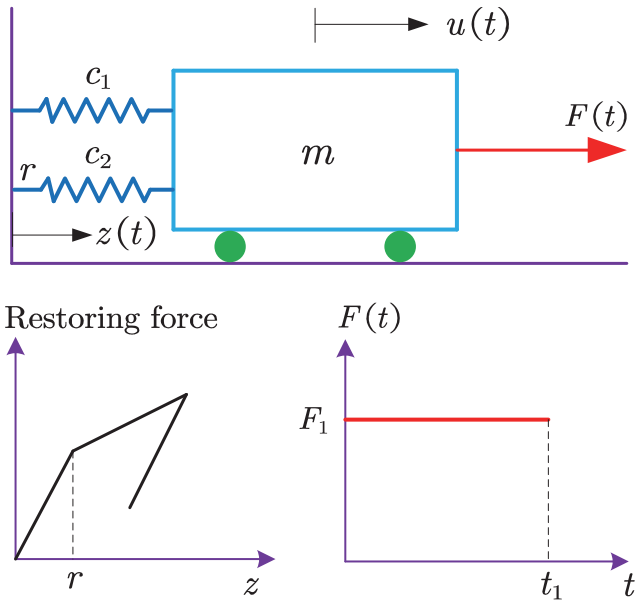


Figure 5.5: An undamped SDOF oscillator with nonlinear restoring force subject to pulse load for Example 2.

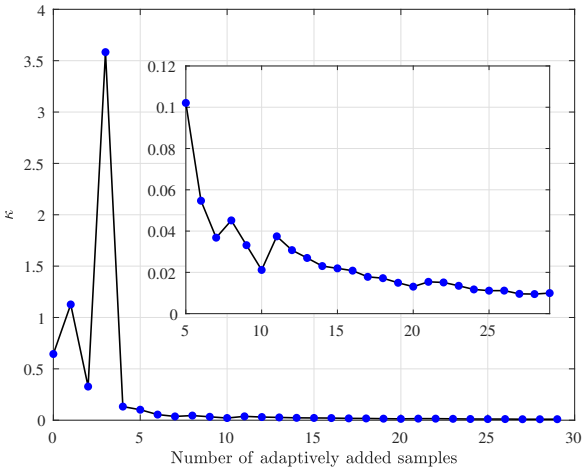


Figure 5.6: Upper bound of the posterior COV of $\hat{\mathcal{P}}_{f,\mathrm{RS},0}$ against the number of adaptively added samples for Example 2.

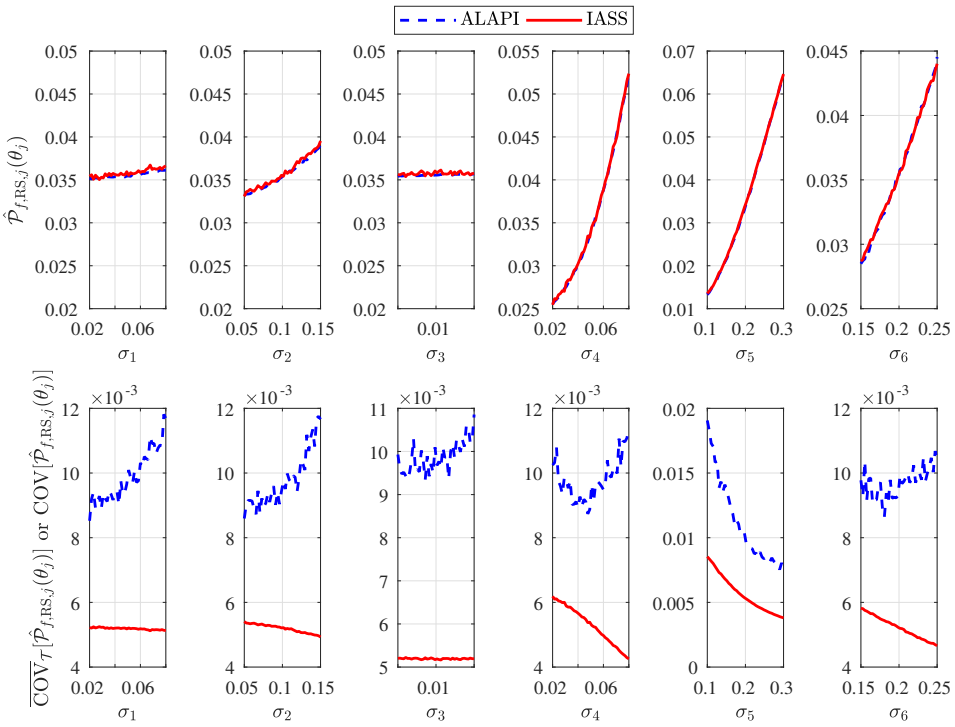


Figure 5.7: Six first-order RS-HDMR component functions for Example 2.

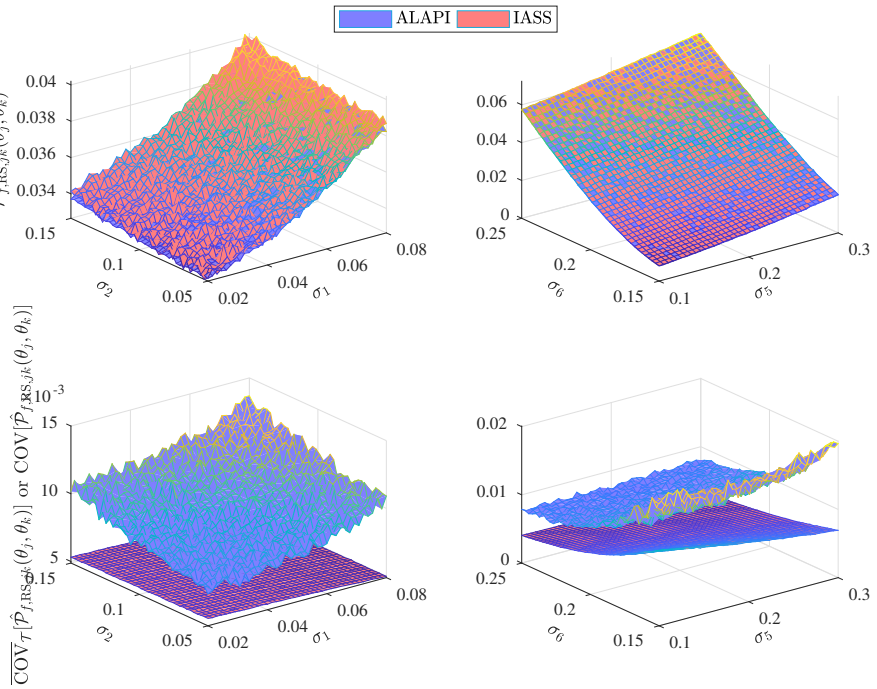


Figure 5.8: Two second-order RS-HDMR component functions for Example 2.

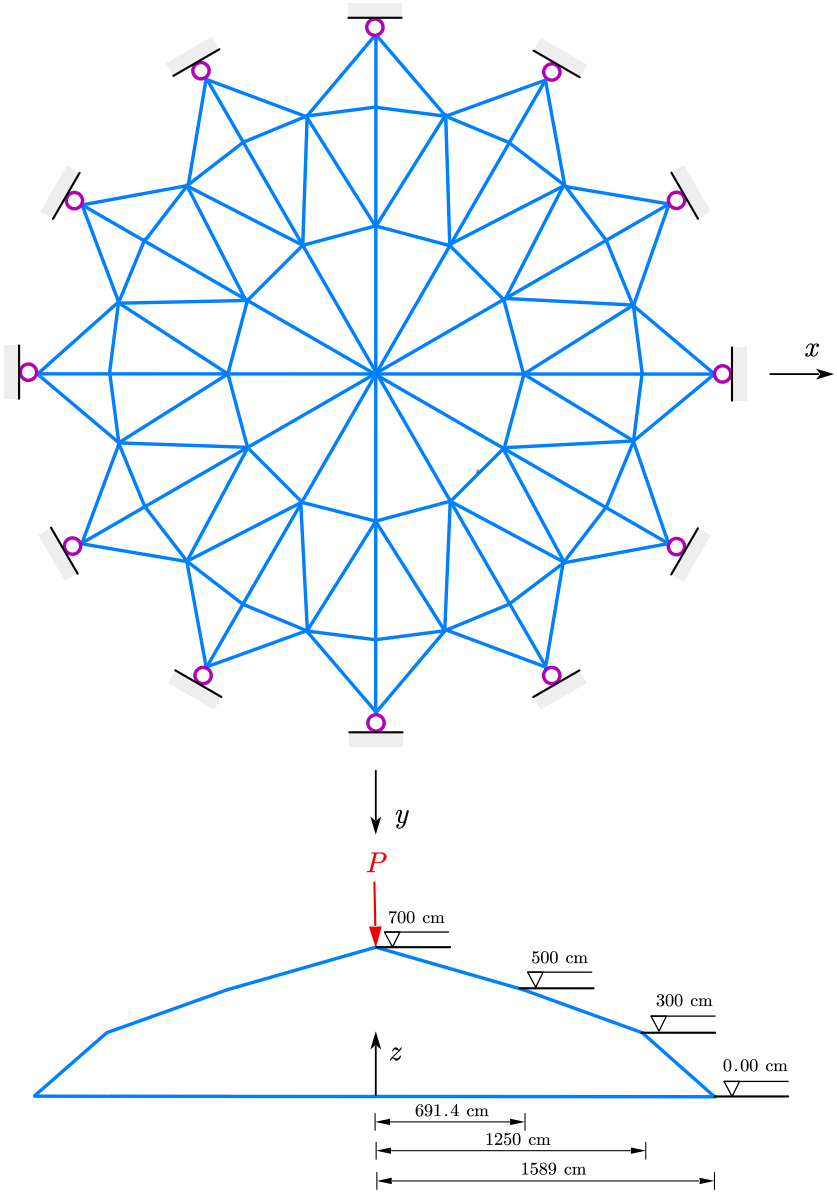


Figure 5.9: A 120-bar space truss structure.

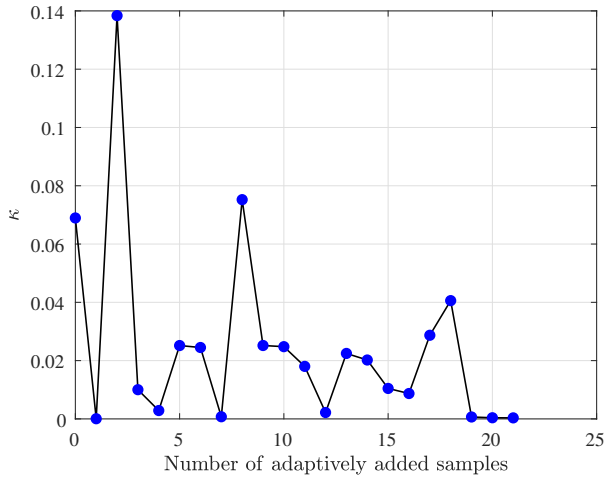


Figure 5.10: Upper bound of the posterior COV of $\hat{\mathcal{P}}_{f,\mathrm{RS},0}$ against the number of adaptively added samples for Example 3.

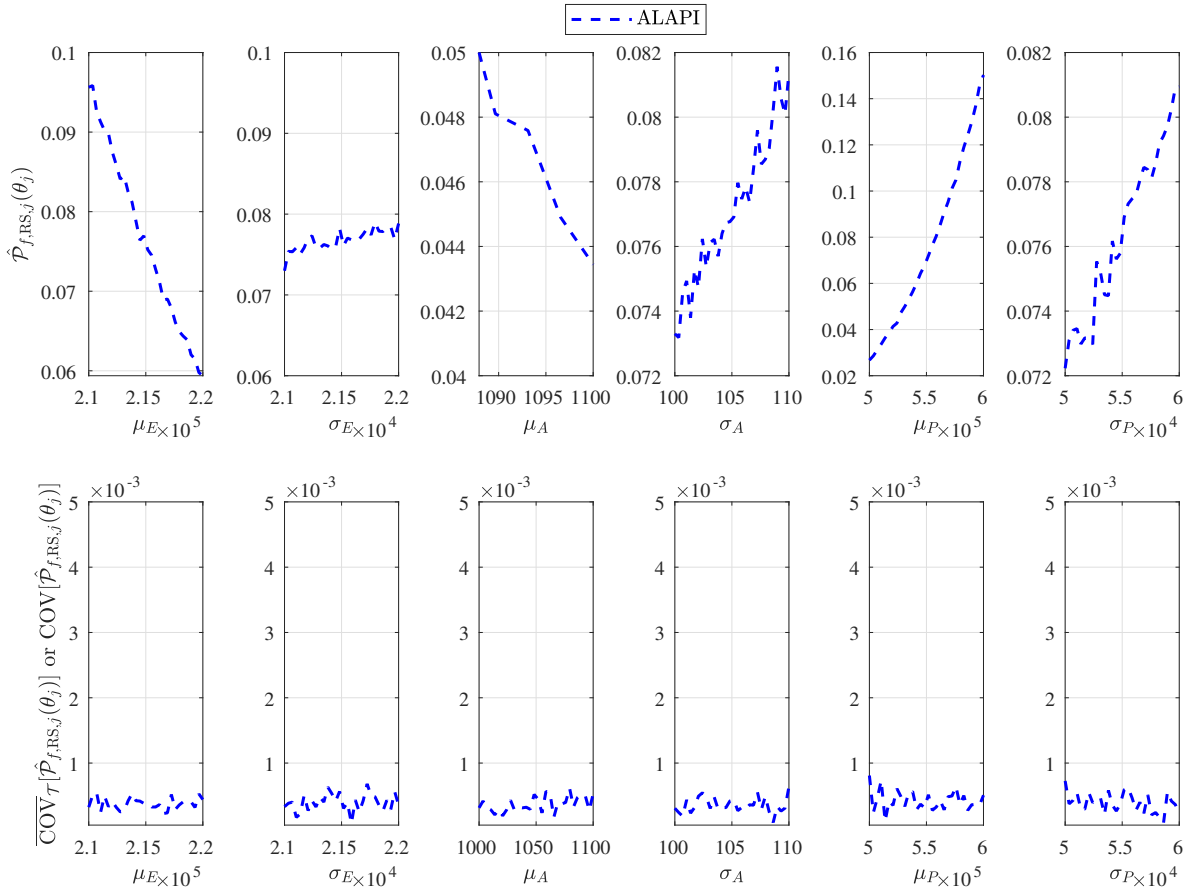


Figure 5.11: Six first-order RS-HDMR component functions for Example 3.

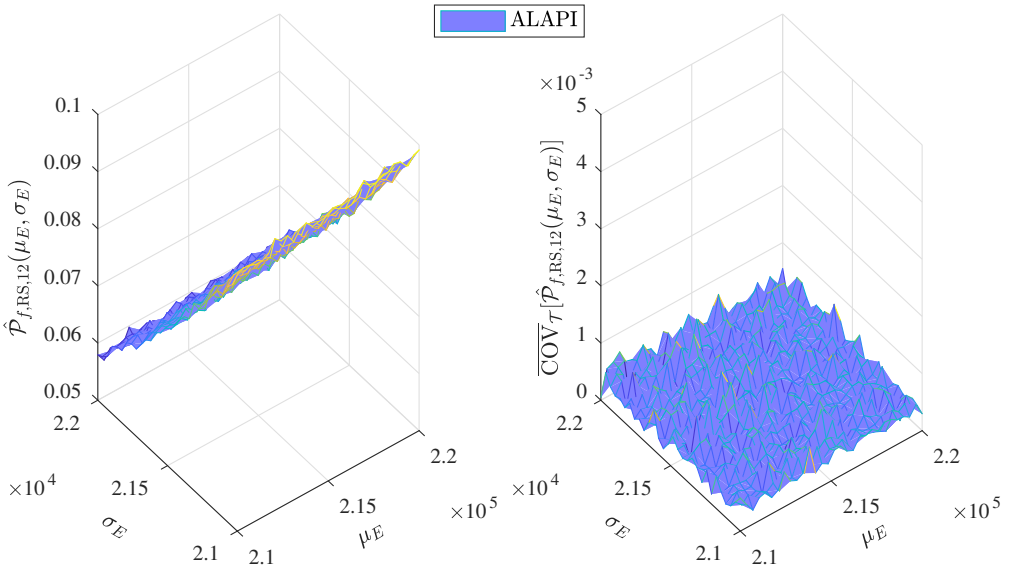


Figure 5.12: One second-order RS-HDMR component function for Example 3.

5.5 Conclusions

The main contribution of this work is to present a novel non-intrusive method, termed as *Active Learning Augmented Probabilistic Integration* (ALAPI), for efficiently estimating the failure probability function in the presence of imprecise probability models. Specifically, the probability-box models are taken as an example for characterizing aleatory uncertainty and epistemic uncertainty by a hierarchical structure. However, all the developments can be conveniently extended to the case with other imprecise probability models. For our purposes, an active learning probabilistic integration (ALPI) method is firstly presented by interpreting the failure probability integral with Bayesian inference, rather than a frequentist view. Further, a imprecise augmented stochastic simulation (IASS) method is proposed based on the ideas of RS-HDMR and augmented uncertainty space. Finally, the ALAPI is formed by a elegant combination of ALPI and IASS. The main feature of ALAPI is that the epistemic uncertainty resulted from discretization error is properly quantified and propagated from the computational pipelines, allowing properly qualifying the accuracy of RS-HDMR component functions of the FPF.

Three numerical examples are investigated to exemplify and validate the proposed method. It is shown that the proposed method can produce very accurate estimates of the RS-HDMR components up to a second order with a small number of g-function calls when the failure probability is relatively larger (typically, with $\hat{\mathcal{P}}_{f,\mathrm{RS},0} > 10^{-3}$). Besides, as revealed by Example 1 the proposed method could be not very sensitive to the level of epistemic uncertainty, which is in contrast to the NISS method. To make the paper concise, only the component functions are presented in the examples, but one can also easily compute the failure probability bounds or sensitivity indices based on the proposed method if interested [6, 19].

While the findings are encouraging, the proposed method is still suffered from some limitations, e.g., small failure probabilities and high dimensions (in terms of the augmented uncertainty vector). These problems will be addressed in the future work.

5.6 Data Availability Statement

Some or all data, models, or code that support the findings of this study are available from the corresponding author upon reasonable request.

5.7 Acknowledgments

The first author would like to appreciate the financial support from China Scholarship Council (CSC). The second author is grateful to the support from the National Natural Science Foundation of China (grant no. NSFC 51905430) and the Alexander von Humboldt Foundation. The second and forth authors would also like to show their thankfulness to the support of Mobility Program 2020 from Sino-German Center (grant number M-0175).

Appendix A: Active learning non-intrusive imprecise stochastic simulation

According to [19], the active learning procedure can be injected into the general NISS framework so as to further reduce the computational burden. Depending on the HDMR used, two kinds of active learning NISS methods, i.e., AK-LEMCS-cut-HDMR and AK-GEMCS-RS-HDMR, have been developed. In the present study, we only compare AK-GEMCS-RS-HDMR with the proposed method, and hence only this method is revisited. Since the RS-HDMR component functions that need to be estimated in the proposed method are somewhat different from those in [19], the original AK-GEMCS-RS-HDMR should be slightly modified for our purposes, and the revised procedures are briefly given as follows.

Step I.1: Generate a set of N simple random samples $\boldsymbol{V} = \{\boldsymbol{X}, \boldsymbol{S}\} = \{\boldsymbol{x}^{(s)}, \boldsymbol{\theta}^{(i)}\}_{i=1}^{N}$ from the augmented PDF $w(\boldsymbol{v})$, which serves as a sample pool for training a GP model for the g-function $g(\boldsymbol{x})$.

Step I.2: Randomly select N_0 (e.g., $N_0 = 12$) samples from $\boldsymbol{X}$, and compute the corresponding g-function values. Attribute these N_0 samples to the training sample set $\boldsymbol{Q}$.

Step I.3: Train or update the GP model, denoted as $\hat{g}(\boldsymbol{x})$, for the g-function $g(\boldsymbol{x})$ based on $\boldsymbol{Q}$.

Step I.4: Compute the GP predictions $\mathbb{E}_{\mathcal{Q}}[\hat{g}(\boldsymbol{x})]$ and $\mathbb{V}_{\mathcal{Q}}[\hat{g}(\boldsymbol{x})]$ based on the trained GP model $\hat{g}(\boldsymbol{x})$ for all the samples in $\boldsymbol{X}$, and judge whether the stopping condition is satisfied with the principle that $\min_{i=1}^{N} U(\boldsymbol{x}^{(i)}) \geq 2$, where $U(\boldsymbol{x}) = \frac{|\mathbb{E}_{\mathcal{Q}}[\hat{g}(\boldsymbol{x})]|}{\sqrt{\mathbb{V}_{\mathcal{Q}}[\hat{g}(\boldsymbol{x})]}}$. If the inequality is satisfied, go to **Step I.4**; otherwise, find the sample $\boldsymbol{x}^{\star}$ with the smallest U value among $\boldsymbol{X}$, compute the corresponding

g-function value $y^\star = g(\boldsymbol{x}^\star)$, add $\{\boldsymbol{x}^\star, y^\star\}$ to the training sample set $\mathcal{Q}$, and go to **Step I.3**;

Step I.4: Based on the well-trained GP model $\hat{g}(\boldsymbol{x})$, obtain a estimate $\hat{P}_f(\boldsymbol{\theta})$ for the FPF. The estimators for the RS-HDMR component functions defined in Eq. (5.22) are given by:

$$\hat{\mathcal{P}}_{f,\mathrm{RS},0} = \frac{1}{N}\sum_{i=1}^{N} \hat{I}_F(\boldsymbol{x}^{(i)}), \tag{5.41}$$

$$\hat{\mathcal{P}}_{f,\mathrm{RS},j}\left(\theta_j\right) = \frac{1}{N}\sum_{i=1}^{N} \hat{I}_F(\boldsymbol{x}^{(i)}) \frac{f(\boldsymbol{x}^{(i)}|\theta_j, \boldsymbol{\theta}_{-j}^{(i)})}{f(\boldsymbol{x}^{(i)}|\boldsymbol{\theta}^{(i)})}, \tag{5.42}$$

$$\hat{\mathcal{P}}_{f,\mathrm{RS},jk}\left(\theta_j, \theta_k\right) = \frac{1}{N}\sum_{i=1}^{N} \hat{I}_F(\boldsymbol{x}^{(i)}) \frac{f(\boldsymbol{x}^{(i)}|\theta_j, \theta_k, \boldsymbol{\theta}_{-jk}^{(i)})}{f(\boldsymbol{x}^{(i)}|\boldsymbol{\theta}^{(i)})}. \tag{5.43}$$

The sampling variability contained in the above estimators can be measured by the following variances:

$$\mathbb{V}\left[\hat{\mathcal{P}}_{f,\mathrm{RS},0}\right] = \frac{1}{(N-1)N}\sum_{i=1}^{N}\left[I_F(\boldsymbol{x}^{(\boldsymbol{i})}) - \hat{\mathcal{P}}_{f,\mathrm{RS},0}\right]^2, \tag{5.44}$$

$$\mathbb{V}\left[\hat{\mathcal{P}}_{f,\mathrm{RS},j}\left(\theta_j\right)\right] = \frac{1}{(N-1)N}\sum_{i=1}^{N}\left[\hat{I}_F(\boldsymbol{x}^{(i)}) \frac{f(\boldsymbol{x}^{(i)}|\theta_j, \boldsymbol{\theta}_{-j}^{(i)})}{f(\boldsymbol{x}^{(i)}|\boldsymbol{\theta}^{(i)})} - \hat{\mathcal{P}}_{f,\mathrm{RS},j}\left(\theta_j\right)\right]^2, \tag{5.45}$$

$$\mathbb{V}\left[\hat{\mathcal{P}}_{f,\mathrm{RS},jk}\left(\theta_j, \theta_k\right)\right] = \frac{1}{(N-1)N}\sum_{i=1}^{N}\left[\hat{I}_F(\boldsymbol{x}^{(i)}) \frac{f(\boldsymbol{x}^{(i)}|\theta_j, \theta_k, \boldsymbol{\theta}_{-jk}^{(i)})}{f(\boldsymbol{x}^{(i)}|\boldsymbol{\theta}^{(i)})} - \hat{\mathcal{P}}_{f,\mathrm{RS},jk}\left(\theta_j, \theta_k\right)\right]^2. \tag{5.46}$$

Bibliography

[1] Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural Safety*, 31(2):105–112, 2009.

[2] Michael Beer, Scott Ferson, and Vladik Kreinovich. Imprecise probabilities in engineering analyses. *Mechanical Systems and Signal Processing*, 37(1-2):4–29, 2013.

[3] Matthias Faes and David Moens. Recent trends in the modeling and quantification of non-probabilistic uncertainty. *Archives of Computational Methods in Engineering*, pages 1–39, 2019.

[4] Chao Jiang, RG Bi, GY Lu, and X Han. Structural reliability analysis using non-probabilistic convex model. *Computer Methods in Applied Mechanics and Engineering*, 254:83–98, 2013.

[5] Bernd Möller and Michael Beer. *Fuzzy randomness: uncertainty in civil engineering and computational mechanics*. Springer Science & Business Media, 2004.

[6] Pengfei Wei, Jingwen Song, Sifeng Bi, Matteo Broggi, Michael Beer, Zhenzhou Lu, and Zhufeng Yue. Non-intrusive stochastic analysis with parameterized imprecise probability models: I. performance estimation. *Mechanical Systems and Signal Processing*, 124:349–368, 2019.

[7] Kari Sentz, Scott Ferson, et al. *Combination of evidence in Dempster-Shafer theory*, volume 4015. Sandia National Laboratories Albuquerque, 2002.

[8] Ronald R Yager and Vladik Kreinovich. Decision making under interval probabilities. *International Journal of Approximate Reasoning*, 22(3):195–215, 1999.

[9] Siao Sun, Guangtao Fu, Slobodan Djordjević, and Soon-Thiam Khu. Separating aleatory and epistemic uncertainties: Probabilistic sewer flooding evaluation using probability box. *Journal of Hydrology*, 420:360–372, 2012.

[10] James J Buckley. *Fuzzy probabilities: new approach and applications*, volume 115. Springer Science & Business Media, 2005.

[11] Hao Zhang, Robert L Mullen, and Rafi L Muhanna. Interval monte carlo methods for structural reliability. *Structural Safety*, 32(3):183–190, 2010.

[12] Hao Zhang, Hongzhe Dai, Michael Beer, and Wei Wang. Structural reliability analysis on the basis of small samples: an interval quasi-monte carlo method. *Mechanical Systems and Signal Processing*, 37(1-2):137–151, 2013.

[13] Hao Zhang. Interval importance sampling method for finite element-based structural reliability assessment under parameter uncertainties. *Structural Safety*, 38:1–10, 2012.

[14] Diego A Alvarez, Felipe Uribe, and Jorge E Hurtado. Estimation of the lower and upper bounds on the probability of failure using subset simulation and random set theory. *Mechanical Systems and Signal Processing*, 100:782–801, 2018.

[15] HB Liu, C Jiang, XY Jia, XY Long, Zhaoyong Zhang, and FJ Guan. A new uncertainty propagation method for problems with parameterized probability-boxes. *Reliability Engineering & System Safety*, 172:64–73, 2018.

[16] HB Liu, C Jiang, J Liu, and JZ Mao. Uncertainty propagation analysis using sparse grid technique and saddlepoint approximation based on parameterized p-box representation. *Structural and Multidisciplinary Optimization*, 59(1):61–74, 2019.

[17] SK Au. Reliability-based design sensitivity by efficient simulation. *Computers & Structures*, 83(14):1048–1061, 2005.

[18] Pengfei Wei, Zhenzhou Lu, and Jingwen Song. Extended monte carlo simulation for parametric global sensitivity analysis and optimization. *AIAA Journal*, 52(4):867–878, 2014.

[19] Pengfei Wei, Jingwen Song, Sifeng Bi, Matteo Broggi, Michael Beer, Zhenzhou Lu, and Zhufeng Yue. Non-intrusive stochastic analysis with parameterized imprecise probability models: II. reliability and rare events analysis. *Mechanical Systems and Signal Processing*, 126:227–247, 2019.

[20] Jingwen Song, Marcos Valdebenito, Pengfei Wei, Michael Beer, and Zhenzhou Lu. Non-intrusive imprecise stochastic simulation by line sampling. *Structural Safety*, 84:101936, 2020.

[21] Jingwen Song, Pengfei Wei, Marcos Valdebenito, and Michael Beer. Adaptive reliability analysis for rare events evaluation with global imprecise line sampling. *Computer Methods in Applied Mechanics and Engineering*, 372:113344, 2020.

[22] Xiukai Yuan, Zhenxuan Zheng, and Baoqiang Zhang. Augmented line sampling for approximation of failure probability function in reliability-based analysis. *Applied Mathematical Modelling*, 80:895–910, 2020.

[23] Matthias GR Faes, Marcos A Valdebenito, David Moens, and Michael Beer. Bounding the first excursion probability of linear structures subjected to imprecise stochastic loading. *Computers & Structures*, 239:106320, 2020.

[24] Matthias GR Faes, Marcos A Valdebenito, David Moens, and Michael Beer. Operator norm theory as an efficient tool to propagate hybrid uncertainties and calculate imprecise probabilities. *Mechanical Systems and Signal Processing*, 152:107482, 2021.

[25] Xiukai Yuan, Matthias GR Faes, Shaolong Liu, Marcos A Valdebenito, and Michael Beer. Effi-

cient imprecise reliability analysis using the augmented space integral. *Reliability Engineering & System Safety*, page 107477, 2021.

[26] Matthias Faes, Marcos Valdebenito, Xiukai Yuan, Pengfei Wei, and Michael Beer. Augmented reliability analysis for estimating imprecise first excursion probabilities in stochastic linear dynamics. *Advances in Engineering Software*, 03 2021.

[27] Pengfei Wei, Fuchao Liu, Marcos Valdebenito, and Michael Beer. Bayesian probabilistic propagation of imprecise probabilities with large epistemic uncertainty. *Mechanical Systems and Signal Processing*, 149:107219, 2021.

[28] Pengfei Wei, Fuchao Liu, Zhenzhou Lu, and Zuotao Wang. A probabilistic procedure for quantifying the relative importance of model inputs characterized by second-order probability models. *International Journal of Approximate Reasoning*, 98:78–95, 2018.

[29] Wang-Sheng Liu and Sai Hung Cheung. Reliability based design optimization with approximate failure probability function in partitioned design space. *Reliability Engineering & System Safety*, 167:602–611, 2017.

[30] Chunyan Ling, Zhenzhou Lu, and Xiaobo Zhang. An efficient method based on ak-mcs for estimating failure probability function. *Reliability Engineering & System Safety*, page 106975, 2020.

[31] Anthony O'Hagan. Bayes–hermite quadrature. *Journal of Statistical Planning and Inference*, 29(3):245–260, 1991.

[32] Carl Edward Rasmussen and Zoubin Ghahramani. Bayesian monte carlo. *Advances in Neural Information Processing Systems*, pages 505–512, 2003.

[33] François-Xavier Briol, Chris J Oates, Mark Girolami, Michael A Osborne, Dino Sejdinovic, et al. Probabilistic integration: A role in statistical computation? *Statistical Science*, 34(1):1–22, 2019.

[34] Pengfei Wei, Xing Zhang, and Michael Beer. Adaptive experiment design for probabilistic integration. *Computer Methods in Applied Mechanics and Engineering*, 365:113035, 2020.

[35] Carl Edward Rasmussen. Gaussian processes in machine learning. In *Summer School on Machine Learning*, pages 63–71. Springer, 2003.

[36] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[37] Vincent Dubourg, Bruno Sudret, and Franois Deheeger. Metamodel-based importance sampling for structural reliability analysis. *Probabilistic Engineering Mechanics*, 33:47–57, 2013.

[38] Sangjune Bae, Chanyoung Park, and Nam H. Kim. Estimating Effect of Additional Sample on Uncertainty Reduction in Reliability Analysis Using Gaussian Process. *Journal of Mechanical Design*, 142(11), 2020. 111706.

[39] B Echard, N Gayton, and M Lemaire. Ak-mcs: an active learning reliability method combining kriging and monte carlo simulation. *Structural Safety*, 33(2):145–154, 2011.

[40] Benjamin Echard, Nicolas Gayton, Maurice Lemaire, and Nicolas Relun. A combined importance sampling and kriging reliability method for small failure probabilities with time-demanding numerical models. *Reliability Engineering & System Safety*, 111:232–240, 2013.

[41] Pengfei Wei, Chenghu Tang, and Yuting Yang. Structural reliability and reliability sensitivity analysis of extremely rare failure events by combining sampling and surrogate model methods. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 233(6):943–957, 2019.

[42] Jingwen Song, Pengfei Wei, Marcos Valdebenito, and Michael Beer. Active learning line sampling for rare event analysis. *Mechanical Systems and Signal Processing*, 147:107113, 2020.

[43] Genyuan Li, Sheng-Wei Wang, and Herschel Rabitz. Practical approaches to construct rs-hdmr component functions. *The Journal of Physical Chemistry A*, 106(37):8721–8733, 2002.

[44] Fengkun Cui and Michel Ghosn. Implementation of machine learning techniques into the subset simulation method. *Structural Safety*, 79:12–25, 2019.

[45] Christian G Bucher and Ulrich Bourgund. A fast and efficient response surface approach for structural reliability problems. *Structural Safety*, 7(1):57–66, 1990.

Chapter 6

Parallel Bayesian active learning for rare event estimation

Parallel adaptive Bayesian quadrature for rare event estimation

Chao Dang[a,⋆], Pengfei Wei[b], Matthias G.R. Faes[c], Marcos A. Valdebenito[d], Michael Beer[a,e,f]

[a] *Institute for Risk and Reliability, Leibniz University Hannover, Callinstr. 34, Hannover 30167, Germany*

[b] *School of Power and Energy, Northwestern Polytechnical University, Xi'an 710072, PR China*

[c] *Chair for Reliability Engineering, TU Dortmund University, Leonhard-Euler-Str. 5, Dortmund 44227, Germany*

[d] *Faculty of Engineering and Sciences, Universidad Adolfo Ibáñez, Av. Padre Hurtado 750, Viña del Mar 2562340, Chile*

[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 20 0 092, PR China*

Published in Reliability Engineering & System Safety in September 2022

Abstract: Various numerical methods have been extensively studied and used for reliability analysis over the past several decades. However, how to understand the effect of numerical uncertainty (i.e., numerical errors due to the discretization of the performance function) on the failure probability is still a challenging issue. The active learning probabilistic integration (ALPI) method offers a principled approach to quantify, propagate and reduce the numerical uncertainty via computation within a Bayesian framework, which has not been fully investigated in context of probabilistic reliability analysis. In this study, a novel method termed 'Parallel Adaptive Bayesian Quadrature' (PABQ) is proposed on the theoretical basis of ALPI, and is aimed at broadening its scope of application. First, the Monte Carlo method used in ALPI is replaced with an importance ball sampling technique so as to reduce the sample size that is needed for rare failure event estimation. Second, a multi-point selection criterion is proposed to enable parallel distributed processing. Four numerical examples are studied to demonstrate the effectiveness and efficiency of the proposed method. It is shown that PABQ can effectively assess small failure probabilities (e.g., as low as 10^{-7}) with a minimum number of iterations by taking advantage of parallel computing.

*Corresponding Author
E-mail address: chao.dang@irz.uni-hannover.de (C. Dang)

Keywords: Reliability analysis, Gaussian process, Numerical uncertainty, Bayesian quadrature, Parallel computing

6.1 Introduction

In many fields, reliability analysis has manifested itself as an essential tool to study the performance of a physical or an engineering system in the presence of uncertainties. A fundamental task in reliability analysis is to compute the probability of a predefined failure event, which is referred as failure probability. Let $\boldsymbol{X} = [X_1, X_2, \cdots, X_d] \in \mathcal{X} \subseteq \mathbb{R}^d$ denote a vector of d random variables with known joint probability density function (PDF) $f_{\boldsymbol{X}}(\boldsymbol{x})$. The performance function (also known as limit state function) is given as $y = g(\boldsymbol{x}) : \mathcal{X} \mapsto \mathcal{Y}$, by which the failure event $F = \{\boldsymbol{x} \in \mathcal{X} : g(\boldsymbol{x}) \leq 0\}$ is defined. The associated failure probability P_f is defined by the following multi-dimensional integral:

$$P_f = \int_{\mathcal{X}} I(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{6.1}$$

where $I(\boldsymbol{x})$ is the failure indicator function, which is defined as:

$$I(\boldsymbol{x}) = \begin{cases} 1, \text{if } g(\boldsymbol{x}) \leq 0 \\ 0, \text{otherwise} \end{cases}. \tag{6.2}$$

To assess the failure probability defined in Eq. (6.1), a variety of numerical methods have been extensively studied and applied by researchers and engineers over the past several decades. In general, the existing methods can be roughly classified into five categories:

1. Stochastic simulation methods, e.g., Monte Carlo simulation (MCS) and its variants (e.g., Subset Simulation (SS) [1] and Importance Sampling (IS) [2, 3]). Despite of their relative robustness to the dimension and complexity of the problem at hand, most of the stochastic simulation methods involve a considerable number of deterministic simulations, and hence are still very computationally demanding, especially for an expensive computational model with a small failure probability;

2. Asymptotic approximation methods, such as first-order reliability method (FORM) [4, 5] and second-order reliability method (SORM) [6, 7]. This kind of methods relies on the first- or second-order Taylor expansion of the limit state surface at the most probable point (MPP).

Hence, its application is challenging whenever one must deal with multiple MPPs and highly nonlinear problems. Besides, FORM and SORM only yield approximate results in general cases, and provide no measure of the error introduced by the expansion;

3. Moment methods, for instance, integer moment based methods [8–10], fractional moment based methods [11–13], moment-generating function (or Laplace transform) based methods [14–16]. The basic idea of these methods is to fit a proper probability distribution to the output variable of a performance function based on the knowledge of its estimated moments of certain type, which, however, typically leads to an ill-posed inverse problem (i.e., the so-called classical moment problem). Moreover, the estimation errors arising from both the estimated moments and assumed probability distribution model could be intractable to assess and handle;

4. Probability-conservation based methods, including, e.g., probability density evolution method [17–20] and direct probability integral method [21–24]. These methods are established on rigorous theoretical fundamentals, but may still suffer from numerical difficulty especially for problems with high-dimensional inputs and/or rare failure events;

5. Surrogate assisted methods. This type of methods is of special interest in the present paper since the proposed method also falls in this category in some sense.

Surrogate assisted methods aim at constructing an inexpensive-to-evaluate surrogate model in place of the original expensive-to-evaluate performance function based on a limited number of its observations. Then, for example, stochastic simulation methods can be directly applied in conjunction with the surrogate model to produce a failure probability estimate. Typical surrogate models for reliability analysis include response surface methods [25–27], support vector machines [28–30], polynomial chaos expansions [31, 32], Gaussian process regression (GPR, also known as Kriging) [33–35], etc. In addition to developing new surrogate models, there has been growing attention paid to adaptive (optimal) design of experiments for training these surrogates. In this line, the GPR model is of particular interest for constructing an adaptive meta-model due to its attractive features, especially for active learning sampling strategies. Representative learning functions consist of the expected feasibility [36], U [37], expected risk [38], H [39], least improvement [40], reliability-based expected improvement [41], folded normal based expected improvement [42], upper-bound

posterior variance contribution (UPVC) [43] and so forth. Besides, the following three aspects have also been paid special attention to in recent publications:

1. assessing small failure probabilities. In addition to MCS, other stochastic simulation methods requiring less samples are combined with active learning Kriging (AK) to evaluate small failure probabilities.

   Representative works include, e.g., AK-IS [44], meta-IS [34], AK-SS [45] and AK-MCMC (Markov chain Monte Carlo) [46], etc;

2. addressing high-dimensional problems. This aspect is mainly tackled by using some dimension-reduction techniques, e.g., active subspace methods [47, 48], principal component analysis [49–51], sufficient dimension reduction [52] and sliced inverse regression [53], etc;

3. enabling parallel computing. Most existing learning functions can only identify one point at each iteration, hindering the use of ever-increasing parallel-computing facilities. To overcome this obstacle, tailored strategies have been proposed, which are mainly based on applying clustering algorithms, such as k-means clustering [34], density clustering [54], spectral clustering [55] and k-medoids clustering [56].

The interested reader can refer to [57] for a comprehensive review. Despite great efforts, most existing Kriging assisted methods still possess respective limitations, and leave room for further improvement in terms of applicability, efficiency and accuracy.

In fact, Gaussian process model can be used in a different way, instead of a pure surrogate model. The first author and his co-workers proposed an active learning probabilistic integration (ALPI) method in a recent paper [43]. In this method, a Bayesian perspective is advocated to reinterpret failure probability integral estimation. By placing a prior distribution (i.e., Gaussian process) over the performance function, we finally arrive at a posterior distribution over the failure probability conditional on some observations of the performance function. The induced posterior distribution of the failure probability reflects the fact that the performance function has been discretised, and hence numerical uncertainty arises due to discretization error. A novel feature of ALPI is that the numerical uncertainty can be properly quantified, propagated and reduced via computation, which distinguishes it from other existing methods. Unfortunately, the idea is only investigated in the context of imprecise probabilities, and lacks of comprehensive studies for probabilistic reliability

analysis.

In this paper, the ALPI method is specially studied under the framework of precise probabilities. The basic idea is explained in a detailed way, and some limitations existing in the previous numerical algorithm are identified. Most importantly, we propose a new method called 'Parallel Adaptive Bayesian Quadrature' (PABQ) on the theoretical basis of the original ALPI method, while alleviating its main limitations. Compared to ALPI, PABQ has two significant advantages. First, PABQ can select multiple points at each iteration, and as such supports parallel distributed processing. Second, PABQ can assess very small failure probabilities without generating a prohibitively large number of candidate samples. Additionally, the Matlab code of the developed method is freely available to the public [1].

The outline of the remaining paper is as follows. The original ALPI method is revisited in Section 6.2 and the theoretical foundations are deepened. Section 6.3 gives the newly developed PABQ method. Four numerical examples are investigated in Section 6.4 to illustrate the performance of the PABQ method. Section 6.5 gives some concluding remarks of the present study.

6.2 Active learning probabilistic integration

This section gives a review of the ALPI method. In comparison to [43], we will explain the basic idea of ALPI in a more detailed and rigorous way, and provide its numerical algorithm that was omitted in [43]. Besides, the advantages and disadvantages of the method will be discussed.

6.2.1 Theoretical background

The ALPI method offers a Bayesian approach to approximating the intractable failure probability integral, which is defined in Eq. (6.1). The method is strongly motivated by Bayesian (probabilistic) integration (also well known as Bayesian quadrature or cubature) [58–60]. To be specific, the ALPI method turns the task of failure probability estimation into a Bayesian inference problem from limited data, as opposed to classical frequentist inference. To do so, we think of the g-function as being random. This is understandable in the Bayesian sense that the numerical value of $g(\boldsymbol{x})$ is always unknown until we actually evaluate $g(\cdot)$ at some point $\boldsymbol{x}$, though the g-function is said to be deterministic. Such interpretation is justified since we can not afford to compute $g(\cdot)$

[1] https://github.com/Chao-Dang/Rare-Event-Estimation-by-Parallel-Adaptive-Bayesian-Quadrature

at every possible location. In this regard, epistemic uncertainty due to discretisation error arises where the g-function is not evaluated. This kind of uncertainty will propagate into the failure indicator function $I(\boldsymbol{x})$ and will therefore affect the failure probability estimate. Consequently, the epistemic uncertainty should be properly treated within our computational framework, because it is not always negligible, especially when the available observations are scarce. Following a standard Bayesian approach, the ALPI method is intended to quantify, propagate and reduce the epistemic uncertainty. Specifically, ALPI first assigns a prior distribution over the g-function. Then, conditioning on some observations $\mathscr{D} = \{\mathscr{X}, \mathscr{Y}\}$ ($\mathscr{X} = \left\{\boldsymbol{x}^{(i)}\right\}_{i=1}^{n}$ with $\boldsymbol{x}^{(i)}$ being the i-th row of $\mathscr{X}$ and $\mathscr{Y} = \left\{y^{(i)}\right\}_{i=1}^{n}$ with $y^{(i)} = g(\boldsymbol{x}^{(i)})$ being the i-th row of $\mathscr{Y}$), gives arise to a posterior distribution of g according to Bayes' rule. This will in turn imply a posterior distribution over $I(\boldsymbol{x})$, and so does over P_f. Technical details of ALPI will be discussed below.

ALPI starts by placing a Gaussian process (GP) prior over the g-function, which is written as:

$$\hat{g}_0 \sim \mathcal{GP}(m_{\hat{g}_0}(\boldsymbol{x}), k_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{x}')), \tag{6.3}$$

where $\hat{g}_0$ denotes the prior distribution of g before seeing any observations; $m_{\hat{g}_0}(\boldsymbol{x})$ and $k_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{x}')$ are the prior mean and covariance functions respectively, by which the GP model can be completely characterized. Among many options for $m_{\hat{g}_0}(\boldsymbol{x})$ and $k_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{x}')$ in the literature, without loss of generality the constant prior mean is adopted (i.e., $m_{\hat{g}_0}(\boldsymbol{x}) = \beta$), and the prior covariance function takes the squared exponential kernel:

$$k_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp\left(-\frac{1}{2}\left(\boldsymbol{x} - \boldsymbol{x}'\right)\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{x} - \boldsymbol{x}'\right)^{\top}\right), \tag{6.4}$$

where σ^2 with $\sigma > 0$ denotes the process variance; $\boldsymbol{\Sigma} = \mathrm{diag}(l_1^2, l_2^2, \cdots, l_d^2)$ with $l_i > 0$ being the length scale in the i-th dimension, and $\mathrm{diag}(\cdot)$ forms a diagonal matrix whose diagonal elements are its arguments. The $d + 2$ parameters collected in $\boldsymbol{\vartheta} = \{\beta, \sigma, l_1, l_2, \cdots, l_d\}$ are referred to hyper-parameters to be determined. In a fully Bayesian fashion, those hyper-parameters should also be specified by Bayesian inference (see, e.g., [61]). However, this will render the posterior distribution of g analytically intractable. For this reason, it was not explored in ALPI.

Alternatively, given the data $\mathscr{D}$, the hyper-parameters are fitted by minimizing the negative

log marginal likelihood (NLML) $\mathcal{L}(\boldsymbol{\vartheta})$:

$$\hat{\boldsymbol{\vartheta}} = \arg\min_{\boldsymbol{\vartheta}} \mathcal{L}(\boldsymbol{\vartheta}), \tag{6.5}$$

with the NLML $\mathcal{L}(\boldsymbol{\vartheta})$ being:

$$\mathcal{L}(\boldsymbol{\vartheta}) = -\log\left[p(\mathscr{Y}|\mathscr{X}, \boldsymbol{\vartheta})\right] = \frac{1}{2}(\mathscr{Y} - \beta)^\top \boldsymbol{K}_{\hat{g}_0}^{-1}(\mathscr{Y} - \beta) + \frac{1}{2}\log\left[|\boldsymbol{K}_{\hat{g}_0}|\right] + \frac{n}{2}\log[2\pi], \tag{6.6}$$

where $p(\mathscr{Y}|\mathscr{X}, \boldsymbol{\vartheta})$ is the marginal likelihood following a normal distribution; $\boldsymbol{K}_{\hat{g}_0}$ is the covariance matrix with (i, j)-th entry $[\boldsymbol{K}_{\hat{g}_0}]_{i,j} = k_{\hat{g}_0}(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$.

Once the point estimate hyper-parameters $\hat{\boldsymbol{\vartheta}}$ are obtained, it turns out that the posterior distribution of g can be derived in closed form, i.e., another GP:

$$\hat{g}_n \sim \mathcal{GP}(m_{\hat{g}_n}(\boldsymbol{x}), k_{\hat{g}_n}(\boldsymbol{x}, \boldsymbol{x}')), \tag{6.7}$$

where $\hat{g}_n$ denotes the posterior distribution of g conditional on $\mathscr{D}$; $m_{\hat{g}_n}(\boldsymbol{x})$ and $k_{\hat{g}_n}(\boldsymbol{x}, \boldsymbol{x}')$ are the posterior mean and covariance functions respectively, which are analytically available:

$$m_{\hat{g}_n}(\boldsymbol{x}) = m_{\hat{g}_0}(\boldsymbol{x}) + \boldsymbol{k}_{\hat{g}_0}(\boldsymbol{x}, \mathscr{X})^\top \boldsymbol{K}_{\hat{g}_0}^{-1}\left(\mathscr{Y} - \boldsymbol{m}_{\hat{g}_0}(\mathscr{X})\right), \tag{6.8}$$

$$k_{\hat{g}_n}\left(\boldsymbol{x}, \boldsymbol{x}'\right) = k_{\hat{g}_0}\left(\boldsymbol{x}, \boldsymbol{x}'\right) - \boldsymbol{k}_{\hat{g}_0}(\boldsymbol{x}, \mathscr{X})^\top \boldsymbol{K}_{\hat{g}_0}^{-1} \boldsymbol{k}_{\hat{g}_0}(\boldsymbol{x}', \mathscr{X}), \tag{6.9}$$

where $\boldsymbol{m}_{\hat{g}_0}(\mathscr{X})$ is an $n \times 1$ mean vector with i-th element being $m_{\hat{g}_0}(\boldsymbol{x}^{(i)})$; $\boldsymbol{k}_{\hat{g}_0}(\boldsymbol{x}, \mathscr{X})$ is an $n \times 1$ covariance vector with i-th entry being $k_{\hat{g}_0}\left(\boldsymbol{x}, \boldsymbol{x}^{(i)}\right)$; $\boldsymbol{k}_{\hat{g}_0}(\boldsymbol{x}', \mathscr{X})$ is defined in a way similar to $\boldsymbol{k}_{\hat{g}_0}(\boldsymbol{x}, \mathscr{X})$. Note that in Eqs. (6.8) and (6.9) $\boldsymbol{\vartheta}$ should be updated with $\hat{\boldsymbol{\vartheta}}$.

It can be deduced that the posterior distribution of failure indicator function I follows a generalized Bernoulli process [2] (GBP):

$$\hat{I}_n \sim \mathcal{GBP}(m_{\hat{I}_n}(\boldsymbol{x}), k_{\hat{I}_n}\left(\boldsymbol{x}, \boldsymbol{x}'\right)), \tag{6.10}$$

where $\hat{I}_n$ denotes the posterior distribution of I conditional on $\mathscr{D}$; $m_{\hat{I}_n}(\boldsymbol{x})$ and $k_{\hat{I}_n}\left(\boldsymbol{x}, \boldsymbol{x}'\right)$ are the posterior mean and covariance functions respectively. The posterior mean of I can be derived in

[2] 'generalized' indicates that the Bernoulli process considered here is location-dependent, in contrast to not considering the dependence in conventional definition of a Bernoulli process.

closed form [43]:

$$m_{\hat{I}_n}(\boldsymbol{x}) = \Phi\left(-\frac{m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right), \tag{6.11}$$

where Φ is the cumulative distribution function of the standard normal distribution; $\sigma_{\hat{g}_n}(\boldsymbol{x})$ is the posterior standard derivation (STD) function of g, i.e., $\sigma_{\hat{g}_n}(\boldsymbol{x}) = \sqrt{k_{\hat{g}_n}(\boldsymbol{x}, \boldsymbol{x})}$. The posterior covariance function of I, however, is not analytically tractable. Only closed-form expression for its posterior variance function $\sigma^2_{\hat{I}_n}(\boldsymbol{x})$ is available [43]:

$$\sigma^2_{\hat{I}_n}(\boldsymbol{x}) = \Phi\left(-\frac{m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right)\Phi\left(\frac{m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right). \tag{6.12}$$

The posterior distribution $\hat{P}_{f,n}$ of failure probability P_f conditional on the data $\mathscr{D}$ should thus follow a random variable, which reflects our epistemic uncertainty about P_f, due to the limited number of observations. Note that the exact posterior distribution of P_f, however, is not known. Instead, the posterior mean and variance of P_f should be more of interest, where the posterior mean corresponds to the failure probability predictor and the posterior variance measures the prediction uncertainty. By applying Fubini's theorem, the posterior mean and variance of P_f can be derived as [43]:

$$\begin{aligned} m_{\hat{P}_{f,n}} &= \mathbb{E}_{\hat{I}_n}\left[\hat{P}_{f,n}\right] \\ &= \mathbb{E}_{\hat{I}_n}\left[\int_{\mathcal{X}} \hat{I}_n(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}\right] \\ &= \int_{\mathcal{X}} \mathbb{E}_{\hat{I}_n}\left[\hat{I}_n(\boldsymbol{x})\right] f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \\ &= \int_{\mathcal{X}} m_{\hat{I}_n}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \\ &= \int_{\mathcal{X}} \Phi\left(-\frac{m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \end{aligned} \tag{6.13}$$

$$
\begin{aligned}
\sigma^2_{\hat{P}_{f,n}} &= \mathbb{V}_{\hat{I}_n}\left[\hat{P}_{f,n}\right] \\
&= \mathbb{E}_{\hat{I}_n}\left[\left(\hat{P}_{f,n} - \mathbb{E}_{\hat{I}_n}\left[\hat{P}_{f,n}\right]\right)^2\right] \\
&= \mathbb{E}_{\hat{I}_n}\left[\left(\int_{\mathcal{X}} \hat{I}_n(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int_{\mathcal{X}} \mathbb{E}_{\hat{I}_n}\left[\hat{I}_n(\boldsymbol{x})\right] f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right)^2\right] \\
&= \mathbb{E}_{\hat{I}_n}\left[\left(\int_{\mathcal{X}} \left(\hat{I}_n(\boldsymbol{x}) - \mathbb{E}_{\hat{I}_n}\left[\hat{I}_n(\boldsymbol{x})\right]\right) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right)^2\right] \\
&= \mathbb{E}_{\hat{I}_n}\left[\left(\int_{\mathcal{X}} \left(\hat{I}_n(\boldsymbol{x}) - \mathbb{E}_{\hat{I}_n}\left[\hat{I}_n(\boldsymbol{x})\right]\right) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right) \times \left(\int_{\mathcal{X}} \left(\hat{I}_n(\boldsymbol{x}') - \mathbb{E}_{\hat{I}_n}\left[\hat{I}_n(\boldsymbol{x}')\right]\right) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}'\right)\right] \\
&= \mathbb{E}_{\hat{I}_n}\left[\int_{\mathcal{X}}\int_{\mathcal{X}} \left(\hat{I}_n(\boldsymbol{x}) - \mathbb{E}_{\hat{I}_n}\left[\hat{I}_n(\boldsymbol{x})\right]\right)\left(\hat{I}_n(\boldsymbol{x}') - \mathbb{E}_{\hat{I}_n}\left[\hat{I}_n(\boldsymbol{x}')\right]\right) f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}'\right] \\
&= \int_{\mathcal{X}}\int_{\mathcal{X}} \mathbb{E}_{\hat{I}_n}\left[\left(\hat{I}_n(\boldsymbol{x}) - \mathbb{E}_{\hat{I}_n}\left[\hat{I}_n(\boldsymbol{x})\right]\right)\left(\hat{I}_n(\boldsymbol{x}') - \mathbb{E}_{\hat{I}_n}\left[\hat{I}_n(\boldsymbol{x}')\right]\right)\right] f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}' \\
&= \int_{\mathcal{X}}\int_{\mathcal{X}} k_{\hat{I}_n}(\boldsymbol{x}, \boldsymbol{x}')\, f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}',
\end{aligned}
\tag{6.14}
$$

where $\mathbb{E}_{\hat{I}_n}[\cdot]$ and $\mathbb{V}_{\hat{I}_n}[\cdot]$ denote expectation and variance operators taken over $\hat{I}_n$ respectively. For computational purposes, Eq. (6.14) is further simplified by considering its upper bound. According to the Cauchy- Schwarz inequality ($k_{\hat{I}_n}(\boldsymbol{x}, \boldsymbol{x}') \leq \sigma_{\hat{I}_n}(\boldsymbol{x})\,\sigma_{\hat{I}_n}(\boldsymbol{x}')$), an upper-bound of the posterior variance (UPV) $\sigma^2_{\hat{P}_{f,n}}$ is given as [43]:

$$
\sigma^2_{\hat{P}_{f,n}} \leq \overline{\sigma}^2_{\hat{P}_{f,n}} = \left(\int_{\mathcal{X}} \sqrt{\Phi\left(-\frac{m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right)\Phi\left(\frac{m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right)} f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right)^2, \tag{6.15}
$$

where the equality holds when the correlation of $\hat{I}_n$ between any two locations $(\boldsymbol{x}, \boldsymbol{x}')$ is always equal to 1, and $\overline{\sigma}^2_{\hat{P}_{f,n}}$ denotes the upper-bound of the posterior variance.

At the theoretical level, ALPI provides two important benefits. First, it offers a principled approach to the quantification and propagation of numerical uncertainty via computation within the Bayesian framework. Second, it gives the possibility to reduce the numerical uncertainty by using an active learning strategy (see next subsection).

6.2.2 Numerical algorithm

For practical reliability analysis, the failure probability estimate should be inferred using as few observations as possible, with the premise of limiting its numerical uncertainty within a pre-specified tolerance. Besides, as the posterior mean and UPV of failure probability (Eqs. (6.13)

and (6.15)) lack of closed-form solutions, a numerical integrator is necessary to make the method practically feasible. The numerical algorithm of the ALPI method for failure probability estimation is summarized in 6.5.

When it comes to numerical implementation, ALPI shows two main limitations. First, it is not applicable to problems with extremely small failure probabilities (typically, less than 10^{-4}) as a large number of Monte Carlo (MC) samples (typically, more than 10^6) are required, making each iteration computationally cumbersome and even infeasible. Second, it is not suitable for parallel computing since only one point is identified at each iteration, resulting in a waste of useful information and computational resources for engineering applications.

6.3 Parallel adaptive Bayesian quadrature

The major limitations of ALPI at implementation level will be addressed in this section. Further, a novel method, called 'Parallel Adaptive Bayesian Quadrature' (PABQ), is presented on the theoretical basis of ALPI. As its name indicates, the proposed PABQ method can support parallel distributed processing. Most importantly, PABQ is able to estimate very small failure probabilities (e.g., 10^{-7}).

6.3.1 General remarks

As we did not imply any distribution types for $\boldsymbol{X}$ when making Bayesian inference about the failure probability in the last section, it means that the ALPI framework is naturally applicable in the standard normal space. In view of this, let us transform $g(\boldsymbol{x})$ from the physical space $\mathcal{X}$ to the standard normal space $\mathcal{U}$, i.e., $g(\boldsymbol{x}) = g(T^{-1}(\boldsymbol{u})) = \mathcal{G}(\boldsymbol{u})$, where $\boldsymbol{u}$ is a realization of the standard normal vector $\boldsymbol{U} = [U_1, U_2, \cdots, U_d] \in \mathcal{U} \subseteq \mathbb{R}^d$ and T^{-1} is the inverse transformation (e.g., iso-probabilistic, Nataf, and Rosenblatt transformation, etc.). For clarification, the transformed performance function is denoted as $Z = \mathcal{G}(\boldsymbol{U})$. Different from ALPI, the proposed PABQ method will be implemented with the $\mathcal{G} = g \circ T^{-1}$-function.

6.3.2 Importance ball sampling

In this subsection, we propose an importance ball sampling (IBS) technique to replace the MC method used in the conventional ALPI method. Let us first introduce a ball, a region enclosed by

a sphere or hypersphere. The d-ball of radius $r > 0$ in the standard normal space $\mathcal{U}$ can be defined as $B^d(r) = \left\{\boldsymbol{u} \in \mathbb{R}^d : ||\boldsymbol{u}||_2 \leq r\right\}$, where $||\cdot||_2$ denotes the 2-norm. The ball is said to be 'important' when it can cover the standard normal space with relatively large probability content (in case that r is appropriately chosen). The uniform PDF over $B^d(r)$ takes the form:

$$f_B(\boldsymbol{u}) = \begin{cases} \frac{1}{V_d(r)}, \text{ if } ||\boldsymbol{u}||_2 \leq r; \\ 0, \text{ otherwise} \end{cases}, \tag{6.16}$$

where $V_d(r) = \frac{\pi^{d/2}}{\Gamma\left(\frac{d}{2}+1\right)} r^d$ is the volume of $B^d(r)$, $\Gamma(\cdot)$ is Euler's gamma function. To generate random points uniformly distributed within the d-ball, there are many methods available in the literature. In this study, one algorithm reported in [62] is adopted, as summarized in Algorithm 3.

Algorithm 3 Generate uniform samples within the d-ball [62]

1: **Input:** dimension d, radius r and sample size N_{ibs}
2: **for** $i = 1, 2, \cdots, N_{ibs}$ **do**
3: Generate d normally distributed samples, $\boldsymbol{w} = [w^{(1)}, w^{(2)}, \cdots, w^{(d)}], w^{(i)} \sim \mathcal{N}(0, 1)$
4: Generate a uniformly distributed sample v from the interval $[0, 1]$
5: Return the i-th vector $\overline{\boldsymbol{u}}^{(i)} = \frac{r v^{1/d} \boldsymbol{w}}{||\boldsymbol{w}||_2}$
6: **end for**
7: **Output:** $\overline{\mathscr{U}} = \left\{\overline{\boldsymbol{u}}^{(i)}\right\}_{i=1}^{N_{ibs}}$: N_{ibs} uniform samples in $B^d(r)$

Then, consider an auxiliary PDF constructed as follows:

$$f_0(\boldsymbol{u}) = \begin{cases} (1 - \Delta) f_B(\boldsymbol{u}), ||\boldsymbol{u}||_2 \leqslant r \\ f_{\boldsymbol{U}}(\boldsymbol{u}), \text{otherwise} \end{cases}, \tag{6.17}$$

where $f_{\boldsymbol{U}}(\boldsymbol{u})$ is the joint PDF of $\boldsymbol{U}$; Δ is a normalizing constant that ensures that the PDF $f_0(\boldsymbol{u})$ integrates to one, which is actually equal to the probability of $f_{\boldsymbol{U}}(\boldsymbol{u})$ outside $B^d(r)$, i.e., $\Delta = \int_{\mathcal{U} \setminus \mathcal{B}} f_{\boldsymbol{U}}(\boldsymbol{u}) \, \mathrm{d}\boldsymbol{u}$. The posterior mean $m_{\hat{P}_{f,n}}$ and upper-bound of posterior standard deviation

(UPSTD) $\overline{\sigma}_{\hat{P}_{f,n}}$ with respect to the $\mathcal{G}$-function can be reformulated respectively as:

$$\begin{aligned}
m_{\hat{P}_{f,n}} =& \int_{\mathcal{U}} \Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) f_{\boldsymbol{U}}(\boldsymbol{u})\mathrm{d}\boldsymbol{u} \\
=& \int_{\mathcal{U}} \Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) \frac{f_{\boldsymbol{U}}(\boldsymbol{u})}{f_0(\boldsymbol{u})} f_0(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u} \\
=& \int_{\mathcal{B}} \Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) \frac{f_{\boldsymbol{U}}(\boldsymbol{u})}{(1-\Delta) f_B(\boldsymbol{u})} (1-\Delta) f_B(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u} \\
&+ \int_{\mathcal{U}\backslash\mathcal{B}} \Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) \frac{f_{\boldsymbol{U}}(\boldsymbol{u})}{f_{\boldsymbol{U}}(\boldsymbol{u})} f_{\boldsymbol{U}}(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u} \\
=& V_d(r) \int_{\mathcal{B}} \Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) f_{\boldsymbol{U}}(\boldsymbol{u}) f_B(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u} \\
&+ \int_{\mathcal{U}\backslash\mathcal{B}} \Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) f_{\boldsymbol{U}}(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u},
\end{aligned} \tag{6.18}$$

$$\begin{aligned}
\overline{\sigma}_{\hat{P}_{f,n}} =& \int_{\mathcal{U}} \sqrt{\Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) \Phi\left(\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right)} f_{\boldsymbol{U}}(\boldsymbol{u})\mathrm{d}\boldsymbol{u} \\
=& \int_{\mathcal{U}} \sqrt{\Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) \Phi\left(\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right)} \frac{f_{\boldsymbol{U}}(\boldsymbol{u})}{f_0(\boldsymbol{u})} f_0(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u} \\
=& \int_{\mathcal{B}} \sqrt{\Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) \Phi\left(\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right)} \frac{f_{\boldsymbol{U}}(\boldsymbol{u})}{(1-\Delta) f_B(\boldsymbol{u})} (1-\Delta) f_B(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u} \\
&+ \int_{\mathcal{U}\backslash\mathcal{B}} \sqrt{\Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) \Phi\left(\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right)} \frac{f_{\boldsymbol{U}}(\boldsymbol{u})}{f_{\boldsymbol{U}}(\boldsymbol{u})} f_{\boldsymbol{U}}(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u} \\
=& V_d(r) \int_{\mathcal{B}} \sqrt{\Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) \Phi\left(\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right)} f_{\boldsymbol{U}}(\boldsymbol{u}) f_B(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u} \\
&+ \int_{\mathcal{U}\backslash\mathcal{B}} \sqrt{\Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) \Phi\left(\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right)} f_{\boldsymbol{U}}(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u},
\end{aligned} \tag{6.19}$$

where $m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})$ and $\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})$ are the posterior mean and STD functions of $\mathcal{G}$ conditional on n observations. Note that if one chooses a sufficiently small Δ (i.e., r is sufficiently large), $f_{\boldsymbol{U}}(\boldsymbol{u})$ over $\mathcal{U}\backslash\mathcal{B}$ will approach to zero. In this case, the last terms in both Eqs. (6.18) and (6.19) can be neglected, and hence $m_{\hat{P}_{f,n}}$ and $\overline{\sigma}_{\hat{P}_{f,n}}$ are approximately equal to:

$$m_{\hat{P}_{f,n}} \approx V_d(r) \int_{\mathcal{B}} \Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right) f_{\boldsymbol{U}}(\boldsymbol{u}) f_B(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u}, \tag{6.20}$$

$$\overline{\sigma}_{\hat{P}_{f,n}} \approx V_d\left(r\right) \int_{\mathcal{B}} \sqrt{\Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}\left(\boldsymbol{u}\right)}\right) \Phi\left(\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}\left(\boldsymbol{u}\right)}\right)} f_{\boldsymbol{U}}(\boldsymbol{u}) f_B\left(\boldsymbol{u}\right) \mathrm{d}\boldsymbol{u}. \tag{6.21}$$

The above two equations are the basic of the proposed IBS method, and $f_B(\boldsymbol{u})$ is regarded as the importance sampling density. The IBS estimators of Eqs. (6.20) and (6.21) are given as:

$$\tilde{m}_{\hat{P}_{f,n}} = \frac{V_d\left(r\right)}{N_{ibs}} \sum_{i=1}^{N_{ibs}} \Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\overline{\boldsymbol{u}}^{(i)})}{\sigma_{\hat{\mathcal{G}}_n}\left(\overline{\boldsymbol{u}}^{(i)}\right)}\right) f_{\boldsymbol{U}}(\overline{\boldsymbol{u}}^{(i)}), \tag{6.22}$$

$$\tilde{\overline{\sigma}}_{\hat{P}_{f,n}} = \frac{V_d\left(r\right)}{N_{ibs}} \sum_{i=1}^{N_{ibs}} \sqrt{\Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\overline{\boldsymbol{u}}^{(i)})}{\sigma_{\hat{\mathcal{G}}_n}\left(\overline{\boldsymbol{u}}^{(i)}\right)}\right) \Phi\left(\frac{m_{\hat{\mathcal{G}}_n}(\overline{\boldsymbol{u}}^{(i)})}{\sigma_{\hat{\mathcal{G}}_n}\left(\overline{\boldsymbol{u}}^{(i)}\right)}\right)} f_{\boldsymbol{U}}(\overline{\boldsymbol{u}}^{(i)}), \tag{6.23}$$

where $\overline{\boldsymbol{u}}^{(i)} \sim f_{\mathcal{B}}\left(\boldsymbol{u}\right)$. The variances of the estimators are formulated as follows:

$$\mathbb{V}\left[\tilde{m}_{\hat{P}_{f,n}}\right] = \frac{V_d^2\left(r\right)}{(N_{ibs}-1)N_{ibs}} \sum_{i=1}^{N_{ibs}} \left[\Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\overline{\boldsymbol{u}}^{(i)})}{\sigma_{\hat{\mathcal{G}}_n}\left(\overline{\boldsymbol{u}}^{(i)}\right)}\right) f_{\boldsymbol{U}}(\overline{\boldsymbol{u}}^{(i)}) - \tilde{m}_{\hat{P}_{f,n}}\right]^2, \tag{6.24}$$

$$\mathbb{V}\left[\tilde{\overline{\sigma}}_{\hat{P}_{f,n}}\right] = \frac{V_d^2\left(r\right)}{(N_{ibs}-1)N_{ibs}} \sum_{i=1}^{N_{ibs}} \left[\sqrt{\Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\overline{\boldsymbol{u}}^{(i)})}{\sigma_{\hat{\mathcal{G}}_n}\left(\overline{\boldsymbol{u}}^{(i)}\right)}\right) \Phi\left(\frac{m_{\hat{\mathcal{G}}_n}(\overline{\boldsymbol{u}}^{(i)})}{\sigma_{\hat{\mathcal{G}}_n}\left(\overline{\boldsymbol{u}}^{(i)}\right)}\right)} f_{\boldsymbol{U}}(\overline{\boldsymbol{u}}^{(i)}) - \tilde{\overline{\sigma}}_{\hat{P}_{f,n}}\right]^2. \tag{6.25}$$

Similar to the MC population in ALPI, the population generated from $f_{\mathcal{B}}\left(\boldsymbol{u}\right)$ also plays two roles. First, the posterior mean and UPSTD of the failure probability should be evaluated numerically based on those samples at each iteration, as shown in Eqs. (6.22) and (6.23). Second, it will be used as a candidate sample pool by which multiple promising points can be identified at each iteration (see next subsection).

Given dimension d, the IBS method has two parameters to be specified appropriately, i.e., radius r and simple size N_{ibs}. As we have mentioned, r should be large enough to ensure that Δ is small enough. By doing so, (1) the bias between Eqs. (6.20) and (6.18), and also Eqs. (6.21) and (6.19) can be neglected; (2) candidate samples can reach the failure domain characterized by a small probability. The probability of $f_{\boldsymbol{U}}(\boldsymbol{u})$ within the d-ball can be given as (see Appendix E of [63]):

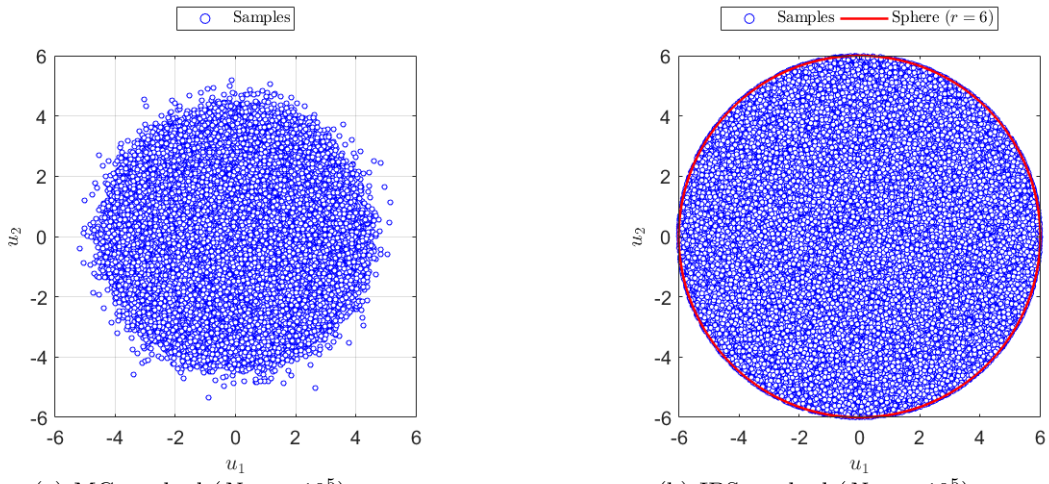
$$F(d,r) = \frac{1}{\Gamma(d/2)} \int_0^{r^2/2} x^{d/2-1} \exp(-x) \mathrm{d}x. \tag{6.26}$$

Based on this, r can be determined as the solution to:

$$F(d, r) = 1 - \Delta, \tag{6.27}$$

It should be noted that given a fixed Δ, r increases with d. For example, if we set Δ as 10^{-8}, $r \approx 6.07$ for $d = 2$, and $r \approx 6.77$ for $d = 5$. As for N_{ibs}, it cannot be too small otherwise the estimators of $m_{\hat{P}_{f,n}}$ and $\overline{\sigma}_{\hat{P}_{f,n}}$ will process relatively large variances, and also cannot fill the d-ball well. On the contrary, a too large N_{ibs} can lead to numerical difficulty and memory problems.

As an illustration, Fig. 6.1 shows two populations generated respectively by MC and IBS in two dimensions with the same sample size 10^5. Obviously, the IBS method can produce a better space-filling population and cover a larger area than that of MC method. If one would like the MC population to cover as large space as the IBS population, the sample size should be increased many times ($> 10^3$).



(a) MC method ($N_{mc} = 10^5$) (b) IBS method ($N_{ibs} = 10^5$)

Figure 6.1: Comparison between MC and IBS methods in two dimensions.

6.3.3 Multi-point UPVC criterion

In order to enable parallel processing, a batch of informative points should be identified to evaluate on the $\mathcal{G}$-function at each iteration, rather than only one single point. For this purpose, we propose a multi-point UPVC criterion, which leverages the advantages of both the UPVC function

[43] and k-means clustering [64].

Suppose that we have inferred a GP posterior $\hat{\mathcal{G}}_n \sim \mathcal{GP}(m_{\hat{\mathcal{G}}_n}(\boldsymbol{u}), k_{\hat{\mathcal{G}}_n}(\boldsymbol{u}, \boldsymbol{u}'))$ of $\mathcal{G}$ at a certain step of the proposed PABQ method. Analogous to Eq. (6.35), the corresponding UPVC function can be defined as:

$$\mathrm{UPVC}(\boldsymbol{u}) = \sqrt{\Phi\left(-\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right)\Phi\left(\frac{m_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}{\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u})}\right)} \times f_{\boldsymbol{U}}(\boldsymbol{u}), \tag{6.28}$$

where $\sigma_{\hat{\mathcal{G}}_n}(\boldsymbol{u}) = \sqrt{k_{\hat{\mathcal{G}}_n}(\boldsymbol{u}, \boldsymbol{u})}$ is the posterior STD function of $\mathcal{G}$. Note that $\overline{\sigma}^2_{\hat{P}_{f,n}} = \left[\int_{\mathcal{U}} \mathrm{UPVC}(\boldsymbol{u})\mathrm{d}\boldsymbol{u}\right]^2$ holds, and hence the UPVC function is a measure of the contribution of numerical uncertainty at the point $\boldsymbol{u}$ to the UPV of the failure probability. The traditional UPVC criterion, however, only selects the point among a MC population that has the maximum UPVC value as the best next point to evaluate on the $\mathcal{G}$-function. As such, other information provided by the UPVC function that might be still useful is discarded at each iteration. This drawback can be alleviated by identifying multiple points. The conventional k-means clustering technique can be used to partition $\overline{\boldsymbol{\mathcal{U}}}$ into k clusters, but it cannot take the UPVC measure into account. In this study, a weighted clustering algorithm is proposed by combining the UPVC function with k-means clustering, which is referred as 'UPVC-weighted k-means clustering'. Suppose that we wish to select q points among $\overline{\boldsymbol{\mathcal{U}}}$ at each iteration, and hence evaluation of the $\mathcal{G}$-function at these q points can be distributed on q processors simultaneously. The number of points q also corresponds to the number of clusters. A compact pseudocode of the proposed algorithm is given in Algorithm 4. The selected q points correspond to the q centroids produced by the proposed UPVC-weighted k-means clustering. It should be pointed out that the identified points usually do not belong to $\overline{\boldsymbol{\mathcal{U}}}$ any more due to the weighted averaging operator embedded in the proposed algorithm.

A test example is considered here to illustrate the proposed multi-point UPVC criterion. The performance function is given as $Z = \mathcal{G}(\boldsymbol{U}) = U_1^2 - U_2 + 2$, where U_1 and U_2 are two independent standard normal variables. For reproducibility, we specify the initial observed locations as $\boldsymbol{\mathcal{U}} = \left\{(-\sqrt{5}, 0), (0, 0), (\sqrt{5}, 0), (0, -\sqrt{5}), (0, \sqrt{5})\right\}$. Based on these five initial observations, we can obtain a posterior GP over the $\mathcal{G}$-function and also the UPVC function. Additional $q = 5$ points are then identified by the proposed UPVC-weighted k-means clustering algorithm from 10^5 uniform samples within the 2-ball of $r = 6$. As shown in Fig. 6.2, the newly selected points are sparsely located in areas where the UPVC values are not very small. Therefore, the total information gained from those 5 points could be more than that of the one with the maximum UPVC value.

Algorithm 4 UPVC-weighted k-means clustering algorithm

Input: the number of clusters q, UPVC($\boldsymbol{u}$) and $\overline{\mathscr{U}}$

1. **Initialization**. Randomly select q points among $\overline{\mathscr{U}}$ as the initial centroids , denoted by $\boldsymbol{E}^{(1)} = \left\{\boldsymbol{e}^{(i)}\right\}_{i=1}^{q}$;

2. **Assignment step**. Each point among $\overline{\mathscr{U}}$ is assigned to a cluster for which the squared Euclidean distance between the point and the cluster centroid is shortest. The i-th cluster is denoted as $\boldsymbol{C}^{(i)} = \left\{\boldsymbol{c}_j^{(i)}\right\}_{j=1}^{N_i}$, where $\boldsymbol{c}_j^{(i)}$ is the j-th point in the i-th cluster ($j = 1, 2, \cdots, N_i$);

3. **Update step**. Each centroid is then updated by UPVC-weighted mean of the cluster:

$$\boldsymbol{e}^{(i)} = \frac{\sum_{j=1}^{N_i} \mathrm{UPVC}(\boldsymbol{c}_j^{(i)}) \times \boldsymbol{c}_j^{(i)}}{\sum_{j=1}^{N_i} \mathrm{UPVC}(\boldsymbol{c}_j^{(i)})}$$

4. **Iteration**. Repeat the assignment step and update step until the centroids do not change or the predefined number of iterations is reached.

Output: q centroids

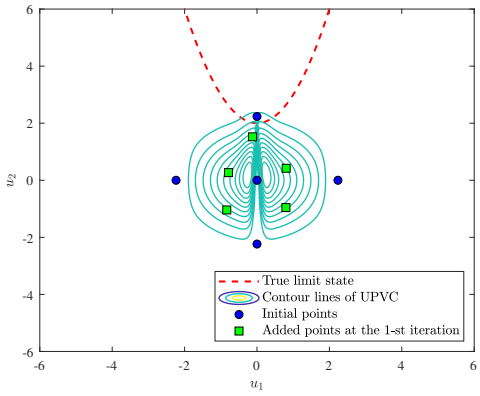


Figure 6.2: Illustration of the proposed multi-points UPVC criterion by a test example.

6.3.4 Summary of the proposed method

The numerical implementation procedure of the proposed PABQ method for reliability analysis, which is also shown in Fig. 6.3, consists of the following main steps:

Step B.1: Generate uniformly distributed samples within the d-ball

Generate N_{ibs} uniform samples within the d-ball of radius r, using Algorithm 3, denoted as $\overline{\mathscr{U}} = \left\{\overline{\boldsymbol{u}}^{(i)}\right\}_{i=1}^{N_{ibs}}$.

Step B.2: Get initial observations

Randomly select N_0 samples among $\overline{\mathscr{U}}$, denoted by $\mathscr{U} = \left\{\boldsymbol{u}^{(i)}\right\}_{i=1}^{N_0}$. These samples are eval-

uated on the $\mathcal{G}$-function in parallel to obtain the corresponding observations $\mathscr{Z} = \left\{z^{(i)}\right\}_{i=1}^{N_0}$ ($z^{(i)} = \mathcal{G}(\boldsymbol{u}^{(i)})$). The initial dataset is constructed by $\mathscr{D} = \{\mathscr{U}, \mathscr{Z}\}$. Let $n = N_0$.

Step B.3: Make Bayesian inference about the failure probability

By assigning a GP prior for the $\mathcal{G}$-function, we finally arrive at the posterior mean and UPSTD of the failure probability conditional on $\mathscr{D}$. In this study, the prior mean and covariance of $\hat{\mathcal{G}}_0 \sim \mathcal{GP}(m_{\hat{\mathcal{G}}_0}(\boldsymbol{x}), k_{\hat{\mathcal{G}}_0}(\boldsymbol{x}, \boldsymbol{x}'))$ are assumed to be a constant and the squared exponential kernel respectively. The involved hyper-parameters are tuned by using the maximum likelihood estimation, and this stage is implemented with the *fitrgp* function in Statistics and Machine Learning Toolbox of Matlab. The posterior mean and UPSTD of the failure probability are then evaluated based on Eqs. (6.22) and (6.23).

Step B.4: Check the stopping criterion

If $\frac{\tilde{\bar{\sigma}}_{\hat{P}_{f,n}}}{\tilde{m}_{\hat{P}_{f,n}}} < \epsilon$ is satisfied, go to **Step B.6**; Else, go to **Step B.5**. Here $\frac{\tilde{\bar{\sigma}}_{\hat{P}_{f,n}}}{\tilde{m}_{\hat{P}_{f,n}}}$ denotes the estimated upper-bound of the posterior COV of the failure probability, and ϵ is a user-specified threshold.

Step B.5: Enrich the previous dataset

Identify additional q points by using the proposed multi-point UPVC criterion (see Algorithm 4), denoted by $\mathscr{U}_+ = \left\{\boldsymbol{u}_+^{(i)}\right\}_{i=1}^{q}$. Then, the corresponding observations of the $\mathcal{G}$-function at those q identified points $\mathscr{U}_+$ should be obtained using parallel computing, denoted by $\mathscr{Z}_+ = \left\{z_+^{(i)}\right\}_{i=1}^{q}$ with $z_+^{(i)} = \mathcal{G}(\boldsymbol{u}_+^{(i)})$. The previous dataset $\mathscr{D}$ is enriched with $\mathscr{D}_+ = \{\mathscr{U}_+, \mathscr{Z}_+\}$, i.e., $\mathscr{D} = \mathscr{D} \cup \mathscr{D}_+$. Let $n = n + q$, and go to **Step B.3**.

Step B.6: End the algorithm

Return $\tilde{m}_{\hat{P}_{f,n}}$ as the estimated failure probability and end the algorithm.

For practical implementation, it is necessary to set proper values for constants N_{ibs}, r, N_0, ϵ and q. The selection of these parameters is problem-dependent. However, according to our experience some general guidelines for selecting them are the following: $N_{ibs} = 5 \times 10^5 \sim 1 \times 10^6$, $r = 6$, $N_0 = 10$, $\epsilon = 5\% \sim 10\%$ and $q =$ the number of available processors for parallel computing.

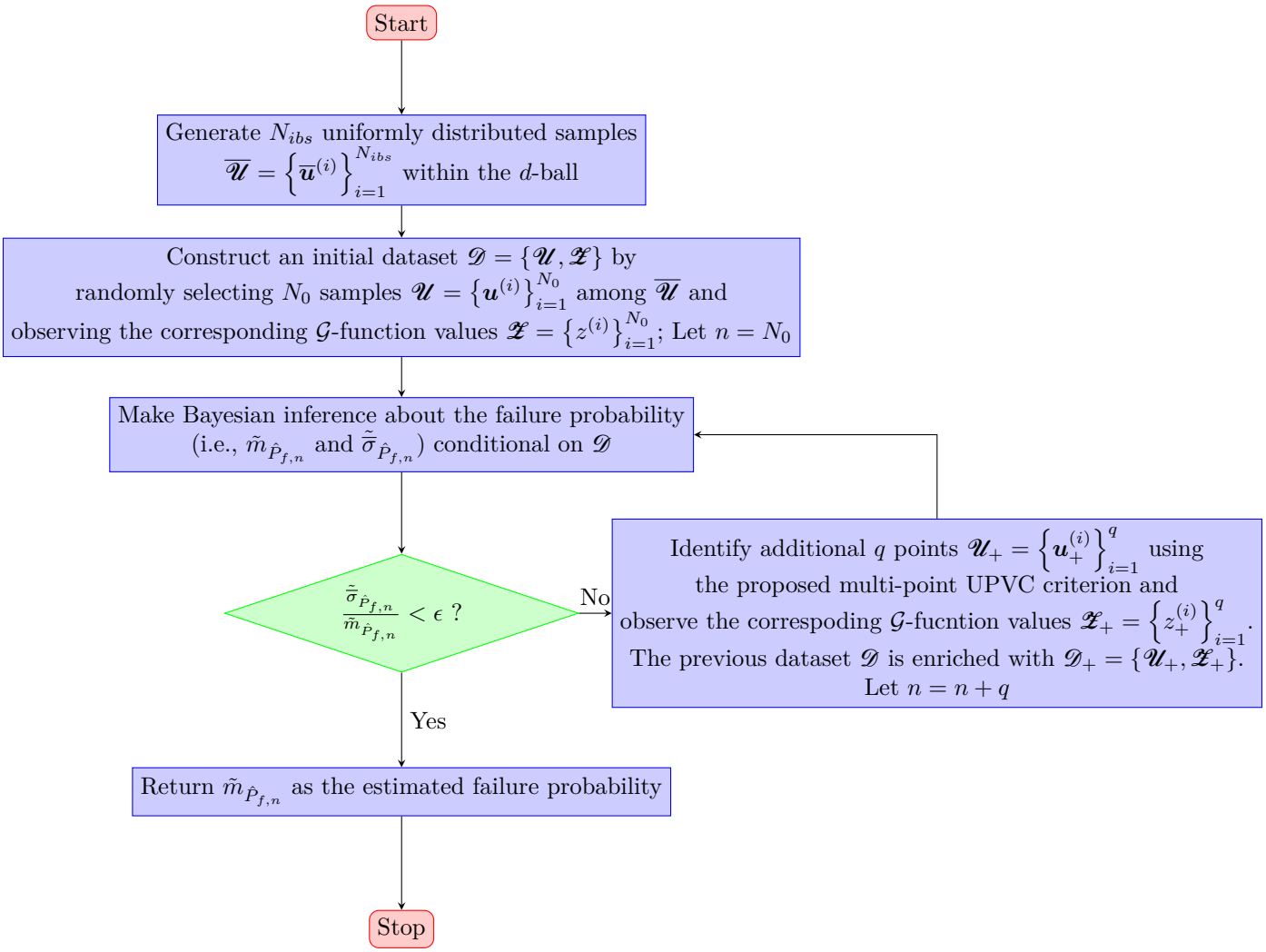


Figure 6.3: Flowchart of the proposed PABQ method.

6.4 Numerical examples

The performance of the proposed PABQ method is investigated by means of four numerical examples with varying complexity in this section. Several different parameter settings of PABQ are experimented in each example to study their effect on the results. For comparison, several state-of-the-art methods, i.e., FORM [65], SORM [65], AK-MCS [37], ALPI [43], AK-MCMC [46] and Polynomial-Chaos Kriging (PC-Kriging) [66], are also implemented when applicable.

6.4.1 Example 1: A series system with four branches

The first numerical example consists of a series system with four branches, which has been a classical benchmark example in structural reliability analysis (see, e.g., [37, 67, 68]). The performance function is given by:

$$Y = g(X_1, X_2) = \min \begin{cases} a + \frac{(X_1 - X_2)^2}{10} - \frac{(X_1 + X_2)}{\sqrt{2}}; \\ a + \frac{(X_1 - X_2)^2}{10} + \frac{(X_1 + X_2)}{\sqrt{2}}; \\ (X_1 - X_2) + \frac{b}{\sqrt{2}}; \\ (X_2 - X_1) + \frac{b}{\sqrt{2}} \end{cases}, \tag{6.29}$$

where X_1 and X_2 are two i.i.d. standard normal variables; a and b are two constant parameters. In this example, two cases by varying those two constant parameters are considered, where $a = 3$, $b = 6$ for the first case, and $a = 5$, $b = 10$ for the second case.

6.4.1.1 Results of Case I

In this case, the proposed PABQ method is compared to several other methods, i.e., AK-MCS+U [37], ALPI [43] and PC-Kriging [66]. Table 6.1 summarizes the results given as the number of iterations N_{iter}, the total number of performance function calls N_{call}, the estimated failure probability $\hat{P}_f$, and the COV of $\hat{P}_f$ (i.e., COV$[\hat{P}_f]$). As seen, the proposed method with different q only takes a very few iterations in average to converge, which are less than that of PC-Kriging, and far less than that of AK-MCS+U and ALPI. This indicates that the proposed method could offer significant time savings when parallel computing is available. Furthermore, the computational advantage may still exist even in case of non-parallel computing since the average number of performance function calls is also reduced a lot by using the proposed method, especially when q is small (e.g., $q = 6$). The results of $\hat{P}_f$ and COV$[\hat{P}_f]$ also imply that the proposed method has an accuracy similar to other methods being compared. By increasing N_{ibs} from 5×10^5 to 1×10^6 and decreasing ϵ from 10% to 8%, the PABQ method can slightly reduce the COVs of failure probability estimates, at the cost of marginally increased computation in an average sense.

To illustrate the proposed method visually, Fig. 6.4(a) depicts the points selected at two stages of an exemplary run, as well as the true limit state curve. It is shown that most of the added points are sparsely located, and some of them are close to the four important parts of the limit state curve

that are crucial for accurate failure probability estimation. These results indicate the effectiveness of the proposed multi-point selection strategy.

Table 6.1: Reliability results for Example 1 (Case I).

Method		N_{iter}	N_{call}	$\hat{P}_f$	COV$[\hat{P}_f]$/%
MCS		-	10^8	4.46×10^{-3}	0.15
AK-MCS+U		$1 + 96.55 = 97.55$	$12 + 96.55 = 108.55$	4.44×10^{-3}	1.54
ALPI		$1 + 72.95 = 73.95$	$12 + 72.95 = 84.95$	4.44×10^{-3}	1.79
PC-Kriging [66]	$q = 6$	$1 + 14.40 = 15.40$	$12 + 86.40 = 98.40$	4.46×10^{-3}	1.50
	$q = 6$	$1 + 5.60 = 6.60$	$10 + 33.60 = 43.60$	4.44×10^{-3}	2.53
Proposed PABQ	$q = 10$	$1 + 4.20 = 5.20$	$10 + 42.00 = 52.00$	4.40×10^{-3}	2.22
($N_{ibs} = 5 \times 10^5, \epsilon = 10\%$)	$q = 15$	$1 + 3.65 = 4.65$	$10 + 54.75 = 64.75$	4.44×10^{-3}	1.35
	$q = 20$	$1 + 3.05 = 4.05$	$10 + 61.00 = 71.00$	4.44×10^{-3}	1.29
	$q = 6$	$1 + 8.64 = 9.64$	$10 + 43.20 = 53.20$	4.43×10^{-3}	2.17
Proposed PABQ	$q = 10$	$1 + 4.55 = 5.55$	$10 + 45.50 = 55.50$	4.40×10^{-3}	1.25
($N_{ibs} = 1 \times 10^6, \epsilon = 8\%$)	$q = 15$	$1 + 3.70 = 4.70$	$10 + 55.50 = 65.50$	4.43×10^{-3}	1.02
	$q = 20$	$1 + 3.45 = 4.45$	$10 + 69.00 = 79.00$	4.45×10^{-3}	0.91

Note: For AK-MCS+U and ALPI, the MC population size is set as 10^6. AK-MCS+U, ALPI and PABQ are performed 20 independent runs. PC-Kriging was performed 50 independent runs. Thus, for those methods, average results are reported for N_{iter}, N_{call}, and $\hat{P}_f$. Besides, COV$[\hat{P}_f]$ is also approximated accordingly.

6.4.1.2 Results of Case II

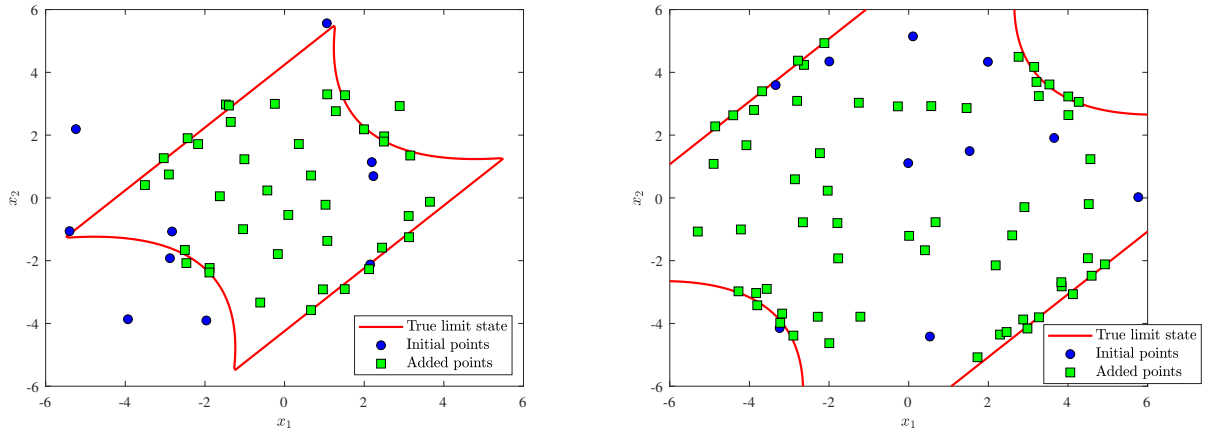
The failure probability is quite small (in the order of 10^{-7}) in Case II, and hence some methods, like AK-MCS and ALPI, are not applicable anymore. For this reason, the proposed method is mainly compared with AK-MCMC [46], which is capable of assessing extremely small failure probabilities. As can be seen from Table 6.2, the proposed method can not only reduce the average number of iterations greatly (especially when q is large, e.g., $q = 20$), but also the total number of calls to the performance function (especially when q is small, e.g., $q = 5$), in comparison to AK-MCMC. Besides, the proposed PABQ method is also able to yield fairly good average results for the failure probability. It is noted that the COVs of the failure probability estimates can be reduced by a more strict parameter setting (i.e., $N_{ibs} = 1 \times 10^6, \epsilon = 8\%$). This case study demonstrates the efficiency and accuracy of the proposed method for such a case with an extremely rare failure event.

Fig. 6.4(b) depicts the points selected at two stages of the proposed method ($q = 10$) via an exemplary run, along with the real limit state curve. It is encouraging to see that the added points are relatively sparsely distributed, and most of them are located in the vicinity of true limit state curve.

Table 6.2: Reliability results for Example 1 (Case II).

Method		N_{iter}	N_{call}	$\hat{P}_f$	COV$[\hat{P}_f]$/%
MCS		-	10^9	8.84×10^{-7}	3.36
AK-MCMC		$1 + 134.00 = 135.00$	$12 + 134.00 = 146.00$	8.85×10^{-7}	1.62
	$q = 5$	$1 + 8.80 = 9.80$	$10 + 44.00 = 54.00$	8.82×10^{-7}	2.14
Proposed PABQ	$q = 10$	$1 + 5.45 = 6.45$	$10 + 54.50 = 64.50$	8.84×10^{-7}	2.06
($N_{ibs} = 5 \times 10^5, \epsilon = 10\%$)	$q = 15$	$1 + 4.75 = 5.75$	$10 + 71.25 = 81.25$	8.83×10^{-7}	1.24
	$q = 20$	$1 + 4.40 = 5.40$	$10 + 88.00 = 98.00$	8.88×10^{-7}	1.24
	$q = 5$	$1 + 8.80 = 9.80$	$10 + 44.00 = 54.00$	8.80×10^{-7}	1.63
Proposed PABQ	$q = 10$	$1 + 5.95 = 6.95$	$10 + 59.50 = 69.50$	8.83×10^{-7}	0.89
($N_{ibs} = 1 \times 10^6, \epsilon = 8\%$)	$q = 15$	$1 + 4.95 = 5.95$	$10 + 74.25 = 84.25$	8.86×10^{-7}	0.89
	$q = 20$	$1 + 4.80 = 5.80$	$10 + 96.00 = 106.00$	8.86×10^{-7}	0.66

Note: AK-MCMC and PABQ are performed 20 independent runs. Thus, for those methods, average results are reported for N_{iter}, N_{call}, and $\hat{P}_f$. Besides, COV$[\hat{P}_f]$ is also approximated accordingly.



(a) Case I ($q = 10$) (b) Case II ($q = 10$)

Figure 6.4: Points selected at different stages by the proposed PABQ method for Example 2.

6.4.2 Example 2: A nonlinear oscillator

A nonlinear undamped single-degree-of-freedom (SDOF) oscillator subjected to a rectangular pulse load [43] is adopted as the second example, as shown in Fig. 6.5. The performance function is defined as:

$$Y = g\left(m, c_1, c_2, r, F_1, t_1\right) = 3r - \left|\frac{2F_1}{c_1 + c_2} \sin\left(\frac{t_1}{2}\sqrt{\frac{c_1 + c_2}{m}}\right)\right|, \tag{6.30}$$

where m, c_1, c_2, r, F_1, t_1 are six random variables, as described in Table 6.3.

The reference value of the failure probability is 5.17×10^{-6} (with COV being small, i.e., 1.39%), provided by MCS with 10^9 samples. As the failure probability is quite small, the proposed method

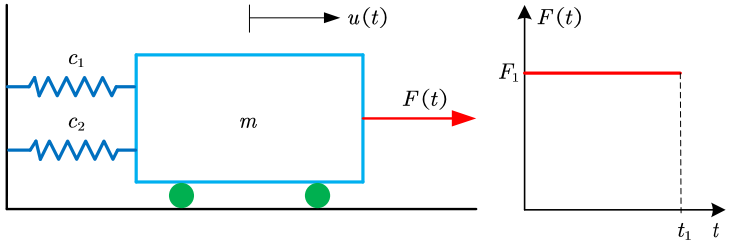


Figure 6.5: A nonlinear undamped SDOF oscillator subjected to pulse load.

Table 6.3: Random variables for Example 2.

Variable	Distribution	Mean	STD
m	Normal	1.0	0.05
c_1	Lognormal	1.0	0.10
c_2	Lognormal	0.1	0.01
r	Normal	0.5	0.05
F_1	Lognormal	0.5	0.10
t_1	Normal	1.0	0.20

is only compared to FORM [65], SORM [65] and AK-MCMC [46]. As can be seen from Table 6.4, the required average number of iterations by the proposed method is less than all the methods being compared, especially for AK-MCMC. This implies the parallel computing advantage of the proposed method. Besides, the proposed method is still more advantageous than FORM, SORM and AK-MCMC in computational efficiency in case of non-parallel computing, since the average number of performance function evaluations can also be reduced a lot (especially when q is small, e.g., $q = 5$). Although $\mathrm{COV}[\hat{P}_f]$ given by the proposed method ($N_{ibs} = 5 \times 10^5, \epsilon = 10\%$) is around 5%, it can still be acceptable in practical applications. If one would like to reduce $\mathrm{COV}[\hat{P}_f]$, one can increase N_{ibs} and decrease ϵ. For example, the last four rows of Table 6.4 give the results by of PABQ ($N_{ibs} = 1 \times 10^6, \epsilon = 5\%$). It can be seen that $\mathrm{COV}[\hat{P}_f]$ is reduced to about 3% at the expense of increased N_{iter} and N_{call} in some cases ($q = 5, 10, 20$), while still much less than those of FORM, SORM and AK-MCMC.

6.4.3 Example 3: A simple bracket model

A simple bracket model that is available in the Partial Differential Equation Toolbox of Matlab is considered as the third example. The schematic diagram of the bracket is shown in Figs. 6.6(a) and 6.6(b), and more details of the model can be found in the description in the toolbox. The bracket

Table 6.4: Reliability results for Example 2.

Method		N_{iter}	N_{call}	$\hat{P}_f$	COV$[\hat{P}_f]$/%
MCS		-	10^9	5.17×10^{-6}	1.39
FORM		10	80	5.45×10^{-6}	-
SORM		10	160	5.25×10^{-6}	-
AK-MCMC		$1 + 109.20 = 110.20$	$12 + 109.20 = 121.20$	5.23×10^{-6}	0.69
	$q = 5$	$1 + 3.15 = 4.15$	$10 + 15.75 = 25.75$	5.19×10^{-6}	5.68
Proposed PABQ	$q = 10$	$1 + 2.05 = 3.05$	$10 + 20.50 = 30.50$	5.21×10^{-6}	4.30
$(N_{ibs} = 5 \times 10^5, \epsilon = 10\%)$	$q = 15$	$1 + 1.65 = 2.65$	$10 + 24.75 = 34.75$	5.17×10^{-6}	4.30
	$q = 20$	$1 + 1.70 = 2.70$	$10 + 34.00 = 44.00$	5.21×10^{-6}	4.79
	$q = 5$	$1 + 4.05 = 5.05$	$10 + 20.25 = 30.25$	5.15×10^{-6}	3.08
Proposed PABQ	$q = 10$	$1 + 2.40 = 3.40$	$10 + 24.00 = 34.00$	5.15×10^{-6}	2.41
$(N_{ibs} = 1 \times 10^6, \epsilon = 5\%)$	$q = 15$	$1 + 2.00 = 3.00$	$10 + 30.00 = 40.00$	5.15×10^{-6}	3.53
	$q = 20$	$1 + 1.95 = 2.95$	$10 + 39.00 = 49.00$	5.20×10^{-6}	3.44

Note: AK-MCMC and PABQ are performed 20 independent runs. Thus, for those methods, average results are reported for N_{iter}, N_{call}, and $\hat{P}_f$. Besides, COV$[\hat{P}_f]$ is also approximated accordingly.

is fixed at the back face (face 4) and subjected to a distributed load in the negative z-direction in the front face (face 8). It is assumed that the Young's modulus E, Poisson's ratio μ, distributed load q and thickness h of the horizontal plate with hole are characterized as independent random variables, whose statistical information is summarized in Table. 6.5. The 10-node tetrahedral element is used to discretize the model, as shown in Figs. 6.6(c) and 6.6(d). The maximal deflection of the bracket in the z direction is of concern in this example. The limit state function is defined as:

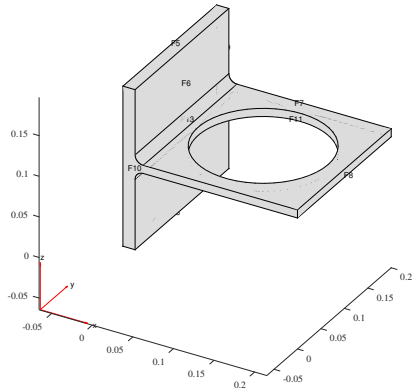
$$Y = G(E, \mu, q, h) = \Delta - \bar{V}(E, \mu, q, h), \tag{6.31}$$

where Δ is the deterministic threshold, which is specified as $\Delta = 140$ μm; $\bar{V}$ denotes the maximum displacement of the bracket in the z-direction.

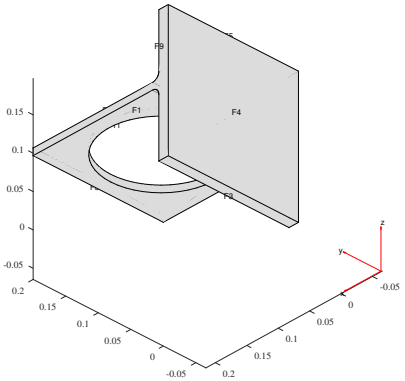
Table 6.5: Random variables for Example 3.

Variable	Distribution	Mean	COV
E (Gpa)	Lognormal	200	0.15
μ	Uniform	0.3	0.10
q (Pa)	Lognormal	10^4	0.20
h (mm)	Lognormal	10	0.10

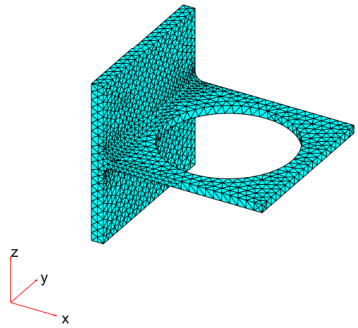
We implement several methods to assess the failure probability corresponding to the limit state function defined in Eq. (6.31). The results are reported in Table 6.6. FORM does not converge within 100 iterations, so its results are not included. The reference value of the failure probability is



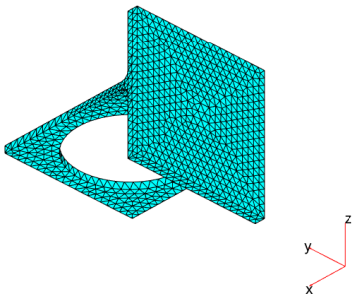
(a) Bracket with face labels (side view)



(b) Bracket with face labels (rear view)



(c) Finite-element mesh (side view)



(d) Finite-element mesh (rear view)

Figure 6.6: A simple bracket model: Geometry and finite-element mesh.

taken as the average result of AK-MCMC, i.e., 1.90×10^{-6} (with a COV of 1.15%). It can be seen from Table 6.6 that the proposed PABQ method can significantly reduce the number of iterations N_{iter} compared to AK-MCMC, while maintaining reasonable accuracy. This indicates that our method could greatly outperform AK-MCMC in terms of computational efficiency when parallel computing is available. One can also notice that the proposed method requires less performance function calls in average than AK-MCMC. Therefore, the proposed method could be still more efficient than AK-MCMC in case that parallel computing is unavailable. The variability of the failure probability estimate given by the proposed method can be reduced to a certain level by setting a large N_{ibs} and a small ϵ.

Table 6.6: Reliability results for Example 3.

Method		N_{iter}	N_{call}	$\hat{P}_f$	COV$[\hat{P}_f]$/%
FORM		-	-	-	-
AK-MCMC		$1 + 44.60 = 45.60$	$12 + 44.60 = 56.60$	1.90×10^{-6}	1.15
Proposed PABQ ($N_{ibs} = 5 \times 10^5, \epsilon = 10\%$)	$q = 2$	$1 + 1.85 = 2.85$	$10 + 3.70 = 13.70$	1.93×10^{-6}	4.99
	$q = 3$	$1 + 1.45 = 2.45$	$10 + 4.35 = 14.35$	1.88×10^{-6}	6.19
	$q = 4$	$1 + 1.40 = 2.40$	$10 + 5.60 = 15.60$	1.93×10^{-6}	5.89
	$q = 5$	$1 + 1.35 = 2.35$	$10 + 6.75 = 16.75$	1.91×10^{-6}	8.99
Proposed PABQ ($N_{ibs} = 1 \times 10^6, \epsilon = 5\%$)	$q = 2$	$1 + 2.45 = 3.45$	$10 + 4.90 = 14.90$	1.91×10^{-6}	3.74
	$q = 3$	$1 + 1.95 = 2.95$	$10 + 5.85 = 15.85$	1.90×10^{-6}	2.19
	$q = 4$	$1 + 1.65 = 2.65$	$10 + 6.60 = 16.60$	1.89×10^{-6}	3.70
	$q = 5$	$1 + 1.55 = 2.55$	$10 + 7.75 = 17.75$	1.93×10^{-6}	3.16

Note: AK-MCMC and PABQ are performed 20 independent runs. Thus, for those methods, average results are reported for N_{iter}, N_{call}, and $\hat{P}_f$. Besides, COV$[\hat{P}_f]$ is also approximated accordingly.

6.4.4 Example 4: A 120-bar space truss structure

A 120-bar space truss structure [43], as shown in Fig. 6.7, is investigated in the last example to further demonstrate the proposed method. The structure is modelled as a three-dimensional (3D) finite-element model with 49 nodes and 120 elements in OpenSees. Nodes 0, 1, 4, 7 and 10 withstand concentrated loads along the negative z-axis, denoted as P_0, P_1, P_4, P_7 and P_{10} respectively. All elements are assumed to have the same cross-sectional area A and Young's modulus E. The structure is expected to be in a linear elastic state, so we simply employ linear finite element analysis. The performance function is defined as:

$$Y = g(P_0, P_1, P_4, P_7, A, E) = \Delta - V_{0,z}, \tag{6.32}$$

where $V_{0,z}$ denotes the vertical displacement of node 0; and Δ is the threshold, specified as 90 mm. The random variables considered in this examples are summarized in Table 6.7.

Table 6.7: Random variables for Example 4.

Variable	Distribution	Mean	COV
P_0	Lognormal	500 kN	0.20
P_1	Lognormal	200 kN	0.20
P_4	Lognormal	200 kN	0.20
P_7	Lognormal	200 kN	0.20
P_{10}	Lognormal	200 kN	0.20
A	Normal	2000 mm^2	0.15
E	Normal	2.00×10^5 MPa	0.15

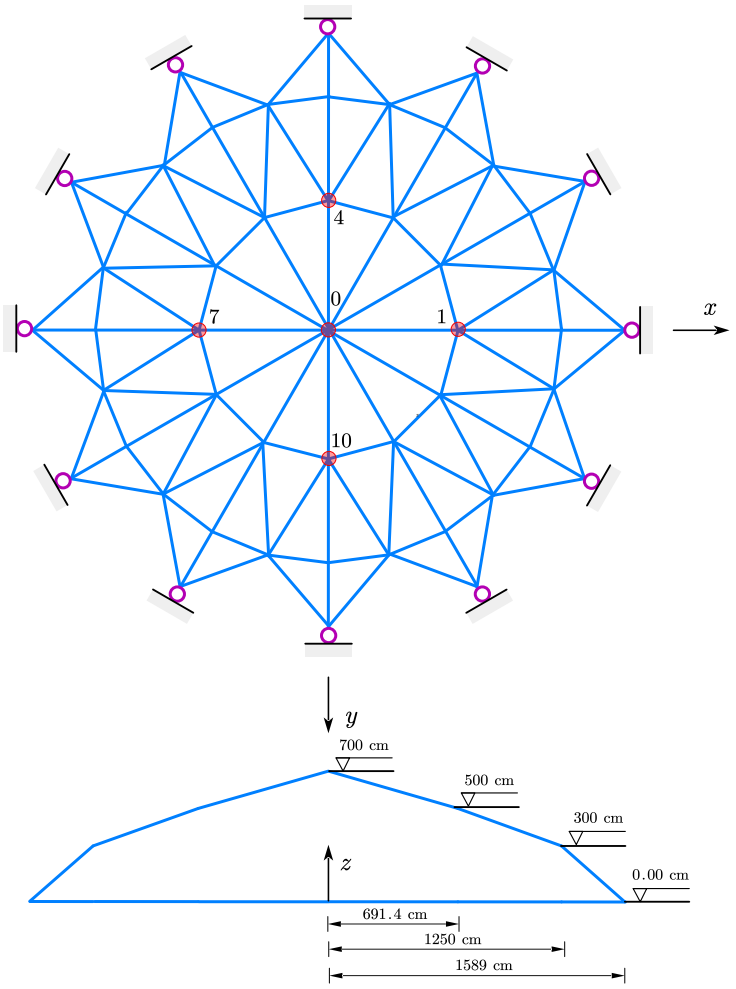


Figure 6.7: A 120-bar space truss structure.

In this example, several methods, i.e., MCS, FORM [65], SORM [65], AK-MCS+U [37], ALPI [43], AK-MCMC [46] and PABQ, are implemented to assess the failure probability. The results are listed in Table 6.8. The reference value for the failure probability is 5.08×10^{-4} with COV being 4.44, provied by MCS with 10^6 samples. The results of AK-MCMC are not reported because it fails to converge in multiple trials. FORM only requires 7 iterations and a total number of 65 performance function calls, which, however, results in an inaccurate result. SORM can provide more accurate failure probability estimate than FORM at the expense of 172 calls to the performance function (hence the finite-element model). Compared to AK-MCS+U and ALPI, the proposed PABQ method performs better in terms of N_{call} (especially when q is small, e.g., $q = 5$), and

much better in terms of N_{iter} (especially when q is large, e.g., $q = 20$). This implies that PABQ can be much more efficient than AK-MCS+U and ALPI in cases of both parallel and non-parallel computing. Besides, the proposed method still has a acceptable accuracy, as indicated by $\hat{P}_f$ and COV$[\hat{P}_f]$. As shown in the last four rows of Table 6.8, COV$[\hat{P}_f]$ can be further reduced by increasing N_{ibs} and decreasing ϵ at the cost of slightly increased N_{iter} and N_{call}.

Table 6.8: Reliability results for Example 4.

Method		N_{iter}	N_{call}	$\hat{P}_f$	COV$[\hat{P}_f]$/%
MCS		-	10^6	5.08×10^{-4}	4.44
FORM		7	65	3.16×10^{-4}	-
SORM		7	172	5.23×10^{-4}	-
AK-MCS+U		$1 + 60.75 = 61.75$	$12 + 60.75 = 72.75$	5.16×10^{-4}	4.84
ALPI		$1 + 47.45 = 48.45$	$12 + 47.45 = 59.45$	5.10×10^{-4}	3.54
AK-MCMC		-	-	-	-
	$q = 5$	$1 + 5.90 = 6.90$	$10 + 29.50 = 39.50$	4.93×10^{-4}	4.74
Proposed PABQ	$q = 10$	$1 + 3.80 = 4.80$	$10 + 38.00 = 48.00$	4.98×10^{-4}	3.31
($N_{ibs} = 5 \times 10^5, \epsilon = 10\%$)	$q = 15$	$1 + 2.65 = 3.65$	$10 + 39.75 = 49.75$	4.99×10^{-4}	4.68
	$q = 20$	$1 + 2.40 = 3.40$	$10 + 48.00 = 58.00$	4.98×10^{-4}	6.22
	$q = 5$	$1 + 8.65 = 9.65$	$10 + 43.25 = 53.25$	5.04×10^{-4}	3.41
Proposed PABQ	$q = 10$	$1 + 4.80 = 5.80$	$10 + 48.00 = 58.00$	5.06×10^{-4}	2.40
($N_{ibs} = 1 \times 10^6, \epsilon = 5\%$)	$q = 15$	$1 + 3.70 = 4.70$	$10 + 55.50 = 65.50$	5.07×10^{-4}	2.15
	$q = 20$	$1 + 2.90 = 3.90$	$10 + 58.00 = 68.00$	5.02×10^{-4}	4.27

Note: For AK-MCS+U and ALPI, the MC population size is set as 10^6. AK-MCS+U, ALPI and PABQ are performed 20 independent runs. Thus, for those methods, average results are reported for N_{iter}, N_{call}, and $\hat{P}_f$. Besides, COV$[\hat{P}_f]$ is also approximated accordingly.

6.5 Conclusions

This paper presents a 'Parallel Adaptive Bayesian Quadrature' (PABQ) method for rare failure event estimation. As it is rooted in ALPI, PABQ offers an alternative framework to the quantification, propagation and reduction of numerical uncertainty for assessing failure probabilities. Besides, compared to ALPI, two important improvements are made in PABQ to enable the use of ever-increasing parallel computing facilities and enhance the capability of assessing small failure probabilities. The parallelism of PABQ is achieved by developing a multi-point selection strategy, while the capableness for rare failure event estimation is realized by proposing an importance ball sampling technique. The performance of the proposed method is illustrated by means of four numerical examples. In most studied cases, it is found that PABQ can not only significantly reduce the average number of iterations (especially when q is large), but also lower the average total num-

ber of performance function calls (especially when q is small) compared to several selected existing methods. This indicates the computational efficiency advantage of PABQ in both parallel and non-parallel computing. In addition, PABQ is able to produce accurate estimates for small failure probabilities (e.g., in the order of 10^{-7}).

The proposed method, in its current form, is not applicable to high-dimensional and/or strongly non-linear problems. The former, one one hand, is due to the challenges of implementing GP models in high dimensions. On the other hand, IBS should not lead to significant improvement for a high-dimensional case. The latter is caused by the fact that the GP model is typically suitable for modelling smooth or moderately nonlinear functions. These drawbacks will be addressed in future work.

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). Marcos Valdebenito acknowledges the support by ANID (National Agency for Research and Development, Chile) under its program FONDECYT, grant number 1180271. Chao Dang, Pengfei Wei and Michael Beer also would like to appreciate the support of Sino-German Mobility Program under grant number M-0175.

Appendix A: Numerical algorithm of the active learning probabilistic integration

The procedure for numerical implementation of the ALPI method includes the following steps:

Step A.1: Generate a Monte Carlo population

Generate a MC population comprising N_{mc} samples according to $f_{\boldsymbol{X}}(\boldsymbol{x})$, denoted by $\overline{\mathscr{X}} = \left\{\overline{\boldsymbol{x}}^{(i)}\right\}_{i=1}^{N_{mc}}$. This population has two functions: (1) It serves as a candidate sample pool among which the next best point is identified to evaluate on the g-function; and (2) It is used to evaluate the posterior mean and UPV of the failure probability (Eqs. (6.13) and (6.15)).

Step A.2: Get initial observations

Randomly select N_0 (e.g., 12) samples among $\overline{\mathscr{X}}$, denoted by $\mathscr{X}$. Those points are then evaluated on the g-function to get N_0 observations, denoted by $\mathscr{Y}$. As such, an initial dataset can be constructed, i.e., $\mathscr{D} = \{\mathscr{X}, \mathscr{Y}\}$. Let $n = N_0$.

Step A.3: Infer the posterior failure probability

The prior mean and variance functions of $\hat{g}_0 \sim \mathcal{GP}(m_{\hat{g}_0}(\boldsymbol{x}), k_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{x}'))$ are assumed to be a constant and the squared exponential kernel in this study, respectively. Based on $\mathscr{D}$, a posterior GP $\hat{g}_n \sim \mathcal{GP}(m_{\hat{g}_n}(\boldsymbol{x}), k_{\hat{g}_n}(\boldsymbol{x}, \boldsymbol{x}'))$ for the g-function can be obtained. This step mainly consists of tuning the hyper-parameters via maximum likelihood estimation. For convenience, one can use the *fitrgp* function in Statistics and Machine Learning Toolbox of Matlab. Afterwards, the posterior mean of failure probability can be estimated by:

$$\tilde{m}_{\hat{P}_{f,n}} = \frac{1}{N_{mc}} \sum_{i=1}^{N_{mc}} \boldsymbol{\Phi}\left(-\frac{m_{\hat{g}_n}(\overline{\boldsymbol{x}}^{(i)})}{\sigma_{\hat{g}_n}\left(\overline{\boldsymbol{x}}^{(i)}\right)}\right), \tag{6.33}$$

and the upper-bound of posterior standard deviation (UPSTD):

$$\tilde{\bar{\sigma}}_{\hat{P}_{f,n}} = \frac{1}{N_{mc}} \sum_{i=1}^{N_{mc}} \sqrt{\boldsymbol{\Phi}\left(-\frac{m_{\hat{g}_n}(\overline{\boldsymbol{x}}^{(i)})}{\sigma_{\hat{g}_n}\left(\overline{\boldsymbol{x}}^{(i)}\right)}\right) \boldsymbol{\Phi}\left(\frac{m_{\hat{g}_n}(\overline{\boldsymbol{x}}^{(i)})}{\sigma_{\hat{g}_n}\left(\overline{\boldsymbol{x}}^{(i)}\right)}\right)}. \tag{6.34}$$

Step A.4: Check the stopping criterion

Only if the posterior failure probability processes a sufficiently low level of epistemic uncertainty, its mean can be used to predict the failure probability. To this end, we propose to examine the estimated upper bound of posterior COV of the failure probability as described next. If $\frac{\tilde{\bar{\sigma}}_{\hat{P}_{f,n}}}{\tilde{m}_{\hat{P}_{f,n}}} < \epsilon$ is satisfied, go to **Step A.6**; Else, go to **Step A.5**. Here ϵ is a user-specified threshold, which takes the value of 0.02 in all numerical examples.

Step A.5: Enrich the previous dataset

At this stage, the best next point to evaluate on the g-function should be identified by a learning

function. By exploring the structure of UPV of failure probability (Eq. (6.15)), the so-called upper-bound posterior variance contribution (UPVC) function is introduced [43]:

$$\mathrm{UPVC}(\boldsymbol{x}) = \sqrt{\Phi\left(-\frac{m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right)\Phi\left(\frac{m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right)} \times f_{\boldsymbol{X}}(\boldsymbol{x}), \tag{6.35}$$

where $\overline{\sigma}^2_{\hat{P}_{f,n}} = \left[\int_{\mathcal{X}} \mathrm{UPVC}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right]^2$ holds. The best next point $\boldsymbol{x}^\star$ is selected by:

$$\boldsymbol{x}^\star = \arg\max_{\overline{\boldsymbol{x}}\in\overline{\mathcal{X}}} \mathrm{UPVC}(\boldsymbol{x}). \tag{6.36}$$

The g-function is then evaluated at the point $\boldsymbol{x}^\star$, i.e., $y^\star = g(\boldsymbol{x}^\star)$. The dataset $\mathscr{D}$ is enriched by $\mathscr{D} = \mathscr{D} \cup (\boldsymbol{x}^\star, y^\star)$. Let $n = n + 1$, and go to **Step A.3**.

Step A.6: End the algorithm

Return $\tilde{m}_{\hat{P}_{f,n}}$ as the estimated failure probability and end the algorithm.

Bibliography

[1] Siu-Kui Au and James L Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, 16(4):263–277, 2001.

[2] A Harbitz. Efficient and accurate probability of failure calculation by the use of importance sampling technique. In *Proceedings of the 4-th International Conference on Applications of Statistics and Probability in Soil and Structural Engineering*, volume 4, pages 825–836, 1983.

[3] Armin Tabandeh, Gaofeng Jia, and Paolo Gardoni. A review and assessment of importance sampling methods for reliability analysis. *Structural Safety*, 97:102216, 2022.

[4] Abraham M Hasofer and Niels C Lind. Exact and invariant second-moment code format. *Journal of the Engineering Mechanics Division*, 100(1):111–121, 1974.

[5] Rüdiger Rackwitz and Bernd Flessler. Structural reliability under combined random load sequences. *Computers & Structures*, 9(5):489–494, 1978.

[6] Karl Breitung. Asymptotic approximations for multinormal integrals. *Journal of Engineering Mechanics*, 110(3):357–366, 1984.

[7] Zhangli Hu, Rami Mansour, Mårten Olsson, and Xiaoping Du. Second-order reliability methods: a review and comparative study. *Structural and Multidisciplinary Optimization*, pages 1–31, 2021.

[8] YG Zhao and ZH Lu. *Structural reliability: approaches from perspectives of statistical moments*. John Wiley & Sons Ltd Chichester, 2021.

[9] Jun Xu and Chao Dang. A new bivariate dimension reduction method for efficient structural reliability analysis. *Mechanical Systems and Signal Processing*, 115:281–300, 2019.

[10] Tong Zhou and Yongbo Peng. Adaptive bayesian quadrature based statistical moments estimation for structural reliability analysis. *Reliability Engineering & System Safety*, 198:106902, 2020.

[11] Xufang Zhang and Mahesh D Pandey. Structural reliability analysis based on the concepts of entropy, fractional moment and dimensional reduction method. *Structural Safety*, 43:28–40, 2013.

[12] Jun Xu and Fan Kong. Adaptive scaled unscented transformation for highly efficient structural reliability analysis by maximum entropy method. *Structural Safety*, 76:123–134, 2019.

[13] Chao Dang and Jun Xu. A mixture distribution with fractional moments for efficient seismic reliability analysis of nonlinear structures. *Engineering Structures*, 208:109912, 2020.

[14] Jun Xu and Chao Dang. A novel fractional moments-based maximum entropy method for high-dimensional reliability analysis. *Applied Mathematical Modelling*, 75:749–768, 2019.

[15] Chao Dang and Jun Xu. Unified reliability assessment for problems with low-to high-dimensional random inputs using the Laplace transform and a mixture distribution. *Reliability Engineering & System Safety*, 204:107124, 2020.

[16] Chao Dang, Pengfei Wei, and Michael Beer. An approach to evaluation of EVD and small failure probabilities of uncertain nonlinear structures under stochastic seismic excitations. *Mechanical Systems and Signal Processing*, 152:107468, 2021.

[17] Jian-Bing Chen and Jie Li. Dynamic response and reliability analysis of non-linear stochastic structures. *Probabilistic Engineering Mechanics*, 20(1):33–44, 2005.

[18] Jie Li and Jian-Bing Chen. Dynamic response and reliability analysis of structures with uncertain parameters. *International Journal for Numerical Methods in Engineering*, 62(2):289–315, 2005.

[19] Jian-Bing Chen and Jie Li. The extreme value distribution and dynamic reliability analysis of nonlinear structures with uncertain parameters. *Structural Safety*, 29(2):77–93, 2007.

[20] Jie Li, Jian-bing Chen, and Wen-liang Fan. The equivalent extreme-value event and evaluation of the structural system reliability. *Structural Safety*, 29(2):112–131, 2007.

[21] Xiaolan Li, Guohai Chen, Haichao Cui, and Dixiong Yang. Direct probability integral method for static and dynamic reliability analysis of structures with complicated performance functions. *Computer Methods in Applied Mechanics and Engineering*, 374:113583, 2021.

[22] Guohai Chen and Dixiong Yang. A unified analysis framework of static and dynamic structural reliabilities based on direct probability integral method. *Mechanical Systems and Signal Processing*, 158:107783, 2021.

[23] Luxin Li, Guohai Chen, Mingxuan Fang, and Dixiong Yang. Reliability analysis of structures with multimodal distributions based on direct probability integral method. *Reliability Engineering & System Safety*, 215:107885, 2021.

[24] Guohai Chen, Dixiong Yang, Yunhe Liu, and Hongchao Guo. System reliability analyses of static and dynamic structures via direct probability integral method. *Computer Methods in Applied Mechanics and Engineering*, 388:114262, 2022.

[25] Christian G Bucher and Ulrich Bourgund. A fast and efficient response surface approach for structural reliability problems. *Structural Safety*, 7(1):57–66, 1990.

[26] Soo-Chang Kang, Hyun-Moo Koh, and Jinkyo F Choo. An efficient response surface method using moving least squares approximation for structural reliability analysis. *Probabilistic Engineering Mechanics*, 25(4):365–371, 2010.

[27] Ali Hadidi, Bahman Farahmand Azar, and Amin Rafiee. Efficient response surface method for high-dimensional structural reliability analysis. *Structural Safety*, 68:15–27, 2017.

[28] Claudio M Rocco and José Alı Moreno. Fast monte carlo reliability evaluation using support vector machine. *Reliability Engineering & System Safety*, 76(3):237–243, 2002.

[29] Kai Cheng and Zhenzhou Lu. Adaptive bayesian support vector regression model for structural reliability analysis. *Reliability Engineering & System Safety*, 206:107286, 2021.

[30] Jinsheng Wang, Chenfeng Li, Guoji Xu, Yongle Li, and Ahsan Kareem. Efficient structural reliability analysis based on adaptive bayesian support vector regression. *Computer Methods in Applied Mechanics and Engineering*, 387:114172, 2021.

[31] Stefano Marelli and Bruno Sudret. An active-learning algorithm that combines sparse polynomial chaos expansions and bootstrap for structural reliability analysis. *Structural Safety*, 75:67–74, 2018.

[32] Biswarup Bhattacharyya. Structural reliability analysis by a bayesian sparse polynomial chaos expansion. *Structural Safety*, 90:102074, 2021.

[33] Irfan Kaymaz. Application of kriging method to structural reliability problems. *Structural Safety*, 27(2):133–151, 2005.

[34] Vincent Dubourg, Bruno Sudret, and Franois Deheeger. Metamodel-based importance sampling for structural reliability analysis. *Probabilistic Engineering Mechanics*, 33:47–57, 2013.

[35] Ivan Depina, Thi Minh Hue Le, Gordon Fenton, and Gudmund Eiksund. Reliability analysis with metamodel line sampling. *Structural Safety*, 60:1–15, 2016.

[36] Barron J Bichon, Michael S Eldred, Laura Painton Swiler, Sandaran Mahadevan, and John M McFarland. Efficient global reliability analysis for nonlinear implicit performance functions. *AIAA Journal*, 46(10):2459–2468, 2008.

[37] Benjamin Echard, Nicolas Gayton, and Maurice Lemaire. AK-MCS: an active learning reliability method combining kriging and monte carlo simulation. *Structural Safety*, 33(2):145–154, 2011.

[38] Xufeng Yang, Yongshou Liu, Yi Gao, Yishang Zhang, and Zongzhan Gao. An active learning kriging model for hybrid reliability analysis with both random and interval variables. *Structural and Multidisciplinary Optimization*, 51(5):1003–1016, 2015.

[39] Zhaoyan Lv, Zhenzhou Lu, and Pan Wang. A new learning function for kriging and its applications to solve reliability problems in engineering. *Computers & Mathematics with Applications*, 70(5):1182–1197, 2015.

[40] Zhili Sun, Jian Wang, Rui Li, and Cao Tong. Lif: A new kriging based learning function and its application to structural reliability analysis. *Reliability Engineering & System Safety*, 157:152–165, 2017.

[41] Xufang Zhang, Lei Wang, and John Dalsgaard Sørensen. REIF: a novel active-learning function toward adaptive kriging surrogate models for structural reliability analysis. *Reliability Engineering & System Safety*, 185:440–454, 2019.

[42] Yan Shi, Zhenzhou Lu, Ruyang He, Yicheng Zhou, and Siyu Chen. A novel learning function based on kriging for reliability analysis. *Reliability Engineering & System Safety*, 198:106857, 2020.

[43] Chao Dang, Pengfei Wei, Jingwen Song, and Michael 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):04021054, 2021.

[44] Benjamin Echard, Nicolas Gayton, Maurice Lemaire, and Nicolas Relun. A combined importance sampling and kriging reliability method for small failure probabilities with time-demanding numerical models. *Reliability Engineering & System Safety*, 111:232–240, 2013.

[45] Xiaoxu Huang, Jianqiao Chen, and Hongping Zhu. Assessing small failure probabilities by AK–SS: An active learning method combining kriging and subset simulation. *Structural Safety*, 59:86–95, 2016.

[46] Pengfei Wei, Chenghu Tang, and Yuting Yang. Structural reliability and reliability sensitivity analysis of extremely rare failure events by combining sampling and surrogate model methods. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 233(6):943–957, 2019.

[47] Tong Zhou and Yongbo Peng. Structural reliability analysis via dimension reduction, adaptive sampling, and monte carlo simulation. *Structural and Multidisciplinary Optimization*, 62(5):2629–2651, 2020.

[48] Tong Zhou and Yongbo Peng. Active learning and active subspace enhancement for pdem-based high-dimensional reliability analysis. *Structural Safety*, 88:102026, 2021.

[49] Yushan Liu, Luyi Li, Sihan Zhao, and Shufang Song. A global surrogate model technique based on principal component analysis and kriging for uncertainty propagation of dynamic systems. *Reliability Engineering & System Safety*, 207:107365, 2021.

[50] Yongbo Peng, Tong Zhou, and Jie Li. Surrogate modeling immersed probability density evolution method for structural reliability analysis in high dimensions. *Mechanical Systems and Signal Processing*, 152:107366, 2021.

[51] Tong Zhou and Yongbo Peng. Kernel principal component analysis-based gaussian process regression modelling for high-dimensional reliability analysis. *Computers & Structures*, 241:106358, 2020.

[52] M Munoz Zuniga, Achille Murangira, and Timothée Perdrizet. Structural reliability assessment through surrogate based importance sampling with dimension reduction. *Reliability Engineering & System Safety*, 207:107289, 2021.

[53] Jianhua Yin and Xiaoping Du. Active learning with generalized sliced inverse regression for high-dimensional reliability analysis. *Structural Safety*, 94:102151, 2022.

[54] Rui Teixeira, Maria Nogal, Alan O'Connor, and Beatriz Martinez-Pastor. Reliability assessment with density scanned adaptive kriging. *Reliability Engineering & System Safety*, 199:106908, 2020.

[55] Rui Li, Xihui Liang, and Qingjin Peng. A selection strategy for kriging based design of experiments by spectral clustering and learning function. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering*, 7(2):020902, 2021.

[56] Zequan Chen, Guofa Li, Jialong He, Zhaojun Yang, and Jili Wang. A new parallel adaptive

structural reliability analysis method based on importance sampling and k-medoids clustering. *Reliability Engineering & System Safety*, 218:108124, 2022.

[57] Rui Teixeira, Maria Nogal, and Alan O'Connor. Adaptive approaches in metamodel-based reliability analysis: A review. *Structural Safety*, 89:102019, 2021.

[58] Anthony O'Hagan. Bayes–hermite quadrature. *Journal of Statistical Planning and Inference*, 29(3):245–260, 1991.

[59] Carl Edward Rasmussen and Zoubin Ghahramani. Bayesian monte carlo. *Advances in Neural Information Pprocessing Systems*, pages 505–512, 2003.

[60] Pengfei Wei, Xing Zhang, and Michael Beer. Adaptive experiment design for probabilistic integration. *Computer Methods in Applied Mechanics and Engineering*, 365:113035, 2020.

[61] Vidhi Lalchand and Carl Edward Rasmussen. Approximate inference for fully bayesian gaussian process regression. In *Symposium on Advances in Approximate Bayesian Inference*, pages 1–12. PMLR, 2020.

[62] Aaron R Voelker, Jan Gosmann, and Terrence C Stewart. Efficiently sampling vectors and coordinates from the n-sphere and n-ball. *Centre for Theoretical Neuroscience-Technical Report*, 2017.

[63] Jie Li and Jianbing Chen. *Stochastic dynamics of structures*. John Wiley & Sons, 2009.

[64] David JC MacKay and David JC Mac Kay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[65] S. Marelli, R. Schöbi, and B. Sudret. UQLab user manual – Structural reliability (Rare event estimation). Technical report, Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland, 2022. Report UQLab-V2.0-107.

[66] Roland Schöbi, Bruno Sudret, and Stefano Marelli. Rare event estimation using polynomial-chaos kriging. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, 3(2):D4016002, 2017.

[67] Antonio Borri and Emanuela Speranzini. Structural reliability analysis using a standard deterministic finite element code. *Structural Safety*, 19(4):361–382, 1997.

[68] Iason Papaioannou, Sebastian Geyer, and Daniel Straub. Improved cross entropy-based importance sampling with a flexible mixture model. *Reliability Engineering & System Safety*, 191:106564, 2019.

Chapter 7

Bayesian perspective for structural reliability analysis

Structural reliability analysis: A Bayesian perspective

Chao Dang[a,⋆], Marcos A. Valdebenito[b], Matthias G.R. Faes[c], Pengfei Wei[d] ,Michael Beer[a,e,f]

[a]*Institute for Risk and Reliability, Leibniz University Hannover, Callinstr. 34, Hannover 30167, Germany*

[b]*Faculty of Engineering and Sciences, Universidad Adolfo Ibáñez, Av. Padre Hurtado 750, Viña del Mar 2562340, Chile*

[c]*Chair for Reliability Engineering, TU Dortmund University, Leonhard-Euler-Str. 5, Dortmund 44227, Germany*

[d]*School of Power and Energy, 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*

Published in Structural Safety in November 2022

Abstract: Numerical methods play a dominant role in structural reliability analysis, and the goal has long been to produce a failure probability estimate with a desired level of accuracy using a minimum number of performance function evaluations. In the present study, we attempt to offer a Bayesian perspective on the failure probability integral estimation, as opposed to the classical frequentist perspective. For this purpose, a principled Bayesian Failure Probability Inference (BFPI) framework is first developed, which allows to quantify, propagate and reduce numerical uncertainty behind the failure probability due to discretization error. Especially, the posterior variance of the failure probability is derived in a semi-analytical form, and the Gaussianity of the posterior failure probability distribution is investigated numerically. Then, a Parallel Adaptive-Bayesian Failure Probability Learning (PA-BFPL) method is proposed within the Bayesian framework. In the PA-BFPL method, a variance-amplified importance sampling technique is presented to evaluate the posterior mean and variance of the failure probability, and an adaptive parallel active learning strategy is proposed to identify multiple updating points at each iteration. Thus, a novel advantage of PA-BFPL is that both prior knowledge and parallel computing can be used to make inference about the failure probability. Four numerical examples are investigated, indicating the potential

*Corresponding Author
E-mail address: chao.dang@irz.uni-hannover.de (C. Dang)

benefits by advocating a Bayesian approach to failure probability estimation.

Keywords: Failure probability, Bayesian inference, Gaussian process, Numerical uncertainty, Parallel computing

7.1 Introduction

A fundamental problem in structural reliability analysis is to assess the likelihood that a structure attains an unsatisfactory performance in the presence of uncertainties. Within a probabilistic framework, the primary objective is to compute the so-called failure probability P_f, defined by the following multifold integral:

$$P_f = \text{Prob}\left[g(\boldsymbol{X}) \leq 0\right] = \int_{\mathcal{X}} I(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{7.1}$$

where $\text{Prob}\left[\cdot\right]$ denotes the probability operator; $\boldsymbol{X} = [X_1, X_2, \cdots, X_d] \in \mathcal{X} \subseteq \mathbb{R}^d$ is a vector of d random variables with known joint probability density function (PDF) $f_{\boldsymbol{X}}(\boldsymbol{x})$; $Y = g(\boldsymbol{X}) : \mathbb{R}^d \to \mathbb{R}$ is the performance function (or limit state function) with $y = g(\boldsymbol{x}) \leq 0$ indicating a failure state and a safe state otherwise; $I(\boldsymbol{x})$ is the failure indicator function such that:

$$I\left(\boldsymbol{x}\right) = \begin{cases} 1, g\left(\boldsymbol{x}\right) \leqslant 0 \\ 0, \text{otherwise} \end{cases}. \tag{7.2}$$

Except for some special cases, it is impossible to derive the analytical solution to the failure probability (defined by Eq. (7.1)). Besides, the g-function in practical applications is typically dependent on a simulation model (e.g., a finite element model) so that each evaluation can be computationally demanding. Therefore, numerical methods that minimize the number of g-function evaluations are highly desirable to approximate the failure probability. Even though various methods following different paradigms have been developed over the past several decades (e.g., as summarized in [1]), it seems that they never reach the end of being efficient while accurate and generally applicable. The present paper is also concerned with developing a new reliability analysis method, but putting more emphasis on how to interpret the problem of failure probability estimation.

In fact, the problem of evaluating the failure probability integral (Eq. (7.1)) can be treated as a statistical problem, though it does not mean that all methods must follow this perspective.

Specifically, the failure probability P_f is an unknown quantity of interest, about which we wish to make inference using a set of g-function observations (equivalently, I-function observations), say $g(\boldsymbol{x}^{(1)}), g(\boldsymbol{x}^{(2)}), \cdots, g(\boldsymbol{x}^{(n)})$. Further, a statistical inference rule approximates P_f as a function of those observations.

In the classical frequentist viewpoint, the sample $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \cdots, \boldsymbol{x}^{(n)}$ might be supposed to draw at random from a population distributed according to $f_{\boldsymbol{X}}(\boldsymbol{x})$. Taking the Monte Carlo simulation (MCS) method as an example, the MCS estimator for the failure probability is given by the sample mean:

$$\hat{P}_f^{MCS} = \frac{1}{n}\sum_{i=1}^{n} I(\boldsymbol{x}^{(i)}). \tag{7.3}$$

The law of large numbers implies that $\hat{P}_f^{MCS}$ converges to P_f with probability 1 as $n \to \infty$. The estimator is viewed as a random variable since $\boldsymbol{x}^{(i)}$ is random. Besides, by the central limit theorem, $\hat{P}_f^{MCS}$ asymptotically follows a normal distribution for a large n. In practical applications, one can only afford a finite sample size to approximate the failure probability. Hence, the uncertainty associated with $\hat{P}_f^{MCS}$ due to the sampling variability may not be neglected. Such uncertainty can be measured by the variance of the estimator [2]:

$$\mathbb{V}\left[\hat{P}_f^{MCS}\right] = \frac{\hat{P}_f^{MCS}(1-\hat{P}_f^{MCS})}{n}, \tag{7.4}$$

where $\mathbb{V}[\cdot]$ denotes the variance operator. Despite its conceptual and algorithmic simplicity, the MCS method is often criticized by many authors for its unreasonable effectiveness and theoretical unsoundness [3, 4]. In addition, some variants of the MCS method, e.g., subset simulation [5, 6], importance sampling [7–10], have been developed and are able to offer improved efficiency. These methods, however, can still be regarded as more advanced frequentist approaches, and hence may be subject to the same criticism as MCS.

In contrast to the classical frequentist perspective, we seek to interpret the problem of failure probability integral estimation as a Bayesian inference problem. For this context, a central role is played by numerical integration (also known as quadrature) that is widely encountered in scientific computing. The study of numerical integration from a point of view of Bayesian dates back to at least the work of Diaconis [11] and has led to the commonly known Bayesian quadrature, Bayesian cubature or probabilistic integration [12–15]. In such methods, our uncertainty about the true in-

tegral value resulted from a limited number of integrand observations (i.e., discretization error) is regarded as a kind of epistemic uncertainty, which can be modelled following a Bayesian approach. The Bayesian approach to numerical integration has demonstrated many promising advantages with respect to the classical approach (e.g., see [11, 16]). However, only a few studies have investigated the Bayesian approach to failure probability estimation, which requires a slightly different treatment compared to a common quadrature problem. Loosely speaking, the popular active learning reliability methods [17, 18], e.g., efficient global reliability analysis [19] and AK-MCS [20], have almost reached the idea of being Bayesian. That is, the surrogate models (e.g., Kriging) used in those methods allow a Bayesian interpretation. In spite of that, the existing methods do not count as fully Bayesian in the strict sense because they provide no probabilistic uncertainty measure over the failure probability. A truly Bayesian interpretation was, to the best of our knowledge, first clearly reported in the work [21], where the Bayesian Monte Carlo method developed in [13] was applied. However, it is challenging to directly place a Gaussian process (GP) prior over the failure indicator function with a large discontinuity. The first author and his co-workers continued the idea of re-interpreting the failure probability integral estimation with Bayesian inference in a recent work [22], and then it was further improved in [1]. In [22], the posterior mean and an upper-bound of the posterior variance of the failure probability were derived, given that a GP prior was assigned to the performance function. Nevertheless, the posterior variance and posterior distribution of the failure probability are still not available, which are undoubtedly of interest and importance in a Bayesian framework.

This paper aims to present a fully Bayesian perspective on failure probability estimation, complementing the work in [1, 22]. The main contributions of this work are summarized as follows. First, to the best of the authors' knowledge, a complete and principled Bayesian framework for failure probability estimation is developed for the first time. The framework is termed 'Bayesian Failure Probability Inference' (BFPI), in which the posterior variance of the failure probability is derived in a semi-analytical form. Besides, the posterior distribution of the failure probability is also empirically investigated by several numerical examples. Second, we illustrate how the BFPI framework can be used to make inference about the failure probability in an adaptive scheme. The resulting method is called 'Parallel Adaptive-Bayesian Failure Probability Learning' (PA-BFPL). In the PA-BFPL method, a variance-amplified importance sampling (VAIS) method is proposed to approximate the posterior mean and variance of the failure probability and an adaptive paral-

lel learning strategy based on the concepts of expected misclassification probability contribution (EMPC) and k-means clustering is presented to enable multipoint selection (hence parallel distributed processing). In addition, we also suggest a new stopping criterion in order to achieve a desired level of accuracy for the failure probability estimate.

The rest of this paper is organized as follows. The proposed BFPI framework is introduced in Section 7.2. Section 7.3 presents the proposed PA-BFPL method. Four numerical examples are investigated in Section 7.4 to demonstrate the proposed method. The Gaussianity of the posterior failure probability is numerically studied in Section 7.5. The paper is closed with some concluding remarks in Section 7.6.

7.2 Bayesian failure probability inference

In this section, the problem of failure probability estimation is interpreted as a Bayesian inference problem, leading to a framework of Bayesian failure probability inference (BFPI). As shown in Fig. 7.1, the proposed BFPI framework begins with a prior distribution over the g-function. Conditional on the observations that arise from evaluating the g-function at some points, we arrive at a posterior distribution over g. This in turn implies a posterior distribution over the failure indicator function I, as well as the failure probability P_f.

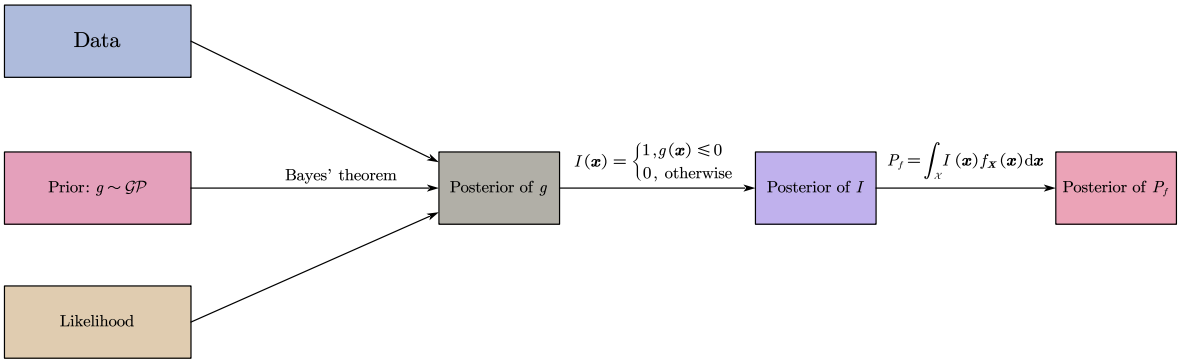


Figure 7.1: A schematic illustration of the proposed BFPI framework.

7.2.1 Prior distributions

A convenient way of putting a prior over the g-function is through GP. A GP can be viewed as a collection of random variables indexed by time or space, any finite number of which have a

multivariate Gaussian distribution.

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, where Ω is a sample space, $\mathcal{F}$ is a set of events equipped with σ-algebra and $\mathbb{P}$ is a probability measure. A GP can be defined as $Z(\varpi, \boldsymbol{x})$: $\Omega \times \mathcal{X} \to \mathbb{R}$. For a fixed location $\boldsymbol{x} \in \mathcal{X}$, $Z(\varpi, \cdot)$ is Gaussian. Conversely, for every fixed elementary event $\varpi \in \Omega$, $Z(\cdot, \boldsymbol{x})$ is a realization of the GP. As a GP can be completely characterized by its mean and covariance functions, our prior assumption is thus rewritten as follows:

$$\hat{g}_0(\varpi, \boldsymbol{x}) \sim \mathcal{GP}(m_{\hat{g}_0}(\boldsymbol{x}), c_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{x}')), \tag{7.5}$$

where $\hat{g}_0$ denotes the prior distribution of g before any observations are obtained; the prior mean function $m_{\hat{g}_0}(\boldsymbol{x}) : \mathcal{X} \to \mathcal{Z}$ and prior covariance function $c_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{x}') : \mathcal{X} \times \mathcal{X} \to \mathcal{Z}$ are respectively defined as:

$$m_{\hat{g}_0}(\boldsymbol{x}) = \mathbb{E}_{\varpi}\left[\hat{g}_0(\varpi, \boldsymbol{x})\right], \tag{7.6}$$

$$c_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}_{\varpi}\left[\left(\hat{g}_0(\varpi, \boldsymbol{x}) - m(\boldsymbol{x})\right)\left(\hat{g}_0(\varpi, \boldsymbol{x}') - m(\boldsymbol{x}')\right)\right], \tag{7.7}$$

in which $\mathbb{E}_{\varpi}[\cdot]$ denotes the expectation operation taken over Ω. The prior mean function reflects the general trend of the GP prior, whereas the prior covariance function encodes our key beliefs on the similarity of the g-function between two points. Among many options available in the literature, this study adopts the commonly used constant mean and squared exponential kernel functions:

$$m_{\hat{g}_0}(\boldsymbol{x}) = \beta, \tag{7.8}$$

$$c_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{x}') = s^2 \exp\left[-\frac{1}{2}\left(\boldsymbol{x} - \boldsymbol{x}'\right)\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{x} - \boldsymbol{x}'\right)^{\mathrm{T}}\right], \tag{7.9}$$

where β is a constant; s^2 denote the process variance; $\boldsymbol{\Sigma} = \mathrm{diag}(l_1^2, l_2^2, ..., l_d^2)$ is a diagonal matrix, whose i-th diagonal entry is l_i^2 with $l_i > 0$ being the length scale in the i-th dimension. Note that the choice of the prior and covariance functions does not affect the generality of our developments, and other forms can also be employed. In Eqs. (7.8) and (7.9), there exist $d + 2$ hyper-parameters to be determined (collected in $\boldsymbol{\vartheta} = [\beta, s, l_1, l_2, \cdots, l_d]$) in total.

Remark 1. Corresponding to the GP prior for g, this also implies implicitly prior distributions for the failure indicator function I, and the failure probability P_f. They are not given here because our main concern is their posterior distributions. However, one still can easily obtain these prior

distributions referring to Subsection 7.2.3.

7.2.2 Learning the hyper-parameters

Suppose that now we observe the g-function at some locations. Let $\boldsymbol{\mathscr{X}} = \left\{\boldsymbol{x}^{(i)}\right\}_{i=1}^{n}$ denote an $n \times d$ matrix containing n design points. The corresponding g-function values at $\boldsymbol{\mathscr{X}}$ are collected in an $n \times 1$ vector $\boldsymbol{\mathscr{Y}} = \left\{y^{(i)}\right\}_{i=1}^{n}$ with $y^{(i)} = g(\boldsymbol{x}^{(i)})$. The hyper-parameters should be learned from the given data $\mathscr{D} = \{\boldsymbol{\mathscr{X}}, \boldsymbol{\mathscr{Y}}\}$, and three approaches are typically considered [23]: (1) maximum likelihood estimation (MLE); (2) maximum a posteriori (MAP) estimation; (3) fully Bayesian approach. In this study, we use the MLE method as follows.

Under the GP prior, the marginal likelihood of $\boldsymbol{\mathscr{Y}}$ is a multivariate normal density:

$$p(\boldsymbol{\mathscr{Y}}|\boldsymbol{\mathscr{X}}, \boldsymbol{\vartheta}) = \frac{1}{\sqrt{(2\pi)^n |\boldsymbol{C}_{\hat{g}_0}|}} \exp\left[-\frac{1}{2}(\boldsymbol{\mathscr{Y}} - \beta)\boldsymbol{C}_{\hat{g}_0}^{-1}(\boldsymbol{\mathscr{Y}} - \beta)^{\mathrm{T}}\right], \tag{7.10}$$

where $\boldsymbol{C}_{\hat{g}_0}$ is an n-by-n covariance matrix, whose (i,j)-th entry is $[\boldsymbol{C}_{\hat{g}_0}]_{ij} = c_{\hat{g}_0}(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$; $|\cdot|$ is the determinant operator. The hyper-parameters are tuned by minimizing the negative log marginal likelihood:

$$\hat{\boldsymbol{\vartheta}} = \arg\min_{\boldsymbol{\vartheta}} -\log\left[p(\boldsymbol{\mathscr{Y}}|\boldsymbol{\mathscr{X}}, \boldsymbol{\vartheta})\right], \tag{7.11}$$

where

$$\log\left[p(\boldsymbol{\mathscr{Y}}|\boldsymbol{\mathscr{X}}, \boldsymbol{\vartheta})\right] = -\frac{1}{2}\left[(\boldsymbol{\mathscr{Y}} - \beta)\boldsymbol{C}_{\hat{g}_0}^{-1}(\boldsymbol{\mathscr{Y}} - \beta)^{\mathrm{T}} + \log|\boldsymbol{C}_{\hat{g}_0}| + n\log 2\pi\right]. \tag{7.12}$$

7.2.3 Posterior distributions

Conditional on the data $\mathscr{D}$, the induced posterior distribution of g is also a GP:

$$\hat{g}_n(\varpi, \boldsymbol{x}) \sim \mathcal{GP}(m_{\hat{g}_n}(\boldsymbol{x}), c_{\hat{g}_n}(\boldsymbol{x}, \boldsymbol{x}')), \tag{7.13}$$

where $\hat{g}_n$ denotes the posterior distribution of g given n observations; $m_{\hat{g}_n}(\boldsymbol{x})$ and $c_{\hat{g}_n}(\boldsymbol{x}, \boldsymbol{x}')$ are the posterior mean and covariance functions respectively, which can be analytically derived as:

$$m_{\hat{g}_n}(\boldsymbol{x}) = m_{\hat{g}_0}(\boldsymbol{x}) + \boldsymbol{c}_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{\mathscr{X}})\boldsymbol{C}_{\hat{g}_0}^{-1}(\boldsymbol{\mathscr{Y}} - \boldsymbol{m}_{\hat{g}_0}(\boldsymbol{\mathscr{X}})), \tag{7.14}$$

$$c_{\hat{g}_n}(\boldsymbol{x}, \boldsymbol{x}') = c_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{c}_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{\mathscr{X}})\boldsymbol{C}_{\hat{g}_0}^{-1}\boldsymbol{c}_{\hat{g}_0}(\boldsymbol{\mathscr{X}}, \boldsymbol{x}'), \tag{7.15}$$

where $\boldsymbol{m}_{\hat{g}_0}(\boldsymbol{\mathcal{X}})$ is an n-by-1 vector with i-th element being $m_{\hat{g}_0}(\boldsymbol{x}^{(i)})$; $\boldsymbol{c}_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{\mathcal{X}})$ is a 1-by-n covariance vector with i-th element being $c_{\hat{g}_0}(\boldsymbol{x}, \boldsymbol{x}^{(i)})$; $\boldsymbol{c}_{\hat{g}_0}(\boldsymbol{\mathcal{X}}, \boldsymbol{x}')$ is an n-by-1 covariance vector with i-th element being $\boldsymbol{c}_{\hat{g}_0}(\boldsymbol{x}^{(i)}, \boldsymbol{x}')$. It should be pointed out that: (1) For any $\boldsymbol{x}^{(i)} \in \boldsymbol{\mathcal{X}}$, the posterior GP is an exact predictor. This means that if a prediction is carried out at an observed point $\boldsymbol{x}^{(i)}$, the posterior mean is exactly equal to the corresponding observation (i.e., $m_{\hat{g}_n}(\boldsymbol{x}^{(i)}) = y^{(i)}$) and the posterior variance is equal to zero (i.e., $\sigma^2_{\hat{g}_n}(\boldsymbol{x}^{(i)}) = c_{\hat{g}_n}(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(i)}) = 0$). (2) For any $\boldsymbol{x} \notin \boldsymbol{\mathcal{X}}$, the posterior GP at $\boldsymbol{x}$ is Gaussian. In this case, the posterior mean $m_{\hat{g}_n}(\boldsymbol{x})$ is a natural estimate of the g-function value, whereas the posterior variance $\sigma^2_{\hat{g}_n}(\boldsymbol{x}) = c_{\hat{g}_n}(\boldsymbol{x}, \boldsymbol{x})$ can measure our uncertainty of the estimate.

The posterior distribution of the failure indicator function I has the following relationship with $\hat{g}_n$:

$$\hat{I}_n(\varpi, \boldsymbol{x}) = \begin{cases} 1, \hat{g}_n(\varpi, \boldsymbol{x}) \leqslant 0 \\ 0, \text{otherwise} \end{cases}, \tag{7.16}$$

where $\hat{I}_n$ denote the posterior distribution of I conditional on $\boldsymbol{\mathcal{D}}$. Based on Eqs. (7.13) and (7.16), the induced posterior distribution $\hat{I}_n$ should follow a generalized Bernoulli process [1] (GBP):

$$\hat{I}_n(\varpi, \boldsymbol{x}) \sim \mathcal{GBP}(m_{\hat{I}_n}(\boldsymbol{x}), c_{\hat{I}_n}(\boldsymbol{x}, \boldsymbol{x}')), \tag{7.17}$$

where $m_{\hat{I}_n}(\boldsymbol{x})$ and $c_{\hat{I}_n}(\boldsymbol{x}, \boldsymbol{x}')$ are the posterior mean and covariance functions, respectively. They can be derived as follows:

$$\begin{aligned} m_{\hat{I}_n}(\boldsymbol{x}) &= \mathbb{E}_{\varpi}\left[\hat{I}_n(\varpi, \boldsymbol{x})\right] \\ &= \mathbb{P}\left[\hat{g}_n(\varpi, \boldsymbol{x}) \leq 0\right] \\ &= \Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right), \end{aligned} \tag{7.18}$$

[1] In the conventional way, a Bernoulli process is defined as a finite or infinite sequence of binary random variables that are independently and identically distributed. Here we use 'generalized' to indicate that the possible correlation among the sequence is considered. For more information on this topic, one can refer to, e.g., [24].

$$
\begin{aligned}
c_{\hat{I}_n}(\boldsymbol{x},\boldsymbol{x}') =& \mathbb{E}_{\varpi}\left[\left(\hat{I}_n(\varpi,\boldsymbol{x})-m_{\hat{I}_n}(\boldsymbol{x})\right)\left(\hat{I}_n(\varpi,\boldsymbol{x}')-m_{\hat{I}_n}(\boldsymbol{x}')\right)\right]\\
=& \mathbb{E}_{\varpi}\left[\hat{I}_n(\varpi,\boldsymbol{x})\hat{I}_n(\varpi,\boldsymbol{x}')\right]-\mathbb{E}_{\varpi}\left[\hat{I}_n(\varpi,\boldsymbol{x})\right]\mathbb{E}_{\varpi}\left[\hat{I}_n(\varpi,\boldsymbol{x}')\right]\\
=& \mathbb{P}\left[\hat{g}_n(\varpi,\boldsymbol{x})\le 0,\hat{g}_n(\varpi,\boldsymbol{x}')\le 0\right]-m_{\hat{I}_n}(\boldsymbol{x})m_{\hat{I}_n}(\boldsymbol{x}')\\
=& F\left([0\ 0];\boldsymbol{m}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}'),\boldsymbol{C}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}')\right)-\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right)\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x}')}{\sigma_{\hat{g}_n}(\boldsymbol{x}')}\right),
\end{aligned}
\tag{7.19}
$$

where Φ is the cumulative distribution function (CDF) of the standard normal distribution; F is the joint CDF of a bivariate normal distribution; The terms $\boldsymbol{m}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}')$ and $\boldsymbol{C}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}')$ are expressed as:

$$
\boldsymbol{m}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}') = [m_{\hat{g}_n}(\boldsymbol{x}), m_{\hat{g}_n}(\boldsymbol{x}')] \tag{7.20}
$$

$$
\boldsymbol{C}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}') = \begin{bmatrix} c_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}) & c_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}') \\ c_{\hat{g}_n}(\boldsymbol{x}',\boldsymbol{x}) & c_{\hat{g}_n}(\boldsymbol{x}',\boldsymbol{x}') \end{bmatrix} = \begin{bmatrix} \sigma^2_{\hat{g}_n}(\boldsymbol{x}) & c_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}') \\ c_{\hat{g}_n}(\boldsymbol{x}',\boldsymbol{x}) & \sigma^2_{\hat{g}_n}(\boldsymbol{x}') \end{bmatrix}. \tag{7.21}
$$

Note that though no closed form is available for F, there are a number of software packages that evaluate it numerically.

The posterior distribution of the failure probability P_f is defined as:

$$
\hat{P}_{f,n}(\varpi) = \int_{\mathcal{X}} \hat{I}_n(\varpi,\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}, \tag{7.22}
$$

where $\hat{P}_{f,n}$ denotes the posterior distribution of $P_{f,n}$ conditional on $\mathscr{D}$. Eq. (7.22) implies that $\hat{P}_f$ is a random variable, whose exact distribution is not known yet. To this end, we investigate empirically the posterior failure probability distribution by several numerical examples in Section 7.5. By applying Fubini's theorem, the posterior mean and variance of P_f can be derived as:

$$
\begin{aligned}
m_{\hat{P}_{f,n}} =& \mathbb{E}_{\varpi}\left[\hat{P}_{f,n}(\varpi)\right]\\
=& \int_{\Omega}\int_{\mathcal{X}} \hat{I}_n(\varpi,\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathbb{P}\left(\mathrm{d}\varpi\right)\\
=& \int_{\mathcal{X}}\int_{\Omega} \hat{I}_n(\varpi,\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x})\mathbb{P}\left(\mathrm{d}\varpi\right)\mathrm{d}\boldsymbol{x}\\
=& \int_{\mathcal{X}} \mathbb{E}_{\varpi}\left[\hat{I}_n(\varpi,\boldsymbol{x})\right] f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\\
=& \int_{\mathcal{X}} \Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x},
\end{aligned}
\tag{7.23}
$$

$$
\begin{aligned}
\sigma^2_{\hat{P}_{f,n}} =& \mathbb{V}_{\varpi}\left[\hat{P}_{f,n}(\varpi)\right] \\
=& \mathbb{E}_{\varpi}\left[\left(\hat{P}_f(\varpi) - \mathbb{E}_{\varpi}\left[\hat{P}_f(\varpi)\right]\right)^2\right] \\
=& \mathbb{E}_{\varpi}\left[\left(\int_{\mathcal{X}} \hat{I}_n(\varpi,\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int_{\mathcal{X}} \mathbb{E}_{\varpi}\left[\hat{I}_n(\varpi,\boldsymbol{x})\right] f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right)^2\right] \\
=& \int_{\Omega}\left(\int_{\mathcal{X}}\left(\hat{I}_n(\varpi,\boldsymbol{x}) - \mathbb{E}_{\varpi}\left[\hat{I}_n(\varpi,\boldsymbol{x})\right]\right) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right)^2 \mathbb{P}\left(\mathrm{d}\varpi\right) \\
=& \int_{\Omega}\int_{\mathcal{X}}\int_{\mathcal{X}}\left(\hat{I}_n(\varpi,\boldsymbol{x}) - \mathbb{E}_{\varpi}\left[\hat{I}_n(\varpi,\boldsymbol{x})\right]\right)\left(\hat{I}_n(\varpi,\boldsymbol{x}') - \mathbb{E}_{\varpi}\left[\hat{I}_n(\varpi,\boldsymbol{x}')\right]\right) f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}'\mathbb{P}\left(\mathrm{d}\varpi\right) \\
=& \int_{\mathcal{X}}\int_{\mathcal{X}}\int_{\Omega}\left(\hat{I}_n(\varpi,\boldsymbol{x}) - \mathbb{E}_{\varpi}\left[\hat{I}_n(\varpi,\boldsymbol{x})\right]\right)\left(\hat{I}_n(\varpi,\boldsymbol{x}') - \mathbb{E}_{\varpi}\left[\hat{I}_n(\varpi,\boldsymbol{x}')\right]\right) f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathbb{P}\left(\mathrm{d}\varpi\right)\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}' \\
=& \int_{\mathcal{X}}\int_{\mathcal{X}} \mathbb{E}_{\varpi}\left[\left(\hat{I}_n(\varpi,\boldsymbol{x}) - m_{\hat{I}_n}(\boldsymbol{x})\right)\left(\hat{I}_n(\varpi,\boldsymbol{x}') - m_{\hat{I}_n}(\boldsymbol{x}')\right)\right] f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}' \\
=& \int_{\mathcal{X}}\int_{\mathcal{X}} c_{\hat{I}_n}(\boldsymbol{x},\boldsymbol{x}') f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}' \\
=& \int_{\mathcal{X}}\int_{\mathcal{X}}\left[F\left([0\ 0]; \boldsymbol{m}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}'), \boldsymbol{C}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}')\right) - \Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right)\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x}')}{\sigma_{\hat{g}_n}(\boldsymbol{x}')}\right)\right] f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}' \\
=& \int_{\mathcal{X}}\int_{\mathcal{X}} F\left([0\ 0]; \boldsymbol{m}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}'), \boldsymbol{C}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}')\right) f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}' - \left(\int_{\mathcal{X}} \Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right)^2 \\
=& \int_{\mathcal{X}}\int_{\mathcal{X}} F\left([0\ 0]; \boldsymbol{m}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}'), \boldsymbol{C}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}')\right) f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}' - m^2_{\hat{P}_{f,n}},
\end{aligned}
\tag{7.24}
$$

where $\mathbb{V}_{\varpi}[\cdot]$ denotes the variance operation taken over Ω. The posterior distribution $\hat{P}_f$ reflects our uncertainty about P_f, which arises from the discretization error resulting from the fact that the g-function is only observed at a finite number of discrete locations. From this perspective, the proposed BFPI framework offers a principled approach to quantifying and propagating the numerical uncertainty behind the failure probability. Once given the data $\mathscr{D}$, the posterior mean $m_{\hat{P}_{f,n}}$ is a natural estimate for the failure probability P_f, while the posterior variance $\sigma^2_{\hat{P}_{f,n}}$ can measure our uncertainty about the estimate.

Remark 2. The posterior mean function $m_{\hat{I}_n}(\boldsymbol{x})$ of the failure indicator function I (defined in Eq. (7.18)) is the same as that given in our recent work [22]. In that work, we also derived the closed-form expressions of the posterior variance function of I and an upper bound of the posterior covariance $c_{\hat{I}_n}(\boldsymbol{x},\boldsymbol{x}')$ (Eq. (7.19)) by using Cauchy-Schwarz inequality.

Remark 3. The posterior mean $m_{\hat{P}_{f,n}}$ of the failure probability P_f (defined in Eq. (7.23)) was previously given in [22]. In additional, an upper bound of the posterior variance $\sigma^2_{\hat{P}_{f,n}}$ (Eq. (7.24)) was derived based on the upper bound of $c_{\hat{I}_n}(\boldsymbol{x},\boldsymbol{x}')$.

Remark 4. Numerical integration techniques are required to evaluate $m_{\hat{P}_{f,n}}$ and $\sigma^2_{\hat{P}_{f,n}}$ due to their analytical intractability. It is interesting that in our context the failure probability estimate $m_{\hat{P}_{f,n}}$ (Eq. (7.23)) is a integral over the whole domain $\mathcal{X}$ (both failure and safe domains), which is in contrast to the classical definition of failure probability (Eq. (7.1)) that is essentially a integral over the failure domain only. The former could be explained by the fact that Eq. (7.23) accounts for the numerical uncertainty at any $\boldsymbol{x} \in \mathcal{X}$ no matter where it is. Besides, if we assume that the numerical uncertainty approaches to zero (i.e., $\sigma^2_{\hat{P}_{f,n}}(\boldsymbol{x}) \to 0^+$ and $m_{\hat{P}_{f,n}}(\boldsymbol{x}) \to g(\boldsymbol{x})$), then there exits $\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right) \to I(\boldsymbol{x})$. In this regard, Eq. (7.1) can be seen as a limiting case of Eq. (7.23) when the numerical uncertainty disappears.

Remark 5. From the Bayesian perspective, the computation of failure probability estimate can be interpreted as a process aiming at reducing the numerical uncertainty that prevents us from inferring the true value. Therefore, an optimal inference about the failure probability requires an optimal decision on where to observe the g-function that leads to maximum reduction of the numerical uncertainty on the failure probability with as less g-function evaluations as possible.

7.3 Parallel adaptive-Bayesian failure probability learning

This section presents a novel method, termed 'parallel adaptive–Bayesian failure probability learning' (PA-BFPL), to make inference about the failure probability. The proposed method builds upon the BFPI framework, and aims at producing a reasonably accurate failure probability estimate using a limited number of observations from the g-function. This objective is achieved mainly by developing a variance-amplified importance sampling (VAIS) method and an adaptive parallel active learning (APAL) strategy, as described in what follows.

7.3.1 Variance-amplified importance sampling

In the BFPI framework, one open task consists of approximating the analytically intractable integrals ($m_{\hat{P}_{f,n}}$ in Eq. (7.23) and $\sigma^2_{\hat{P}_{f,n}}$ in Eq. (7.24)). The most straightforward solution would be to use the crude MCS due to its broad applicability. However, a considerably large number of samples are needed to achieve a reasonable accuracy in certain conditions, which can make each iteration of our method time-consuming and even cause the problem of computer memory loss. Taking $m_{\hat{P}_{f,n}}$ as an example, $\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right)$ might be small where $f_{\boldsymbol{X}}(\boldsymbol{x})$ is large and vice versa. In

these cases, directly sampling according to $f_{\boldsymbol{X}}(\boldsymbol{x})$ could be less efficient. If we turn to importance sampling, the optimal sampling density should be proportional to $\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right) f_{\boldsymbol{X}}(\boldsymbol{x})$, but is not practically achievable since it requires knowledge of the quantity we are trying to estimate. Similar problems also exist for $\sigma^2_{\hat{P}_{f,n}}$, and we will not repeat too much herein.

The present study proposes a VAIS technique to assess $m_{\hat{P}_{f,n}}$ and $\sigma^2_{\hat{P}_{f,n}}$. Let us reformulate $m_{\hat{P}_{f,n}}$ and $\sigma^2_{\hat{P}_{f,n}}$ as follows:

$$\begin{aligned} m_{\hat{P}_{f,n}} &= \int_{\mathcal{X}} \Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \\ &= \int_{\mathcal{X}} \Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right) \frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})} h_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}, \end{aligned} \tag{7.25}$$

$$\begin{aligned} \sigma^2_{\hat{P}_{f,n}} &= \int_{\mathcal{X}}\int_{\mathcal{X}} \left[F\left([0\ 0]; \boldsymbol{m}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}'), \boldsymbol{C}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}')\right) - \Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right)\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x}')}{\sigma_{\hat{g}_n}(\boldsymbol{x}')}\right)\right] f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}' \\ &= \int_{\mathcal{X}}\int_{\mathcal{X}} \left[F\left([0\ 0]; \boldsymbol{m}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}'), \boldsymbol{C}_{\hat{g}_n}(\boldsymbol{x},\boldsymbol{x}')\right) - \Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x})}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right)\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x}')}{\sigma_{\hat{g}_n}(\boldsymbol{x}')}\right)\right] \\ &\quad \frac{f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}')}{h_{\boldsymbol{X}}(\boldsymbol{x}) h_{\boldsymbol{X}}(\boldsymbol{x}')} h_{\boldsymbol{X}}(\boldsymbol{x}) h_{\boldsymbol{X}}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}', \end{aligned} \tag{7.26}$$

where $h_{\boldsymbol{X}}(\boldsymbol{x})$ is the so-called 'importance sampling density' (ISD). In this study, we do not intend to approach a nearly optimal ISD (whose formulation may be challenging), yet a simple but effective one. The concept of increasing the variances of random variables has been used in the different contexts, such as [25–28]. In particular, it has been reported in [29] that such an approach was used within Importance Sampling as early as 1983. Following those ideas, the ISD $h_{\boldsymbol{X}}(\boldsymbol{x})$ is simply constructed by amplifying the standard deviations $\boldsymbol{\sigma}_{\boldsymbol{X}}$ (or equivalently variances $\boldsymbol{\sigma}^2_{\boldsymbol{X}}$) of $f_{\boldsymbol{X}}(\boldsymbol{x})$ (keep the mean $\boldsymbol{m}_{\boldsymbol{X}}$ unchanged), i.e., $h_{\boldsymbol{X}}(\boldsymbol{x}) = f_{\boldsymbol{X}}(\boldsymbol{x}; \boldsymbol{m}_{\boldsymbol{X}}, \alpha\boldsymbol{\sigma}_{\boldsymbol{X}})$, where $\alpha \geq 1$ denotes the amplification factor of standard deviation. Note that for any X_i that follows a uniform distribution, its standard deviation does not need to be enlarged. Besides, one can use different amplification factors for different random variables, but for the sake of convenience we just consider a single amplification factor in this work. The unbiased VAIS estimators of $m_{\hat{P}_{f,n}}$ and $\sigma^2_{\hat{P}_{f,n}}$ are simply given as their sample means:

$$\hat{m}_{\hat{P}_{f,n}} = \frac{1}{N_1}\sum_{i=1}^{N_1}\left[\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x}^{(i)})}{\sigma_{\hat{g}_n}(\boldsymbol{x}^{(i)})}\right)\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}\right], \tag{7.27}$$

$$\hat{\sigma}^2_{\hat{P}_{f,n}} = \frac{1}{N_2}\sum_{j=1}^{N_2}\left[F\left([0\ 0]; \boldsymbol{m}_{\hat{g}_n}(\boldsymbol{x}^{(j)}, \boldsymbol{x}'^{,(j)}), \boldsymbol{C}_{\hat{g}_n}(\boldsymbol{x}^{(j)}, \boldsymbol{x}'^{,(j)})\right) - \Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x}^{(j)})}{\sigma_{\hat{g}_n}(\boldsymbol{x}^{(j)})}\right)\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x}'^{,(j)})}{\sigma_{\hat{g}_n}(\boldsymbol{x}'^{,(j)})}\right)\right] \times \frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(j)}) f_{\boldsymbol{X}}(\boldsymbol{x}'^{,(j)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(j)}) h_{\boldsymbol{X}}(\boldsymbol{x}'^{,(j)})}, \tag{7.28}$$

where $\left\{\boldsymbol{x}^{(i)}\right\}_{i=1}^{N_1}$ is a set of N_1 random samples generated according to $h_{\boldsymbol{X}}(\boldsymbol{x})$; $\left\{\boldsymbol{x}^{(j)}\right\}_{j=1}^{N_2}$ and $\left\{\boldsymbol{x}'^{,(j)}\right\}_{j=1}^{N_2}$ are two sets of N_2 random samples generated according to $h_{\boldsymbol{X}}(\boldsymbol{x})$ and $h_{\boldsymbol{X}}(\boldsymbol{x}')$ respectively. The variances of the VAIS estimators $\hat{m}_{\hat{P}_{f,n}}$ and $\hat{\sigma}^2_{\hat{P}_{f,n}}$ are given by:

$$\mathrm{Var}\left[\hat{m}_{\hat{P}_{f,n}}\right] = \frac{1}{N_1(N_1-1)}\sum_{i=1}^{N_1}\left[\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x}^{(i)})}{\sigma_{\hat{g}_n}(\boldsymbol{x}^{(i)})}\right)\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})} - \hat{m}_{\hat{P}_{f,n}}\right]^2, \tag{7.29}$$

$$\mathrm{Var}\left[\hat{\sigma}^2_{\hat{P}_{f,n}}\right] = \frac{1}{N_2(N_2-1)}\sum_{j=1}^{N_2}\left\{\left[F\left([0\ 0]; \boldsymbol{m}_{\hat{g}_n}(\boldsymbol{x}^{(j)}, \boldsymbol{x}'^{,(j)}), \boldsymbol{C}_{\hat{g}_n}(\boldsymbol{x}^{(j)}, \boldsymbol{x}'^{,(j)})\right) - \Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x}^{(j)})}{\sigma_{\hat{g}_n}(\boldsymbol{x}^{(j)})}\right)\Phi\left(\frac{-m_{\hat{g}_n}(\boldsymbol{x}'^{,(j)})}{\sigma_{\hat{g}_n}(\boldsymbol{x}'^{,(j)})}\right)\right] \times \frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(j)}) f_{\boldsymbol{X}}(\boldsymbol{x}'^{,(j)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(j)}) h_{\boldsymbol{X}}(\boldsymbol{x}'^{,(j)})} - \hat{\sigma}^2_{\hat{P}_{f,n}}\right\}^2, \tag{7.30}$$

where $\mathrm{Var}[\cdot]$ means to take variance of its argument.

When $\alpha = 1$, the proposed VAIS method reduces to crude MCS. In case that $\alpha > 1$, $h_{\boldsymbol{X}}(\boldsymbol{x})$ can be viewed as an auxiliary sampling density formed by redistributing the density of $f_{\boldsymbol{X}}(\boldsymbol{x})$. Typically, $h_{\boldsymbol{X}}(\boldsymbol{x})(\alpha > 1)$ is more dispersedly distributed than $f_{\boldsymbol{X}}(\boldsymbol{x})$ over $\mathcal{X}$. As an illustration, Fig. 7.2 compares the density change of a standard normal density $\varphi(x)$ before and after its variance is amplified, where two amplification factors are considered, i.e., $\alpha = 1.5, 2.0$. It is shown that as α increases, $h_{\boldsymbol{X}}(\boldsymbol{x})$ becomes more flatter than $\varphi(x)$, and hence enlarges density where $\varphi(x)$ is small, while lowers the density where $\varphi(x)$ is large. Consequently, the variance amplification will have an effect on random sampling. To be specific, the random samples generated from $h_{\boldsymbol{X}}(\boldsymbol{x})(\alpha > 1)$ are more dispersedly distributed than those of $f_{\boldsymbol{X}}(\boldsymbol{x})$ over $\mathcal{X}$. If we take $f_{\boldsymbol{X}}(\boldsymbol{x}) \sim \varphi(x_1)\varphi(x_2)$ as an example, the random samples generated before and after variance amplification are depicted in Fig. 7.3, where two cases (i.e., $\alpha = 1.5, 2.0$) are also considered. As can been seen, as the amplification factor increases, the random samples will reach the area where $f_{\boldsymbol{X}}(\boldsymbol{x})$ is relatively small. Thus, sampling from $h_{\boldsymbol{X}}(\boldsymbol{x})$ instead of $f_{\boldsymbol{X}}(\boldsymbol{x})$ could alleviate some of limitations discussed at the beginning of this subsection. The effect of variance amplification on random sampling is also useful for our APAL strategy (see next subsection). The optimal α values for $\hat{m}_{\hat{P}_{f,n}}$ and $\hat{\sigma}^2_{\hat{P}_{f,n}}$

could be determined by minimizing their corresponding variances (Eqs. (7.29) and (7.30)), which, however, is still a tricky task. To determine the appropriate sample sizes N_1 and N_2 for $\hat{m}_{\hat{P}_{f,n}}$ and $\hat{\sigma}^2_{\hat{P}_{f,n}}$, one can first assign them two small values, and then gradually increase the sample sizes until $\text{Var}\left[\hat{m}_{\hat{P}_{f,n}}\right]$ and $\text{Var}\left[\hat{\sigma}^2_{\hat{P}_{f,n}}\right]$ are acceptable.

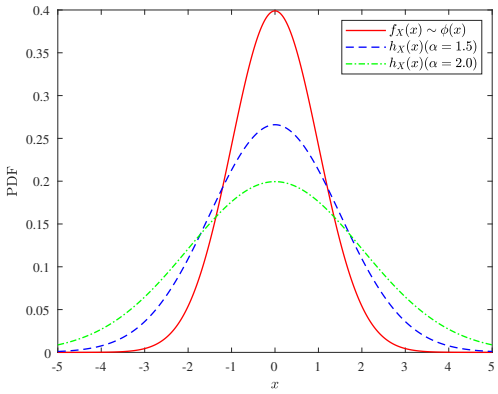
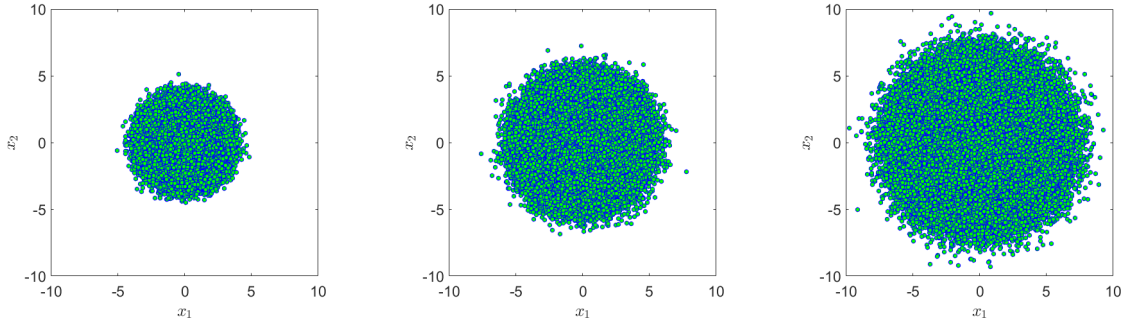


Figure 7.2: Illustration the effects of variance amplification from density change.



(a) (b) (c)

Figure 7.3: Illustration the effects of variance amplification from random sampling: (a) 10^6 random samples drawn from $f_{\boldsymbol{X}}(\boldsymbol{x}) \sim \varphi(x_1)\varphi(x_2)$; (b) 10^6 random samples drawn from $h_{\boldsymbol{X}}(\boldsymbol{x})$ ($\alpha = 1.5$); (c) 10^6 random samples drawn from $h_{\boldsymbol{X}}(\boldsymbol{x})$ ($\alpha = 2.0$).

Remark 6. As a common limitation in Importance Sampling [30], the proposed VAIS method could not be directly applied to high-dimensional problems (e.g., larger than 20). Besides, a premise of the proposed VAIS method is that all random variables process variances. In case that there exist a random variable without variance (e.g., Cauchy distribution), some pre-processing steps are needed in order to apply the VAIS method, e.g., transforming it to a random variable with variance if possible.

Remark 7. In [1], the authors developed a importance ball sampling (IBS) method to approximate $m_{\hat{P}_{f,n}}$ and an upper bound of $\sigma^2_{\hat{P}_{f,n}}$. However, the method is biased in nature and has to work in the standard normal space.

Remark 8. It is interesting to note that VAIS itself is a purely frequentist approach. As in many Bayesian methodologies, the frequentist methods also play a significant role [31].

7.3.2 Adaptive parallel active learning strategy

Another issue to be solved in the BFPI framework is how to select design points $\boldsymbol{\mathcal{X}}$, which is commonly known as design of experiments (DOE). Although the BFPI framework itself does not impose any restrictions on the DOE, an optimal DOE can yield an accurate estimate for the failure probability with a minimum number of g-function evaluations. In view of this, we propose an APAL strategy to sequentially select a batch of points, which attempts to make the fullest possible use of all previous g-function evaluations and parallel computing simultaneously. The core of the APAL strategy is a weighted clustering technique.

Considering the posterior distribution $\hat{g}_n$ defined in Eq. (7.13), the probability of making a wrong prediction on the sign of g at $\boldsymbol{x}$ is given by [20]:

$$\pi(\boldsymbol{x}) = \Phi\left(\frac{-|m_{\hat{g}_n}(\boldsymbol{x})|}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right). \tag{7.31}$$

For simplicity, we refer to $\pi(\boldsymbol{x})$ as the probability of misclassification (POM). The well known U function [20] (i.e., $U(\boldsymbol{x}) = \frac{|m_{\hat{g}_n}(\boldsymbol{x})|}{\sigma_{\hat{g}_n}(\boldsymbol{x})}$) is proposed based on the concept of POM, and the best next point to evaluate on the g-function is identified by minimizing the U function (equivalently maximizing the POM). However, only the misclassification probability at a single point that minimizes the U function is considered, without taking other points and the probability distribution information of $\boldsymbol{X}$ into account. This may lead to underutilization of useful information and is not suitable for parallel distributed processing.

To overcome these limitations, a new concept, called 'expected misclassification probability' (EMP), is first introduced as follows:

$$\mathit{\Pi} = \mathbb{E}_{\boldsymbol{X}}\left[\pi(\boldsymbol{X})\right] = \int_{\mathcal{X}} \Phi\left(\frac{-|m_{\hat{g}_n}(\boldsymbol{x})|}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right) f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}, \tag{7.32}$$

which is actually defined as an expectation of the POM $\pi(\boldsymbol{x})$ under the density $f_{\boldsymbol{X}}(\boldsymbol{x})$. Hence, the EMP can be interpreted as the posterior expected probability that $\hat{g}_n$ makes a mistake on the sign of g. In order to have an accurate failure probability estimate, an alternative way is to reduce $\mathit{\Pi}$ instead of the maximum value of $\pi(\boldsymbol{x})$.

Let us rewrite $\mathit{\Pi}$ with respect to $h_{\boldsymbol{X}}(\boldsymbol{x})$ as:

$$\mathit{\Pi} = \int_{\mathcal{X}} \Phi\left(\frac{-|m_{\hat{g}_n}(\boldsymbol{x})|}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right) \frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})} h_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}. \tag{7.33}$$

To reduce $\mathit{\Pi}$ defined in Eq. (7.33), one potential solution is to find out the locations that contribute the most to $\mathit{\Pi}$. Here, we introduce a measure (i.e., the proposed learning function), called 'expected misclassification probability contribution' (EMPC), as follows:

$$\mathrm{EMPC}(\boldsymbol{x}) = \Phi\left(\frac{-|m_{\hat{g}_n}(\boldsymbol{x})|}{\sigma_{\hat{g}_n}(\boldsymbol{x})}\right) \frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}. \tag{7.34}$$

It is straightforward to observe that the EMPC function provides a natural measure of the contribution of the misclassification probability at $\boldsymbol{x}$ to $\mathit{\Pi}$, where $\boldsymbol{x} \sim h_{\boldsymbol{X}}(\boldsymbol{x})$. Besides, one should note that the probability density $f_{\boldsymbol{X}}(\boldsymbol{x})$ is properly included in the EMPC function.

Now, we consider the problem of how to identify a batch of informative points among a set of points generated from $h_{\boldsymbol{X}}(\boldsymbol{x})$, e.g., $\left\{\boldsymbol{x}^{(l)}\right\}_{l=1}^{N_3}$. This objective is realized by developing a weighted clustering algorithm, called 'EMPC-weighted k-means clustering'. As its name indicates, the proposed algorithm actually combines EMPC with k-means clustering [32]. As mentioned before, the EMPC function can measure the contribution of the misclassification probability at $\boldsymbol{x}^{(l)}$ to $\mathit{\Pi}$. On the other hand, the k-means clustering algorithm can partition a dataset into k clusters that are represented by k centroids. However, the conventional k-means clustering algorithm does not account for the weight information of data. The proposed EMPC-weighted k-means clustering enables to identify k centroids by using the data $\left\{\boldsymbol{x}^{(l)}\right\}_{l=1}^{N_3}$ while considering their EMPC values as weights. The k centroids correspond to the batch of points we wish to select. Once the k points are obtained, computation of the corresponding g-function values can be distributed on k CPU cores simultaneously. A compact pseudocode of the proposed algorithm is given in Algorithm 5.

The reason why we introduce the ISD $h_{\boldsymbol{X}}(\boldsymbol{x})$ to Eq. (7.33) (and hence in Eq. (7.34)) is because with the same sample size $h_{\boldsymbol{X}}(\boldsymbol{x})$ can generate much more dispersed samples than $f_{\boldsymbol{X}}(\boldsymbol{x})$, making it

Algorithm 5 Proposed EMPC-weighted k-means clustering algorithm

Input: The EMPC function, number of clusters k and dataset $\left\{\boldsymbol{x}^{(l)}\right\}_{l=1}^{N_3}$

1. **Initialization**. Randomly choose k points from the dataset $\left\{\boldsymbol{x}^{(l)}\right\}_{l=1}^{N_3}$ as the initial centroids, denoted by $\boldsymbol{S}=\left\{\boldsymbol{s}^{(i)}\right\}_{i=1}^{k}$;

2. **Assignment step**. Assign each point among the dataset $\left\{\boldsymbol{x}^{(l)}\right\}_{l=1}^{N_3}$ to the nearest cluster: that with the least squared Euclidean distance. The i-th cluster is denoted as $\boldsymbol{R}^{(i)}=\left\{\boldsymbol{r}_j^{(i)}\right\}_{j=1}^{N^{(i)}}$, where $\boldsymbol{r}_j^{(i)}$ is the j-th point in the i-th cluster and $N^{(i)}$ is the number of points in the i-th cluster;

3. **Update step**. The i-th centroid is updated by the EMPC-weighted mean of the points belonging to i-th cluster:

$$\boldsymbol{s}^{(i)}=\frac{\sum_{j=1}^{N^{(i)}}\mathrm{EMPC}(\boldsymbol{r}_j^{(i)})\times\boldsymbol{r}_j^{(i)}}{\sum_{j=1}^{N^{(i)}}\mathrm{EMPC}(\boldsymbol{r}_j^{(i)})}$$

4. **Iteration**. Repeat steps 2 and 3 until the centroids do not change or the pre-specified number of iterations is reached.

Output: k centroids

possible to reach the failure domain characterized with a small failure probability. Besides, by doing so, the random samples generated for evaluating, e.g., $\hat{m}_{\hat{P}_{f,n}}$, can be reused in the proposed weighted clustering algorithm. To illustrate the proposed weighted clustering method, let us consider the case that: $\mathrm{EMPC}(\boldsymbol{x})=\Phi\left(-\left(x_1^2+x_2^4-4\right)^2\right)\frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})}$, $f_{\boldsymbol{X}}(\boldsymbol{x})\sim\varphi(x_1)\varphi(x_2)$, $\alpha=1.5$, $N_3=10^6$ and $k=5$. As shown in Fig. 7.4, the identified points are sparsely located in the region with relatively large EMPC values, and hence informative in our context.

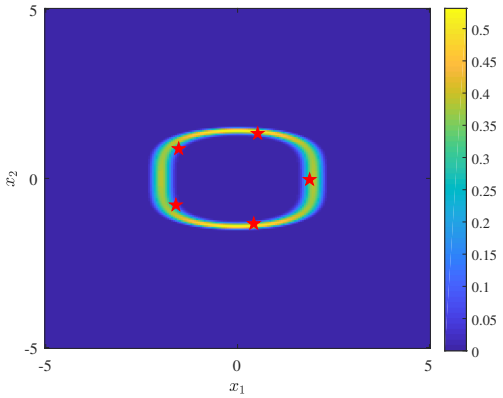


Figure 7.4: Five points identified by the proposed weighted clustering algorithm: the colormap represents the EMPC values and the pentagrams denote the identified centroids.

Remark 9. For reducing the numerical uncertainty of $\hat{P}_{f,n}(\varpi)$, one obvious way is to minimize its variance $\sigma^2_{\hat{P}_{f,n}}$. However, the variance itself is analytically intractable, in contrast to the proposed EMPC function.

Remark 10. The basic idea of proposed APAL is similar to the one in [1], while different learning functions are used.

7.3.3 Numerical implementation procedure

The numerical implementation procedure of the proposed PA-BFPL consists of the following main steps:

Step 1: Generate samples according to the ISD $h_{\boldsymbol{X}}(\boldsymbol{x})$

In order to approximate $m_{\hat{P}_{f,n}}$ and $\sigma^2_{\hat{P}_{f,n}}$ by the proposed VAIS method, random samples need to be generated according to the ISD $h_{\boldsymbol{X}}(\boldsymbol{x})$. First, draw a set of N_1 random samples from $h^{(1)}_{\boldsymbol{X}}(\boldsymbol{x})$ (that corresponds to α_1), denoted by $\left\{\boldsymbol{x}^{(i)}\right\}_{i=1}^{N_1}$. Then, draw two sets of N_2 random samples from $h^{(2)}_{\boldsymbol{X}}(\boldsymbol{x})$ and $h^{(2)}_{\boldsymbol{X}}(\boldsymbol{x}')$ (that correspond to α_2) respectively, denoted by $\left\{\boldsymbol{x}^{(j)}\right\}_{j=1}^{N_2}$ and $\left\{\boldsymbol{x}'^{,(j)}\right\}_{j=1}^{N_2}$.

The reason why we introduce two amplification factors α_1 and α_2 is because $\sigma^2_{\hat{P}_{f,n}}$ is much more time consuming to evaluate than $m_{\hat{P}_{f,n}}$. By doing so, we can use a larger α_2 and hence a smaller N_2 for $\sigma^2_{\hat{P}_{f,n}}$ in order to save computational time. Once α_1 and α_2 are properly chosen, one can specify N_1 and N_2 either adaptively or as large as possible.

Step 2: Obtain an initial dataset $\mathscr{D}$ from the g-function

To perform the proposed PA-BFPL method, an initial dataset observed from the g-function is required. First, generate a set of n_0 samples from $f_{\boldsymbol{X}}(\boldsymbol{x})$ by using Latin hypercube sampling (LHS), which is denoted by $\mathscr{X} = \left\{\boldsymbol{x}^{(l)}\right\}_{l=1}^{n_0}$. Then, these points are evaluated on the g-function in parallel, and the corresponding observations are denoted by $\mathscr{Y} = \left\{y^{(l)}\right\}_{l=1}^{n_0}$ with $y^{(l)} = g(\boldsymbol{x}^{(l)})$. At last, the initial dataset is constructed by $\mathscr{D} = \{\mathscr{X}, \mathscr{Y}\}$. Let $n = n_0$.

Step 3: Compute the posterior mean and variance of P_f

The posterior distribution of g conditional on $\mathscr{D}$ can be inferred as Eq. (7.13), which mainly involves learning the hyper-parameters using maximum likelihood estimation. This in turn implies posterior distributions over I and P_f. The posterior mean $m_{\hat{P}_{f,n}}$ and variance $\sigma^2_{\hat{P}_{f,n}}$ are approximated by the proposed VAIS method (Eqs. (7.27) and (7.28)) using samples $\left\{\boldsymbol{x}^{(i)}\right\}_{i=1}^{N_1}$, $\left\{\boldsymbol{x}^{(j)}\right\}_{j=1}^{N_2}$ and

$\left\{\boldsymbol{x}'^{,(j)}\right\}_{j=1}^{N_2}$, respectively.

Step 4: Check the stopping criterion

A stopping criterion that determines when to stop the iteration is needed. In this study, we present a hybrid convergence measure consisting of two indices. The first index is defined as the relative error of $\hat{m}_{\hat{P}_{f,n}}$ between two consecutive iterations:

$$e_1 = \frac{|\hat{m}^{(q)}_{\hat{P}_{f,n}} - \hat{m}^{(q-1)}_{\hat{P}_{f,n-k}}|}{\hat{m}^{(q-1)}_{\hat{P}_{f,n-k}}}, \tag{7.35}$$

where $\hat{m}^{(q-1)}_{\hat{P}_{f,n-k}}$ and $\hat{m}^{(q)}_{\hat{P}_{f,n}}$ are the estimated failure probabilities at the $(q-1)$-th and q-th iterations, respectively. The first index e_1 can measure the stability of the estimated failure probability. The estimated posterior coefficient of variation (COV) of the failure probability is considered as the second index such that:

$$e_2 = \frac{\hat{\sigma}^{(q)}_{\hat{P}_{f,n}}}{\hat{m}^{(q)}_{\hat{P}_{f,n}}}, \tag{7.36}$$

where $\hat{m}^{(q)}_{\hat{P}_{f,n}}$ and $\hat{\sigma}^{(q)}_{\hat{P}_{f,n}}$ represent the estimated posterior mean and standard deviation of the failure probability at the q-th iteration. The second index e_2 implies the level of epistemic uncertainty of the failure probability estimate. Based on these two indices, this step proceeds as follows:

If both $e_1 < \epsilon_1$ and $e_2 < \epsilon_2$ are satisfied twice in succession, go to **Step 6**; Else, go **Step 5**. Here ϵ_1 and ϵ_2 are two user-specified thresholds. 'Twice in succession' is adopted here to avoid possible fake convergence.

Step 5: Enrich the dataset by the proposed APAL

This stage consists of identifying k new points $\boldsymbol{\mathcal{X}}^+ = \left\{\boldsymbol{x}^{+,(i)}\right\}_{i=1}^{k}$ from $\left\{\boldsymbol{x}^{(i)}\right\}_{i=1}^{N_1}$ using the proposed APAL (i.e., EMPC-weighted k-means clustering). Then, the g-function is evaluated in parallel at $\boldsymbol{\mathcal{X}}^+$ to produce the corresponding observations $\boldsymbol{\mathcal{Y}}^+ = \left\{y^{+,(i)}\right\}_{i=1}^{k}$ with $y^{+,(i)} = g(\boldsymbol{x}^{+,(i)})$. Finally, the previous dataset is enriched by $\boldsymbol{\mathcal{D}}^+ = \left\{\boldsymbol{\mathcal{X}}^+, \boldsymbol{\mathcal{Y}}^+\right\}$, i.e., $\boldsymbol{\mathcal{D}} = \boldsymbol{\mathcal{D}} \cup \boldsymbol{\mathcal{D}}^+$. Let $n = n + k$, and go to **Step 3**.

Step 6: End the algorithm

The proposed method stops and the last failure probability estimate is considered as the result of the method.

There remain several parameters in the proposed algorithm to be specified. In all numerical examples of this study, unless otherwise specified these parameters except for k are set to: $N_1 = 10^6$, $N_2 = 5 \times 10^4$, $\alpha_1 = 1.6$, $\alpha_2 = 2.1$, $n_0 = 10$, $\epsilon_1 = 15\%$ and $\epsilon_2 = 5\%$. The parameter k will be varied to test the performance of the proposed method.

7.4 Numerical examples

The performance of the proposed PA-BFPL method will be illustrated in this section by means of four numerical examples. These examples cover a variety of problems with varying dimensions, non-linearity and failure probabilities, etc. The reference results for the target failure probabilities are provided by MCS when there is no (semi-) analytical solution. For comparison, AK-MCS [20], Active Learning Probabilistic Integration (ALPI) [22], Active learning Kriging Markov Chain Monte Carlo (AK-MCMC) [33] and other methods are also considered if applicable. In particular, the active learning reliability (ALR) method in UQLab (version 2.0.0) [34], denoted as ALR in UQLab 2.0.0, is compared to the proposed method in all four numerical examples. If not further specified, the ALR method runs with its default setting [35]. The efficiency of these methods is measured by the number of iterations N_{iter}, the total number of g-function calls N_{call}, while the accuracy is measured by the failure probability estimate $\hat{P}_f$ and its COV denoted by COV $\left[\hat{P}_f\right]$. Except for MCS and the (semi-) analytical method, these performance measures are estimated from the average results over 10 independent runs unless otherwise specified.

7.4.1 Example 1: A test problem with four beta points

The first example considers a test problem with multiple beta points, which is modified from [36]. The performance function is given by:

$$Y = g\left(X_1, X_2\right) = \beta^2 - |X_1 \cdot X_2|, \tag{7.37}$$

where β is a constant parameter, specified as 3; X_1 and X_2 are two standard normal variables. It is easy to know that the limit state surface $g\left(x_1, x_2\right) = 0$ has four beta points: (β, β), $(-\beta, \beta)$, $(\beta, -\beta)$ and $(-\beta, -\beta)$. Another characteristic of the problem is that the semi-analytic formula of

the failure probability can be derived as:

$$P_f = 1 - \frac{2}{\pi} \int_0^{\beta^2} K_0 \left(u \right) \mathrm{d}u, \tag{7.38}$$

where $K_0(\cdot)$ is the modified Bessel function of the second kind of order zero. By applying some numerical integration techniques, it is trivial to obtain the result of Eq. (7.38) with sufficient accuracy.

The proposed method with different k is implemented to assess the failure probability, together with several other existing methods. The results are summarized in Table. 7.1. It can be seen that the failure probability provided by the the semi-analytic formula accords well with that of MCS, and hence we take 3.09×10^{-5} as the reference result. The proposed method is able to yield unbiased estimates with COVs less than 3%. The two AK-MCS methods are less accurate than the proposed method in terms of the average failure probabilities and their COVs. On the other hand, the proposed method greatly outperforms the non-parallel counterparts (i.e., AK-MCS+U [20], AK-MCS+EFF [20] and AK-MCMC [33]) in terms of N_{iter}, especially compared with AK-MCMC. This implies that the proposed method can be much more efficient than those non-parallel counterparts when parallel computing is available. When it comes to the parallel counterpart, i.e., ALR in UQLab 2.0.0 [35], the proposed method needs slightly more computational efforts than it, regarding both N_{iter} and N_{call}. However, the ALR method produces obvious biases for the failure probabilities for all three cases (i.e., $k = 5, 10, 15$). For $k = 10, 15$, the COVs of the ALR method even reach up to 38.32% and 34.06% respectively. The reason for why the biased results are yielded could be due to the limitation of subset simulation used in ALR, as has been revealed in [36]. In view of these, the proposed method also shows better overall performance than the ALR method in this example.

For illustration purposes, Fig. 7.5 depicts the identified points resulted from an exemplary run of the proposed method ($k = 10$), along with the true limit state. It is shown that as the iteration goes on, the points identified by the proposed method gradually move towards the vicinity of the true limit state. Moreover, the selected points, at least most of them, are sparsely distributed in the design space, but slightly denser around the true beta points. These results imply that the proposed APAL strategy are informative for accurately inferring the failure probability.

Table 7.1: Results for Example 1 by different methods.

Method		N_{iter}	N_{call}	$\hat{P}_f$	COV$[\hat{P}_f]$/%
Semi-analytic		-	-	3.09×10^{-5}	-
MCS		-	10^9	3.09×10^{-5}	0.57
AK-MCS+U		$1 + 41.50 = 42.50$	$12 + 41.50 = 53.50$	3.13×10^{-5}	7.15
AK-MCS+EFF		$1 + 45.90 = 46.90$	$12 + 45.90 = 57.90$	3.03×10^{-5}	6.65
AK-MCMC		$1 + 100.90 = 101.90$	$12 + 100.90 = 112.90$	3.09×10^{-5}	0.72
	$k = 5$	$1 + 5.20 = 6.20$	$10 + 26.00 = 36.00$	1.64×10^{-5}	3.80
ALR in UQLab 2.0.0	$k = 10$	$1 + 4.30 = 5.30$	$10 + 43.00 = 53.00$	2.16×10^{-5}	38.32
	$k = 15$	$1 + 3.70 = 4.70$	$10 + 55.50 = 65.50$	2.19×10^{-5}	34.16
	$k = 5$	$1 + 7.40 = 8.40$	$10 + 37.00 = 47.00$	3.07×10^{-5}	2.55
Proposed PA-BFPL	$k = 10$	$1 + 5.20 = 6.20$	$10 + 52.00 = 62.00$	3.08×10^{-5}	1.41
	$k = 15$	$1 + 4.80 = 5.80$	$10 + 72.00 = 82.00$	3.07×10^{-5}	0.97

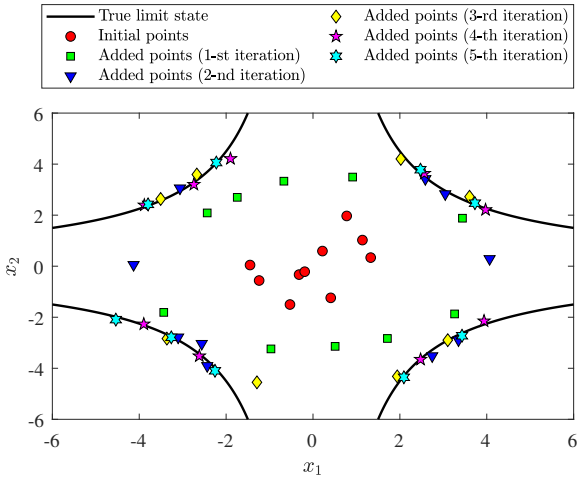


Figure 7.5: Selected points by the proposed method ($k = 10$) for Example 1.

7.4.2 Example 2: A series system with four branches

The second example consists of a series system with four branches, which has been extensively studied as a benchmark [9, 20, 22]. The performance function is given by:

$$Y = g(X_1, X_2) = \min \begin{cases} a + \frac{(X_1 - X_2)^2}{10} - \frac{(X_1 + X_2)}{\sqrt{2}}, \\ a + \frac{(X_1 - X_2)^2}{10} + \frac{(X_1 + X_2)}{\sqrt{2}}, \\ (X_1 - X_2) + \frac{b}{\sqrt{2}}, \\ (X_2 - X_1) + \frac{b}{\sqrt{2}} \end{cases}, \tag{7.39}$$

where a and b are two constant parameters, which can be used to adjust the failure probability; X_1 and X_2 are normally distributed with zero means and unit variances. In the following, two cases

are considered.

Case I: $a = 3$ and $b = 7$

In this case, the target failure probability is in the order of magnitude 10^{-3}, as indicated by the reference result from MCS (i.e., $\hat{P}_f = 2.22 \times 10^{-3}$ with COV $\left[\hat{P}_f\right] = 0.21\%$). The proposed method is also compared with several other methods, as listed in Table 7.2. It is found that all methods can give close average failure probabilities to the reference result with COVs less than 5%, except for ALR that processes COVs larger than 10%. The latter is due to the fact that ALR cannot always identify all failure domains. As for N_{iter}, the proposed method is significantly advantageous against these non-parallel methods (i.e., AK-MCS+U [20], AK-MCS+EFF [20] and ALPI [22]), and still fairly better than these parallel methods (i.e., ISKRA (KB) [37], ISKRA (k-means) [37] and ALR [35]). In addition, the average number of g-function calls required by the proposed method is also less than the other methods, especially when k is small, say $k = 5$. These results demonstrate the accuracy and efficiency of the proposed method in this case.

Table 7.2: Results of Example 2 (Case I) by different methods.

Method		N_{iter}	N_{call}	$\hat{P}_f$	COV$[\hat{P}_f]$/%
MCS		-	10^8	2.22×10^{-3}	0.21
AK-MCS+U		1 + 82.20 = 83.20	12 + 82.20 = 94.20	2.22×10^{-3}	1.35
AK-MCS+EFF		1 + 103.20 = 104.20	12 + 103.20 = 115.20	2.21×10^{-3}	1.20
ALPI		1 + 70.70 = 71.70	12 + 70.70 = 72.70	2.22×10^{-3}	2.25
ISKRA (KB) [37]	$k = 12$	1 + 6.68 = 7.68	12 + 80.16 = 92.16	2.23×10^{-3}	1.50
ISKRA (k-means) [37]	$k = 12$	1 + 8.62 = 9.62	12 + 103.44 = 115.44	2.22×10^{-3}	1.50
	$k = 5$	1 + 10.90 = 11.90	10 + 54.50 = 64.50	2.05×10^{-3}	11.95
ALR in UQLab 2.0.0	$k = 12$	1 + 6.40 = 7.40	10 + 76.80 = 86.80	2.07×10^{-3}	18.55
	$k = 15$	1 + 5.20 = 6.20	10 + 78.00 = 88.00	1.94×10^{-3}	17.92
	$k = 5$	1 + 6.50 = 7.50	10 + 32.50 = 42.50	2.13×10^{-3}	3.07
Proposed PA-BFPL	$k = 12$	1 + 4.30 = 5.30	10 + 51.60 = 61.60	2.24×10^{-3}	1.59
	$k = 15$	1 + 3.40 = 4.40	10 + 51.00 = 61.00	2.22×10^{-3}	1.08

Note: The results of ISKRA (KB) and ISKRA (k-means) are directly taken from [37], and they were averaged over 50 independent runs.

Fig. 7.6(a) shows the points selected by the proposed method ($k = 10$) with an exemplary run. It is observed that most of the points identified from iterations 1-5 are scattered in the vicinity of true limit state, indicating the effectiveness of the proposed APAL strategy.

Case II: $a = 5$ and $b = 9$

The second case is more challenging than the first one since the target failure probability is relatively small, i.e., in the order of 10^{-6} as provided by MCS with 10^{10} samples. Table 7.3 compares

the results from MCS, AK-MCMC [33], ALR [35] and the proposed method. As can be seen, fairly accurate results for such a small failure probability can still be produced by the proposed method with different k. Besides, the proposed method requires far less N_{iter} and N_{call} than those of AK-MCMC, especially for N_{iter}. The ALR method still produces biased results with considerably large COVs in this case, though it requires similar N_{iter} and N_{call} to those the proposed method. The results indicate that the proposed method is accurate and efficient in such a case.

Table 7.3: Results of Example 2 (Case II) by different methods.

Method		N_{iter}	N_{call}	$\hat{P}_f$	COV$[\hat{P}_f]$/%
MCS		-	10^{10}	7.09×10^{-6}	0.38
AK-MCMC		$1 + 127.50 = 128.50$	$12 + 127.50 = 139.50$	7.10×10^{-6}	1.37
	$k = 5$	$1 + 9.90 = 10.90$	$10 + 49.50 = 59.50$	4.82×10^{-6}	79.59
ALR in UQLab 2.0.0	$k = 10$	$1 + 6.70 = 7.70$	$10 + 67.00 = 77.00$	4.42×10^{-6}	81.79
	$k = 15$	$1 + 5.30 = 6.30$	$10 + 79.50 = 89.50$	6.50×10^{-6}	49.85
	$k = 5$	$1 + 10.00 = 11.00$	$10 + 50.00 = 60.00$	7.04×10^{-6}	2.17
Proposed PA-BFPL	$k = 10$	$1 + 5.80 = 6.80$	$10 + 58.00 = 68.00$	7.13×10^{-6}	2.01
	$k = 15$	$1 + 5.10 = 6.10$	$10 + 76.50 = 86.50$	7.06×10^{-6}	1.20

Once again, we depict the points selected at different stages of the proposed method ($k = 10$) via an exemplary run in Fig. 7.6(b). One can see that the identified points gradually approach to the four important parts of the true limit state that are relatively important for failure probability estimation. This demonstrates the effectiveness of the proposed method.

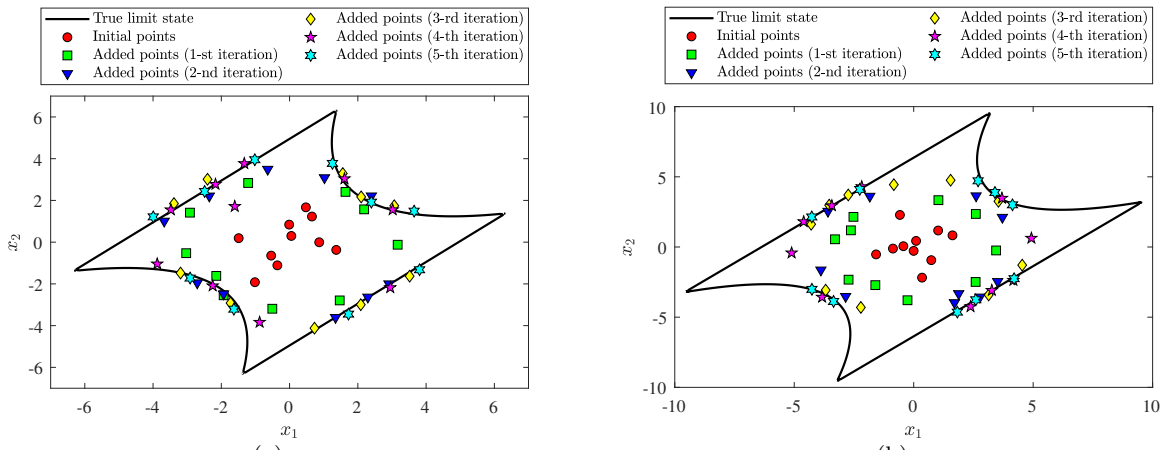


Figure 7.6: Selected points by the proposed method ($k = 10$) for Example 1: (a) Case I; (b) Case II.

7.4.3 Example 3: A slender column

The third example considers a sufficiently slender column subject to an axial compressive force [38], as shown in Fig. 7.7. The performance function corresponding to the buckling failure is given by:

$$Z = g(\boldsymbol{X}) = \frac{\pi^2 E}{L^2}\left\{\frac{\pi}{64}\left[(D+T)^4 - D^4\right]\right\} - P, \tag{7.40}$$

where $\boldsymbol{X} = [E, D, T, L, P]$, as listed in Table 7.4.

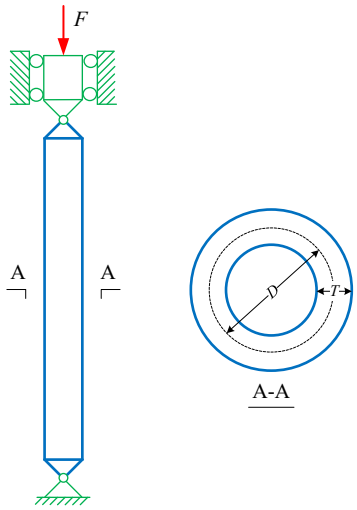


Figure 7.7: A slender column subject to an axial compressive force.

Table 7.4: Random variables of Example 3.

Variable	Description	Distribution	Mean	Standard deviation
E/Gpa	Young's modulus	Normal	203	30.45
D/mm	Dimension of the section	Normal	23.5	2.0
T/mm	Dimension of the section	Normal	4	1
L/mm	Height of the column	Normal	2500	50
P/N	Axial load	Lognormal	1000	200

The proposed method is compared in Table 7.5 with several other methods, i.e., MCS, AK-MCS+U [20], ALPI [22] and ALR [35]. The MCS with 10^7 samples can produce a failure probability estimate with a very small COV, and hence it is taken as a reference. The results of $\hat{P}_f$ and COV$[\hat{P}_f]$ show that AK-MCS+U, ALPI, ALR and the proposed method have similar accuracy. However, the proposed method is much more efficient than AK-MCS+U, ALPI and ALR in terms of N_{iter}. Besides, when $k = 5$ the proposed method also requires less calls to the g-function in average than

all those methods being compared. Overall, this example demonstrates the potential high-efficiency advantage of PA-BFPL when parallel computing facilities are available.

Table 7.5: Reliability results of Example 3 by different methods.

Method		N_{iter}	N_{call}	$\hat{P}_f$	COV$[\hat{P}_f]$/%
MCS		-	10^7	5.80×10^{-3}	0.41
AK-MCS+U		1 + 68.00 = 69.00	12 + 68.00 = 80.00	5.76×10^{-3}	1.74
ALPI		1 + 40.50 = 41.50	12 + 40.50 = 42.50	5.71×10^{-3}	1.56
	$k = 5$	1 + 17.10 = 18.10	10 + 85.50 = 95.50	5.97×10^{-3}	1.79
ALR in UQLab 2.0.0	$k = 10$	1 + 10.20 = 11.20	10 + 102.00 = 112.00	5.90×10^{-3}	3.08
	$k = 15$	1 + 6.90 = 7.90	10 + 103.50 = 113.50	5.94×10^{-3}	2.08
	$k = 5$	1 + 5.40 = 6.40	10 + 27.00 = 37.00	5.71×10^{-3}	1.05
Proposed PA-BFPL	$k = 10$	1 + 3.70 = 4.70	10 + 37.00 = 47.00	5.74×10^{-3}	1.50
	$k = 15$	1 + 3.00 = 4.00	10 + 45.00 = 55.00	5.77×10^{-3}	0.97

7.4.4 Example 4: A transmission tower

To illustrate the practical applicability of the proposed method, a transmission tower structure subject to horizontal loads (Fig. 7.8) is considered as the last example, which is modified from [39]. The structure is modelled as a three-dimensional (3-D) truss using the finite element software OpenSees. The finite element model consists of 24 joints and 80 truss members. As schematized in Fig. 7.8(c), the constitute law of the material adopts the bi-linear model. Ten horizontal forces along the x-axis are applied to the structure, which are shown in Fig. 7.8(a) and 7.8(b). The performance function is defined as:

$$Y = g(\boldsymbol{X}) = \Delta - U(P_1, P_2, P_3, P_4, P_5, E, A, b, F_y), \tag{7.41}$$

where $U(\cdot)$ denotes the horizontal displacement at the top of the structure along the x-axis, which is a function of nine random variables (see Table 7.6); Δ is the threshold of U, specified as 50 mm in this study.

In this example, the reference failure probability is 6.25×10^{-4} with COV being 1.26%, which given by MCS with 10^7 samples. As summarized in Table 7.7, the proposed method is compared with several other methods, i.e., AK-MCS+U [20], ALPI [22], and ALR [35]. One can see that AK-MCS+U, ALPI and the proposed method can produce fairly good average failure probability estimates with small COVs (say less than 4%). However, the ALR method produces biased results for $k = 5, 10, 15$. When it comes to the computational efficiency, the proposed method outperforms

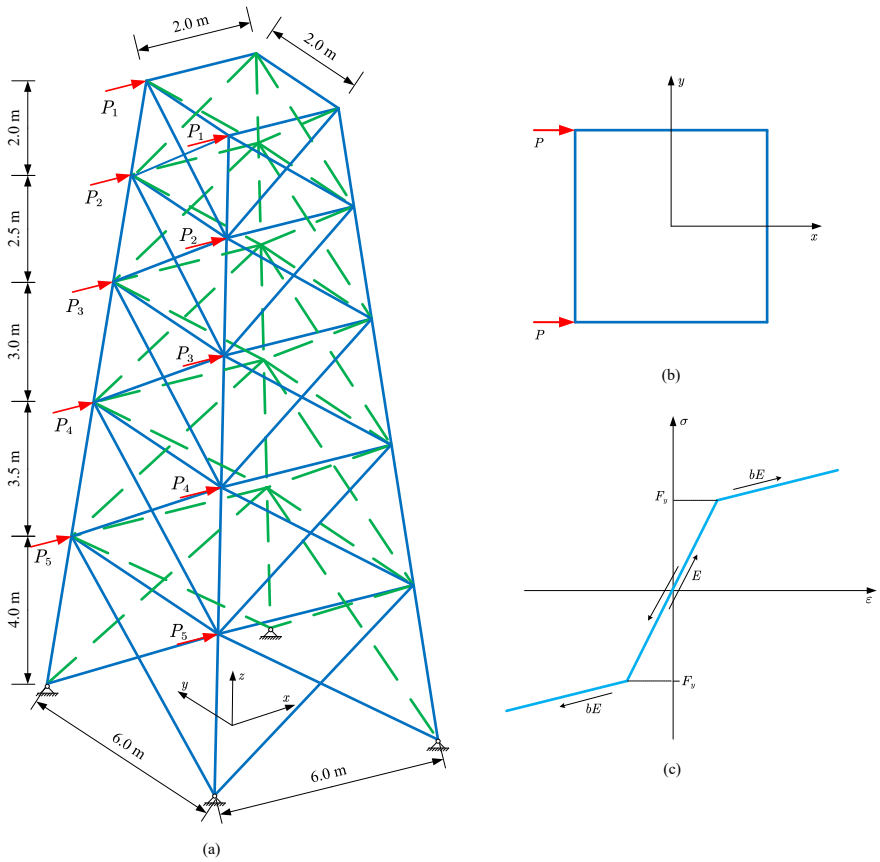


Figure 7.8: A transmission tower subject to horizontal loads: (a) 3-D finite element model; (b) Schematic diagram of load direction; (c) Schematic diagram of bi-linear constitute law.

other methods in terms of the average number of iterations N_{iter}. In addition, when k is small (e.g., 5), the average total number of calls N_{call} required by PA-BFPL is also slightly less than that of ALPI, and far less than those of AK-MCS+U and ALR.

7.4.5 Final remarks

As can be seen from the above numerical studies, the parameter k greatly affects the performance of the proposed PA-BFPL method, especially for efficiency. Typically, the average number of iterations decreases, while the average number of g-function evaluations when k increases from 5 to 15. However, this should not be regarded as a general conclusion because we only investigated three cases for k in each example. As a rule of thumb, one can choose a small k for non-parallel computing, whereas a large k when parallel computing is available.

Table 7.6: Random variables of Example 4.

Variable	Description	Distribution	Mean	COV
P_1/kN	Horizontal load	Lognormal	100	0.20
P_2/kN	Horizontal load	Lognormal	80	0.20
P_3/kN	Horizontal load	Lognormal	60	0.20
P_4/kN	Horizontal load	Lognormal	40	0.20
P_5/kN	Horizontal load	Lognormal	20	0.20
E/GPa	Young's modulus	Normal	200	0.15
A/mm^2	Cross-sectional area	Normal	5000	0.15
F_y/MPa	Yield stress	Normal	400	0.15
b	Strain-hardening ratio	Normal	0.02	0.10

Table 7.7: Reliability results of Example 4 by different methods.

Method		N_{iter}	N_{call}	$\hat{P}_f$	COV$[\hat{P}_f]$/%
MCS		-	10^7	6.25×10^{-4}	1.26
AK-MCS+U		1 + 68.00 = 69.00	12 + 113.80 = 125.80	6.17×10^{-4}	1.83
ALPI		1 + 46.60 = 47.60	12 + 46.60 = 58.60	6.12×10^{-4}	4.28
	$k = 5$	1 + 37.30 = 38.30	10 + 186.50 = 196.50	2.27×10^{-3}	225.88
ALR in UQLab 2.0.0	$k = 10$	1 + 18.80 = 19.80	10 + 188.00 = 198.00	2.50×10^{-2}	215.52
	$k = 15$	1 + 12.50 = 13.50	10 + 187.50 = 197.50	7.20×10^{-4}	25.03
	$k = 5$	1 + 6.80 = 7.80	10 + 34.00 = 44.00	6.32×10^{-4}	3.34
Proposed PA-BFPL	$k = 10$	1 + 4.90 = 5.90	10 + 49.00 = 59.00	6.30×10^{-4}	2.09
	$k = 15$	1 + 4.70 = 5.70	10 + 70.50 = 80.50	6.25×10^{-4}	1.78

Note: For most runs, the ALR method cannot converge even for $N_{call} > 200$. For this reason, the maximum value of N_{call} is set to be 200 for $k = 5, 10$, while 205 for $k = 15$.

7.5 Numerical investigation on the posterior distribution of failure probability

In addition to the posterior mean and variance, the posterior distribution of failure probability could be of interest for a complete Bayesian framework. For example, one can offer a confidence interval for the failure probability when the posterior distribution is available. However, it cannot be obtained analytically according to its definition (Eq. (7.22)). In this section, we attempt to numerically investigate the posterior distribution of failure probability through the four numerical examples given in the preceding section.

According to the proposed VAIS method, Eq. (7.22) can be rewritten as follows:

$$\hat{P}_{f,n}(\varpi) = \int_{\mathcal{X}} \hat{I}_n(\varpi, \boldsymbol{x}) \frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h_{\boldsymbol{X}}(\boldsymbol{x})} h_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}. \tag{7.42}$$

The reformulation actually allows us to evaluate the above integral numerically as:

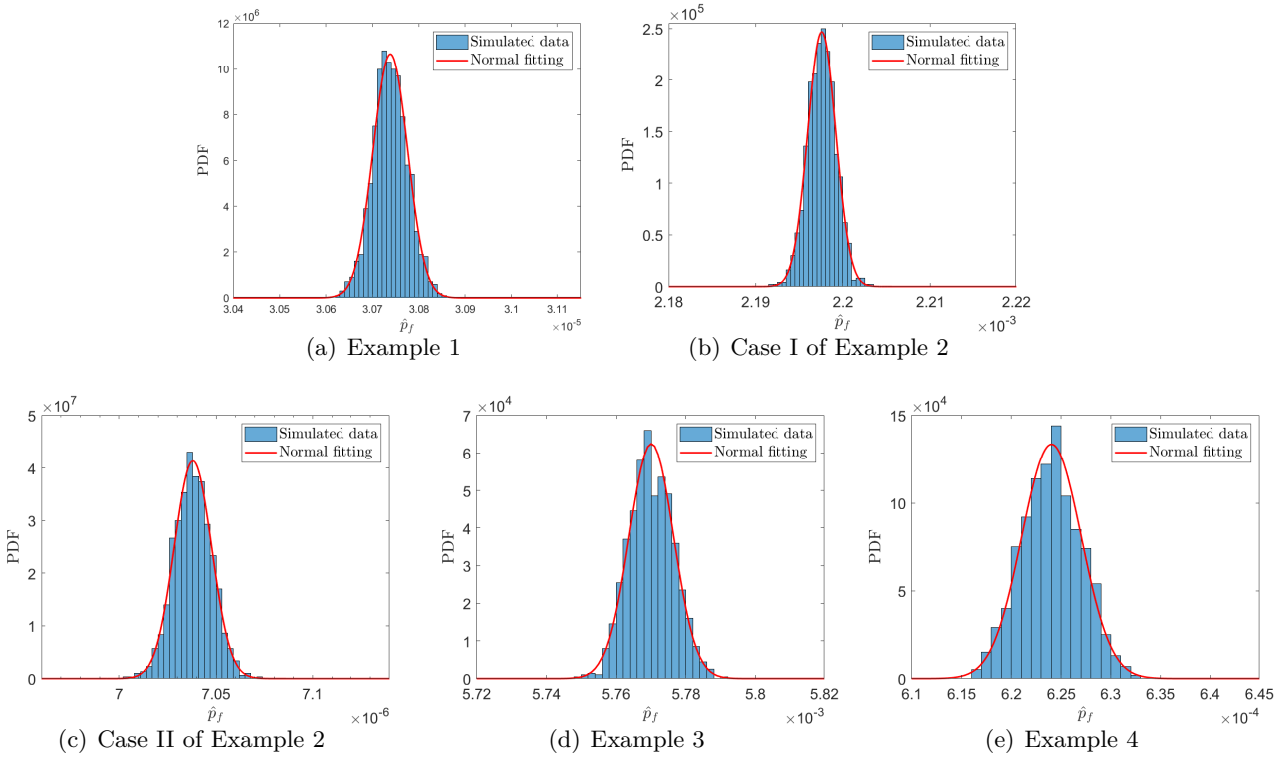
$$\hat{P}_{f,n}(\varpi) \approx \frac{1}{N_4}\sum_{i=1}^{N_4}\hat{I}_n(\varpi,\boldsymbol{x}^{(i)})\frac{f_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}{h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)})}h_{\boldsymbol{X}}(\boldsymbol{x}^{(i)}), \tag{7.43}$$

where $\left\{\boldsymbol{x}^{(i)}\right\}_{i=1}^{N_4}$ is a set of N_4 random samples generated according to $h_{\boldsymbol{X}}(\boldsymbol{x})$. Given that N_4 is sufficiently large, we can approximately generate random numbers for the posterior failure probability $\hat{P}_f(\varpi)$ via Eq. (7.43). The key is to sample from correlated Bernoulli random variables $\left\{\hat{I}_n(\varpi,\boldsymbol{x}^{(1)}),\hat{I}_n(\varpi,\boldsymbol{x}^{(2)}),\cdots,\hat{I}_n(\varpi,\boldsymbol{x}^{(N_4)})\right\}$ defined by Eq. (7.17). Nevertheless, this task is still challenging, especially when N_4 is large. For simplicity, the Bernoulli random variables are assumed to be independent and it is shown from some numerical experiments that this assumption does not affect our finial conclusion. Under these settings, posterior failure probability samples can be generated at each step of the proposed PA-BFPL method. To limit the length of the paper, only the results from the last step of the proposed method (an exemplary run, $k = 10$) are reported for those four numerical examples. The number of posterior failure probability samples is set to be 10^3. Other parameters are specified as: $\alpha = 2$ and $N_4 = 5 \times 10^5$.

The results of the normality tests for the simulated data of the posterior failure probabilities are depicted in Fig. 7.9. It is shown that the posterior failure probability samples can be well-modelled by normal distributions for all the cases studied. The results indicate that the posterior distribution of the failure probability might be approximated by a Gaussian distribution $\mathcal{N}(m_{\hat{P}_{f,n}},\sigma^2_{\hat{P}_{f,n}})$.

7.6 Concluding remarks

In the present paper, the task of failure probability estimation is interpreted from a perspective of Bayesian inference, in contrast to the classical frequentist inference. The proposed Bayesian failure probability inference (BFPI) framework regards the discretization error as a kind of epistemic uncertainty, and allows it to be properly modelled. To be specific, a prior Gaussian process is assumed for the performance function, a posterior distribution is then derived for the performance function, failure indicator function and failure probability conditional on observations arising from evaluating the performance function at a set of points. Numerical investigation indicates that the posterior failure probability could be approximated by a normal distribution. In addition, the parallel adaptive Bayesian failure probability learning (PA-BFPL) method is developed to make inference



(a) Example 1 (b) Case I of Example 2

(c) Case II of Example 2 (d) Example 3 (e) Example 4

Figure 7.9: Normality tests of the simulated data from the posterior failure probabilities: (a) Example 1; (b) Case I of Example 2; (c) Case II of Example 2; (d) Example 3; (e) Example 4.

about the failure probability within the Bayesian framework in a parallel adaptive scheme. The proposed PA-BFPL enables to make the fullest possible use of prior evaluations on the performance function evaluation, and can take advantage of parallel computing. Compared to several existing methods, the proposed method shows improved performance for structural reliability analysis regarding robustness, accuracy and efficiency. The advantage in computational efficiency is significant especially when parallel computing facilities are available.

The proposed PA-BFPL method is supposed to work well in linear, weakly nonlinear and moderately nonlinear problems with up to medium-dimensional random variables. For highly nonlinear and/or high-dimensional problems, additional research efforts are still needed in the future.

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). Marcos Valdebenito acknowledges the support by ANID (National Agency for Research and Development, Chile) under its program FONDECYT, grant number 1180271. Chao Dang, Pengfei Wei and Michael Beer also would like to appreciate the support of Sino-German Mobility Program under grant number M-0175.

Bibliography

[1] Chao Dang, Pengfei Wei, Matthias GR Faes, Marcos A Valdebenito, and Michael Beer. Parallel adaptive bayesian quadrature for rare event estimation. *Reliability Engineering & System Safety*, 225:108621, 2022.

[2] Reuven Y Rubinstein and Dirk P Kroese. *Simulation and the Monte Carlo method*. John Wiley & Sons, 2016.

[3] Art B Owen. Comment: Unreasonable effectiveness of Monte Carlo. *Statistical Science*, 34(1):29–33, 2019.

[4] Anthony O'Hagan. Monte carlo is fundamentally unsound. *The Statistician*, pages 247–249, 1987.

[5] Siu-Kui Au and James L Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, 16(4):263–277, 2001.

[6] Siu-Kui Au and Yu Wang. *Engineering risk assessment with subset simulation*. John Wiley & Sons, 2014.

[7] Siu-Kui Au and James L Beck. A new adaptive importance sampling scheme for reliability calculations. *Structural Safety*, 21(2):135–158, 1999.

[8] Ziqi Wang and Junho Song. Cross-entropy-based adaptive importance sampling using von Mises-Fisher mixture for high dimensional reliability analysis. *Structural Safety*, 59:42–52, 2016.

[9] Sebastian Geyer, Iason Papaioannou, and Daniel Straub. Cross entropy-based importance sampling using Gaussian densities revisited. *Structural Safety*, 76:15–27, 2019.

[10] Iason Papaioannou, Sebastian Geyer, and Daniel Straub. Improved cross entropy-based importance sampling with a flexible mixture model. *Reliability Engineering & System Safety*, 191:106564, 2019.

[11] Persi Diaconis. *Bayesian Numerical Analysis*. Statistical Decision Theory and Related Topics, IV, Vol. 1 (West Lafayette, Ind., 1986), 1988.

[12] Anthony O'Hagan. Bayes–Hermite quadrature. *Journal of Statistical Planning and Inference*, 29(3):245–260, 1991.

[13] Carl Edward Rasmussen and Zoubin Ghahramani. Bayesian Monte Carlo. *Advances in Neural Information Processing Systems*, pages 505–512, 2003.

[14] Piyush Pandita, Ilias Bilionis, and Jitesh Panchal. Bayesian optimal design of experiments for inferring the statistical expectation of expensive black-box functions. *Journal of Mechanical Design*, 141(10), 2019.

[15] Pengfei Wei, Xing Zhang, and Michael Beer. Adaptive experiment design for probabilistic integration. *Computer Methods in Applied Mechanics and Engineering*, 365:113035, 2020.

[16] François-Xavier Briol, Chris J Oates, Mark Girolami, Michael A Osborne, and Dino Sejdinovic. Probabilistic integration: a role in statistical computation? *Statistical Science*, 34(1):1–22, 2019.

[17] Rui Teixeira, Maria Nogal, and Alan O'Connor. Adaptive approaches in metamodel-based reliability analysis: A review. *Structural Safety*, 89:102019, 2021.

[18] Maliki Moustapha, Stefano Marelli, and Bruno Sudret. Active learning for structural reliability: Survey, general framework and benchmark. *Structural Safety*, page 102174, 2022.

[19] Barron J Bichon, Michael S Eldred, Laura Painton Swiler, Sandaran Mahadevan, and John M McFarland. Efficient global reliability analysis for nonlinear implicit performance functions. *AIAA Journal*, 46(10):2459–2468, 2008.

[20] Benjamin Echard, Nicolas Gayton, and Maurice Lemaire. AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation. *Structural Safety*, 33(2):145–154, 2011.

[21] Francesco Cadini and A Gioletta. A Bayesian Monte Carlo-based algorithm for the estimation of small failure probabilities of systems affected by uncertainties. *Reliability Engineering & System Safety*, 153:15–27, 2016.

[22] Chao Dang, Pengfei Wei, Jingwen Song, and Michael 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):04021054, 2021.

[23] Peter I Frazier. A tutorial on Bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

[24] Jeonghwa Lee. Generalized Bernoulli process with long-range dependence and fractional binomial distribution. *Dependence Modeling*, 9(1):1–12, 2021.

[25] Mahdi Teimouri Sichani, Søren RK Nielsen, and Christian Bucher. Applications of asymptotic sampling on high dimensional structural dynamic problems. *Structural Safety*, 33(4-5):305–316, 2011.

[26] Shupeng Sun, Xin Li, Hongzhou Liu, Kangsheng Luo, and Ben Gu. Fast statistical analysis of rare circuit failure events via scaled-sigma sampling for high-dimensional variation space. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 34(7):1096–1109, 2015.

[27] Mohsen Rashki. Hybrid control variates-based simulation method for structural reliability analysis of some problems with low failure probability. *Applied Mathematical Modelling*, 60:220–234, 2018.

[28] Kai Cheng, Zhenzhou Lu, Sinan Xiao, and Jingyu Lei. Estimation of small failure probability using generalized subset simulation. *Mechanical Systems and Signal Processing*, 163:108114, 2022.

[29] Yaacob Ibrahim. Observations on applications of importance sampling in structural reliability analysis. *Structural Safety*, 9(4):269–281, 1991.

[30] Siu-Kui Au and JL Beck. Important sampling in high dimensions. *Structural Safety*, 25(2):139–163, 2003.

[31] Ming-Hui Chen, Qi-Man Shao, and Joseph G Ibrahim. *Monte Carlo methods in Bayesian computation*. Springer Science & Business Media, 2012.

[32] David JC MacKay and David JC Mac Kay. *Information theory, inference and learning algorithms*. Cambridge University Press, 2003.

[33] Pengfei Wei, Chenghu Tang, and Yuting Yang. Structural reliability and reliability sensitivity analysis of extremely rare failure events by combining sampling and surrogate model methods. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 233(6):943–957, 2019.

[34] Stefano Marelli and Bruno Sudret. Uqlab: A framework for uncertainty quantification in Matlab. In *Vulnerability, uncertainty, and risk: quantification, mitigation, and management*, pages 2554–2563. 2014.

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

[36] Karl Breitung. The geometry of limit state function graphs and subset simulation: Counterexamples. *Reliability Engineering & System Safety*, 182:98–106, 2019.

[37] Zhixun Wen, Haiqing Pei, Hai Liu, and Zhufeng Yue. A sequential Kriging reliability analysis method with characteristics of adaptive sampling regions and parallelizability. *Reliability Engineering & System Safety*, 153:170–179, 2016.

[38] Xianzhen Huang and Yimin Zhang. Reliability–sensitivity analysis using dimension reduction methods and saddlepoint approximations. *International Journal for Numerical Methods in Engineering*, 93(8):857–886, 2013.

[39] Chao Dang, Pengfei Wei, Matthias GR Faes, Marcos A Valdebenito, and Michael Beer. Interval uncertainty propagation by a parallel bayesian global optimization method. *Applied Mathematical Modelling*, 108:220–235, 2022.

Chapter 8

Partially Bayesian active learning line sampling for rare event estimation

Estimation of small failure probabilities by partially Bayesian active learning line sampling: Theory and algorithm

Chao Dang[a,⋆], Marcos A. Valdebenito[b], Jingwen Song[c], Pengfei Wei[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 Mechanical Engineering, Northwestern Polytechnical University, Xi'an 710072, PR China*

[d]*School of Power and Energy, 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*

Published in Computer Methods in Applied Mechanics and Engineering in July 2023

Abstract: Line sampling (LS) has proved to be a highly promising advanced simulation technique for assessing small failure probabilities. Despite the great interest in practical engineering applications, many efforts from the research community have been devoted to improving the standard LS. This paper aims at offering some new insights into the LS method, leading to an innovative method, termed 'partially Bayesian active learning line sampling' (PBAL-LS). The problem of evaluating the failure probability integral in the LS method is treated as a Bayesian, rather than frequentist, inference problem, which allows to incorporate our prior knowledge and model the discretization error. The Gaussian process model is used as the prior distribution for the distance function, and the posterior mean, and an upper bound of the posterior variance of the failure probability are derived. Based on the posterior statistics of the failure probability, we also put forward a learning function and a stopping criterion, which enable us to use active learning. Besides, an efficient algorithm is also designed to implement the PBAL-LS method, with the ability to automat-

*Corresponding Author
E-mail address: chao.dang@irz.uni-hannover.de (C. Dang)

ically adjust the important direction and efficiently process the lines. Five numerical examples are studied to demonstrate the performance of the proposed PBAL-LS method against several existing methods.

Keywords: Line sampling; Failure probability; Bayesian inference; Active learning; Gaussian process

8.1 Introduction

Structural reliability analysis has been recognized as a central task for the design and analysis of safety-critical engineering systems in the presence of uncertainties, since the pioneering work of Freudenthal [1] in 1956. For more than half a century, great efforts have been devoted to developing suitable methods to assess the so-called failure probability. Since most engineering systems are expected to be highly reliable, failure should be a rare event, and therefore the probability of failure should be sufficiently small. Besides, very often the failure probability analysis involves evaluating a computationally intensive model multiple times. Among various numerical methods proposed in literature to address the computational challenges, stochastic simulation techniques hold a prominent position. Examples of such techniques include the direct Monte Carlo simulation (MCS) method [2], Importance Sampling [3, 4], Directional Sampling [5], Subset Simulation (SubSim) [6, 7], Line Sampling (LS) [8, 9], etc. However, it is usually computationally expensive to directly apply these stochastic simulation methods. This promotes the development of surrogate-assisted stochastic simulation, e.g., efficient global reliability analysis [10] and AK-MCS (active learning Kriging-MCS) [11]. In this study, we will restrict ourselves to the LS method and its variants.

As a stand-alone simulation technique, the preliminary idea of LS appeared in a talk [12], and then evolved into an expanded version that is normally known to us in [8]. Note that a similar but not quite the same method, known as axis orthogonal sampling, was introduced earlier in [13]. The underlying idea behind LS is to employ lines, instead of random points, to probe the failure domain. At implementation level, a so-called 'important direction' that points towards the failure domain is first required to be identified, and then one has to solve a number of one-dimensional reliability problems conditional on the random line samples parallel to the important direction. The failure probability is finally obtained as an average of the failure probabilities along all the lines. The LS method has proved to be a highly promising approach to evaluating small failure probabilities of

weakly or moderately nonlinear problems, which can be found in a number of practical cases. It has been successfully applied to various areas of engineering, such as aerospace [14, 15], automotive [16], nuclear [17, 18] and civil engineering [19, 20], etc. Besides, the standard LS method is also available in a general-purpose software suite for uncertainty quantification and risk management, called COSSAN [21].

In recent years, increasing attention from the research community has been devoted to expanding the application scope of the original LS method and further improving its performance. As far as the first aspect is concerned, the standard LS method with possible modifications has been used for non-original purposes, such as sensitivity analysis [22–25], estimation of failure probability function or its bounds in the context of reliability-based design optimization [26] and polymorphic uncertainty propagation [27–30]. The second aspect consists of, e.g., adapting the important direction [27, 31], reformulating the LS estimator [31, 32], processing the lines sequentially [27] and introducing multiple important directions [33]. To reduce the computational burden for expensive reliability analysis, surrogate models, such as artificial neural network [18] and Gaussian process (also known as Kriging) [34–36], have also been combined with LS. As a representative, the adaptive Gaussian process regression - line sampling (APGR-LS) method [36] has shown to be capable of assessing very small failure probabilities with a reduced number of performance function evaluations. However, the practical performance of APGR-LS is dependent on a good important direction that is determined by the first-order reliability method, which may suffer from non-convergence and unnecessary computational demands. Besides, it also has a hard-to-tune parameter that is closely related to the learning function and stopping criterion. One can refer to [37] for the recent advancements of the LS method.

The objective of this work is to provide some new insights into the LS method mainly from the perspective of Bayesian active learning, at least partially. More specifically, we develop an innovative LS method, called 'partially Bayesian active learning line sampling' (PBAL-LS). The main contributions of the present work can be summarized as follows. First, the problem of evaluating the failure probability integral in the LS method is reinterpreted by a Bayesian, rather than frequentist, inference problem for the first time. This will enable to incorporate our prior knowledge about the function we wish to learn, which is not allowed for frequentist inference. Second, we present a principled Bayesian approach that is able to reflect our epistemic uncertainty about the failure probability stemming from the discretization error. In this context, the Gaussian process is used as

a prior distribution. The induced posterior statistics of the failure probability is derived. Third, the Bayesian approach is further cast in an active learning setting. Two essential components, namely learning function and stopping criterion, are proposed based on the uncertainty representation of the failure probability. Fourth, we also offer a tailored algorithm for implementing the PBAL-LS in a strategic manner. Two novel features of the algorithm are the adaption of important direction and efficient line processing.

The rest of the paper is structured as follows. Section 8.2 briefly reviews the standard LS method, followed by a discussion on its limitations. Theoretical aspects of the proposed PBAL-LS method are given in Section 8.3. In Section 8.4, we present the proposed PBAL-LS algorithm in a step-by-step manner. Five benchmark examples are investigated in Section 8.5 to demonstrate the developed method. The paper ends with some concluding remarks in Section 8.6.

8.2 Brief review of the standard line sampling

In this section, we revisit the standard LS method after introducing the failure probability definition in the standard normal space. Besides, the key factors affecting the practical performance of the standard LS method are also identified and discussed, as they motivate us to offer a partially Bayesian active learning counterpart in Section 8.3 and its algorithm in Section 8.4.

8.2.1 Failure probability definition in standard normal space

Let $\boldsymbol{X} = X_1, X_2, \cdots, X_d] \in \mathcal{X} \subseteq \mathbb{R}^d$ denote a vector of d physical random variables, which are used to model the uncertain inputs of a performance function $g : \mathcal{X} \to \mathcal{Y}$ associated with the behavior of a system. It is assumed that these physical random variables have a known jointed probability density function (PDF), which is denoted as $f_{\boldsymbol{X}}(\boldsymbol{x})$. In addition, a failure event occurs whenever the performance function takes a value smaller than zero, i.e., $y = g(\boldsymbol{x}) < 0$. It is important to note that for most cases of practical interest the performance function is an expensive-to-evaluate black box, and non-Gaussian inputs might be encountered. Due to the latter, without loss of generality we further assume that $\boldsymbol{X}$ is a vector containing d non-Gaussian random variables. As the LS method typically operates in the standard normal space, it is of necessity to transform the vector $\boldsymbol{X}$ into a standard normal one $\boldsymbol{U} = [U_1, U_2, \cdots, U_d] \in \mathcal{U} \subseteq \mathbb{R}^d$ using an appropriate method (e.g., iso-probabilistic transformation, Nataf transformation and Rosenblatt transformation, etc.).

Such transformation is denoted by $\boldsymbol{U} = T(\boldsymbol{X}) : \mathcal{X} \to \mathcal{U}$, and hence the transformed performance function with respect to $\boldsymbol{U}$ is given by $Z = \mathcal{G}(\boldsymbol{U}) : \mathcal{U} \to \mathcal{Z}$ such that $\mathcal{G} = g \circ T^{-1}$, where $T^{-1} : \mathcal{U} \to \mathcal{X}$ represents the inverse transformation. Under these settings, the failure probability P_F can be defined as:

$$P_F = \int_{\mathbb{R}^d} I_{\mathcal{G}}(\boldsymbol{u}) \varphi_{\boldsymbol{U}}(\boldsymbol{u}) \mathrm{d}\boldsymbol{u}, \tag{8.1}$$

where $I_{\mathcal{G}}(\boldsymbol{u})$ is the failure indicator function corresponding to $\mathcal{G}$: $I_{\mathcal{G}}(\boldsymbol{u}) = 1$ if $\mathcal{G}(\boldsymbol{u}) < 0$ and $I_{\mathcal{G}}(\boldsymbol{u}) = 0$ otherwise; $\varphi_{\boldsymbol{U}}(\boldsymbol{u})$ is the joint PDF of $\boldsymbol{U}$, i.e., $\varphi_{\boldsymbol{U}}(\boldsymbol{u}) = \prod_{i=1}^{d} \varphi_{U_i}(u_i) = \frac{1}{(2\pi)^{d/2}} \exp\left(-\frac{\boldsymbol{u}\boldsymbol{u}^{\mathrm{T}}}{2}\right)$, where $(\cdot)^{\mathrm{T}}$ denotes transpose.

8.2.2 Failure probability estimation by line sampling

The basic idea of LS is to reformulate the d-dimensional failure probability integral defined in Eq. (8.1) into a nested integral, with the inner being a one-dimensional integral along an important direction $\boldsymbol{\alpha}$, and the outer being a $(d-1)$-dimensional integral over a hyperplane orthogonal to $\boldsymbol{\alpha}$ [8, 12]. The so-called important direction $\boldsymbol{\alpha}$ is a unit vector that points towards the failure domain $F = \{\boldsymbol{u} \in \mathcal{U} : \mathcal{G}(\boldsymbol{u}) < 0\}$. In practical implementation, several strategies have been suggested to choose an important direction, e.g., the normalized gradient vector of the $\mathcal{G}$-function at a certain point [14] and the unit vector pointing towards the design point [38]. Once $\boldsymbol{\alpha}$ is given, the original standard normal vector $\boldsymbol{U}$ can be expressed in a new orthogonal coordinate system such that:

$$\boldsymbol{U} = U^{\parallel}\boldsymbol{\alpha} + \boldsymbol{U}^{\perp}\boldsymbol{r}, \tag{8.2}$$

where $\boldsymbol{r}$ is a $(d-1) \times d$ matrix consisting of $d-1$ orthogonal basis vectors of the hyperplane perpendicular to $\boldsymbol{\alpha}$; $U^{\parallel}$ is a standard normal variable and $\boldsymbol{U}^{\perp}$ is a $(d-1)$-dimensional standard normal vector because of the rotational invariance of the standard normal vector. In the new orthogonal coordinate system, the failure probability defined in Eq. (8.1) can be rewritten as:

$$\begin{aligned} P_F &= \int_{\mathbb{R}^{d-1}} \int_{\mathbb{R}} I_{\mathcal{G}}(u^{\parallel}\boldsymbol{\alpha} + \boldsymbol{u}^{\perp}\boldsymbol{r}) \varphi_{U^{\parallel}}(u^{\parallel}) \varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}) \mathrm{d}u^{\parallel} \mathrm{d}\boldsymbol{u}^{\perp} \\ &= \int_{\mathbb{R}^{d-1}} \left(\int_{\mathbb{R}} I_{\mathcal{G}}(u^{\parallel}\boldsymbol{\alpha} + \boldsymbol{u}^{\perp}\boldsymbol{r}) \varphi_{U^{\parallel}}(u^{\parallel}) \mathrm{d}u^{\parallel} \right) \varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}) \mathrm{d}\boldsymbol{u}^{\perp}, \end{aligned} \tag{8.3}$$

where $\varphi_{U^{\parallel}}(u^{\parallel})$ and $\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})$ are the (joint) PDF of $U^{\parallel}$ and $\boldsymbol{U}^{\perp}$, respectively. Under the assumption that the failure domain is simply a half-open region (see Fig. 8.1), the inner integral in Eq. (8.3) is

equal to $\Phi(-\beta(\boldsymbol{u}^{\perp}))$, where $\beta(\boldsymbol{u}^{\perp})$ denotes the Euclidean distance between the limit state surface $\mathcal{G}=0$ and a point $\boldsymbol{u}^{\perp}$ on the orthogonal hyperplane along the direction $\boldsymbol{\alpha}$. In this case, Eq. (8.3) can be further simplified as:

$$P_F = \int_{\mathbb{R}^{d-1}} \Phi(-\beta(\boldsymbol{u}^{\perp}))\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}\boldsymbol{u}^{\perp}. \tag{8.4}$$

Note that in Eq. (8.4) P_F is actually defined as an expectation of $\Phi(-\beta(\boldsymbol{U}^{\perp}))$ with respect to the standard normal vector $\boldsymbol{U}^{\perp}$. In addition, since the previous presumption is often the case for most component reliability problems with linear, weakly nonlinear or moderately nonlinear performance functions, Eq. (8.4) rather than Eq. (8.3) is commonly used in literature. Thereafter, we also refer to Eq. (8.4) when mentioning LS, unless otherwise specified.

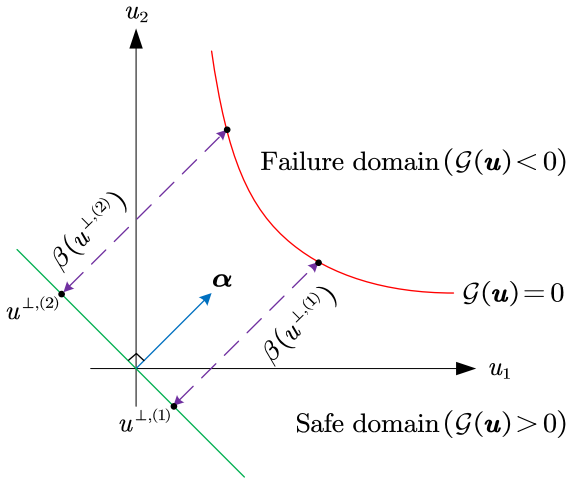


Figure 8.1: Schematic illustration of the standard LS in two dimensions.

In the standard LS method, the failure probability integral defined in Eq. (8.4) is solved by mainly using a purely frequentist procedure – MCS method. The MCS estimator of P_F is given by:

$$\hat{P}_F = \frac{1}{N}\sum_{i=1}^{N} \Phi(-\beta(\boldsymbol{u}^{\perp,(i)})), \tag{8.5}$$

where $\left\{\boldsymbol{u}^{\perp,(i)}\right\}_{i=1}^{N}$ is a set of N random samples drawn from $\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})$; $\beta(\boldsymbol{u}^{\perp,(i)})$ is the distance between $\boldsymbol{u}^{\perp,(i)}$ and the limit state surface $\mathcal{G}=0$ along the important direction $\boldsymbol{\alpha}$, as illustrated in Fig. 8.1. To avoid the specification of $\boldsymbol{r}$, a convenient way is to directly obtain $\boldsymbol{u}^{\perp,(i)}\boldsymbol{r}$ rather than $\boldsymbol{u}^{\perp,(i)}$ by $\boldsymbol{u}^{\perp,(i)}\boldsymbol{r} = \boldsymbol{u}^{(i)} - (\boldsymbol{\alpha}\cdot\boldsymbol{u}^{(i)})\boldsymbol{\alpha}$, where $\boldsymbol{u}^{(i)}$ is a random sample generated according to

$\varphi_{\boldsymbol{U}}(\boldsymbol{u})$. The term $\beta(\boldsymbol{u}^{\perp,(i)})$ can be solved by means of any appropriate root-finding algorithms, such as polynomial interpolation [14] and Newton's method [27]. Note that for each $\boldsymbol{u}^{\perp,(i)}$ finding the value $\beta(\boldsymbol{u}^{\perp,(i)})$ usually requires a handful of $\mathcal{G}$-function evaluations. The variance associated with the MCS estimator is:

$$\sigma^2_{\hat{P}_F} = \frac{1}{N(N-1)} \sum_{i=1}^{n} \left(\Phi(-\beta(\boldsymbol{u}^{\perp,(i)})) - \hat{P}_F \right)^2 . \tag{8.6}$$

In general, the practical performance of the standard LS method is affected by three main aspects. The first aspect consists in the choice of the important direction $\boldsymbol{\alpha}$. A good choice will lead to a fast convergence rate of the subsequent MCS procedure, and hence reduces the sample size N. However, this in turn requires a good knowledge of the limit state surface, which is usually at the expense of many additional $\mathcal{G}$-function evaluations. Another aspect is associated with the MCS method used to solve Eq. (8.4). In case that a poor important direction is chosen and/or the variance of $\Phi(-\beta(\boldsymbol{U}^{\perp}))$ is significant, a large size N is required for the MCS method to achieve a desirable level of accuracy. What is more, the MCS method as a typical frequentist approach suffers from several major limitations, e.g., brute force feature and inability to incorporate prior knowledge, though it has many undeniable advantages. As for the last aspect, the accuracy and efficiency related to searching each $\beta(\boldsymbol{u}^{\perp,(i)})$ may also influence the overall performance of the standard LS method.

8.3 Partially Bayesian active learning line sampling: Theory

In view of the limitations of the standard LS method, especially when examined from frequentist interpretation, we offer a Bayesian active learning treatment in this section, at least partially. The resulting methodology is termed 'partially Bayesian active learning line sampling' (PBAL-LS). Our objective here is to approximate the intractable integral in Eq. (8.4) from a Bayesian active learning perspective, under the premise that a suitable important direction has been determined. However, the premise is only for convenience and not necessary in practice, as shown in Section 8.4. The key to achieving the objective relies on approaching the problem of evaluating the integral in Eq. (8.4) using Bayesian inference, as opposed to frequentist inference. By doing so, it is possible to incorporate our prior knowledge and model the discretization error. Such a Bayesian idea is actually consistent with the spirit of a class of Bayesian probabilistic numerical methods, i.e., Bayesian probabilistic

numerical integration [39, 40]. Further, the Bayesian treatment makes it possible to address the underlying problem in an active learning manner.

8.3.1 Prior distribution

Recall that the class of reliability problems we are interested in the present study involves only weakly to mildly nonlinear behavior. In other words, the limit state surface $\mathcal{G}(\boldsymbol{u}) = 0$ is not expected to be extremely rough, and hence the distance function $\beta(\boldsymbol{u}^{\perp})$ exhibits a smooth behavior. Our prior beliefs about $\beta(\boldsymbol{u}^{\perp})$ can thus be reflected by defining a proper prior distribution over it. A popular choice for a prior distribution over functions is a Gaussian process (GP) due to its conjugate properties and modelling power. In this study, we also assume a GP prior for $\beta(\boldsymbol{u}^{\perp})$, which is expressed as:

$$\hat{\beta}_0(\boldsymbol{u}^{\perp}) \sim \mathcal{GP}(m_{\hat{\beta}_0}(\boldsymbol{u}^{\perp}), k_{\hat{\beta}_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{8.7}$$

where $\hat{\beta}_0$ denotes the prior distribution of β; $m_{\hat{\beta}_0}(\boldsymbol{u}^{\perp})$ and $k_{\hat{\beta}_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$ denote respectively the prior mean and covariance functions, by which the GP model is completely specified. The prior mean function reflects our expected value of $\beta(\boldsymbol{u}^{\perp})$, while the prior covariance function encodes our key assumptions about $\beta(\boldsymbol{u}^{\perp})$, such as smoothness or periodicity. For convenience and without loss of generality, in this study the prior mean function is chosen as a constant denoted as b and the prior covariance function is assumed to be the squared exponential kernel:

$$m_{\hat{\beta}_0}(\boldsymbol{u}^{\perp}) = b, \tag{8.8}$$

$$k_{\hat{\beta}_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) = \sigma_f^2 \exp\left(-\frac{1}{2}(\boldsymbol{u}^{\perp} - \boldsymbol{u}^{\perp\prime})\boldsymbol{\Sigma}^{-1}(\boldsymbol{u}^{\perp} - \boldsymbol{u}^{\perp\prime})^{\mathrm{T}}\right), \tag{8.9}$$

where $\sigma_f > 0$ is the process standard deviation; $\boldsymbol{\Sigma} = \mathrm{diag}\{l_1, l_2, \cdots, l_d\}$ with $l_i > 0$ being the characteristic length scale in i-th dimension; $\mathrm{diag}\{\cdot\}$ means to form a square diagonal matrix with its arguments on the main diagonal. All these parameters $\{b, \sigma_f, l_1, l_2, \cdots, l_d\}$ are collectively referred to as hyperparameters.

8.3.2 Hyperparameters tuning and posterior statistics

Given a observation dataset $\boldsymbol{\mathcal{D}} = \left\{\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{\mathcal{H}}\right\}$, where $\boldsymbol{\mathcal{U}}^{\perp} = \left\{\boldsymbol{u}^{\perp,(i)}\right\}_{i=1}^{n}$ is an n-by-$(d-1)$ matrix with $\boldsymbol{u}^{\perp,(i)}$ being the i-th observing location and $\boldsymbol{\mathcal{H}} = \left[h^{(1)}, h^{(2)}, \cdots, h^{(n)}\right]^{\mathrm{T}}$ is an n-by-1 vector with

i-th element being $h^{(i)} = \beta(\boldsymbol{u}^{\perp,(i)})$. The hyper-parameters can be learned from the observed data $\boldsymbol{\mathcal{D}}$ by some suitable methods, e.g., maximum likelihood estimation [41].

The posterior distribution of β conditioning the GP prior in Eq. (8.7) on $\boldsymbol{\mathcal{D}}$ is again a GP:

$$\hat{\beta}_n(\boldsymbol{u}^{\perp}) \sim \mathcal{GP}(m_{\hat{\beta}_n}(\boldsymbol{u}^{\perp}), k_{\hat{\beta}_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{8.10}$$

where $\hat{\beta}_n$ denotes the posterior distribution of β conditioned on $\boldsymbol{\mathcal{D}}$; $m_{\hat{\beta}_n}(\boldsymbol{u}^{\perp})$ and $k_{\hat{\beta}_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$ are the posterior mean and covariance functions of β, which can be given by:

$$m_{\hat{\beta}_n}(\boldsymbol{u}^{\perp}) = m_{\hat{\beta}_0}(\boldsymbol{u}^{\perp}) + \boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{\mathrm{T}} \boldsymbol{K}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{-1}(\boldsymbol{\mathcal{H}} - \boldsymbol{m}_{\hat{\beta}_0}(\boldsymbol{\mathcal{U}}^{\perp})), \tag{8.11}$$

$$k_{\hat{\beta}_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) = k_{\hat{\beta}_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) - \boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{\mathrm{T}} \boldsymbol{K}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{-1} \boldsymbol{k}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{u}^{\perp\prime}), \tag{8.12}$$

where $\boldsymbol{m}_{\hat{\beta}_0}(\boldsymbol{\mathcal{U}}^{\perp})$ is an n-by-1 mean vector with i-th element being $m_{\hat{\beta}_0}(\boldsymbol{u}^{\perp,(i)})$; $\boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})$ is an n-by-1 covariance vector with i-th element being $k_{\hat{\beta}_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp,(i)})$; $\boldsymbol{k}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{u}^{\perp\prime})$ is an n-by-1 covariance vector with i-th element being $\boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp,(i)}, \boldsymbol{u}^{\perp\prime})$; $\boldsymbol{K}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})$ is an n-by-n covariance matrix with (i,j)-th entry being $k_{\beta_0}(\boldsymbol{u}^{\perp,(i)}, \boldsymbol{u}^{\perp,(j)})$.

The GP posterior of β as shown in Eq. (8.10) can be employed together with the normal cumulative distribution function to produce a posterior on $\Phi(-\beta)$. However, the induced posterior distribution $\hat{\Phi}_n(-\hat{\beta})$ of $\Phi(-\beta)$ is not exactly known in closed form. Fortunately, the posterior mean and variance functions that are needed in the present work can be derived as follows:

$$\begin{aligned} m_{\hat{\Phi}_n(-\hat{\beta})}\left(\boldsymbol{u}^{\perp}\right) =& \mathbb{E}_{\hat{\beta}_n}\left[\hat{\Phi}\left(-\hat{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right)\right] \\ =& \mathbb{E}_U\left[\hat{\Phi}\left(-\left(m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right) + \sigma_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right) U\right)\right)\right] \\ =& \Phi\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right), \end{aligned} \tag{8.13}$$

$$
\begin{aligned}
\sigma^2_{\hat{\Phi}_n(-\hat{\beta})}\left(\boldsymbol{u}^{\perp}\right) =& \mathbb{V}_{\hat{\beta}_n}\left[\hat{\Phi}\left(-\hat{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right)\right] \\
=& \mathbb{E}_{\hat{\beta}_n}\left[\hat{\Phi}^2\left(-\hat{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right)\right] - \mathbb{E}_{\hat{\beta}_n}\left[\hat{\Phi}\left(-\hat{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right)\right]^2 \\
=& \mathbb{E}_U\left[\hat{\Phi}^2\left(-\left(m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)+\sigma_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)U\right)\right)\right] - \Phi^2\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right) \\
=& \Phi\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right) - 2\mathcal{T}\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}, \frac{1}{\sqrt{1+2\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right) - \Phi^2\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right) \\
=& \Phi\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)\Phi\left(\frac{m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right) - 2\mathcal{T}\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}, \frac{1}{\sqrt{1+2\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right),
\end{aligned}
\tag{8.14}
$$

where $\mathbb{E}_{\hat{\beta}_n}[\cdot]$ and $\mathbb{V}_{\hat{\beta}_n}[\cdot]$ denote the expectation and variance operators taken with respect to $\hat{\beta}_n$; $\mathbb{E}_U[\cdot]$ denotes the expectation operator taken with respect to the standard normal variable U; $\sigma_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)$ is the posterior standard function of β, i.e., $\sigma_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right) = \sqrt{k_{\hat{\beta}_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp})}$; $\mathcal{T}(\cdot,\cdot)$ is the Owen's T function, which is defined by an analytically intractable integral. Note that the above derivation makes partial use of the results in [42].

The posterior distribution of $\Phi(-\beta)$ will give arise to a posterior distribution of P_F via the integral operator. Since $\hat{\Phi}_n(-\hat{\beta})$ is not known in closed form, we cannot arrive at the exact posterior distribution $\hat{P}_{F,n}$. This, however, does not impose significant restrictions on the proposed method because the low-order moments other than the exact distribution could be more useful. Analogous to the results of previous studies (e.g., [43]), the posterior mean and variance of P_F turn out to be:

$$
\begin{aligned}
m_{\hat{P}_{F,n}} =& \mathbb{E}_{\boldsymbol{U}^{\perp}}\left[m_{\hat{\Phi}_n(-\hat{\beta})}\left(\boldsymbol{U}^{\perp}\right)\right] \\
=& \int_{\mathbb{R}^{d-1}} \Phi\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}\boldsymbol{u}^{\perp},
\end{aligned}
\tag{8.15}
$$

$$
\begin{aligned}
\sigma^2_{\hat{P}_{F,n}} =& \mathbb{E}_{\boldsymbol{U}^{\perp},\boldsymbol{U}^{\perp\prime}}\left[k_{\hat{\Phi}_n(-\hat{\beta})}(\boldsymbol{U}^{\perp},\boldsymbol{U}^{\perp\prime})\right] \\
=& \int_{\mathbb{R}^{d-1}}\int_{\mathbb{R}^{d-1}} k_{\hat{\Phi}_n(-\hat{\beta})}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime})\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\varphi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime})\mathrm{d}\boldsymbol{u}^{\perp}\mathrm{d}\boldsymbol{u}^{\perp\prime},
\end{aligned}
\tag{8.16}
$$

where $\mathbb{E}_{\boldsymbol{U}^{\perp}}[\cdot]$ means to take expectation of its argument with respect to $\boldsymbol{U}^{\perp}$; $\mathbb{E}_{\boldsymbol{U}^{\perp},\boldsymbol{U}^{\perp\prime}}[\cdot]$ denotes the expectation taken with respect to $\boldsymbol{U}^{\perp}$ and $\boldsymbol{U}^{\perp\prime}$ ($\boldsymbol{U}^{\perp}$ and $\boldsymbol{U}^{\perp\prime}$ are independent and identically distributed); $k_{\hat{\Phi}_n(-\hat{\beta})}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime})$ is the posterior covariance function of $\Phi(-\beta)$. Eq. (8.15) implies that the posterior mean of P_F is the expectation of the posterior mean function of $\Phi(-\beta)$ taken

with respect to $\boldsymbol{U}^{\perp}$, while Eq. (8.16) suggests that the posterior variance of P_F is the expectation of the posterior covariance function of $\Phi(-\beta)$ taken with respect to $\boldsymbol{U}^{\perp}$ and $\boldsymbol{U}^{\perp\prime}$. However, as the posterior covariance function of $\Phi(-\beta)$ is unknown, we cannot arrive at the solution to the posterior variance of P_F. To circumvent this issue, our proposal is to consider an upper-bound of $\sigma^2_{\hat{P}_{F,n}}$. Using the Cauchy–Schwarz inequality (i.e., $k_{\hat{\Phi}_n(-\hat{\beta})}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) \leq \sigma_{\hat{\Phi}_n(-\hat{\beta})}\left(\boldsymbol{u}^{\perp}\right)\sigma_{\hat{\Phi}_n(-\hat{\beta})}\left(\boldsymbol{u}^{\perp\prime}\right)$), we have

$$\begin{aligned}\sigma^2_{\hat{P}_{F,n}} \leq \overline{\sigma}^2_{\hat{P}_{F,n}} &= \int_{\mathbb{R}^{d-1}}\int_{\mathbb{R}^{d-1}} \sigma_{\hat{\Phi}_n(-\hat{\beta})}\left(\boldsymbol{u}^{\perp}\right)\sigma_{\hat{\Phi}_n(-\hat{\beta})}\left(\boldsymbol{u}^{\perp\prime}\right)\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\varphi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime})\mathrm{d}\boldsymbol{u}^{\perp}\mathrm{d}\boldsymbol{u}^{\perp\prime} \\ &= \left(\int_{\mathbb{R}^{d-1}} \sqrt{\Phi\left(\frac{-m_{\hat{\beta}_n}(\boldsymbol{u}^{\perp})}{\sqrt{1+\sigma^2_{\hat{\beta}_n}(\boldsymbol{u}^{\perp})}}\right)\Phi\left(\frac{m_{\hat{\beta}_n}(\boldsymbol{u}^{\perp})}{\sqrt{1+\sigma^2_{\hat{\beta}_n}(\boldsymbol{u}^{\perp})}}\right) - 2\mathcal{T}\left(\frac{-m_{\hat{\beta}_n}(\boldsymbol{u}^{\perp})}{\sqrt{1+\sigma^2_{\hat{\beta}_n}(\boldsymbol{u}^{\perp})}}, \frac{1}{\sqrt{1+2\sigma^2_{\hat{\beta}_n}(\boldsymbol{u}^{\perp})}}\right)}\right. \\ &\quad \left. \times\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}\boldsymbol{u}^{\perp}\right)^2,\end{aligned} \tag{8.17}$$

where $\overline{\sigma}^2_{\hat{P}_{F,n}}$ denotes the resulting upper bound of $\sigma^2_{\hat{P}_{F,n}}$ and the equality holds if and only if $\hat{\boldsymbol{\Phi}}_n(-\hat{\beta})$ between any two locations is linearly dependent. The posterior mean $m_{\hat{P}_{F,n}}$ in Eq. (8.15) can be used as a natural estimate for P_F, whereas $\overline{\sigma}^2_{\hat{P}_{F,n}}$ in Eq. (8.17) can measure our maximum possible epistemic uncertainty about the estimate. The epistemic uncertainty is related to the numerical uncertainty arising from the fact that the β-function is only observed at a number of discrete points. As we only offer an upper bound for quantifying the numerical uncertainty (as revealed by Eq. (8.17)), the proposed approach is termed as *partially* Bayesian.

8.3.3 Learning function and stopping criterion

Our general goal with the PBAL-LS method is to produce a failure probability estimate with a desired degree of accuracy using as few β-function evaluations as possible. To achieve such goal, one potential strategy is to implement the Bayesian approach sequentially. Specifically, an active learning procedure can be readily used in conjunction with the Bayesian framework. An essential ingredient for an active learning procedure is a so-called learning (acquisition) function that can suggest the best next observation point based on our prior knowledge. In this work, a new learning function that is extracted from $\overline{\sigma}_{\hat{P}_{F,n}}$, called upper-bound posterior standard deviation contribution

(UPSDC), is proposed as follows:

$$\mathrm{UPSDC}\left(\boldsymbol{u}^{\perp}\right)=\sqrt{\Phi\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)\Phi\left(\frac{m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)-2\mathcal{T}\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}},\frac{1}{\sqrt{1+2\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)}\times\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}). \tag{8.18}$$

It is easy to check that $\overline{\sigma}_{\hat{P}_{F,n}}=\int_{\mathbb{R}^{d-1}}\mathrm{UPSDC}\left(\boldsymbol{u}^{\perp}\right)\mathrm{d}\boldsymbol{u}^{\perp}$ holds. Therefore, the UPSDC function can be seen as a measure of the contribution of epistemic uncertainty at the site $\boldsymbol{u}^{\perp}$ to the upper bound of the posterior standard deviation of P_F. The best next point $\boldsymbol{u}^{\perp,(n+1)}$ where to query the β-function can be selected as the point maximizing the UPSDC function (i.e., $\boldsymbol{u}^{\perp,(n+1)}=\arg\max_{\boldsymbol{u}^{\perp}\in\mathcal{U}^{\perp}}\mathrm{UPSDC}\left(\boldsymbol{u}^{\perp}\right)$), which is expected to reduce $\overline{\sigma}_{\hat{P}_{F,n+1}}$ the most. Another key ingredient for an active learning procedure is associated with a stopping criterion that can determine when to stop the iteration. In this study, the stopping criterion is defined as whether the upper-bound of the posterior COV of the failure probability $\overline{\mathrm{COV}}_{\hat{P}_{F,n}}$ is less than a tolerance ϵ:

$$\overline{\mathrm{COV}}_{\hat{P}_{F,n}}=\frac{\overline{\sigma}_{\hat{P}_{F,n}}}{m_{\hat{P}_{F,n}}}<\epsilon, \tag{8.19}$$

where ϵ is user-specified. A smaller ϵ typically implies a higher accuracy of $m_{\hat{P}_{F,n}}$, but with increased computational cost, and vice versa.

8.4 Partially Bayesian active learning line sampling: Algorithm

In this section, we present an efficient algorithm for applying the proposed PBAL-LS method to practical reliability analysis problems, where several important implementation issues are addressed. The first issue is related to the choice of the important direction $\boldsymbol{\alpha}$ that is assumed to be already known before the Bayesian treatment in the preceding section, which, however, should be appropriately specified in practice. Instead of a fixed and initially determined one, an adaptive strategy is proposed to adjust the important direction on the fly throughout the simulation. As for the second issue, searching for the distance between a given sample $\boldsymbol{u}^{\perp}$ and the limit state surface $\mathcal{G}=0$ along the important direction, is tackled in a strategic manner. The third issue arising from coping with $m_{\hat{P}_{F,n}}$, $\overline{\sigma}^2_{\hat{P}_{F,n}}$ and $\mathrm{UPSDC}\left(\boldsymbol{u}^{\perp}\right)$ is also properly processed from a numerical point of

view.

The proposed algorithm for the PBAL-LS method is illustrated with the flowchart in Fig. 8.3, and it consists of six main steps described as below. Note that for clarity the core tasks of each step are listed in the blue box, followed by detailed descriptions when necessary.

Step 1: Selection of an initial important direction

An initial important direction $\boldsymbol{\alpha}^{(0)}$ must be chosen in this step, and then one has to specify the corresponding matrix $\boldsymbol{r}^{(0)}$ for the hyperplane orthogonal to $\boldsymbol{\alpha}^{(0)}$.

The important direction will be designed to be updated automatically as soon as a more probable one is found, and hence the proposed algorithm does not need to start with an optimal important direction. As a rough guess, the initial important direction can be selected as the negative normalized gradient of the $\mathcal{G}$-function at the origin:

$$\boldsymbol{\alpha}^{(0)} = -\frac{\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0})}{||\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0})||}, \tag{8.20}$$

where $\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0}) = \left[\frac{\partial\mathcal{G}(\mathbf{0})}{\partial u_1}, \frac{\partial\mathcal{G}(\mathbf{0})}{\partial u_2}, \cdots, \frac{\partial\mathcal{G}(\mathbf{0})}{\partial u_d}\right]$; $||\cdot||$ denotes the Euclidean norm. Note that $\boldsymbol{\alpha}^{(0)}$ corresponds to the direction of steepest descent of the $\mathcal{G}$-function at the origin, and is expected to point towards the failure domain for those cases we are interested in. As the $\mathcal{G}$-function is assumed to be known only implicitly with respect to $\boldsymbol{u}$, forward difference is used to estimate $\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0})$ at the expense of $(d+1)$ $\mathcal{G}$-function evaluations.

The matrix $\boldsymbol{r}^{(0)}$ is theoretically not unique given $\boldsymbol{\alpha}^{(0)}$. In practice, however, it needs to be specified from numerical point of view. To do so, one can resort to, e.g., the Gram–Schmidt process, which is also used in the following steps whenever needed.

Step 2: Generation of an initial observation dataset

This step involves generating a small initial observation dataset $\boldsymbol{\mathcal{D}} = \left\{\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{\mathcal{H}}\right\}$ and updating the important direction if possible at the same time.

First, an auxiliary line is deployed from the origin along the direction $\boldsymbol{\alpha}^{(0)}$ (denoted as $u^{\parallel}\boldsymbol{\alpha}^{(0)}$), and we have to find the solution to the equation $\mathcal{G}(u^{\parallel}\boldsymbol{\alpha}^{(0)}) = 0$. Note that since this line is

only designed for providing a rough reference for the following procedure, an accurate solution to $\mathcal{G}(u^{\parallel}\boldsymbol{\alpha}^{(0)}) = 0$ is not necessary. To this end, it is suggested to use a simple interpolation method, e.g., piecewise cubic Hermite interpolating polynomial (PCHIP). The PCHIP method requires evaluating the $\mathcal{G}$-function at three discrete points of $u^{\parallel}$. In this study, the three points are specified as $c_1 = 3$, $c_2 = 6$ and $c_3 = 8$. The interpolated value of $u^{\parallel}$ is denoted as $\beta^{(0)}$. The interpolation procedure is illustrated in Fig. 8.2(a).

Second, the β-function should be observed at several locations so as to form an initial observation dataset and update the important direction if possible. A small number, n_0, of samples $\left\{\boldsymbol{u}^{\perp,(j)}\right\}_{j=1}^{n_0}$ are generated from $\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})$ by using, e.g., Sobol sequence. In this study, n_0 is set to be 5. Here, the Newton's method is used to identify the root of $\mathcal{G}(u^{\parallel,(j)}\boldsymbol{\alpha}^{(l)} + \boldsymbol{u}^{\perp,(j)}\boldsymbol{r}^{(l)}) = 0$. Instead of processing $\left\{\boldsymbol{u}^{\perp,(j)}\right\}_{j=1}^{n_0}$ in the order as they are generated, we rearrange them according to the distance from the origin in ascending order, denoted as $\left\{\tilde{\boldsymbol{u}}^{\perp,(j)}\right\}_{j=1}^{n_0}$. By doing so, one can make use of $\beta^{(0)}$ as the starting point when searching $\tilde{h}^{(1)}$, and $\tilde{h}^{(j)}$ for $\tilde{h}^{(j+1)}$ $(j = 1, 2, \cdots, n_0)$. The intersection point for the line $\tilde{u}^{\parallel,(j)}\boldsymbol{\alpha}^{(l)} + \tilde{\boldsymbol{u}}^{\perp,(j)}\boldsymbol{r}^{(l)}$ with the limit state surface $\mathcal{G}(\boldsymbol{u}) = 0$ is recorded as $\tilde{\boldsymbol{u}}^{(j)} = \tilde{h}^{(j)}\boldsymbol{\alpha}^{(l)} + \tilde{\boldsymbol{u}}^{\perp,(j)}\boldsymbol{r}^{(l)}$. Whenever a nearest intersection point from the origin is found, e.g., $||\tilde{\boldsymbol{u}}^{(j)}|| = \min_{j=1}^{j}||\tilde{\boldsymbol{u}}^{(j)}||$ $(j \geq 2)$, the important direction should be immediately updated as $\boldsymbol{\alpha}^{(l+1)} = \frac{\tilde{\boldsymbol{u}}^{(j)}}{||\tilde{\boldsymbol{u}}^{(j)}||}$, together with $\boldsymbol{r}^{(l+1)}$. One can refer to Figs. 8.2(b) and 8.2(c) for a schematic illustration. Once all the lines have been processed, $\boldsymbol{\mathcal{U}}^{\perp}$ can be obtained by projecting the intersection points $\left\{\tilde{\boldsymbol{u}}^{(j)}\right\}_{j=1}^{n_0}$ onto the last orthogonal hyperplane, and $\boldsymbol{\mathcal{H}}$ contains the corresponding distances between $\left\{\tilde{\boldsymbol{u}}^{(j)}\right\}_{j=1}^{n_0}$ and the last orthogonal hyperplane along the last importance direction.

Step 3: Computation of posterior statistics of the failure probability

Conditional on the observation dataset $\boldsymbol{\mathcal{D}}$, the posterior mean and upper bound on the posterior standard deviation of the failure probability are evaluated.

The GP posterior of β conditional on $\boldsymbol{\mathcal{D}}$ can be obtained analytically as defined in Eq. (8.10). This process can be done with the help of existing software packages, e.g., *fitrgp* function in Statistics and Machine Learning Toolbox of Matlab.

As can be seen in Eqs. (8.15) and (8.17), the posterior mean and upper bound of the posterior standard deviation of the failure probability are not analytically tractable. In view of that, the MCS method is used here. According to Eq. (8.15), an unbiased estimator for the posterior mean

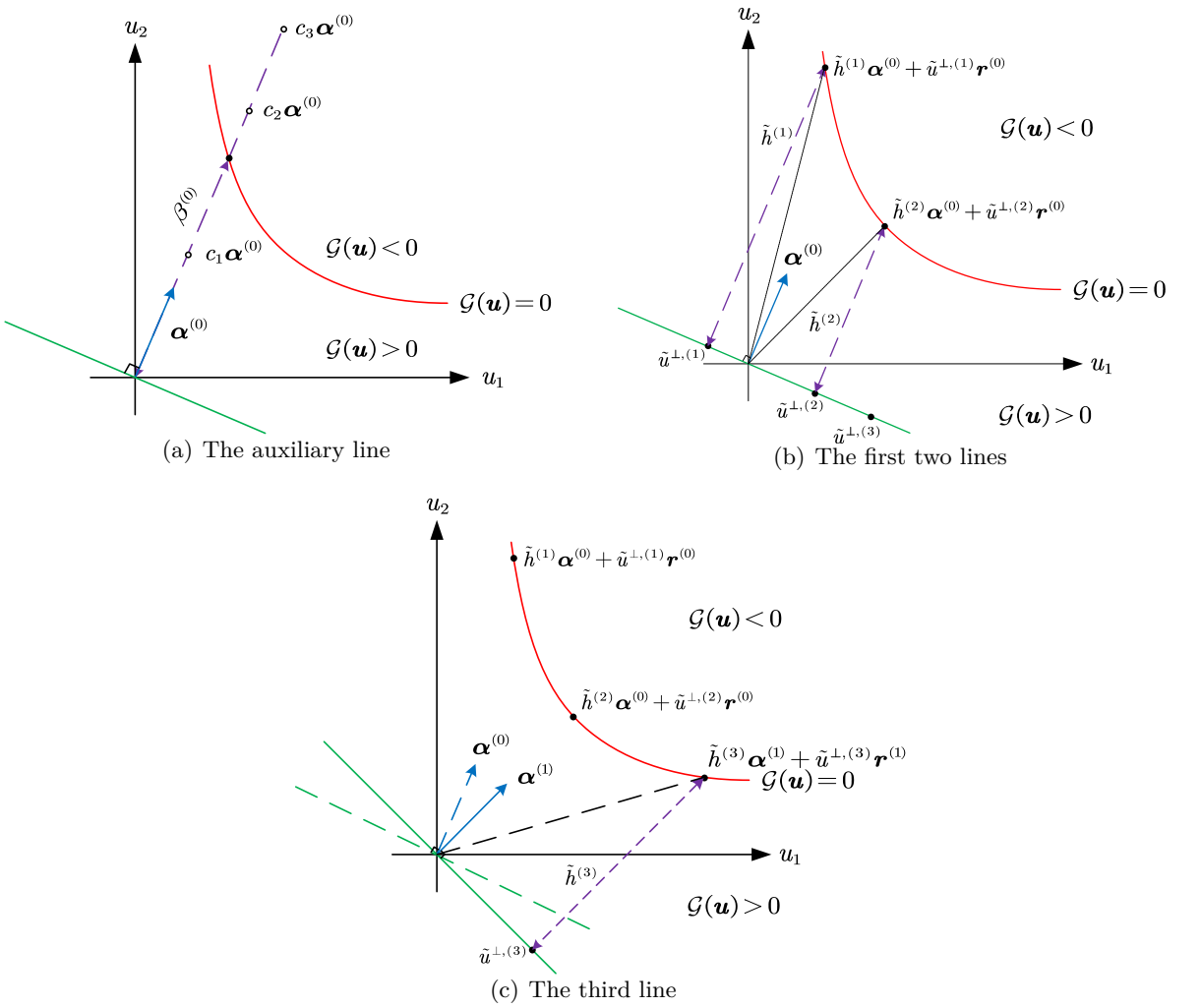


Figure 8.2: Illustration of Step 2 of the proposed PBAL-LS algorithm in two dimensions ($n_0 = 3$).

of the failure probability is given by:

$$\hat{m}_{\hat{P}_{F,n}} = \frac{1}{N}\sum_{i=1}^{N} \Phi\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)}}\right), \tag{8.21}$$

where $\left\{\boldsymbol{u}^{\perp,(i)}\right\}_{i=1}^{N}$ is a set of N random samples generated from $\varphi_{\boldsymbol{U}^\perp}(\boldsymbol{u}^\perp)$. The variance of the estimator can be expressed as:

$$\mathbb{V}\left[\hat{m}_{\hat{P}_{F,n}}\right] = \frac{1}{N(N-1)}\sum_{i=1}^{N}\left[\Phi\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)}}\right) - \hat{m}_{\hat{P}_{F,n}}\right]^2. \tag{8.22}$$

The estimator for the upper bound on the posterior standard deviation of the failure probability is formulated as:

$$\hat{\bar{\sigma}}_{\hat{P}_{F,n}} = \frac{1}{N}\sum_{i=1}^{N}\sqrt{\mathbb{E}_U\left[\hat{\Phi}^2\left(-\left(m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)+\sigma_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)U\right)\right)\right]-\Phi^2\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)}}\right)}. \quad (8.23)$$

It should be pointed out that we do not make use of the final result of $\sigma^2_{\hat{\Phi}_n(-\hat{\beta})}\left(\boldsymbol{u}^{\perp}\right)$ (but the intermediate result) to formulate $\hat{\bar{\sigma}}_{\hat{P}_{F,n}}$ because the involved Owen's T function is not easy to handle numerically. Instead, the expectation term in Eq. (8.23) can be readily approximated by some well-known quadrature rules available in a number of software packages, e.g., the *integral* function in Matlab. The variance of the estimator in Eq. (8.23) is written as:

$$\mathbb{V}\left[\hat{\bar{\sigma}}_{\hat{P}_{F,n}}\right] = \frac{1}{N(N-1)}\sum_{i=1}^{N}\left[\sqrt{\mathbb{E}_U\left[\hat{\Phi}^2\left(-\left(m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)+\sigma_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)U\right)\right)\right]-\Phi^2\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(i)}\right)}}\right)}-\hat{\bar{\sigma}}_{\hat{P}_{F,n}}\right]^2. \quad (8.24)$$

To ensure the accuracy of $\hat{m}_{\hat{P}_{F,n}}$ and $\hat{\bar{\sigma}}_{\hat{P}_{F,n}}$, the sample size N should be large enough. However, a too large sample size may lead to low efficiency and even memory loss when evaluating $\hat{m}_{\hat{P}_{F,n}}$ and $\hat{\bar{\sigma}}_{\hat{P}_{F,n}}$. For this reason, it is suggested to increase the sample size progressively until a stopping criterion is satisfied. A convenient stopping criterion is defined as the maximum value of two COVs ($\sqrt{\mathbb{V}\left[\hat{m}_{\hat{P}_{F,n}}\right]}\Big/\hat{m}_{\hat{P}_{F,n}}$ and $\sqrt{\mathbb{V}\left[\hat{\bar{\sigma}}_{\hat{P}_{F,n}}\right]}\Big/\hat{\bar{\sigma}}_{\hat{P}_{F,n}}$), i.e., $\max\left(\sqrt{\mathbb{V}\left[\hat{m}_{\hat{P}_{F,n}}\right]}\Big/\hat{m}_{\hat{P}_{F,n}}, \sqrt{\mathbb{V}\left[\hat{\bar{\sigma}}_{\hat{P}_{F,n}}\right]}\Big/\hat{\bar{\sigma}}_{\hat{P}_{F,n}}\right) < \delta$, where δ is the tolerance. In this study, δ is set to be 2%.

Step 4: Judgment of the stopping condition on learning

The stopping criterion for active learning should be examined and then decide whether to continue or stop the algorithm.

If $\hat{\overline{\mathrm{COV}}}_{\hat{P}_{F,n}} = \hat{\bar{\sigma}}_{\hat{P}_{F,n}}/\hat{m}_{\hat{P}_{F,n}} < \epsilon$ is satisfied twice in a row, go to **Step 6**; otherwise, go to **Step 5**. The tolerance ϵ can take a value between $5\% - 10\%$.

Step 5: Active updating of the observation dataset

The β-function is observed at a new point identified by our learning function. Then, the important direction can be modified if possible and the previous observation dataset can be updated.

The next best point to query the β-function is identified by maximizing the proposed UPSDC function such that:

$$\boldsymbol{u}^{\perp,(n+1)} = \underset{\boldsymbol{u}^{\perp}\in\mathcal{U}^{\perp}}{\arg\max} \sqrt{\mathbb{E}_U\left[\hat{\Phi}^2\left(-\left(m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)+\sigma_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)U\right)\right)\right]-\Phi^2\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)} \times \varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}). \tag{8.25}$$

It is worth mentioning that the UPSDC function is used in the form given by Eq. (8.25) rather than Eq. (8.18) in order to avoid directly dealing with the Owen's T function. The expectation term on the right-hand side of Eq. (8.25) can be approximated by a suitable quadrature rule, while Eq. (8.25) can be solved by some nature-inspired optimization algorithms.

The distance $\tilde{h}^{(n+1)}$ between $\boldsymbol{u}^{\perp,(n+1)}$ and the limit state surface $\mathcal{G}=0$ along the latest important direction is searched by the Newton's method. To accelerate the search process, $m_{\hat{\beta}_n}(\boldsymbol{u}^{\perp,(n+1)})$ can be taken as a good starting point. Once $\tilde{h}^{(n+1)}$ is available, it is trivial to obtain the corresponding intersection point $\tilde{\boldsymbol{u}}^{(n+1)}$. If $||\tilde{\boldsymbol{u}}^{(n+1)}|| \neq \min_{j=1}^{n+1}||\tilde{\boldsymbol{u}}^{(j)}||$, the previous training dataset is enriched with $\left\{\boldsymbol{u}^{\perp,(n+1)}, \tilde{h}^{(n+1)}\right\}$; otherwise, the important direction is updated as the normalized $\frac{\tilde{\boldsymbol{u}}^{(n+1)}}{||\tilde{\boldsymbol{u}}^{(n+1)}||}$ (as well as the matrix $\boldsymbol{r}$), and the observation dataset should be renewed by projecting these $n+1$ intersection points onto the latest orthogonal hyperplane. Go to **Step 3**.

Step 6: End of the proposed PBAL-LS algorithm

Return the last estimate of the posterior mean of the failure probability and end the proposed PBAL-LS algorithm.

Note that the important direction might be adapted in both **Step 2** and **Step 5**, which do not require any additional evaluations of the $\mathcal{G}$-function. Similar ideas can be found in several improved LS methods [27, 31, 32]. This feature proves to be useful as one does not need to waste extra effort on specifying an optimal important direction. In the active learning process the next location to observe the β-function is chosen in terms of the learning function rather than randomly, which

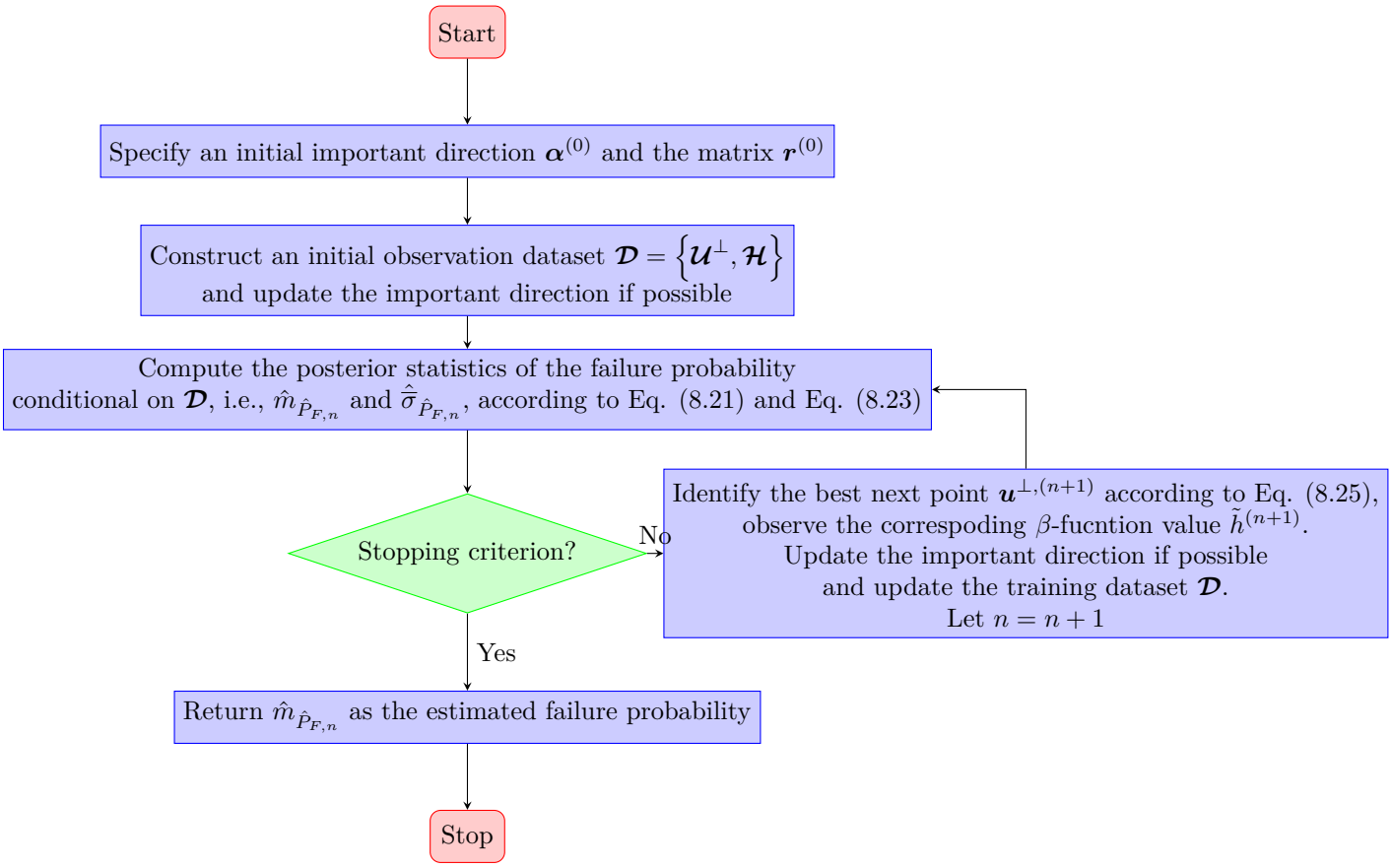


Figure 8.3: Flowchart of the proposed PBAL-LS algorithm.

makes the fullest possible use of our prior knowledge about the β-function, and hence can reduce the number of lines required. In addition, the root-finding procedure in **Step 2** is also tailored as information can be reused to accelerate convergence and reduce the number of $\mathcal{G}$-function calls. The authors of [27] adopted a similar but slightly different strategy. In **Step 5**, the starting point of the root-finding algorithm is provided by the GP prediction, and hence incorporating our prior knowledge. It is also worth mentioning that the final result of Eq. (8.14) and Eq. (8.18) can be directly used in the PBAL-LS algorithm once an appropriate quadrature rule is available to deal with the Owen's T function.

8.5 Numerical examples

In this section, the performance of the proposed PBAL-LS method for assessing small failure probabilities is demonstrated against several existing methods by means of five numerical examples.

These selected methods include the standard LS [8], combination line sampling (CLS) [32], active learning Kriging - importance sampling (AK-IS) [44] and AGPR-LS [36]. The (initial) important direction is specified by the negative normalized gradient of the g-function at the origin for LS and CLS, while by first-order reliability method using Hasofer-Lind-Rackwitz-Fiessler [45] for and AK-IS and AGPR-LS. The studied numerical examples represent a class of moderately nonlinear problems with varying complexity where those LS methods are expected to be applicable.The common feature of these examples is that they are designed to have very small failure probabilities. In all numerical examples, the MCS method with a sufficiently large sample size is employed to provide the reference failure probability if applicable.

8.5.1 Example 1: An illustrative problem

The first example involves an illustrative problem with the performance function:

$$Y = g(\boldsymbol{X}) = a - X_2 + bX_1^3 + c\sin\left(dX_1\right), \tag{8.26}$$

where X_1 and X_2 are two independent standard normal variables; a, b, c and d are four constant parameters, which are specified as: $a = 5.0$, $b = 0.01$, $c = 1.0$ and $d = 1.0$.

The proposed PBAL-LS method is compared in Table 8.1 with several other methods, including MCS, standard LS, CLS, AK-IS and AGPR-LS. The reference value of the failure probability is obtained as 5.87×10^{-6} (with a COV being 0.41%), provided by the MCS method with 10^{10} samples. It is clear that the LS methods without using the GP model (i.e., standard LS and CLS) require a large number of lines and performance function evaluations in order to produce failure probability estimates with small COVs. AK-IS and AGPR-LS can significantly reduce the number of calls to the performance function, while yielding acceptable failure probability estimates. The performance of the AGPR-LS method, however, is highly dependent upon the parameter ϵ, which is not easy to tune. The proposed method gives the same failure probability as the AGPR-LS method. However, it requires fewer performance function evaluations than the AGPR-LS method.

For the sake of illustration, Fig. 8.4 depicts the initial and final important directions, and the intersection points identified by the proposed PBAL-LS method ($\epsilon = 10\%$), along with the true limit state curve. It is shown that the initial important direction chosen by Eq. (8.20) is far from optimal, but the final one is nearly optimal. Besides, the approximate intersection points are

located on the true limit state curve, indicating the accuracy of the root-finding algorithm.

Table 8.1: Reliability results for Example 1 obtained from several methods.

Method	$\hat{P}_f$	COV$[\hat{P}_f]$ or $\overline{\text{COV}}[\hat{P}_f]$	N_{line}	N_{total}
MCS	5.87×10^{-6}	0.41%	-	10^{10}
Standard LS	6.68×10^{-6}	47.72%	100	721
	6.00×10^{-6}	13.46%	1000	7079
CLS	6.18×10^{-6}	7.68%	100	524
	5.92×10^{-6}	2.36%	1000	5103
AK-IS	5.68×10^{-6}	3.46%	-	62
AGPR-LS	5.85×10^{-6}	1.81%	11	64
Proposed PBAL-LS ($\epsilon = 10\%$)	5.85×10^{-6}	2.28%	10	38

Note: The parameter ϵ in the AGPR-LS method is set to be 0.001.

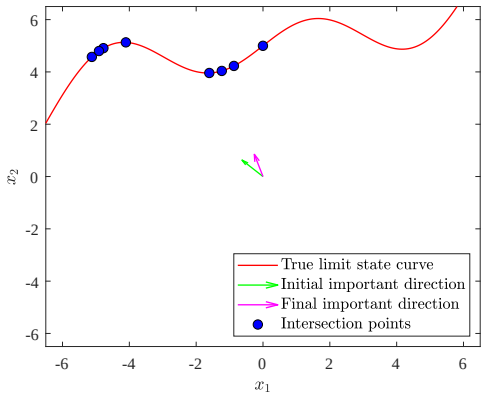


Figure 8.4: Illustration of the proposed PBAL-LS method ($\epsilon = 10\%$) in Example 1.

8.5.2 Example 2: A quadratic function

The second example considers a quadratic performance function of the form [46]:

$$Y = g(\boldsymbol{X}) = a - X_d + b\sum_{i=1}^{d-1} X_i^2, \tag{8.27}$$

where $\boldsymbol{X} = [X_1, X_2, \cdots, X_d]$ is a vector of d independent log-normal variables with a mean of 1 and a standard deviation of 0.2. In this example, the dimension is set to be $d = 21$, while the other two parameters are $a = 2.5$ and $b = 0.02$.

The results of several reliability analysis methods are summarized in Table 8.2. The reference value of the failure probability is 2.00×10^{-8} with a COV being 2.23%, provided by MCS with 10^{11}

samples. In order to produce a failure probability estimate with a small COV, both standard LS and CLS requires a large number of lines, as well as $\mathcal{G}$-function calls. The results of AK-IS and AGPR-LS are empty because FORM-HLRF fails in this example. The proposed PBAL-LS is able to produce an accurate failure probability estimate with only 39 additional lines and a total of 137 performance function evaluations.

Table 8.2: Reliability results for Example 2 obtained from several methods.

Method	$\hat{P}_f$	COV$[\hat{P}_f]$ or $\overline{\text{COV}}[\hat{P}_f]$	N_{line}	N_{total}
MCS	2.00×10^{-8}	2.23%	-	10^{11}
Standard LS	2.01×10^{-8}	7.64%	100	422
	2.04×10^{-8}	4.88%	200	822
CLS	1.94×10^{-8}	7.31%	100	486
	2.17×10^{-8}	4.82%	200	918
AK-IS	-	-	-	-
AGPR-LS	-	-	-	-
Proposed PBAL-LS ($\epsilon = 10\%$)	1.95×10^{-8}	8.38%	39	137

8.5.3 Example 3: A nonlinear oscillator

As a third example, we consider a nonlinear single-degree-of-freedom (SDOF) oscillator under a rectangular-pulse load [47], which is shown in Fig. 8.5. The limit state function is defined by:

$$Y = g\left(m, c_1, c_2, r, F_1, t_1\right) = 3r - \left|\frac{2F_1}{c_1 + c_2} \sin\left(\frac{t_1}{2}\sqrt{\frac{c_1 + c_2}{m}}\right)\right|, \tag{8.28}$$

where m, c_1, c_2, r, F_1, t_1 are six random variables, which are specified according to Table 8.3.

The reference failure probability of this example is taken as 1.15×10^{-7} with COV being 2.94%, provided by the MCS method with 10^{10} samples. Table 8.4 summarizes the results by several other LS methods, along those of MCS. It is seen that with the same number of lines the CLS method is able to give much more better failure probability estimate with a smaller COV than the standard LS method, but at the expense of sightly increased $\mathcal{G}$-function evaluations. Note that both the two methods require considerably more calls to the $\mathcal{G}$-function than AK-IS, AGPR-LS and PBAL-LS in order to achieve a desired level of accuracy, especially for the standard LS method. AK-IS and AGPR-LS method are still much less efficient than the proposed PBAL-LS method in terms of N_{total}, even though they give very similar estimates for the failure probability.

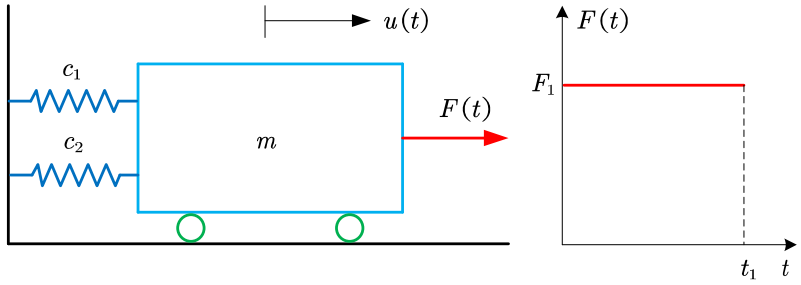


Figure 8.5: A nonlinear SDOF oscillator subject to pulse load.

Table 8.3: Random variables for Example 3.

Variable		Distribution	Mean	COV
m	Mass	Normal	1.0	0.10
c_1	Stiffness	Lognormal	1.0	0.10
c_2	Stiffness	Lognormal	0.1	0.10
r	Yield displacement	Normal	0.5	0.10
F_1	Load amplitude	Lognormal	0.4	0.20
t_1	Load duration	Normal	1.0	0.20

8.5.4 Example 4: A cantilever tube

The third example consists of a cantilever tube subject to three forces and one torque [48], as shown in Fig. 8.6. The performance function is defined as

$$Y = S_y - \sigma_{\max}, \tag{8.29}$$

Table 8.4: Reliability results for Example 3 obtained from several methods.

Method	$\hat{P}_f$	COV$[\hat{P}_f]$ or $\overline{\text{COV}}[\hat{P}_f]$	N_{line}	N_{total}
MCS	1.15×10^{-7}	2.94%	-	10^{10}
Standard LS	6.57×10^{-8}	17.14%	500	1969
	9.29×10^{-8}	13.54%	1000	3909
CLS	1.17×10^{-7}	5.89%	500	2084
	1.14×10^{-7}	2.76%	1000	4113
AK-IS	1.12×10^{-7}	2.61%	-	156
AGPR-LS	1.14×10^{-7}	0.86%	46	103
Proposed PBAL-LS ($\epsilon = 10\%$)	1.17×10^{-7}	7.62%	17	62

Note: The parameter ϵ in the AGPR-LS method is set to be 0.005.

where S_y is the yield stress of the material; $\sigma_{\max}$ denotes the maximum von Mises tress on the top surface of the tube at the origin, which is given by:

$$\sigma_{\max} = \sqrt{\sigma_x^2 + 3\tau_{zx}^2} \tag{8.30}$$

where σ_x and τ_{zx} read:

$$\sigma_x = \frac{P + F_1 \sin\theta_1 + F_2 \sin\theta_2}{A} + \frac{d\left(F_1 L_1 \cos\theta_1 + F_2 L_2 \cos\theta_2\right)}{2I} \tag{8.31}$$

$$\tau_{zx} = \frac{Td}{2J} \tag{8.32}$$

with

$$A = \frac{\pi}{4}\left[d^2 - (d-2t)^2\right], \tag{8.33}$$

$$I = \frac{\pi}{64}\left[d^4 - (d-2t)^4\right], \tag{8.34}$$

$$J = 2I. \tag{8.35}$$

There are a total number of 11 random variables involved in this example, as listed in Table 8.5.

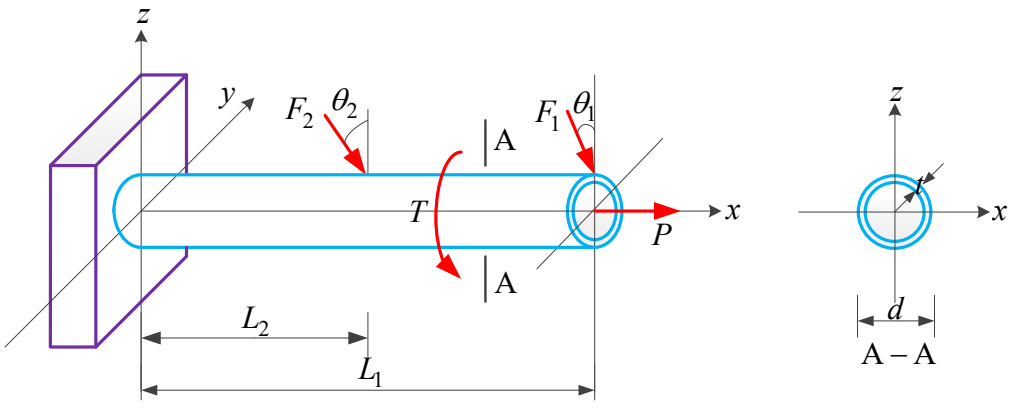


Figure 8.6: A cantilever tube subject to three forces and one torsion.

Table 8.6 compares the results of several reliability analysis methods. The reference value of the failure probability is 5.99×10^{-8} with COV being 1.29%, which is provided by the MCS method with 10^{11} samples. Both standard LS and CLS results in large COVs even using 1000 lines in this example. The number of $\mathcal{G}$-function calls can be significantly reduced by applying AK-IS and

AGPR-LS and PBAL-LS. Among them, the proposed method performs much better in terms of N_{tobal}.

Table 8.5: Random variables for Example 4.

Variable	Distribution	Mean	COV
t	Normal	5 mm	0.05
d	Normal	40 mm	0.05
L_1	Normal	120 mm	0.05
L_2	Normal	60 mm	0.05
F_1	Lognormal	2.0 kN	0.15
F_2	Lognormal	1.5 kN	0.15
P	Lognormal	10 kN	0.20
T	Lognormal	0.2 N·m	0.15
S_y	Normal	300 MPa	0.10
θ_1	Normal	10°	0.05
θ_2	Normal	10°	0.05

Table 8.6: Reliability results for Example 4 obtained from several methods.

Method	$\hat{P}_f$	COV$[\hat{P}_f]$ or $\overline{\text{COV}}[\hat{P}_f]$	N_{line}	N_{total}
MCS	5.99×10^{-8}	1.29%	-	10^{11}
Standard LS	6.62×10^{-8}	15.71%	500	1960
	4.71×10^{-8}	11.72%	1000	3944
CLS	4.47×10^{-8}	10.31%	500	2342
	5.59×10^{-8}	11.22%	1000	4882
AK-IS	5.91×10^{-8}	2.80%	-	481
AGPR-LS	5.88×10^{-8}	1.18%	107	211
Proposed PBAL-LS ($\epsilon = 10\%$)	5.92×10^{-8}	9.16%	60	175

Note: The parameter ϵ in the AGPR-LS method is set to be 1.

8.5.5 Example 5: A transmission tower

The last example involves a transmission tower subjected to lateral loads (as shown in Fig. 8.7), which is modified from [43, 49]. With the aid of a finite-element software called OpenSees, the tower is modelled as a three-dimensional nonlinear truss structure consisting of 24 nodes and 80 truss members. The geometric dimensions of the finite element model are shown in Fig. 8.7(a). For each member, the cross-sectional area is the same, denoted by A. The uniaxial Giuffrè-Menegotto-Pinto material law (Steel02 in OpenSees) is adopted, as schematically depicted in Fig. 8.7(c). At least six parameters are needed for the Steel02 model, i.e., Fy, E_0, b, $R0$, $CR1$ and $CR2$. The detailed

description of these parameters can be found in the command manual of OpenSees. In this study, the last three parameters are specified as: $R0 = 10$, $CR1 = 0.925$ and $CR2 = 0.15$. Ten lateral loads, P_1-P_{10}, are applied to the structure along the x-axis (see Fig. 8.7(a)). The performance function is defined as follows:

$$Y = g(P_1\text{-}P_{10}, A, E_0, F_y, b) = \Delta - \frac{u_{1,x} + u_{2,x}}{2}, \tag{8.36}$$

where Δ is a threshold, specified as 100 mm; $u_{1,x}$ and $u_{2,x}$ represent the horizontal displacements of nodes 1 and 2 (see Fig. 8.7(b)), respectively. $u_{1,x}$ is a function of the 14 random variables described in Table 8.7, and so is $u_{2,x}$.

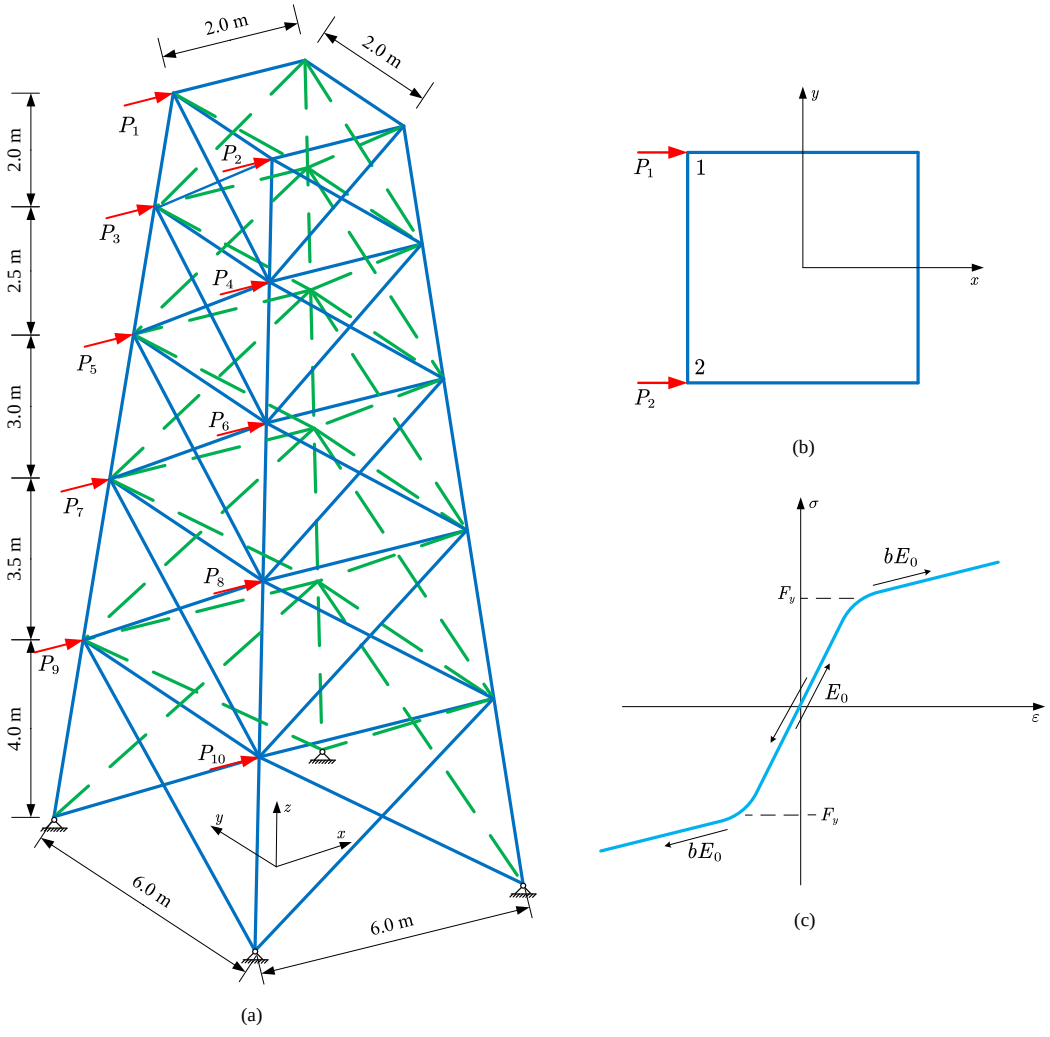


Figure 8.7: A transmission tower subject to horizontal loads.

Table 8.7: Random variables for Example 5.

Variable	Distribution	Mean	COV
P_1, P_2	Lognormal	10 kN	0.20
P_3, P_4	Lognormal	8 kN	0.20
P_5, P_6	Lognormal	6 kN	0.20
P_7, P_8	Lognormal	4 kN	0.20
P_9, P_{10}	Lognormal	2 kN	0.20
A	Normal	5000 mm^2	0.05
E_0	Normal	200 GPa	0.15
F_y	Normal	400 Mpa	0.15
b	Uniform	0.02	0.05

In this example, we cannot afford to perform the MCS method for providing a reference solution. Alternatively, the IS method provided in UQLab [50] is used due to its improved efficiency for an expensive reliability analysis. The failure probability estimate from IS is 5.56×10^{-8} with a COV being 1.00%. The results of IS are reported in Table 8.8, along with those of some other methods. The standard LS, CLS, AK-IS, and AGPR-LS methods produce errors or fail to converge on multiple trivals, so their results are not available. On the contrary, the proposed PBAL-LS method can still work and produce reasonable results with only 112 $\mathcal{G}$-function evaluations.

Table 8.8: Reliability results for Example 5 obtained from several methods.

Method	$\hat{P}_f$	$\mathrm{COV}[\hat{P}_f]$ or $\overline{\mathrm{COV}}[\hat{P}_f]$	N_{line}	N_{total}
IS	5.56×10^{-8}	1.00%	-	61,430
Standard LS	-	-	-	-
CLS	-	-	-	-
AK-IS	-	-	-	-
AGPR-LS	-	-	-	-
Proposed PBAL-LS ($\epsilon = 10\%$)	5.57×10^{-8}	9.36%	40	112

8.6 Conclusions

This paper presents a 'partially Bayesian active learning line sampling ' (PBAL-LS) method for structural reliability analysis, especially when involving small failure probabilities. The proposed method is derived from the Bayesian interpretation of the failure probability integral in the LS method, in which the discretization error is regarded as a kind of epistemic uncertainty that can be modeled explicitly. By assigning a Gaussian process prior over the β-function, the induced posterior statistics of the failure probability conditional on observations is then obtained. Two essential components for active learning, i.e., learning function and stopping criterion, are proposed by taking advantage of the uncertainty representation of the failure probability. In addition to these theoretical developments, we also design a tailored algorithm for the PBAL-LS method, which allows updating the important direction on the fly and efficiently processing the lines. Five numerical studies indicate that the proposed method outperforms several existing LS methods in one or more aspects of efficiency, accuracy and robustness when assessing extremely small failure probabilities in the order of $10^{-6} - 10^{-8}$.

While the application scope of the proposed method is large, it is mostly suitable for assessing (small) failure probabilities of weakly or moderately nonlinear problems in low-to-medium dimensions, where a single main failure domain exits. The proposed method might also be extended to cases with multiple failure domains if multiple initial important directions can be identified and updated. This will be investigated in the future work.

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). Jingwen Song acknowledges the financial support from the National Natural Science Foundation of China (grant no. 12202358 and 12220101002). Pengfei Wei is grateful to the support from the National Natural Science Foundation of China (grant no. 51905430 and 72171194). Chao Dang, Pengfei Wei and

Michael Beer also would like to appreciate the support of Sino-German Mobility Program under grant number M-0175.

Data availability

Data will be made available on request.

Bibliography

[1] Alfred M Freudenthal. Safety and the probability of structural failure. *Transactions of the American Society of Civil Engineers*, 121(1):1337–1375, 1956.

[2] Masanobu Shinozuka. Monte Carlo solution of structural dynamics. *Computers & Structures*, 2(5-6):855–874, 1972.

[3] RE Melchers. Importance sampling in structural systems. *Structural Safety*, 6(1):3–10, 1989.

[4] Svend Engelund and Ruediger Rackwitz. A benchmark study on importance sampling techniques in structural reliability. *Structural Safety*, 12(4):255–276, 1993.

[5] Robert E Melchers. Structural system reliability assessment using directional simulation. *Structural Safety*, 16(1-2):23–37, 1994.

[6] Siu-Kui Au and James L Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, 16(4):263–277, 2001.

[7] Siu-Kui Au and Yu Wang. *Engineering risk assessment with subset simulation*. John Wiley & Sons, 2014.

[8] Phaedon-Stelios Koutsourelakis, Helmuth J Pradlwarter, and Gerhart Iwo Schueller. Reliability of structures in high dimensions, part I: algorithms and applications. *Probabilistic Engineering Mechanics*, 19(4):409–417, 2004.

[9] Phaedon-Stelios Koutsourelakis. Reliability of structures in high dimensions. part II. theoretical validation. *Probabilistic Engineering Mechanics*, 19(4):419–423, 2004.

[10] Barron J Bichon, Michael S Eldred, Laura Painton Swiler, Sandaran Mahadevan, and John M McFarland. Efficient global reliability analysis for nonlinear implicit performance functions. *AIAA Journal*, 46(10):2459–2468, 2008.

[11] Benjamin Echard, Nicolas Gayton, and Maurice Lemaire. AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation. *Structural Safety*, 33(2):145–154, 2011.

[12] PS Koutsourelakis, HJ Pradlwarter, and GI Schuëller. Reliability of structures in high dimensions. In *PAMM: Proceedings in Applied Mathematics and Mechanics*, volume 3, pages 495–496. Wiley Online Library, 2003.

[13] Michael Hohenbichler and Ruediger Rackwitz. Improvement of second-order reliability estimates by importance sampling. *Journal of Engineering Mechanics*, 114(12):2195–2199, 1988.

[14] HJ Pradlwarter, MF Pellissetti, CA Schenk, GI Schueller, A Kreis, S Fransen, A Calvi, and M Klein. Realistic and efficient reliability estimation for aerospace structures. *Computer Methods in Applied Mechanics and Engineering*, 194(12-16):1597–1617, 2005.

[15] MF Pellissetti, GI Schuëller, HJ Pradlwarter, A Calvi, SHJA Fransen, and M Klein. Reliability analysis of spacecraft structures under static and dynamic loading. *Computers & Structures*, 84(21):1313–1325, 2006.

[16] L Hinke, L Pichler, HJ Pradlwarter, BR Mace, and TP Waters. Modelling of spatial variations in vibration analysis with application to an automotive windshield. *Finite Elements in Analysis and Design*, 47(1):55–62, 2011.

[17] Enrico Zio and Nicola Pedroni. Functional failure analysis of a thermal–hydraulic passive system by means of line sampling. *Reliability Engineering & System Safety*, 94(11):1764–1781, 2009.

[18] Enrico Zio and Nicola Pedroni. An optimized line sampling method for the estimation of the failure probability of nuclear passive systems. *Reliability Engineering & System Safety*, 95(12):1300–1313, 2010.

[19] HJ Pradlwarter, Gerhart Iwo Schueller, Phaedon-Stelios Koutsourelakis, and Dimos C Charmpis. Application of line sampling simulation method to reliability benchmark problems. *Structural Safety*, 29(3):208–221, 2007.

[20] Lambros S Katafygiotis and Jia Wang. Reliability analysis of wind-excited structures using domain decomposition method and line sampling. *Structural Engineering and Mechanics*, 32(1):37–53, 2009.

[21] Edoardo Patelli. COSSAN: A multidisciplinary software suite for uncertainty quantification and risk management. In *Handbook of Uncertainty Quantification*, pages 1909–1977. Springer International Publishing Switzerland, 2017.

[22] Zhenzhou Lu, Shufang Song, Zhufeng Yue, and Jian Wang. Reliability sensitivity method by line sampling. *Structural Safety*, 30(6):517–532, 2008.

[23] Marcos A Valdebenito, Hector A Jensen, HB Hernández, and L Mehrez. Sensitivity estimation of failure probability applying line sampling. *Reliability Engineering & System Safety*, 171:99–111, 2018.

[24] Marcos A Valdebenito, Herman B Hernández, and Héctor A Jensen. Probability sensitivity estimation of linear stochastic finite element models applying line sampling. *Structural Safety*, 81:101868, 2019.

[25] Xiaobo Zhang, Zhenzhou Lu, Wanying Yun, Kaixuan Feng, and Yanping Wang. Line sampling-based local and global reliability sensitivity analysis. *Structural and Multidisciplinary Optimization*, 61(1):267–281, 2020.

[26] Xiukai Yuan, Zhenxuan Zheng, and Baoqiang Zhang. Augmented line sampling for approximation of failure probability function in reliability-based analysis. *Applied Mathematical Modelling*, 80:895–910, 2020.

[27] Marco de Angelis, Edoardo Patelli, and Michael Beer. Advanced line sampling for efficient robust reliability analysis. *Structural Safety*, 52:170–182, 2015.

[28] Jingwen Song, Marcos Valdebenito, Pengfei Wei, Michael Beer, and Zhenzhou Lu. Non-intrusive imprecise stochastic simulation by line sampling. *Structural Safety*, 84:101936, 2020.

[29] Jingwen Song, Pengfei Wei, Marcos Valdebenito, and Michael Beer. Adaptive reliability analysis for rare events evaluation with global imprecise line sampling. *Computer Methods in Applied Mechanics and Engineering*, 372:113344, 2020.

[30] Jiaqi Wang, Zhenzhou Lu, Yulong Cheng, and Lu Wang. An efficient method for estimating failure probability bound functions of composite structure under the random-interval mixed uncertainties. *Composite Structures*, page 116011, 2022.

[31] Mohsen Ali Shayanfar, Mohammad Ali Barkhordari, Moien Barkhori, and Mehrollah Rakhshanimehr. An adaptive line sampling method for reliability analysis. *Iranian Journal of Science and Technology, Transactions of Civil Engineering*, 41(3):275–282, 2017.

[32] Iason Papaioannou and Daniel Straub. Combination line sampling for structural reliability analysis. *Structural Safety*, 88:102025, 2021.

[33] Marcos A Valdebenito, Pengfei Wei, Jingwen Song, Michael Beer, and Matteo Broggi. Failure probability estimation of a class of series systems by multidomain line sampling. *Reliability Engineering & System Safety*, 213:107673, 2021.

[34] Zhaoyan Lv, Zhenzhou Lu, and Pan Wang. A new learning function for kriging and its applications to solve reliability problems in engineering. *Computers & Mathematics with Applications*, 70(5):1182–1197, 2015.

[35] Ivan Depina, Thi Minh Hue Le, Gordon Fenton, and Gudmund Eiksund. Reliability analysis with metamodel line sampling. *Structural Safety*, 60:1–15, 2016.

[36] Jingwen Song, Pengfei Wei, Marcos Valdebenito, and Michael Beer. Active learning line sampling for rare event analysis. *Mechanical Systems and Signal Processing*, 147:107113, 2021.

[37] Marcos A Valdebenito, Marco de Angelis, and Edoardo Patelli. Line sampling simulation: Recent advancements and applications. In *Reliability-Based Analysis and Design of Structures and Infrastructure*, pages 215–226. CRC Press, 2021.

[38] Gerhart Iwo Schueller, Helmuth J Pradlwarter, and Phaedon-Stelios Koutsourelakis. A critical appraisal of reliability estimation procedures for high dimensions. *Probabilistic Engineering Mechanics*, 19(4):463–474, 2004.

[39] Anthony O'Hagan. Bayes–Hermite quadrature. *Journal of Statistical Planning and Inference*, 29(3):245–260, 1991.

[40] Carl Edward Rasmussen and Zoubin Ghahramani. Bayesian Monte Carlo. *Advances in Neural Information Processing Systems*, pages 505–512, 2003.

[41] Carl Edward Rasmussen and Christopher KI Williams. *Gaussian processes for machine learning*. The MIT press, 2006.

[42] Donald Bruce Owen. A table of normal integrals: A table. *Communications in Statistics-Simulation and Computation*, 9(4):389–419, 1980.

[43] Chao Dang, Marcos A. Valdebenito, Matthias G.R. Faes, Pengfei Wei, and Michael Beer. Structural reliability analysis: A Bayesian perspective. *Structural Safety*, 99:102259, 2022.

[44] Benjamin Echard, Nicolas Gayton, Maurice Lemaire, and Nicolas Relun. A combined importance sampling and kriging reliability method for small failure probabilities with time-demanding numerical models. *Reliability Engineering & System Safety*, 111:232–240, 2013.

[45] Abraham M Hasofer and Niels C Lind. Exact and invariant second-moment code format. *Journal of the Engineering Mechanics Division*, 100(1):111–121, 1974.

[46] Jun Xu and Fan Kong. A new unequal-weighted sampling method for efficient reliability analysis. *Reliability Engineering & System Safety*, 172:94–102, 2018.

[47] Christian G Bucher and Ulrich Bourgund. A fast and efficient response surface approach for structural reliability problems. *Structural Safety*, 7(1):57–66, 1990.

[48] Xiaoping Du. Unified uncertainty analysis by the first order reliability method. *Journal of Mechanical Design*, 130(9), 08 2008. 091401.

[49] Chao Dang, Pengfei Wei, Matthias GR Faes, Marcos A Valdebenito, and Michael Beer. Interval uncertainty propagation by a parallel Bayesian global optimization method. *Applied Mathematical Modelling*, 108:220–235, 2022.

[50] S. Marelli, R. Schöbi, and B. Sudret. UQLab user manual – Structural reliability (Rare event estimation). Technical report, Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland, 2022. Report UQLab-V2.0-107.

Chapter 9

Bayesian active learning line sampling for rare event estimation

Structural reliability analysis by line sampling: A Bayesian active learning treatment

Chao Dang[a,⋆], Marcos A. Valdebenito[b], Matthias G.R. Faes[b], Jingwen Song[c], Pengfei Wei[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 Mechanical Engineering, Northwestern Polytechnical University, Xi'an 710072, PR China*

[d] *School of Power and Energy, 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*

Published in Structural Safety in September 2023

Abstract: Line sampling has been demonstrated to be a promising simulation method for structural reliability analysis, especially for assessing small failure probabilities. However, its practical performance can still be significantly improved by taking advantage of, for example, Bayesian active learning. Along this direction, a recently proposed 'partially Bayesian active learning line sampling' (PBAL-LS) method has shown to be successful. This paper aims at offering a more complete Bayesian active learning treatment of line sampling, resulting in a new method called 'Bayesian active learning line sampling' (BAL-LS). Specifically, we derive the exact posterior variance of the failure probability, which can measure our epistemic uncertainty about the failure probability more precisely than the upper bound given in PBAL-LS. Further, two essential components (i.e., learning function and stopping criterion) are proposed to facilitate Bayesian active learning, based on the uncertainty representation of the failure probability. In addition, the important direction can be automatically updated throughout the simulation, as one advantage directly inherited from PBAL-

*Corresponding Author
E-mail address: chao.dang@irz.uni-hannover.de (C. Dang)

LS. The performance of BAL-LS is illustrated by four numerical examples. It is shown that the proposed method is capable of evaluating extremely small failure probabilities with desired efficiency and accuracy.

Keywords: Structural reliability analysis; Line sampling; Bayesian active learning; Bayesian inference; Gaussian process

9.1 Introduction

Structural reliability analysis usually involves calculating the complement of the so-called reliability of a structure or component, that is the failure probability P_f, which is formulated as a multiple integral:

$$P_f = \int_{\mathcal{X}} I(g(\boldsymbol{x})) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{9.1}$$

where $\boldsymbol{X} = [X_1, X_2, \cdots, X_d]^{\mathrm{T}} \in \mathcal{X} \subseteq \mathbb{R}^d$ denotes a set of d basic random variables with known joint probability density function (PDF) $f_{\boldsymbol{X}}(\boldsymbol{x})$; $\boldsymbol{x} = [x_1, x_2, \cdots, x_d]^{\mathrm{T}}$ represents a realization of $\boldsymbol{X}$; $g(\cdot)$ is the limit state function (also known as performance function), which takes a value less than zero when a failure occurs; $I(\cdot)$ refers to the failure indicator function: $I(g(\boldsymbol{x})) = 1$ if $g(\boldsymbol{x}) < 0$ and $I(g(\boldsymbol{x})) = 0$ otherwise.

Except for some special cases, the failure probability integral, as defined in Eq. (9.1), is unlikely to be analytically solvable due largely to the underlying complexity of the limit state function (usually in an implicit form) in practice. Therefore, the development of efficient and accurate numerical methods to provide approximate solutions is of central interest from researchers and practitioners. Existing numerical methods for structural reliability analysis can be roughly divided into five categories [1]: stochastic simulation methods, asymptotic approximation methods, methods of moments, probability-conservation based methods and surrogate assisted methods. Among these categories, a prominent position is held by stochastic simulation techniques. They typically involve randomly simulating a large number of independent performance function values and then computing a failure probability estimate via an appropriate estimator. A non-exhaustive list of such techniques includes Monte Carlo simulation (MCS) [2], importance sampling [3, 4], directional sampling [5, 6], subset simulation [7, 8] and line sampling (LS) [9, 10]. As the most classic class of structural reliability analysis approaches, asymptotic approximation methods attempt to derive approximate solutions

to the failure probability integral by using, e.g., Taylor series expansion. The most representative methods in this category are the first- and second order reliability methods (FORM, SORM) [11, 12]. The third category consists of methods of moments, in which the failure probability estimate is obtained by estimating the probability distribution of the state variable (i.e., output variable of the performance function) or structural response of interest from the knowledge of its moments. In this context, the integer moments-based methods [13, 14] and fractional moments-based methods [15–17] are prevalent. As the fourth category, probability-conservation based methods also aim at capturing the probability distribution of the state variable or structural response, but build upon the principle of probability conservation. The probability density evolution method [18, 19] and direct probability integral method [20, 21] are two typical examples under this category. The search for more efficient and accurate methods for structural reliability analysis also promotes the development of surrogate assisted methods, especially combined with active learning. Examples of such methods include (but not limited to) efficient global reliability analysis [22] and active learning method combining Kriging and MCS (AK-MCS) [23]. For more information about surrogate assisted methods, one can refer to, e.g. [24, 25] and references therein. Despite those great efforts over the past several decades, no agreement has been reached so far on which method or kind of methods is better than others. In fact, each method has its own advantages and disadvantages. For practical applications, one should choose the most appropriate method considering the characteristics of both the problem at hand and the candidate reliability analysis methods.

In this study, we shall restrict our attention to LS. As a standard-alone stochastic simulation method, LS was originally developed by Koutsourelakis et al. [9]. The basic idea of it is to probe the failure domain using lines, rather than random points. Specifically, the failure probability is estimated by an average of the conditional failure probabilities corresponding to a set of random lines parallel to an important direction, which points towards the failure domain. LS has been demonstrated to be a promising stochastic simulation technique that is suitable for assessing small failure probabilities of weakly or moderately nonlinear reliability problems [26–29]. However, its performance strongly depends on three main aspects [30]:

(1) The important direction. A poor important direction will lead to a slow convergence rate of the subsequent MCS procedure, and hence unnecessary computational costs in order to achieve an acceptable result. On the contrary, an optimal importance direction is always desirable, which in turn requires a good knowledge about the limit state surface or many additional g-function

evaluations.

(2) The numerical integrator. As a representative frequentist approach, the MCS method used in LS cannot make use of our prior knowledge on the limit state surface. Besides, it also shows a low convergence rate when an improper important direction is adopted and/or the limit surface around the important region is rough.

(3) The line search algorithm. To obtain each conditional failure probability, a root-finding algorithm is usually implemented. Therefore, the accuracy and efficiency of the selected root-finding algorithm also affect the overall performance of LS.

The traditional version of LS has been improved by several studies e.g., [31–33]. However, they still rely on the direct use of MCS, which can be less efficient, as discussed earlier. To further reduce the computational costs, there have been some research efforts to develop surrogate-assisted LS methods, e.g., metamodel LS [34] and adaptive Gaussian process regression-LS (AGPR-LS) [35]. More recently, the first author and his co-workers also proposed a partially Bayesian active learning LS (PBAL-LS) [30]. In PBAL-LS, estimation of the failure probability integral in LS is first interpreted as a Bayesian inference problem, where the posterior mean and an upper bound of the posterior variance for the failure probability are derived. Based on the uncertainty representation of the failure probability, a learning function and a stopping criterion that constitute two critical ingredients of active learning are then proposed to form the PBAL-LS method. Besides, the important direction in PBAL-LS can be updated on the fly throughout the simulation. To the best of knowledge of the authors, PBAL-LS is the first work that explores the Bayesian active learning (a concept originates from machine learning), at least partially, in the context of LS for structural reliability analysis.

The main objective of this work is to present a more complete Bayesian active learning treatment of LS. Specially, a full expression of the posterior variance of the failure probability in LS is deduced, which can measure our uncertainty about the failure probability more precisely than the upper bound given in [30]. The variance amplified importance sampling (VAIS) originally developed in [36] is introduced to approximate the posterior mean and variance of the failure probability, due to their analytical intractability. Based on the posterior statistics of the failure probability, we further propose a stopping criterion and a learning function to enable active learning. Besides, some advantages of PBAL-LS are also inherited, e.g., the adaption of importance direction. We shall refer to this new development as Bayesian active learning LS (BAL-LS). It is expected that the proposed

BAL-LS method can address the challenge of assessing (extremely) small failure probabilities for a class of weakly to moderately nonlinear problems in low to moderate dimensions.

The remaining of this paper is structured as follows. In Section 9.2, a general overview of several existing LS methods is given, among which two methods, i.e., standard LS and PBAL-LS, are briefly introduced. Section 9.3 presents the proposed BAL-LS method in detail. Four numerical examples are investigated in Section 9.4 to demonstrate the proposed method. Some concluding remarks are given in Section 9.5.

9.2 Literature review

This section first provides a general overview of several existing LS methods in the literature. Then, two of them, which are closely related to the proposed method, are briefly introduced.

9.2.1 General overview

LS has received a lot of attention from the structural reliability analysis community since its inception. This has led to the development of many variants of the traditional LS. We will not cover all of them, but only select some of the most important developments. The selected methods include the traditional LS [9], slime mold algorithm-assisted LS (LS-SMA) [37], advanced LS [31], adaptive LS [32], combination LS [33], multidomain LS [38], optimized LS [39], metamodel LS [34], AGPR-LS [35] and PBAL-LS [30]. They are compared in Table 9.1 regarding the important direction, numerical integrator and line search algorithm. Several aspects are worth mentioning:

- Multidomain LS allows for several important directions, while it is only applicable to a special class of series systems involving components whose response is linear with respect to a set of Gaussian random variables;

- Optimized LS adopts the ANN regression model as a surrogate of the original system model code, which is used only at the stage of determining the important direction. The failure probability is finally obtained by using the direct MCS (LHS);

- Metamodel LS formulates the failure probability estimate as a product of a metamodel-based failure probability and a correction coefficient. The former is computed from a properly-

trained Kriging model, while the latter is obtained from both the Kriging model and the original performance function;

- Overall, existing LS methods are only suitable or advantageous for a certain kind of problems with weak to moderate non-linearity.

Table 9.1: Comparison of several selected LS methods for structural reliability analysis.

Method	Important direction			Numerical integrator		Line search algorithm
	Multiple?	Adaptive?	Method	Direct MCS?	MCS on surrogate?	
Traditional LS [9]	✗	✗	MCMC	✓	✗	Not reported
LS-SMA [37]	✗	✗	FORM	✓	✗	SOPI
Advanced LS [31]	✗	✓(see [31])	Gradient (initial)	✓	✗	Newton's method
Adaptive LS [32]	✗	✓(see [32])	Not reported	✓	✗	Not reported
Combination LS [33]	✗	✓(see [33])	Gradient (initial)	✓	✗	Not reported
Multidomain LS [38]	✓(see [38])	✗	Analytical	✓	✗	See [38]
Optimized LS [39]	✗	✗	ANN assisted	✓(LHS)	✗	Not reported
Metamodel LS [34]	✗	✗	Gradient	✓	✓(Kriging)	Not reported
AGPR-LS [35]	✗	✗	FORM	✗	✓(GP)	SOPI
PBAL-LS [30]	✗	✓(see [30])	Gradient (initial)	✗	✓(GP)	Newton's method

MCMC = Markov chain Monte Carlo; SOPI = second-order polynomial interpolation; FORM = first-order reliability method; ANN = artificial neural network; LHS = Latin hypercube sampling; GP = Gaussian process

9.2.2 Brief review of two related methods

9.2.2.1 Traditional line sampling

The failure probability integral defined in Eq. (9.1) can be reformulated in the standard normal space such that:

$$P_f = \int_{\mathcal{U}} I(\mathcal{G}(\boldsymbol{u}))\varphi_{\boldsymbol{U}}(\boldsymbol{u})\mathrm{d}\boldsymbol{u}, \tag{9.2}$$

where $\boldsymbol{U} = [U_1, U_2, \cdots, U_d]^{\mathrm{T}} \in \mathcal{U} \subseteq \mathbb{R}^d$ is a vector of d i.i.d. standard normal variables with joint PDF $\varphi_{\boldsymbol{U}}(\boldsymbol{u}) = (2\pi)^{-d/2}\exp(-\boldsymbol{u}^{\mathrm{T}}\boldsymbol{u}/2)$; $\boldsymbol{u} = [u_1, u_2, \cdots, u_d]^{\mathrm{T}}$ denotes a random realization of $\boldsymbol{U}$; $\mathcal{G} = g \circ T^{-1}$ can be called a transformed limit state function; $T : \mathcal{X} \to \mathcal{U}$ represents an appropriate operator that can transform $\boldsymbol{X}$ to $\boldsymbol{U}$, e.g., an isoprobabilistic transformation.

The formulation of the traditional LS method relies on the assumption that an important direction $\boldsymbol{\alpha}$ can be identified, which is a unit vector pointing towards the failure domain $F = \{\boldsymbol{u} \in \mathcal{U} : \mathcal{G}(\boldsymbol{u}) < 0\}$, as shown in Fig. 9.1. Once $\boldsymbol{\alpha}$ is given, the d-dimensional standard normal vector $\boldsymbol{U}$ can be expressed in a rotated coordinate system such that:

$$\boldsymbol{U} = \boldsymbol{R}\boldsymbol{U}' = \boldsymbol{\alpha}U^{\parallel} + \boldsymbol{Q}\boldsymbol{U}^{\perp}, \tag{9.3}$$

where $\boldsymbol{R}$ is a $d \times d$ rotational matrix with its first row being $\boldsymbol{\alpha}^{\mathrm{T}}$ and the rest rows being $\boldsymbol{Q}^{\mathrm{T}}$; $\boldsymbol{Q}$ is a $d \times (d-1)$ matrix containing $d-1$ orthogonal basis vectors of the hyperplane perpendicular to $\boldsymbol{\alpha}$; $\boldsymbol{U}' = [U^{\parallel}, \boldsymbol{U}^{\perp\mathrm{T}}]^{\mathrm{T}} \in \mathcal{U}' \subseteq \mathbb{R}^d$ is a d-dimensional rotated standard normal vector of $\boldsymbol{U}$, due to the rotational invariance of standard normal vector; $U^{\parallel} \in \mathcal{U}^{\parallel} \subseteq \mathbb{R}$ is a standard normal variable, while $\boldsymbol{U}^{\perp} = [U_1^{\perp}, U_2^{\perp}, \cdots, U_{d-1}^{\perp}]^{\mathrm{T}} \in \mathcal{U}^{\perp} \subseteq \mathbb{R}^{d-1}$ is a $(d-1)$-dimensional standard normal vector.

It follows that the failure probability integral defined in Eq. (9.2) can be reformulated as:

$$\begin{aligned} P_f &= \int_{\mathcal{U}'} I(\mathcal{G}(\boldsymbol{R}\boldsymbol{u}'))\varphi_{\boldsymbol{U}'}(\boldsymbol{u}')\mathrm{d}\boldsymbol{u}' \\ &= \int_{\mathcal{U}^{\perp}}\int_{\mathcal{U}^{\parallel}} I(\mathcal{G}(\boldsymbol{\alpha}u^{\parallel} + \boldsymbol{Q}\boldsymbol{u}^{\perp}))\varphi_{U^{\parallel}}(u^{\parallel})\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}u^{\parallel}\mathrm{d}\boldsymbol{u}^{\perp} \\ &= \int_{\mathcal{U}^{\perp}} \left(\int_{\mathcal{U}^{\parallel}} I(\mathcal{G}(\boldsymbol{\alpha}u^{\parallel} + \boldsymbol{Q}\boldsymbol{u}^{\perp}))\varphi_{U^{\parallel}}(u^{\parallel})\mathrm{d}u^{\parallel}\right)\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}\boldsymbol{u}^{\perp} \\ &= \int_{\mathcal{U}^{\perp}} p(\boldsymbol{u}^{\perp})\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}\boldsymbol{u}^{\perp}, \end{aligned} \tag{9.4}$$

where $\varphi_{U^{\parallel}}(u^{\parallel})$ and $\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})$ are the (joint) PDF of $U^{\parallel}$ and $\boldsymbol{U}^{\perp}$; $p(\boldsymbol{u}^{\perp}) = \int_{\mathcal{U}^{\parallel}} I(\mathcal{G}(\boldsymbol{\alpha}u^{\parallel}+\boldsymbol{Q}\boldsymbol{u}^{\perp}))\varphi_{U^{\parallel}}(u^{\parallel})\mathrm{d}u^{\parallel}$

can be interpreted as a conditional failure probability given $\boldsymbol{U}^{\perp} = \boldsymbol{u}^{\perp}$, which is associated with a one-dimensional reliability problem with performance function $\mathcal{G}(\boldsymbol{\alpha}U^{\parallel} + \boldsymbol{Q}\boldsymbol{u}^{\perp})$. In case that the failure domain F is a simple half-open domain (as shown in Fig. 9.1), the conditional failure probability $p(\boldsymbol{u}^{\perp})$ is equal to:

$$p(\boldsymbol{u}^{\perp}) = \Phi(-\beta(\boldsymbol{u}^{\perp})), \tag{9.5}$$

where Φ denotes the cumulative distribution function of the standard normal variable; $\beta(\boldsymbol{u}^{\perp})$ is the Euclidean distance between $\boldsymbol{u}^{\perp}$ and the limit state surface $\mathcal{G}(\boldsymbol{u}) = 0$ along the direction $\boldsymbol{\alpha}$. Using Eq. (9.5), Eq. (9.4) is simplified as:

$$P_f = \int_{\mathcal{U}^{\perp}} \Phi(-\beta(\boldsymbol{u}^{\perp}))\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}\boldsymbol{u}^{\perp}, \tag{9.6}$$

Note that Eq. (9.6) rather than Eq. (9.4) is commonly considered in the traditional LS method, and also other improved LS methods.

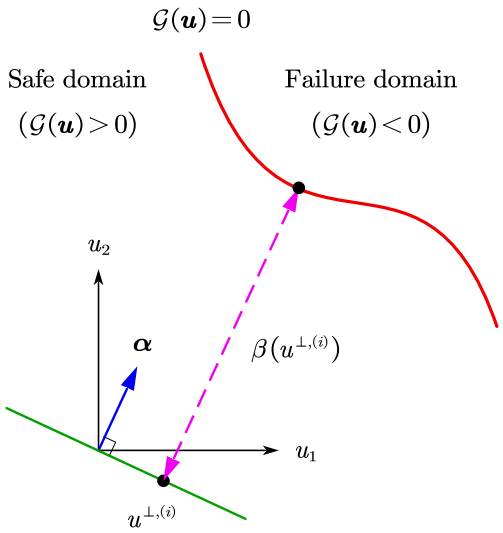


Figure 9.1: Schematic illustration of traditional LS in two dimensions.

In the traditional LS method, the failure probability integral defined in Eq. (9.6) is solved by the MCS method in conjunction with a root-finding technique. A MCS estimator of P_f is given by:

$$\hat{P}_f = \frac{1}{N}\sum_{i=1}^{N} \Phi(-\beta(\boldsymbol{u}^{\perp,(i)})), \tag{9.7}$$

where $\left\{\boldsymbol{u}^{\perp,(i)}\right\}_{i=1}^{N}$ is a set of N random samples generated according to $\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})$. For each sample $\boldsymbol{u}^{\perp,(i)}$, $\beta(\boldsymbol{u}^{\perp,(i)})$ is considered as the solution of $u^{\parallel}$ subject to $\mathcal{G}(\boldsymbol{\alpha}u^{\parallel} + \boldsymbol{Q}\boldsymbol{u}^{\perp,(i)}) = 0$ (as illustrated in Fig. 9.1), which can be solved by a suitable root-finding algorithm. The variance associated with $\hat{P}_f$ can be estimated by:

$$\mathrm{Var}\left[\hat{P}_f\right] = \frac{1}{N(N-1)}\sum_{i=1}^{N}\left(\Phi(-\beta(\boldsymbol{u}^{\perp,(i)})) - \hat{P}_f\right)^2. \tag{9.8}$$

9.2.2.2 Partially Bayesian active learning line sampling

PBAL-LS [30] offers a Bayesian active learning alternative to the traditional LS method and its variants. Specifically, the task of estimating the failure probability integral defined in Eq. (9.6) is first interpreted as a Bayesian inference problem. Then, such a task is further framed in an active learning setting based on the posterior statistics of the failure probability. Besides, another notable feature of PBAL-LS is that the importance direction needs not to be optimal at the very beginning, and it can be updated on the fly through the simulation.

PBAL-LS begins by modeling our uncertainty over the β-function with a Gaussian process (GP):

$$\tilde{\beta}_0(\boldsymbol{u}^{\perp}) \sim \mathcal{GP}(m_{\tilde{\beta}_0}(\boldsymbol{u}^{\perp}), k_{\tilde{\beta}_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{9.9}$$

where $\tilde{\beta}_0$ represents the prior distribution of β before seeing any observations; $m_{\tilde{\beta}_0}(\boldsymbol{u}^{\perp})$ and $k_{\tilde{\beta}_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$ are the prior mean and covariance functions, which are specified as a constant value and square exponential kernel [30], respectively.

Suppose that now we have an observation matrix $\boldsymbol{\mathcal{D}} = \left\{\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{\mathcal{Y}}\right\}$, where $\boldsymbol{\mathcal{U}}^{\perp} = \left\{\boldsymbol{u}^{\perp,(i)}\right\}_{i=1}^{n}$ is a $(d-1)\times n$ matrix consisting of n observed locations on the hyperplane orthogonal to the important direction, and $\boldsymbol{\mathcal{Y}} = \left\{y^{(i)}\right\}_{i=1}^{n}$ is an $n \times 1$ vector with $y^{(i)} = \beta(\boldsymbol{u}^{\perp,(i)})$. Conditioning on data $\boldsymbol{\mathcal{D}}$, the posterior distribution of β turns out to be another GP of the form:

$$\tilde{\beta}_n(\boldsymbol{u}^{\perp}) \sim \mathcal{GP}(m_{\tilde{\beta}_n}(\boldsymbol{u}^{\perp}), k_{\tilde{\beta}_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{9.10}$$

where $\tilde{\beta}_n$ denotes the posterior distribution of β conditional on n observations; $m_{\tilde{\beta}_n}(\boldsymbol{u}^{\perp})$ and $k_{\tilde{\beta}_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$ are respectively the posterior mean and covariance functions, which can be expressed

in closed form [40]:

$$m_{\tilde{\beta}_n}(\boldsymbol{u}^{\perp}) = m_{\tilde{\beta}_0}(\boldsymbol{u}^{\perp}) + \boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{\mathrm{T}}\boldsymbol{K}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{-1}(\boldsymbol{\mathcal{Y}} - \boldsymbol{m}_{\tilde{\beta}_0}(\boldsymbol{\mathcal{U}}^{\perp})), \tag{9.11}$$

$$k_{\tilde{\beta}_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) = k_{\tilde{\beta}_0}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) - \boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{\mathrm{T}}\boldsymbol{K}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{-1}\boldsymbol{k}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp},\boldsymbol{u}^{\perp\prime}), \tag{9.12}$$

where $\boldsymbol{m}_{\tilde{\beta}_0}(\boldsymbol{\mathcal{U}}^{\perp})$ is an n-by-1 mean vector whose i-th element is $m_{\tilde{\beta}_0}(\boldsymbol{u}^{\perp,(i)})$; $\boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})$ is an n-by-1 covariance vector whose i-th entry is $k_{\tilde{\beta}_0}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp,(i)})$; $\boldsymbol{k}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp},\boldsymbol{u}^{\perp\prime})$ is an n-by-1 covariance vector whose i-th entry is $\boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp,(i)},\boldsymbol{u}^{\perp\prime})$; $\boldsymbol{K}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})$ is an n-by-n covariance matrix with (i,j)-th entry being $k_{\beta_0}(\boldsymbol{u}^{\perp,(i)},\boldsymbol{u}^{\perp,(j)})$.

Through some mathematical derivations, we can arrive at the posterior mean and an upper bound of posterior variance for the failure probability [30]:

$$m_{\tilde{P}_{f,n}} = \int_{\mathbb{R}^{d-1}} \Phi\left(\frac{-m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}\boldsymbol{u}^{\perp}, \tag{9.13}$$

$$\begin{aligned}\overline{\sigma}^2_{\tilde{P}_{f,n}} =& \left(\int_{\mathbb{R}^{d-1}} \sqrt{\Phi\left(\frac{-m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)\Phi\left(\frac{m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right) - 2\mathcal{T}\left(\frac{-m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}, \frac{1}{\sqrt{1+2\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)}\right.\\ &\left.\times\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}\boldsymbol{u}^{\perp}\right)^2,\end{aligned} \tag{9.14}$$

where $\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)$ is the posterior variance function of β, i.e., $\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right) = k_{\tilde{\beta}_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp})$; $\mathcal{T}(\cdot,\cdot)$ is the Owen's T function. The posterior mean $m_{\tilde{P}_{f,n}}$ can be used naturally as the failure probability estimate, while the upper bound of posterior variance $\overline{\sigma}^2_{\tilde{P}_{f,n}}$ measures our maximum uncertainty about the estimate.

On the basis of Eq. (9.14), a learning function, called 'upper-bound posterior standard deviation contribution' (UPSDC), is proposed in [30]:

$$\begin{aligned}\mathrm{UPSDC}\left(\boldsymbol{u}^{\perp}\right) =& \sqrt{\Phi\left(\frac{-m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)\Phi\left(\frac{m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right) - 2\mathcal{T}\left(\frac{-m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}, \frac{1}{\sqrt{1+2\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)}\\ &\times\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}).\end{aligned} \tag{9.15}$$

Note that $\overline{\sigma}_{\tilde{P}_{F,n}} = \int_{\mathbb{R}^{d-1}} \mathrm{UPSDC}\left(\boldsymbol{u}^{\perp}\right)\mathrm{d}\boldsymbol{u}^{\perp}$ holds. In case that the prescribed stopping criterion

is not satisfied, the best next point is then selected by maximizing the UPSDC function, i.e., $\boldsymbol{u}^{\perp,(n+1)} = \arg\max_{\boldsymbol{u}^{\perp}\in\mathcal{U}^{\perp}} \text{UPSDC}\left(\boldsymbol{u}^{\perp}\right)$. In PBAL-LS, the stopping criterion is defined based on judging the upper bound of the posterior coefficient of variation of the failure probability [30]:

$$\overline{\text{COV}}_{\tilde{P}_{f,n}} = \frac{\overline{\sigma}_{\tilde{P}_{f,n}}}{m_{\tilde{P}_{f,n}}} < \epsilon, \tag{9.16}$$

where ϵ is a user-specified tolerance.

The interested reader is referred to [30] for theoretical and algorithmic details of PBAL-LS. It is shown that PBAL-LS outperforms most, if not all, of the existing LS methods for several benchmark problems. Despite this, PBAL-LS still belongs to a kind of PBAL method largely due to unavailability of the posterior variance for the failure probability, and a complete Bayesian active learning treatment is worth studying. The main reasons are the following. First, the upper bound of the posterior variance for the failure probability (Eq. (9.14)) might be too loose to reflect our real epistemic uncertainty about the failure probability estimate. In addition, it is difficult and even impossible for us to know to what extent the real epistemic uncertainty is magnified when using Eq. (9.14). Second, the learning function (i.e., the UPSDC function defined in Eq. (9.15)) could be less effective because it comes from the upper bound of the posterior variance of the failure probability (Eq. (9.14)), which is the result of a very strict assumption. Third, it is difficult to specify a proper tolerance ϵ (that is related to the true posterior COV of the failure probability) for the stopping criterion. A conservative choice is to set a small ϵ, which may lead to an accurate estimate for the failure probability, but usually causes unnecessary computational costs.

9.3 Bayesian active learning line sampling

In the present section, BAL-LS as an enhanced version of the previously developed PBAL-LS is introduced. First, the posterior mean and variance of the failure probability defined in Eq. (9.6) are devised so as to offer a more complete Bayesian interpretation of the standard LS. The approximate solutions for the posterior mean and variance are also given, due to their analytical intractability. Based on the posterior statistics of the failure probability, two principal elements, i.e., learning function and stopping criterion, are proposed, which enables us to offer a new Bayesian active learning treatment for the standard LS. Finally, the numerical implementation procedure

of BAL-LS is summarized, where how to adapt the important direction and process each line are explained.

9.3.1 Posterior mean and variance of the failure probability

Proposition 1. *If a GP prior is assigned to the β-function (i.e., Eq. (9.9)), the posterior mean and variance of the failure probability defined in Eq. (9.6) can be expressed as:*

$$m_{\tilde{P}_{f,n}} = \int_{\mathcal{U}^\perp} m_{\tilde{\Phi}_n(-\tilde{\beta})}\left(\boldsymbol{u}^\perp\right) \varphi_{\boldsymbol{U}^\perp}(\boldsymbol{u}^\perp)\mathrm{d}\boldsymbol{u}^\perp, \tag{9.17}$$

$$\sigma^2_{\tilde{P}_{f,n}} = \int_{\mathcal{U}^\perp}\int_{\mathcal{U}^\perp} k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^\perp, \boldsymbol{u}^{\perp\prime})\varphi_{\boldsymbol{U}^\perp}(\boldsymbol{u}^\perp)\varphi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime})\mathrm{d}\boldsymbol{u}^\perp\mathrm{d}\boldsymbol{u}^{\perp\prime} \tag{9.18}$$

where $m_{\tilde{\Phi}_n(-\tilde{\beta})}\left(\boldsymbol{u}^\perp\right)$ and $k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^\perp, \boldsymbol{u}^{\perp\prime})$ are the posterior mean and covariance functions of $\Phi(-\beta)$.

Proof. Analogy to our previous results (see Eqs. (23) and (24) in [36]), the above proposition is easy to be proved by using the Fubini's theorem. Therefore, the detailed proof is omitted here. □

9.3.1.1 Posterior mean of the failure probability

Proposition 2. *If a GP prior is placed over the β-function (i.e., Eq. (9.9)), the posterior mean function of $\Phi(-\beta)$ takes the form:*

$$m_{\tilde{\Phi}_n(-\tilde{\beta})}\left(\boldsymbol{u}^\perp\right) = \Phi\left(\frac{-m_{\tilde{\beta}_n}\left(\boldsymbol{u}^\perp\right)}{\sqrt{1+\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^\perp\right)}}\right). \tag{9.19}$$

Proof. The posterior mean function $m_{\tilde{\Phi}_n(-\tilde{\beta})}\left(\boldsymbol{u}^\perp\right)$ can be further written as:

$$\begin{aligned} m_{\tilde{\Phi}_n(-\tilde{\beta})}\left(\boldsymbol{u}^\perp\right) =& \mathbb{E}_{\tilde{\beta}_n}\left[\Phi(-\tilde{\beta}_n\left(\boldsymbol{u}^\perp\right))\right] \\ =& 1 - \int_{\infty}^{\infty} \Phi\left(-z\right) \frac{1}{\sigma_{\tilde{\beta}_n}\left(\boldsymbol{u}^\perp\right)} \varphi\left(\frac{z - m_{\tilde{\beta}_n}\left(\boldsymbol{u}^\perp\right)}{\sigma_{\tilde{\beta}_n}\left(\boldsymbol{u}^\perp\right)}\right) \mathrm{d}z \\ =& 1 - \int_{-\infty}^{\infty} \Phi\left(m_{\tilde{\beta}_n}\left(\boldsymbol{u}^\perp\right) + \sigma_{\tilde{\beta}_n}\left(\boldsymbol{u}^\perp\right) v\right) \varphi\left(v\right) \mathrm{d}v. \end{aligned} \tag{9.20}$$

Note that the following equation holds

$$\int_{-\infty}^{\infty} \Phi\left(m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)+\sigma_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right) v\right) \varphi(v)\,\mathrm{d}v=\Phi\left(\frac{m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma_{\tilde{\beta}_n}^2\left(\boldsymbol{u}^{\perp}\right)}}\right), \tag{9.21}$$

which has been given repeatedly in the literature, with or without proof. One can refer to, e.g., [41], for a proof. Substituting Eq. (9.21) into Eq. (9.20), Eq. (9.19) can be proved.

□

Substituting Eq. (9.19) into Eq. (9.17), the posterior mean of the failure probability can be obtained as in Eq. (9.13).

9.3.1.2 Posterior variance of the failure probability

Proposition 3. *If a GP prior is assumed for the β-function (i.e., Eq. (9.9)), the posterior covariance function of $\Phi(-\beta)$ is formulated as:*

$$\begin{aligned} k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) =& \mathcal{F}\left(\left[\begin{array}{c} m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right) \\ m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp\prime}\right) \end{array}\right];\left[\begin{array}{c} 0 \\ 0 \end{array}\right],\left[\begin{array}{cc} \sigma_{\tilde{\beta}_n}^2\left(\boldsymbol{u}^{\perp}\right)+1 & k_{\tilde{\beta}_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) \\ k_{\tilde{\beta}_n}(\boldsymbol{u}^{\perp\prime},\boldsymbol{u}^{\perp}) & \sigma_{\tilde{\beta}_n}^2\left(\boldsymbol{u}^{\perp\prime}\right)+1 \end{array}\right]\right) \\ & -\Phi\left(\frac{m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma_{\tilde{\beta}_n}^2\left(\boldsymbol{u}^{\perp}\right)}}\right)\Phi\left(\frac{m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp\prime}\right)}{\sqrt{1+\sigma_{\tilde{\beta}_n}^2\left(\boldsymbol{u}^{\perp\prime}\right)}}\right), \end{aligned} \tag{9.22}$$

where $\mathcal{F}$ denotes the bivariate Gaussian CDF, which does not have a closed form. Alternatively, it can be approximated by several existing numerical methods, e.g., [42].

Proof. The posterior covariance function $k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$ is further expressed as:

$$\begin{aligned}
k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) =& \mathbb{E}_{\tilde{\beta}_n}\left[\left(\Phi\left(-\tilde{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right) - \mathbb{E}_{\tilde{\beta}_n}\left[\Phi\left(-\tilde{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right)\right]\right)\left(\Phi\left(-\tilde{\beta}_n\left(\boldsymbol{u}^{\perp\prime}\right)\right) - \mathbb{E}_{\tilde{\beta}_n}\left[\Phi\left(-\tilde{\beta}_n\left(\boldsymbol{u}^{\perp\prime}\right)\right)\right]\right)\right] \\
=& \mathbb{E}_{\tilde{\beta}_n}\left[\Phi\left(-\tilde{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right)\Phi\left(-\tilde{\beta}_n\left(\boldsymbol{u}^{\perp\prime}\right)\right)\right] - \mathbb{E}_{\tilde{\beta}_n}\left[\Phi\left(-\tilde{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right)\right]\mathbb{E}_{\tilde{\beta}_n}\left[\Phi\left(-\tilde{\beta}_n\left(\boldsymbol{u}^{\perp\prime}\right)\right)\right] \\
=& \mathbb{E}_{\tilde{\beta}_n}\left[\left(1-\Phi\left(\tilde{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right)\right)\left(1-\Phi\left(\tilde{\beta}_n\left(\boldsymbol{u}^{\perp\prime}\right)\right)\right)\right] - \mathbb{E}_{\tilde{\beta}_n}\left[1-\Phi\left(\tilde{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right)\right]\mathbb{E}_{\tilde{\beta}_n}\left[1-\Phi\left(\tilde{\beta}_n\left(\boldsymbol{u}^{\perp\prime}\right)\right)\right] \\
=& \mathbb{E}_{\tilde{\beta}_n}\left[\Phi\left(\tilde{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right)\Phi\left(\tilde{\beta}_n\left(\boldsymbol{u}^{\perp\prime}\right)\right)\right] - \mathbb{E}_{\tilde{\beta}_n}\left[\Phi\left(\tilde{\beta}_n\left(\boldsymbol{u}^{\perp}\right)\right)\right]\mathbb{E}_{\tilde{\beta}_n}\left[\Phi\left(\tilde{\beta}_n\left(\boldsymbol{u}^{\perp\prime}\right)\right)\right] \\
=& \int_{-\infty}^{\infty}\Phi\left(m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)+\sigma_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)v\right)\Phi\left(m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp\prime}\right)+\sigma_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp\prime}\right)w\right)\varphi(v)\,\varphi(w)\,\mathrm{d}v\mathrm{d}w \\
& - \Phi\left(\frac{m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)\Phi\left(\frac{m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp\prime}\right)}{\sqrt{1+\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp\prime}\right)}}\right),
\end{aligned} \tag{9.23}$$

By making use of the result in [43], we have

$$\begin{aligned}
&\int_{-\infty}^{\infty}\Phi\left(m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)+\sigma_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)v\right)\Phi\left(m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp\prime}\right)+\sigma_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp\prime}\right)w\right)\varphi(v)\,\varphi(w)\,\mathrm{d}v\mathrm{d}w \\
=&\mathcal{F}\left(\begin{bmatrix} m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right) \\ m_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp\prime}\right)\end{bmatrix}; \begin{bmatrix} 0 \\ 0\end{bmatrix}, \begin{bmatrix} \sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)+1 & k_{\tilde{\beta}_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) \\ k_{\tilde{\beta}_n}(\boldsymbol{u}^{\perp\prime}, \boldsymbol{u}^{\perp}) & \sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp\prime}\right)+1\end{bmatrix}\right).
\end{aligned} \tag{9.24}$$

The proof of Eq. (9.24) is referred to the supplementary materials for [43]. Substituting Eq. (9.24) into Eq. (9.23), Eq. (9.22) can be proved.

□

The posterior variance of the failure probability can be obtained by substituting Eq. (9.22) into Eq. (9.18). It is worth mentioning that the upper bound of the posterior variance given in Eq. (9.14) can be obtained from Eq. (9.18) by using the Cauchy–Schwarz inequality for $k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})$. Thus, the upper bound is achieved only when the posterior distribution of $\Phi(-\beta)$ between any two locations is linearly dependent. This condition is very strict and hence in most practical cases $\sigma^2_{\tilde{P}_{f,n}} < \overline{\sigma}^2_{\tilde{P}_{f,n}}$ holds true. For this reason, $\sigma^2_{\tilde{P}_{f,n}}$ can be considered as a more appropriate measure of the numerical uncertainty behind the failure probability.

9.3.1.3 Approximation of the posterior mean and variance of the failure probability

The posterior mean and variance of the failure probability involves two analytically intractable integrals, as defined in Eqs. (9.17) and (9.18). In PBAL-LS [30], the posterior mean and upper

bound of the posterior variance (Eqs. (9.13) and (9.14)) are evaluated by the crude MCS in a sequential manner. Aside from the algorithmic simplicity and wide applicability, one disadvantage of the crude MCS technique is its poor sampling efficiency. In order to partially alleviate the problem, this study employs the variance-amplified importance sampling (VAIS) developed in [36] to numerically approximate the posterior mean and variance of the failure probability. Hereafter, the VAIS method is referred to as standard deviation-amplified importance sampling (SDA-IS) to avoid possible misunderstanding.

The unbiased SDA-IS estimators for $m_{\tilde{P}_{f,n}}$ and $\sigma^2_{\tilde{P}_{f,n}}$ can be given by:

$$\hat{m}_{\tilde{P}_{f,n}} = \frac{1}{N}\sum_{j=1}^{N} \Phi\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(j)}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(j)}\right)}}\right)\frac{\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp,(j)})}{\varphi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp,(j)})}, \tag{9.25}$$

$$\hat{\sigma}^2_{\tilde{P}_{f,n}} = \frac{1}{N}\sum_{j=1}^{N} k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^{\perp,(j)},\boldsymbol{u}^{\perp\prime,(j)})\frac{\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp,(j)})\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp\prime,(j)})}{\varphi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp,(j)})\varphi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp\prime,(j)})}, \tag{9.26}$$

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

$$\varphi_{U_i^{\perp},\lambda}(u_i^{\perp}) = \frac{1}{\lambda\sqrt{2\pi}}\exp\left(-\frac{u_i^{\perp,2}}{2\lambda^2}\right), \tag{9.27}$$

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

The variances associated with the above two estimators are expressed as:

$$\mathbb{V}\left[\hat{m}_{\tilde{P}_{f,n}}\right] = \frac{1}{N(N-1)}\sum_{j=1}^{N}\left[\Phi\left(\frac{-m_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(j)}\right)}{\sqrt{1+\sigma^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(j)}\right)}}\right)\frac{\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp,(j)})}{\varphi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp,(j)})} - \hat{m}_{\tilde{P}_{f,n}}\right]^2, \tag{9.28}$$

$$\mathbb{V}\left[\hat{\sigma}^2_{\tilde{P}_{f,n}}\right] = \frac{1}{N(N-1)}\sum_{j=1}^{N}\left[k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^{\perp,(j)},\boldsymbol{u}^{\perp\prime,(j)})\frac{\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp,(j)})\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp\prime,(j)})}{\varphi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp,(j)})\varphi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp\prime,(j)})} - \hat{\sigma}^2_{\tilde{P}_{f,n}}\right]^2. \tag{9.29}$$

Note that even though the SDA-IS method only works with the GP posterior, rather than the typically expensive-to-evaluate β function, it can be relatively time-consuming, especially when approximating the posterior variance due to the necessity of numerically evaluating the bivariate Gaussian CDF. To guarantee the accuracy and efficiency, it is suggested to implement the SDA-IS method in a sequential way. That is, we can sequentially increase the sample size (e.g., 1×10^4,

2×10^4, ...) until the two COVs of the estimators are respectively smaller than the prescribed tolerances δ_1 and δ_2, i.e., $\sqrt{\mathbb{V}\left[\hat{m}_{\tilde{P}_{f,n}}\right]}/\hat{m}_{\tilde{P}_{f,n}} < \delta_1$ and $\sqrt{\mathbb{V}\left[\hat{\sigma}^2_{\tilde{P}_{f,n}}\right]}/\hat{\sigma}^2_{\tilde{P}_{f,n}} < \delta_2$.

9.3.2 Stopping criterion and learning function

In terms of the second-order statistics, we have so far completed a Bayesian treatment of the failure probability integral defined in Eq. (9.6). That is, once given data $\boldsymbol{\mathcal{D}} = \left\{\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{\mathcal{Y}}\right\}$, we can make Bayesian inference about the failure probability, including the posterior mean and variance. It is noted that $\boldsymbol{\mathcal{U}}^{\perp}$ can be arbitrarily chosen without specified restrictions in theory. The Bayesian interpretation also allows us to frame the failure probability integral estimation in a Bayesian active learning setting, based on the full exploitation of the posterior statistics of the failure probability. This framework consists mainly of a stopping criterion and a learning function.

9.3.2.1 Stopping criterion

The stopping criterion can be naturally defined based on the estimated posterior COV of the failure probability such that:

$$\widehat{\mathrm{COV}}_{\tilde{P}_{f,n}} = \frac{\hat{\sigma}_{\tilde{P}_{f,n}}}{\hat{m}_{\tilde{P}_{f,n}}} < \eta, \tag{9.30}$$

where η is a user-defined threshold. As both $\hat{\sigma}_{\tilde{P}_{f,n}}$ and $\hat{m}_{\tilde{P}_{f,n}}$ may process some approximation errors to some extent, Eq. (9.30) is required to satisfy twice in a row in order to avoid possible fake convergence. Compared to the upper bound of posterior COV defined in Eq. (9.16), the posterior COV is a more suitable quality that can measure the extent of variability in relation to the posterior mean of the failure probability. This makes it easier to specify an appropriate threshold for the stopping criterion before running the method.

9.3.2.2 Learning function

In case that the above stopping criterion is not met, a learning function is needed to suggest the best next point to query the β-function, rather than choosing it arbitrarily. Based on the posterior variance of the failure probability, a new learning function, termed 'posterior standard deviation contribution' (PSDC), is proposed:

$$\mathrm{PSDC}\left(\boldsymbol{u}^{\perp}\right) = \varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}) \times \int_{\mathcal{U}^{\perp}} k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) \varphi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime}) \mathrm{d}\boldsymbol{u}^{\perp\prime}. \tag{9.31}$$

It is easy to check that $\int_{\mathcal{U}^\perp} \mathrm{PSDC}\left(\boldsymbol{u}^\perp\right) \mathrm{d}\boldsymbol{u}^\perp = \sigma^2_{\tilde{P}_{f,n}}$ holds true. In this respect, the PSDC function can be regarded as a measure of the contribution of the numerical uncertainty at site $\boldsymbol{u}^\perp$ to the posterior variance (or rather the posterior standard deviation) of the failure probability. It is worth mentioning that the UPSDC function (Eq. (9.15)) is actually an upper bound of the proposed PSDC function. Besides, the UPSDC function only includes the posterior variance function of $\Phi(-\beta)$, not the posterior covariance function, which can reveal spatial correlation, while the proposed PSDC does. Therefore, the PSDC function provides a more informative indicator regarding the degree of contribution of a specific realization in the sample space towards the level of epistemic uncertainty associated with the failure probability. By selecting the point maximizing the PSDC function as the best next point to evaluate the β function, it is expected that $\sigma^2_{\tilde{P}_{f,n+1}}$ will be reduced the most. This involves an optimization problem, where one should note that an analytically intractable integral is involved in the objective function (i.e., the PSDC function).

In this study, we propose to approximate the integral term in Eq. (9.31) by a numerical integration scheme, called unscented transformation [44]. In this context, the PSDC function can be approximated by the following expression:

$$\widehat{\mathrm{PSDC}}\left(\boldsymbol{u}^\perp\right) = \varphi_{\boldsymbol{U}^\perp}(\boldsymbol{u}^\perp) \sum_{i=0}^{2(d-1)} w_i k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^\perp, \boldsymbol{u}^{\perp\prime,(i)}), \tag{9.32}$$

where the $2(d-1)+1$ integration points and weights are given by [44]

$$\begin{aligned} \boldsymbol{u}^{\perp\prime,(0)} &= \boldsymbol{0}, w_0 = \frac{\varrho}{d-1+\varrho}, \\ \boldsymbol{u}^{\perp\prime,(i)} &= \sqrt{d-1+\varrho}\boldsymbol{e}_i, w_i = \frac{1}{2(d-1+\varrho)}, \\ \boldsymbol{u}^{\perp\prime,(i+d-1)} &= -\sqrt{d-1+\varrho}\boldsymbol{e}_i, w_{i+d-1} = \frac{1}{2(d-1+\varrho)}, \end{aligned} \tag{9.33}$$

where ϱ is a free parameter specified as $\varrho = 3-(d-1)- = 4-d$ [44]; $\boldsymbol{e}_i = [0, \cdots, 0, 1, \cdots, 0]$, $i = 1, 2, \cdots, d-1$.

The best next point is identified by maximizing the $\widehat{\mathrm{PSDC}}$ function such that:

$$\boldsymbol{u}^{\perp,(n+1)} = \underset{\boldsymbol{u}^\perp \in \mathcal{U}^\perp}{\arg\max} \log \widehat{\mathrm{PSDC}}\left(\boldsymbol{u}^\perp\right). \tag{9.34}$$

It should be pointed out that each evaluation of $\widehat{\mathrm{PSDC}}\left(\boldsymbol{u}^\perp\right)$ can still be slightly computationally

demanding, though the integral involved in the PSDC $\left(\boldsymbol{u}^{\perp}\right)$ function is approximated by only using $2(d-1)+1$ points. In this work, we apply a commonly-used nature-inspired optimization method, called particle swarm optimization, but other more efficient techniques can also be explored.

Once $\boldsymbol{u}^{\perp,(n+1)}$ is obtained, $y^{(n+1)} = \beta\left(\boldsymbol{u}^{\perp,(n+1)}\right)$ can be evaluated according the method described in the coming subsection. It follows that the previous dataset can be enriched with $\left\{\boldsymbol{u}^{\perp,(n+1)}, y^{(n+1)}\right\}$, and one can make Bayesian inference about the failure probability based on the enriched data.

9.3.3 Step-by-step procedure of the proposed method

The above two subsections only focus on several important ingredients (e.g., the posterior variance, learning function and stopping criterion), while there are still some aspects left for implementing the proposed method, such as the important direction and evaluation of β function. Due to length limitation, these aspects are directly embedded in the numerical implementation procedure of the proposed method in the following.

The procedure of the proposed BAL-LS method consists of six main steps, as illustrated by the flowchart (Fig. 9.3) and summarized below:

Step 1: Choosing an initial important direction

The proposed BAL-LS method has to been initialized with an initial important direction $\boldsymbol{\alpha}^{(0)}$. As suggested in PAL-LS, a convenient choice is the negative normalized gradient of the $\mathcal{G}$-function at the origin [30]:

$$\boldsymbol{\alpha}^{(0)} = -\frac{\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0})}{||\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0})||}, \tag{9.35}$$

where $\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0}) = \left[\frac{\partial\mathcal{G}(\mathbf{0})}{\partial u_1}, \frac{\partial\mathcal{G}(\mathbf{0})}{\partial u_2}, \cdots, \frac{\partial\mathcal{G}(\mathbf{0})}{\partial u_d}\right]$; $||\cdot||$ denotes the 2-norm. In case that the gradient information of $\mathcal{G}$ is not available, one can simply apply the numerical differentiation method at the cost of $(d+1)$ $\mathcal{G}$-function evaluations. After that, the corresponding matrix $\boldsymbol{Q}^{(0)}$ that defines the orthogonal hyperplane perpendicular to $\boldsymbol{\alpha}^{(0)}$ can be specified by means of, e.g., the Gram–Schmidt process.

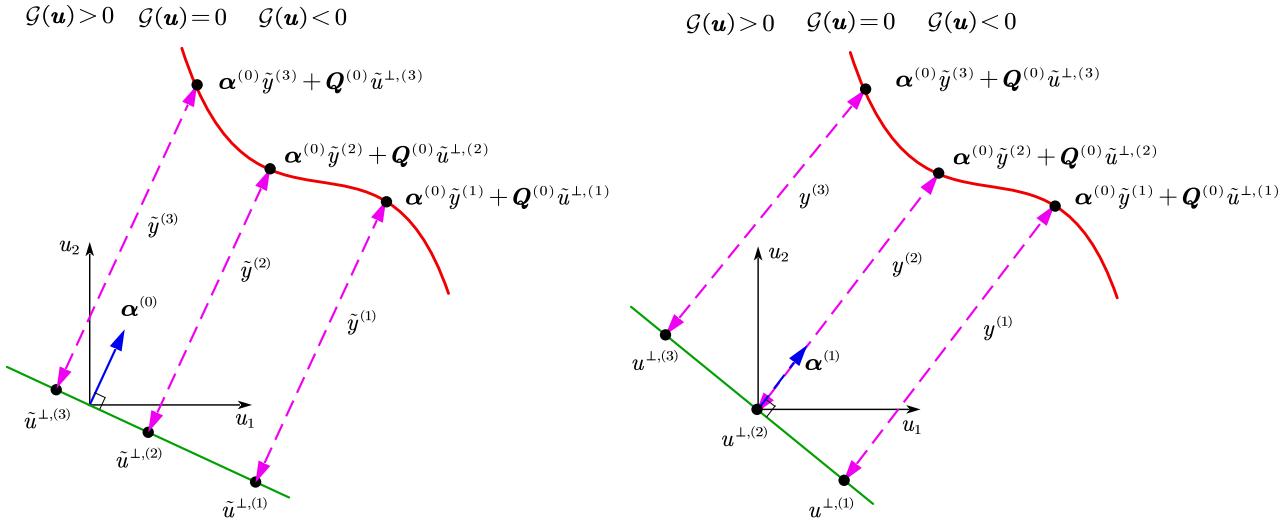
Step 2: Constructing an initial observation dataset and updating the important direction

This step involves generating an initial observation dataset $\boldsymbol{\mathcal{D}}$ from the β function and updating

the important direction. First, a small number of samples (say $n_0 = 5$) on the hyperplane orthogonal to $\boldsymbol{\alpha}^{(0)}$ are generated according to $\varphi_{\boldsymbol{U}^\perp,\lambda}(\boldsymbol{u}^\perp)$ by using the, e.g., Latin hypercube sampling. These samples are denoted as $\tilde{\boldsymbol{\mathcal{U}}}^\perp = \left\{\tilde{\boldsymbol{u}}^{\perp,(i)}\right\}_{i=1}^{n_0}$, each of which can form a line parallel to $\boldsymbol{\alpha}^{(0)}$, i.e., $\boldsymbol{\alpha}^{(0)}u^\parallel + \boldsymbol{Q}^{(0)}\tilde{\boldsymbol{u}}^{\perp,(i)}$. Second, one has to find the distance between $\tilde{\boldsymbol{u}}^{\perp,(i)}$ and the limit state surface $\mathcal{G} = 0$ along $\boldsymbol{\alpha}^{(0)}$, which is identical to fining the root of $\mathcal{G}\left(\boldsymbol{\alpha}^{(0)}u^\parallel + \boldsymbol{Q}^{(0)}\tilde{\boldsymbol{u}}^{\perp,(i)}\right) = 0$. In this study, we develop an adaptive inverse interpolation (AII) method for solving the aforementioned equation. The idea is to use the cubic spline interpolation to approximate the inverse of $\mathcal{G}$ along the direction $\boldsymbol{\alpha}^{(0)}$. To get started, two values $z^{(1)}$ and $z^{(2)}$ of $\mathcal{G}\left(\boldsymbol{\alpha}^{(0)}u^\parallel + \boldsymbol{Q}^{(0)}\tilde{\boldsymbol{u}}^{\perp,(i)}\right)$ at two prescribed points (say $u^{\parallel,(1)}$ and $u^{\parallel,(2)}$) are determined. As a convenient rule of thumb, $u^{\parallel,(1)}$ and $u^{\parallel,(2)}$ in this study are set to be 3 and 7, respectively. A rough root (denote as $u^{\parallel,(3)}$) can be found by performing a cubic spline interpolation of the two data points $\left(z^{(1)}, u^{\parallel,(1)}\right)$ and $\left(z^{(2)}, u^{\parallel,(2)}\right)$ at $z = 0$, and the third value $z^{(3)}$ is obtained by evaluating $\mathcal{G}\left(\boldsymbol{\alpha}^{(0)}u^{\parallel,(3)} + \boldsymbol{Q}^{(0)}\tilde{\boldsymbol{u}}^{\perp,(i)}\right)$. One can identify the next approximate root $u^{\parallel,(4)}$ by interpolating the three data points $\left(z^{(1)}, u^{\parallel,(1)}\right)$, $\left(z^{(2)}, u^{\parallel,(2)}\right)$ and $\left(z^{(3)}, u^{\parallel,(3)}\right)$ at $z = 0$. The process is repeated until the relative distance of two consecutive approximate roots is less than a small threshold γ (e.g., 5%), i.e., $\left|u^{\parallel,(j+1)} - u^{\parallel,(j)}\right| / u^{\parallel,(j)} < \gamma$, $j = 3, 4, \cdots$. Typically, the stopping criterion can be reached after several iterations. The final n_0 roots corresponding to $\tilde{\boldsymbol{\mathcal{U}}}^\perp$ are denoted as $\tilde{\boldsymbol{\mathcal{Y}}} = \left\{\tilde{y}^{(i)}\right\}_{i=1}^{n_0}$, and each approximate intersection point of the line $\boldsymbol{\alpha}^{(0)}u^\parallel + \boldsymbol{Q}^{(0)}\tilde{\boldsymbol{u}}^{\perp,(i)}$ and the limit state surface $\mathcal{G} = 0$ is recorded as $\boldsymbol{\alpha}^{(0)}\tilde{y}^{(i)} + \boldsymbol{Q}^{(0)}\tilde{\boldsymbol{u}}^{\perp,(i)}$. Third, a new important direction $\boldsymbol{\alpha}^{(1)}$ is identified as the normalized vector of the approximate intersection point with the shortest distance to the origin, and the associated matrix $\boldsymbol{Q}^{(1)}$ can be specified. Fourth, one can obtain the initial observation dataset $\boldsymbol{\mathcal{D}} = \left\{\boldsymbol{\mathcal{U}}^\perp, \boldsymbol{\mathcal{Y}}\right\}$ simply by projecting those n_0 approximate intersection points on the hyperplane orthogonal to $\boldsymbol{\alpha}^{(1)}$. Let $n = n_0$ and $q = 1$. It is worth mentioning that one does not need to re-evaluate the $\mathcal{G}$ function, though the important direction is changed in this step. For a schematic illustration of this step, one can refer to Fig. 9.2.

Step 3: Making Bayesian inference about the failure probability

Conditional on the observation dataset $\boldsymbol{\mathcal{D}}$, the posterior mean and variance of the failure probability can be inferred. To do so, the posterior mean and covariance functions of the β-function are first obtained by Eqs. (9.11) and (9.12), and this task can be done by using the *fitrgp* function in Statistics and Machine Learning Toolbox of Matlab. In this study, the prior mean function is assumed to be a constant value and the prior covariance function adopts the squared exponential kernel with a separate length scale per dimension. The hyper-parameters are



(a) Before changing important direction

(b) After changing important direction

Figure 9.2: Schematic illustration of Step 2 of the proposed BAL-LS algorithm in two dimensions ($n_0 = 3$).

determined by the maximum likelihood estimation. The posterior mean and variance estimates of the failure probability are then computed by the SDA-IS method in a sequential manner, as described in section 9.3.1.3. The SDA factor λ is set to be 1.5, and two tolerances δ_1 and δ_2 are specified as 1% and 10%, respectively.

Step 4: Checking the stopping criterion

If the stopping criterion given in Eq. (9.30) is satisfied twice in a row, then go to **Step 5**. Otherwise, go to **Step 6**. In this study, the associated threshold η is taken as 5%.

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

The best next point $\tilde{\boldsymbol{u}}^{\perp,(n+1)}$ to evaluate the β-function is identified by maximizing the proposed $\widehat{\text{PSDC}}$ function, according to Eq. (9.34). The β-function value $\tilde{y}^{(n+1)}$ at $\tilde{\boldsymbol{u}}^{\perp,(n+1)}$ can be obtained by solving the equation $\mathcal{G}\left(\boldsymbol{\alpha}^{(q)}u^{\parallel}+\boldsymbol{Q}^{(q)}\tilde{\boldsymbol{u}}^{\perp,(n+1)}\right)=0$. Different from **Step 2**, the Newton's method is used here with a starting point taken as $m_{\tilde{\beta}_n}\left(\tilde{\boldsymbol{u}}^{\perp,(n+1)}\right)$ [30]. Once $\tilde{y}^{(n+1)}$ is solved, a new approximate intersection point $\boldsymbol{\alpha}^{(q)}\tilde{y}^{(n+1)}+\boldsymbol{Q}^{(q)}\tilde{\boldsymbol{u}}^{\perp,(n+1)}$ is available. As long as the new point is the nearest to the origin among all the $n+1$ intersection points, the important direction is updated to $\boldsymbol{\alpha}^{(q+1)}=\left(\boldsymbol{\alpha}^{(q)}\tilde{y}^{(n+1)}+\boldsymbol{Q}^{(q)}\tilde{\boldsymbol{u}}^{\perp,(n+1)}\right)/||\boldsymbol{\alpha}^{(q)}\tilde{y}^{(n+1)}+\boldsymbol{Q}^{(q)}\tilde{\boldsymbol{u}}^{\perp,(n+1)}||$. After that, a new matrix $\boldsymbol{Q}^{(q+1)}$ can be specified. The enriched observation dataset can be obtained by projecting the $n+1$ intersection points on the latest hyperplane orthogonal to $\boldsymbol{\alpha}^{(q+1)}$ and let $q=q+1$. Otherwise, one

can simply enrich the previous dataset with $(\tilde{\boldsymbol{u}}^{\perp,(n+1)}, \tilde{y}^{(n+1)})$. Let $n = n + 1$ and go to **Step 3**.

Step 6: Ending the algorithm

Return the last posterior mean estimate of the failure probability and end the algorithm.

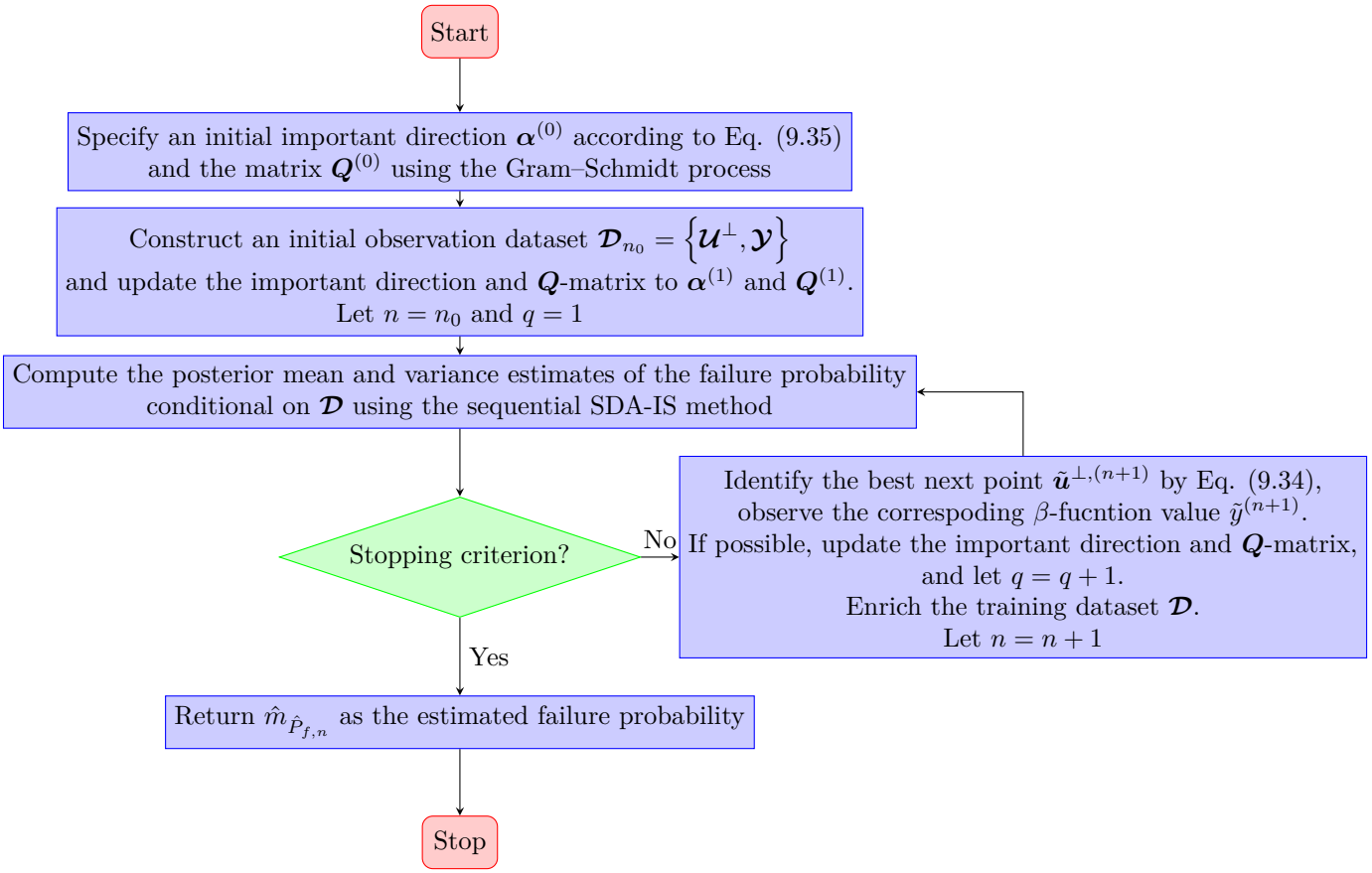


Figure 9.3: Flowchart of the proposed BAL-LS method.

9.4 Numerical examples

The performance of the proposed BAL-LS method is demonstrated in this section by means of four numerical examples. The reference result of the failure probability for each example is produced by the crude MCS method with a sufficiently large sample size when applicable. For comparison purposes, we also implement several existing methods, including sequential quadratic programming (SQP) based FORM [45] (denoted as FORM-SQP), SORM [12], traditional LS [9], AGPR-LS [35] and PBAL-LS [30]. All methods expcet PBAL-LS are based on the use of FORM-SQP to provide the most probable point (MPP) if applicable. Otherwise, FORM-HLRF [11] is applied instead. For

repeatability, the initial points of FORM-SQP and FORM-HLRF are selected as the origin. For traditional LS, the Newton's method is adopted for processing each line. Similar to the proposed BAL-LS, the stopping criterion in PBAL-LS is also required to meet twice in succession and the tolerance is set to be 5%. Note that even though the gradient information for some numerical examples is easy to solve analytically, we treat them as black-box problems.

9.4.1 Example 1: A test function

The first numerical example takes a test function of the form [30]:

$$Z = g(\boldsymbol{X}) = a - X_2 + bX_1^3 + c\sin(dX_1), \tag{9.36}$$

where X_1 and X_2 are two i.i.d. standard normal variables; a, b, c and d are four constant parameters, the values of which are set as $a = 5.5$, $b = 0.02$ and $c = \frac{5}{6}$, $d = \frac{\pi}{3}$.

The reference value of the failure probability is 3.57×10^{-7} (with a COV being 0.53%), which is provided by MCS with 10^{11} samples. The proposed method is compared to several other methods, as summarized in Table 9.2. FORM-SQP only requires 28 performance function evaluations, which, however, produces a poor failure probability estimate. The poor accuracy of FORM-SQP can be significantly improved by using SORM, with 7 additional performance function calls. The traditional LS method is carried out twice with two different numbers of lines (i.e., 100 and 200). In both cases, the traditional LS method is able to yield more accurate results than FORM-SQP, but it requires considerably more computational costs. By using AGPR-LS, PBAL-LS and BAL-LS, the number of lines and performance function calls can be significantly reduced, while maintaining reasonable accuracy. Compared to AGPR-LS and PBAL-LS, the proposed BAL-LS method is more efficient in terms of N_{call}.

For illustration purposes, Fig. 9.4 shows some computational details of the proposed BAL-LS method, including the initial and final importance directions, and approximate intersections points. It can be seen that the initial importance direction given by Eq. (9.35) is far from optimal, while the final one is almost optimal. This indicates the effectiveness of the proposed learning function for suggesting next best points to query, as well as the developed strategy for automatically updating the importance direction. What is more, those approximate intersection points are very close to the true limit state line, implying the accuracy of the proposed line search algorithm.

Table 9.2: Results for Example 1 by several different methods.

Method	$\hat{P}_f$	$\delta_{\hat{P}_f}$ or $\overline{\delta}_{\hat{P}_f}$	N_{line}	N_{total}
MCS	3.57×10^{-7}	0.53%	-	10^{11}
FORM-SQP	7.19×10^{-7}	-	-	28
SORM	3.53×10^{-7}	-	-	35
Traditional LS	3.36×10^{-7}	7.56%	100	376
	3.70×10^{-7}	4.66%	200	706
AGPR-LS	3.63×10^{-7}	2.24%	10	46
PBAL-LS	3.56×10^{-7}	1.60%	14	40
Proposed BAL-LS	3.56×10^{-7}	3.40%	8	30

Note: $\hat{P}_f$ = failure probability estimate; $\delta_{\hat{P}_f}$ = COV of $\hat{P}_f$; $\overline{\delta}_{\hat{P}_f}$ = upper bound of the COV of $\hat{P}_f$, which is only used for PBAL-LS; N_{line} = the number of lines; N_{total} = the total number of performance function calls.

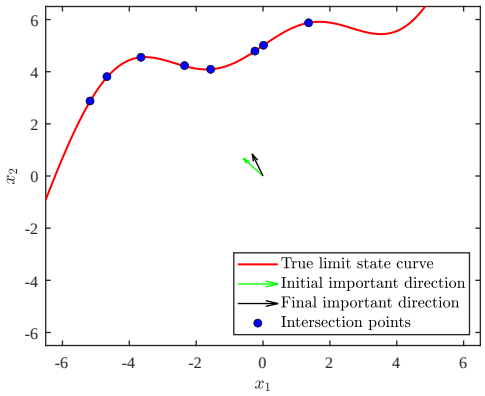


Figure 9.4: Illustration of the proposed BAL-LS method for Example 1.

9.4.2 Example 2: A nonlinear oscillator

A nonlinear single-degree-of-freedom (SDOF) oscillator under a rectangular pulse load [46] is considered as a second example, which is shown in Fig. 9.5. The limit state function is given by:

$$Z = g\left(m, k_1, k_2, r, F_1, t_1\right) = 3r - \left|\frac{2F_1}{k_1 + k_2}\sin\left(\frac{t_1}{2}\sqrt{\frac{k_1 + k_2}{m}}\right)\right|, \tag{9.37}$$

where m, k_1, k_2, r, F_1 and t_1 are six random variables, as listed in Table 9.3.

A reference solution to the failure probability is obtained as 4.01×10^{-8} (with a COV being 0.50%), generated by MCS with 10^{12} samples. The proposed BAL-LS method is conducted to assess

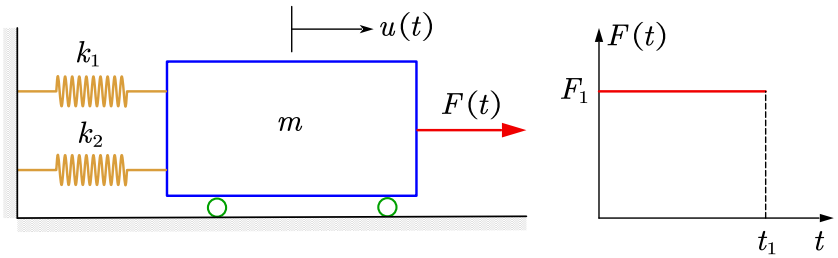


Figure 9.5: A nonlinear SDOF oscillator driven by a rectangular pulse load.

Table 9.3: 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

the failure probability, as well as several other methods, i.e., FORM-SQP, SORM, traditional LS, AGPR-LS and PBAL-LS. The key results of these methods are summarized in Table 9.4. Similar to the first example, FORM-SQP still produces an inaccurate failure probability estimate (i.e., 4.88×10^{-8}) even at the cost of 176 $\mathcal{G}$-function evaluations in this example. With more calls to the $\mathcal{G}$-function, SORM can produce an accurate failure probability estimate, say 4.08×10^{-8}. The traditional LS method can improve the accuracy of FORM-SQP by using a number of additional lines to probe the failure domain, which in turn leads to the significant increase in computational costs. AGPR-LS, PBAL-LS and BAL-LS are able to produce failure probability estimates with desirable accuracy. Among them, AGPR-LS requires the most performance function calls (say 205), while the proposed BAL-LS method requires the fewest (say 39).

9.4.3 Example 3: A reinforced concrete section

For the third example, we consider the bending limit state of a reinforced concrete section [47, 48], as shown in Fig. 9.6. The limit state function is expressed as:

$$Z = g(\boldsymbol{X}) = X_1 X_2 X_3 - \frac{X_1^2 X_2^2 X_4}{X_5 X_6} - X_7, \tag{9.38}$$

where X_1 to X_7 are seven basic random variables, as detailed in Table 9.5.

Table 9.4: Results for Example 2 by several different methods.

Method	$\hat{P}_f$	$\delta_{\hat{P}_f}$ or $\overline{\delta}_{\hat{P}_f}$	N_{line}	N_{total}
MCS	4.01×10^{-8}	0.50%	-	10^{12}
FORM-SQP	4.88×10^{-8}	-	-	176
SORM	4.08×10^{-8}	-	-	219
Traditional LS	4.22×10^{-8}	2.83%	50	376
	4.09×10^{-8}	2.10%	100	576
AGPR-LS	3.93×10^{-8}	0.81%	21	205
PBAL-LS	4.14×10^{-8}	3.76%	22	62
Proposed BAL-LS	4.07×10^{-8}	1.13%	13	39

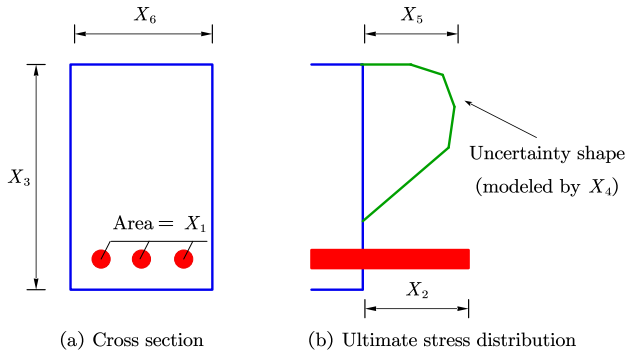


Figure 9.6: Ultimate stress state for the reinforced concrete section.

As indicated by the reference result from the crude MCS method, this example also constitutes a situation where the probability of failure is extremely small, say 1.57×10^{-8}. Table 9.6 reports the main results of several selected methods. As seen, the failure probability estimate given by FORM-SQP is less accurate; however, it requires a total number of 157 $\mathcal{G}$-function calls. With 214 $\mathcal{G}$-function calls, SORM gives a less accurate value of the failure probability estimate, say 1.44×10^{-8}. The accuracy of FORM-SQP can be further improved by the traditional LS method by generating

Table 9.5: Basic random variables for Example 3.

Variable	Description	Distribution	Mean	COV
X_1	Area of reinforcement	Normal	1260 mm^2	0.05
X_2	Yield stress of reinforcement	Lognormal	300 N/mm^2	0.10
X_3	Effective depth of reinforcement	Normal	770 mm	0.05
X_4	Stress–strain factor of concrete	Lognormal	0.35	0.10
X_5	Compressive strength of concrete	Lognormal	30 N/mm^2	0.15
X_6	Width of section	Normal	400 mm	0.05
X_7	Applied bending moment	Lognormal	80 kN·m	0.20

additional lines, which leads to increased $\mathcal{G}$-function evaluations at the same time. AGPR-LS is able to provide an accurate failure probability estimate with 8 additional lines, while relying on the MPP provided by FORM-SQP, which requires 157 additional calls to the $\mathcal{G}$-function. Both PBAL-LS and BAL-LS can give desirable results, but BAL-LS requires less lines and $\mathcal{G}$-function calls.

Table 9.6: Results for Example 3 by several different methods.

Method	$\hat{P}_f$	$\delta_{\hat{P}_f}$ or $\overline{\delta}_{\hat{P}_f}$	N_{line}	N_{total}
MCS	1.57×10^{-8}	2.53%	-	10^{11}
FORM-SQP	1.46×10^{-8}	-	-	157
SORM	1.44×10^{-8}	-	-	214
Traditional LS	1.59×10^{-8}	1.63%	10	164
	1.59×10^{-8}	1.43%	20	204
AGPR-LS	1.53×10^{-8}	0.54%	8	173
PBAL-LS	1.58×10^{-8}	3.72%	15	55
Proposed BAL-LS	1.58×10^{-8}	0.21%	12	40

9.4.4 Example 4: A transmission tower structure

The last example consists of a transmission tower structure subject to horizontal and oblique loads, as shown in Fig. 9.7. Using OpenSees [49], the structure is modeled as a three-dimensional truss with 41 nodes and 148 elements. The geometric dimensions of the model are marked in Fig. 9.7 (a) and (b). The limit state function is defined by:

$$Z = g(\boldsymbol{X}) = \Delta - H_1(F_1, F_2, F_3, F_4, F_5, \theta_1, \theta_2, \theta_3, \theta_4, E, A), \tag{9.39}$$

where Δ denotes a threshold, specified as 50 mm; H_1 represents the horizontal displacement on x-axis of the top node, which is a function of 11 random variables as given in Table 9.7.

The crude MCS method is not likely to be affordable for providing a reference solution in this example. For this reason, we implement important sampling (IS) [50] as an alternative. The failure probability given by IS is 6.04×10^{-6} with a COV being 1.00%. In this example, FORM-SQP does not converge to the correct result, while FORM-HLRF does. The results from IS, FORM-HLRF, SORM, traditional LS, AGPR-LS, PBAL-LS and BAL-LS are reported in Table 9.8. Both FORM-HLRF and SORM give inaccurate failure probability estimates. Traditional LS can improve the accuracy of FORM-HLRF by employing additional lines to probe the failure domain, while requiring

Table 9.7: Basic random variables for Example 4.

Variable	Description	Distribution	Mean	STD
F_1	Oblique load (in xz - plane)	Lognormal	50 kN	10 kN
F_2	Oblique load (in xz - plane)	Lognormal	50 kN	10 kN
F_3	Oblique load (in xz - plane)	Lognormal	60 kN	12 kN
F_4	Oblique load (in xz - plane)	Lognormal	60 kN	12 kN
F_5	Horizontal load (on x-axis)	Lognormal	80 kN	16 kN
θ_1	Angle	Normal	0°	10°
θ_2	Angle	Normal	0°	10°
θ_3	Angle	Normal	0°	10°
θ_4	Angle	Normal	0°	10°
E	Young's modulus	Normal	200 MPa	30 Mpa
A	Sectional area	Normal	8000 mm^2	800 mm^2

many additional $\mathcal{G}$-function evaluations in order to provide a reliable result. AGPR-LS is able to enhance the accuracy of FORM-HLRF, at the cost of many additional computational efforts. PBAL-LS and BAL-LS give reasonably good estimates of the probability of failure. However, BAL-LS is much more efficient than PBAL-LS in this example.

Table 9.8: Results for Example 4 by several different methods.

Method	$\hat{P}_f$	$\delta_{\hat{P}_f}$ or $\overline{\delta}_{\hat{P}_f}$	N_{line}	N_{total}
IS	6.04×10^{-6}	1.00%	-	64,687
FORM-HLRF	4.19×10^{-6}	-	-	288
SORM	4.33×10^{-6}	-	-	421
Traditional LS	5.86×10^{-6}	5.74%	100	810
	6.16×10^{-6}	3.89%	200	1356
AGPR-LS	6.24×10^{-8}	3.49%	172	468
PBAL-LS	5.86×10^{-6}	4.88%	89	272
Proposed BAL-LS	5.95×10^{-6}	4.65%	23	106

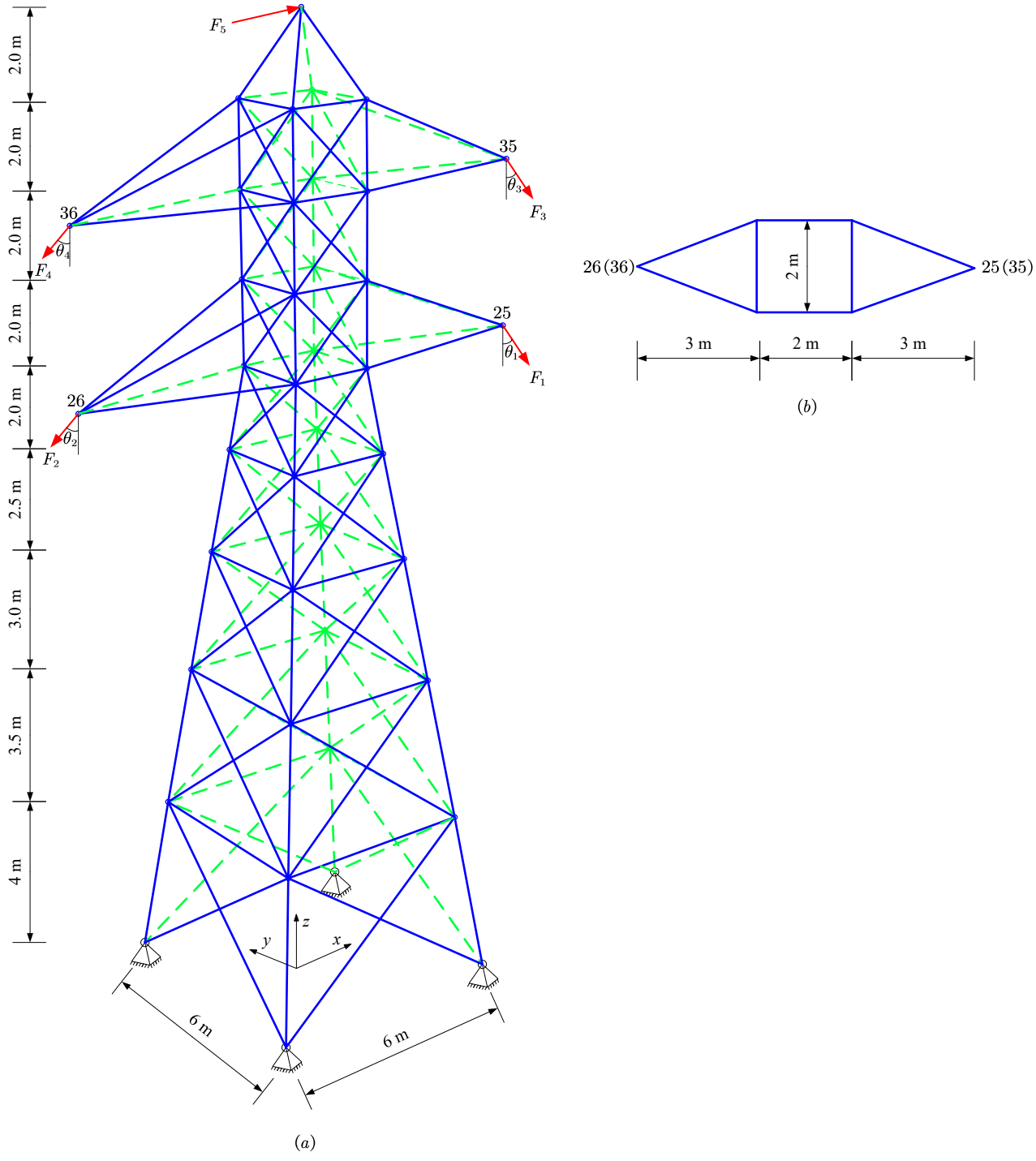


Figure 9.7: A transmission tower structure subject to horizontal and oblique loads.

9.5 Concluding remarks

This paper offers a more complete Bayesian active learning treatment of line sampling in the context of structural reliability analysis. This treatment leads to a new method, called 'Bayesian active learning line sampling' (BAL-LS). In this method, we first complete a Bayesian treatment of the standard line sampling in terms of the second-order posterior statistics. Specially, the posterior variance of the failure probability defined in line sampling is derived, which can measure our epistemic uncertainty about the failure probability resulted from a limited number of observations. Then, the Bayesian active learning treatment is accomplished by proposing a learning function and a stopping criterion based on the posterior statistics of the failure probability. Besides, the proposed method can automatically update the importance direction throughout its course without re-evaluating the performance function. From several numerical studies, it is shown that the proposed BAL-LS method is able to assess extremely small failure probabilities for weakly and moderately nonlinear reliability problems with high efficiency and accuracy. Moreover, BAL-LS exhibits a superior performance when compared with our previously developed PBAL-LS in the studied examples.

The proposed method is only suitable for a class of weakly to moderately nonlinear problems in low to moderate dimensions (<20). For highly nonlinear problems, the failure domain can be quite complex in geometry, far from being half-open. The Bayesian active learning framework based on the GP model in its current form can be quite challenging in high dimensions. These limitations can be addressed in future work.

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). Jingwen Song would like to acknowledge financial support from the National Natural Science Foundation of China (grant no. 12202358 and 12220101002). Pengfei Wei is grateful to the support from the National Natural

Science Foundation of China (grant no. 72171194). 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.

Bibliography

[1] Chao Dang, Pengfei Wei, Matthias GR Faes, Marcos A Valdebenito, and Michael Beer. Parallel adaptive Bayesian quadrature for rare event estimation. *Reliability Engineering & System Safety*, page 108621, 2022.

[2] Reuven Y Rubinstein and Dirk P Kroese. *Simulation and the Monte Carlo method*. John Wiley & Sons, 2016.

[3] Siu-Kui Au and James L Beck. A new adaptive importance sampling scheme for reliability calculations. *Structural Safety*, 21(2):135–158, 1999.

[4] Nolan Kurtz and Junho Song. Cross-entropy-based adaptive importance sampling using gaussian mixture. *Structural Safety*, 42:35–44, 2013.

[5] Ove Ditlevsen, Robert E Melchers, and H Gluver. General multi-dimensional probability integration by directional simulation. *Computers & Structures*, 36(2):355–368, 1990.

[6] Jinsuo Nie and Bruce R Ellingwood. Directional methods for structural reliability analysis. *Structural Safety*, 22(3):233–249, 2000.

[7] Siu-Kui Au and James L Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, 16(4):263–277, 2001.

[8] Siu-Kui Au and Yu Wang. *Engineering risk assessment with subset simulation*. John Wiley & Sons, 2014.

[9] Phaedon-Stelios Koutsourelakis, Helmuth J Pradlwarter, and Gerhart Iwo Schueller. Reliability of structures in high dimensions, part i: algorithms and applications. *Probabilistic Engineering Mechanics*, 19(4):409–417, 2004.

[10] Phaedon-Stelios Koutsourelakis. Reliability of structures in high dimensions. part ii. theoretical validation. *Probabilistic Engineering Mechanics*, 19(4):419–423, 2004.

[11] Abraham M Hasofer and Niels C Lind. Exact and invariant second-moment code format. *Journal of the Engineering Mechanics division*, 100(1):111–121, 1974.

[12] Karl Breitung. Asymptotic approximations for multinormal integrals. *Journal of Engineering Mechanics*, 110(3):357–366, 1984.

[13] Yan-Gang Zhao and Tetsuro Ono. Moment methods for structural reliability. *Structural Safety*, 23(1):47–75, 2001.

[14] Yan-Gang Zhao and Zhao-Hui Lu. *Structural reliability: approaches from perspectives of statistical moments*. John Wiley & Sons, 2021.

[15] Xufang Zhang and Mahesh D Pandey. Structural reliability analysis based on the concepts of entropy, fractional moment and dimensional reduction method. *Structural Safety*, 43:28–40, 2013.

[16] Chao Dang and Jun Xu. A mixture distribution with fractional moments for efficient seismic reliability analysis of nonlinear structures. *Engineering Structures*, 208:109912, 2020.

[17] Chen Ding, Chao Dang, Marcos A Valdebenito, Matthias GR Faes, Matteo Broggi, and Michael Beer. First-passage probability estimation of high-dimensional nonlinear stochastic dynamic systems by a fractional moments-based mixture distribution approach. *Mechanical Systems and Signal Processing*, 185:109775, 2023.

[18] Jie Li and Jian-Bing Chen. Dynamic response and reliability analysis of structures with uncertain parameters. *International Journal for Numerical Methods in Engineering*, 62(2):289–315, 2005.

[19] Jian-Bing Chen and Jie Li. Dynamic response and reliability analysis of non-linear stochastic structures. *Probabilistic Engineering Mechanics*, 20(1):33–44, 2005.

[20] Xiaolan Li, Guohai Chen, Haichao Cui, and Dixiong Yang. Direct probability integral method for static and dynamic reliability analysis of structures with complicated performance functions. *Computer Methods in Applied Mechanics and Engineering*, 374:113583, 2021.

[21] Guohai Chen and Dixiong Yang. A unified analysis framework of static and dynamic structural reliabilities based on direct probability integral method. *Mechanical Systems and Signal Processing*, 158:107783, 2021.

[22] Barron J Bichon, Michael S Eldred, Laura Painton Swiler, Sandaran Mahadevan, and John M McFarland. Efficient global reliability analysis for nonlinear implicit performance functions. *AIAA Journal*, 46(10):2459–2468, 2008.

[23] Benjamin Echard, Nicolas Gayton, and Maurice Lemaire. AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation. *Structural Safety*, 33(2):145–154, 2011.

[24] Rui Teixeira, Maria Nogal, and Alan O'Connor. Adaptive approaches in metamodel-based reliability analysis: A review. *Structural Safety*, 89:102019, 2021.

[25] Maliki Moustapha, Stefano Marelli, and Bruno Sudret. Active learning for structural reliability: Survey, general framework and benchmark. *Structural Safety*, 96:102174, 2022.

[26] MF Pellissetti, GI Schuëller, HJ Pradlwarter, A Calvi, SHJA Fransen, and M Klein. Reliability analysis of spacecraft structures under static and dynamic loading. *Computers & Structures*, 84(21):1313–1325, 2006.

[27] HJ Pradlwarter, Gerhart Iwo Schueller, Phaedon-Stelios Koutsourelakis, and Dimos C Charmpis. Application of line sampling simulation method to reliability benchmark problems. *Structural Safety*, 29(3):208–221, 2007.

[28] Song Shufang, Lu Zhenzhou, Zhang Weiwei, and Ye Zhengyin. Reliability and sensitivity analysis of transonic flutter using improved line sampling technique. *Chinese Journal of Aeronautics*, 22(5):513–519, 2009.

[29] Jia Wang, Lambros S Katafygiotis, and Zhouquan Feng. An efficient simulation method for the first excursion problem of linear structures subjected to stochastic wind loads. *Computers & Structures*, 166:75–84, 2016.

[30] Chao Dang, Marcos A Valdebenito, Jingwen Song, Pengfei Wei, and Michael Beer. Estimation

of small failure probabilities by partially Bayesian active learning line sampling: Theory and algorithm. *Computer Methods in Applied Mechanics and Engineering*, 412:116068, 2023.

[31] Marco de Angelis, Edoardo Patelli, and Michael Beer. Advanced line sampling for efficient robust reliability analysis. *Structural Safety*, 52:170–182, 2015.

[32] Mohsen Ali Shayanfar, Mohammad Ali Barkhordari, Moien Barkhori, and Mehrollah Rakhshanimehr. An adaptive line sampling method for reliability analysis. *Iranian Journal of Science and Technology, Transactions of Civil Engineering*, 41(3):275–282, 2017.

[33] Iason Papaioannou and Daniel Straub. Combination line sampling for structural reliability analysis. *Structural Safety*, 88:102025, 2021.

[34] Ivan Depina, Thi Minh Hue Le, Gordon Fenton, and Gudmund Eiksund. Reliability analysis with metamodel line sampling. *Structural Safety*, 60:1–15, 2016.

[35] Jingwen Song, Pengfei Wei, Marcos Valdebenito, and Michael Beer. Active learning line sampling for rare event analysis. *Mechanical Systems and Signal Processing*, 147:107113, 2021.

[36] Chao Dang, Marcos A Valdebenito, Matthias GR Faes, Pengfei Wei, and Michael Beer. Structural reliability analysis: A Bayesian perspective. *Structural Safety*, 99:102259, 2022.

[37] Jafar Jafari-Asl, Sima Ohadi, Mohamed El Amine Ben Seghier, and Nguyen-Thoi Trung. Accurate structural reliability analysis using an improved line-sampling-method-based slime mold algorithm. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, 7(2):04021015, 2021.

[38] Marcos A Valdebenito, Pengfei Wei, Jingwen Song, Michael Beer, and Matteo Broggi. Failure probability estimation of a class of series systems by multidomain line sampling. *Reliability Engineering & System Safety*, 213:107673, 2021.

[39] Enrico Zio and Nicola Pedroni. An optimized line sampling method for the estimation of the failure probability of nuclear passive systems. *Reliability Engineering & System Safety*, 95(12):1300–1313, 2010.

[40] Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

[41] Bob E Ellison. Two theorems for inferences about the normal distribution with applications in acceptance sampling. *Journal of the American Statistical Association*, 59(305):89–95, 1964.

[42] Alan Genz. Numerical computation of rectangular bivariate and trivariate normal and t probabilities. *Statistics and Computing*, 14(3):251–260, 2004.

[43] Henry R Chai and Roman Garnett. Improving quadrature for constrained integrands. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2751–2759. PMLR, 2019.

[44] Simon J Julier and Jeffrey K Uhlmann. New extension of the kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI*, volume 3068, pages 182–193. Spie, 1997.

[45] Pei-Ling Liu and Armen Der Kiureghian. Optimization algorithms for structural reliability. *Structural Safety*, 9(3):161–177, 1991.

[46] Christian G Bucher and Ulrich Bourgund. A fast and efficient response surface approach for structural reliability problems. *Structural Safety*, 7(1):57–66, 1990.

[47] Jianhua Zhou and Andrzej S Nowak. Integration formulas to evaluate functions of random variables. *Structural Safety*, 5(4):267–284, 1988.

[48] Jun Xu and Chao Dang. A new bivariate dimension reduction method for efficient structural reliability analysis. *Mechanical Systems and Signal Processing*, 115:281–300, 2019.

[49] Frank McKenna. Opensees: a framework for earthquake engineering simulation. *Computing in Science & Engineering*, 13(4):58–66, 2011.

[50] S. Marelli, R. Schöbi, and B. Sudret. UQLab user manual – Structural reliability (Rare event estimation). Technical report, Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland, 2022. Report UQLab-V2.0-107.

Chapter 10

Advanced Bayesian active learning line sampling for rare event estimation

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

Chao Dang[a,⋆], Marcos A. Valdebenito[b], Jingwen Song[c], Pengfei Wei[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 Mechanical Engineering, Northwestern Polytechnical University, Xi'an 710072, PR China*

[d] *School of Power and Energy, 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*

Submitted to Reliability Engineering & System Safety in May 2023

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 a 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

*Corresponding Author
E-mail address: chao.dang@irz.uni-hannover.de (C. Dang)

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

10.1 Introduction

Probabilistic reliability analysis has become an important tool in the modern engineer's toolbox. In this context, one of the main tasks is to calculate the failure probability, which constitutes a typical rare event probability estimation problem. The failure probability is defined by a multiple integral of the form:

$$P_f = \int_{\mathbb{X}} I(g(\boldsymbol{x})) f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{10.1}$$

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. (10.1) is not analytically tractable, and therefore a variety of numerical methods have been developed over the years to approximate it.

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]. All in all, each method has its own advantages and disadvantages, as often being discussed in the literature. In the following, we will limit our attention to the LS method that is closely related to the subject of this study.

As a stand-alone simulation method, the invention of LS is attributed to the work of Kout-

sourelakis 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 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-based 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 simple 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 some locations, the posterior distribution of the logarithmic distance function turns out to be a GP. This implies that the posterior distribution of the distance function follows a 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 10.2, two related methods are briefly reviewed. The proposed method is presented in Section 10.3. Four numerical examples are investigated in Section 10.4 to demonstrate the proposed method. Section 10.5 gives some concluding remarks.

10.2 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 $\mathcal{G}(\boldsymbol{U}) := g(T^{-1}(\boldsymbol{U}))$.

10.2.1 Traditional line sampling

Traditional LS begins by identifying an important direction $\boldsymbol{\alpha}$, which is a unit vector pointing to the failure domain in the standard normal space, i.e., $\mathcal{F} = \{\boldsymbol{u} \in \mathcal{U} : \mathcal{G}(\boldsymbol{u}) < 0\}$. This 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].

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

$$P_f = \int_{\mathbb{R}^{d-1}} \Phi(-\beta(\boldsymbol{u}^\perp))\varphi_{\boldsymbol{U}^\perp}(\boldsymbol{u}^\perp)\mathrm{d}\boldsymbol{u}^\perp, \tag{10.2}$$

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; $\varphi_{\boldsymbol{U}^\perp}(\cdot)$ 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. (10.2) is solved by the crude MCS in conjugation with a root-finding technique. The MCS estimator of P_f is given by:

$$\hat{P}_f = \frac{1}{N}\sum_{i=1}^{N} \Phi(-\hat{\beta}(\boldsymbol{u}^{\perp,(i)})), \tag{10.3}$$

where $\left\{\boldsymbol{u}^{\perp,(i)}\right\}_{i=1}^{N}$ is a set of N random samples generated according to $\varphi_{\boldsymbol{U}^\perp}(\cdot)$; $\hat{\beta}(\boldsymbol{u}^{\perp,(i)})$ denotes

the approximate result of $u^{\parallel}$ subject to $\mathcal{G}(\boldsymbol{\alpha}u^{\parallel} + \boldsymbol{B}\boldsymbol{u}^{\perp,(i)}) = 0$, which can be obtained by a suitable root-finding algorithm such as polynomial interpolation [8] and Newton's method [12]. Note that a small approximation error of $\hat{\beta}(\boldsymbol{u}^{\perp,(i)})$ may cause a large numerical error of $\Phi(-\hat{\beta}(\boldsymbol{u}^{\perp,(i)}))$, and the errors of $\Phi(-\hat{\beta}(\boldsymbol{u}^{\perp,(i)})), i = 1, 2, \cdots, N$ can accumulate when forming the failure probability estimate. However, this aspect is ignored in the traditional LS and its variants.

10.2.2 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. (10.2) 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:

$$\beta_0(\boldsymbol{u}^{\perp}) \sim \mathcal{GP}(m_{\beta_0}(\boldsymbol{u}^{\perp}), k_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{10.4}$$

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:

$$\beta_n(\boldsymbol{u}^{\perp}) \sim \mathcal{GP}(m_{\beta_n}(\boldsymbol{u}^{\perp}), k_{\beta_n}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{10.5}$$

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]:

$$m_{\beta_n}(\boldsymbol{u}^{\perp}) = m_{\beta_0}(\boldsymbol{u}^{\perp}) + \boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{\mathcal{U}}^{\perp})^{\top} \boldsymbol{K}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp}, \boldsymbol{\mathcal{U}}^{\perp})^{-1}(\boldsymbol{\mathcal{Y}} - \boldsymbol{m}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp})), \tag{10.6}$$

$$k_{\beta_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) = k_{\beta_0}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) - \boldsymbol{k}_{\beta_0}(\boldsymbol{u}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{\top}\boldsymbol{K}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{-1}\boldsymbol{k}_{\beta_0}(\boldsymbol{\mathcal{U}}^{\perp},\boldsymbol{u}^{\perp\prime}), \tag{10.7}$$

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]:

$$m_{\Phi_n(-\tilde{\beta})}\left(\boldsymbol{u}^{\perp}\right) = \Phi\left(\frac{-m_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}}\right), \tag{10.8}$$

$$\begin{aligned} k_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) =& \Psi\left(\begin{bmatrix} m_{\beta_n}\left(\boldsymbol{u}^{\perp}\right) \\ m_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right) \end{bmatrix}; \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)+1 & k_{\beta_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) \\ k_{\beta_n}(\boldsymbol{u}^{\perp\prime},\boldsymbol{u}^{\perp}) & \sigma^2_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)+1 \end{bmatrix}\right) \\ & - \Phi\left(\frac{m_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\sigma^2_{\tilde{\beta}_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)\Phi\left(\frac{m_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)}{\sqrt{1+\sigma^2_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)}}\right), \end{aligned} \tag{10.9}$$

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:

$$m_{P_{f,n}} = \int_{\mathbb{R}^{d-1}} m_{\Phi_n(-\beta)}\left(\boldsymbol{u}^{\perp}\right)\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}\boldsymbol{u}^{\perp}, \tag{10.10}$$

$$\sigma^2_{P_{f,n}} = \int_{\mathbb{R}^{d-1}}\int_{\mathbb{R}^{d-1}} k_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime})\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\varphi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime})\mathrm{d}\boldsymbol{u}^{\perp}\mathrm{d}\boldsymbol{u}^{\perp\prime}. \tag{10.11}$$

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:

$$\frac{\sigma_{P_{f,n}}}{m_{P_{f,n}}} < \delta, \tag{10.12}$$

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):

$$\mathrm{PSDC}\left(\boldsymbol{u}^{\perp}\right) = \varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}) \times \int_{\mathcal{U}^{\perp}} k_{\tilde{\Phi}_n(-\tilde{\beta})}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})\varphi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime})\mathrm{d}\boldsymbol{u}^{\perp\prime}, \tag{10.13}$$

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.

10.3 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 a 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.

10.3.1 Theoretical development

10.3.1.1 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:

$$\log\left(\hat{\beta}(\boldsymbol{u}^{\perp})\right) = \log\left(\beta(\boldsymbol{u}^{\perp})\right) + \varepsilon, \tag{10.14}$$

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. (10.14) can be rewritten as:

$$\hat{l}(\boldsymbol{u}^{\perp}) = l(\boldsymbol{u}^{\perp}) + \varepsilon. \tag{10.15}$$

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

$$\beta_0(\boldsymbol{u}^{\perp}) \sim \mathcal{LP}(\boxed{m}_{\beta_0}(\boldsymbol{u}^{\perp}), \boxed{k}_{\beta_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{10.16}$$

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. Using a simple trick, we equate the LP prior over β to a GP prior over $l(\boldsymbol{u}^{\perp})$:

$$l_0(\boldsymbol{u}^{\perp}) \sim \mathcal{GP}(m_{l_0}(\boldsymbol{u}^{\perp}), k_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime})), \tag{10.17}$$

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:

$$m_{l_0}(\boldsymbol{u}^{\perp}) = b, \tag{10.18}$$

$$k_{l_0}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) = \sigma_k^2 \exp\left(-\frac{1}{2}(\boldsymbol{u}^{\perp} - \boldsymbol{u}^{\perp\prime})^{\top} \boldsymbol{\Sigma}^{-1} (\boldsymbol{u}^{\perp} - \boldsymbol{u}^{\perp\prime})\right), \tag{10.19}$$

where $b \in \mathbb{R}$; $\sigma_k > 0$ is the process standard deviation; $\boldsymbol{\Sigma} = \mathrm{diag}\left(w_1^2, w_2^2, \cdots, w_{d-1}^2\right)$ 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:

$$\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2), \tag{10.20}$$

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.

10.3.1.2 Hyper-parameters tuning

Our prior assumptions expressed in Eqs. (10.18)-(10.20) 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:

$$\boldsymbol{\Omega} = \arg\max \log p(\tilde{\boldsymbol{\mathcal{Z}}}|\boldsymbol{\mathcal{U}}^\perp, \boldsymbol{\Omega}), \tag{10.21}$$

in which

$$\log p(\tilde{\boldsymbol{\mathcal{Z}}}|\boldsymbol{\mathcal{U}}^\perp, \boldsymbol{\Omega}) = -\frac{1}{2}\left[\log\left(|\boldsymbol{K}_{l_0} + \sigma_\varepsilon \boldsymbol{I}|\right) + \left(\tilde{\boldsymbol{\mathcal{Z}}} - b\right)^\top \left(\boldsymbol{K}_{l_0} + \sigma_\varepsilon \boldsymbol{I}\right)^{-1}\left(\tilde{\boldsymbol{\mathcal{Z}}} - b\right) + n\log(2\pi)\right], \tag{10.22}$$

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.

10.3.1.3 Posterior distributions

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

$$l_n(\boldsymbol{u}^\perp) \sim \mathcal{GP}(m_{l_n}(\boldsymbol{u}^\perp), k_{l_n}(\boldsymbol{u}^\perp, \boldsymbol{u}^{\perp\prime})), \tag{10.23}$$

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]:

$$m_{l_n}(\boldsymbol{u}^{\perp}) = m_{l_0}(\boldsymbol{u}^{\perp}) + \boldsymbol{k}_{l_0}(\boldsymbol{u}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{\top}\left(\boldsymbol{K}_{l_0} + \sigma_{\varepsilon}\boldsymbol{I}\right)^{-1}\left(\tilde{\boldsymbol{\mathcal{Z}}} - \boldsymbol{m}_{l_0}(\boldsymbol{\mathcal{U}}^{\perp})\right), \tag{10.24}$$

$$k_{l_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) = k_{l_0}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) - \boldsymbol{k}_{l_0}(\boldsymbol{u}^{\perp},\boldsymbol{\mathcal{U}}^{\perp})^{\top}\left(\boldsymbol{K}_{l_0} + \sigma_{\varepsilon}\boldsymbol{I}\right)^{-1}\boldsymbol{k}_{l_0}(\boldsymbol{\mathcal{U}}^{\perp},\boldsymbol{u}^{\perp,\prime}), \tag{10.25}$$

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 a LP:

$$\beta_n(\boldsymbol{u}^{\perp}) \sim \mathcal{LP}(\boxed{m}_{\beta_n}(\boldsymbol{u}^{\perp}), \boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime})), \tag{10.26}$$

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:

$$\boxed{m}_{\beta_n}(\boldsymbol{u}^{\perp}) = \exp\left(m_{l_n}(\boldsymbol{u}^{\perp}) + \frac{1}{2}\sigma_{l_n}^2(\boldsymbol{u}^{\perp})\right), \tag{10.27}$$

$$\boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) = \left[\exp\left(k_{l_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime})\right) - 1\right]\exp\left(m_{l_n}(\boldsymbol{u}^{\perp}) + m_{l_n}(\boldsymbol{u}^{\perp\prime}) + \frac{1}{2}\left(\sigma_{l_n}^2(\boldsymbol{u}^{\perp}) + \sigma_{l_n}^2(\boldsymbol{u}^{\perp\prime})\right)\right), \tag{10.28}$$

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

With the LP posterior of β, it is non-trivial 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., $\mathcal{GP}(\boxed{m}_{\beta_n}(\boldsymbol{u}^{\perp}), \boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}))$. This approximation makes it possible to 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:

$$\boxed{m}_{\tilde{\Phi}_n(-\beta)}\left(\boldsymbol{u}^{\perp}\right) = \Phi\left(\frac{-\boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\boxed{\sigma}_{\beta_n}^2\left(\boldsymbol{u}^{\perp}\right)}}\right), \tag{10.29}$$

$$\begin{aligned}\boxed{k}_{\tilde{\Phi}_n(-\beta)}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime}) =& \Psi\left(\begin{bmatrix}\boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)\\ \boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)\end{bmatrix};\begin{bmatrix}0\\0\end{bmatrix},\begin{bmatrix}\boxed{\sigma}^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)+1 & \boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime})\\ \boxed{k}_{\beta_n}(\boldsymbol{u}^{\perp\prime},\boldsymbol{u}^{\perp}) & \boxed{\sigma}^2_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)+1\end{bmatrix}\right)\\ &-\Phi\left(\frac{\boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}{\sqrt{1+\boxed{\sigma}^2_{\beta_n}\left(\boldsymbol{u}^{\perp}\right)}}\right)\Phi\left(\frac{\boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)}{\sqrt{1+\boxed{\sigma}^2_{\beta_n}\left(\boldsymbol{u}^{\perp\prime}\right)}}\right),\end{aligned} \tag{10.30}$$

where $\boxed{\sigma}^2_{\beta_n}(\cdot)=\boxed{k}_{\beta_n}(\cdot,\cdot)$. For proofs of Eqs. (10.29) and (10.30), please refer to [31]. Note that Eqs. (10.29) and (10.30) are respectively different from Eqs. (10.8) and (10.9) 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:

$$\boxed{m}_{P_{f,n}} = \int_{\mathbb{R}^{d-1}} \boxed{m}_{\Phi_n(-\beta)}\left(\boldsymbol{u}^{\perp}\right)\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\mathrm{d}\boldsymbol{u}^{\perp}, \tag{10.31}$$

$$\boxed{\sigma}^2_{P_{f,n}} = \int_{\mathbb{R}^{d-1}}\int_{\mathbb{R}^{d-1}} \boxed{k}_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp},\boldsymbol{u}^{\perp\prime})\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp})\varphi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime})\mathrm{d}\boldsymbol{u}^{\perp}\mathrm{d}\boldsymbol{u}^{\perp\prime}. \tag{10.32}$$

Eqs. (10.31) and (10.32) are easy to be proved by using the Fubini's theorem, hence the proofs are omitted. It is noted that Eqs. (10.31) and (10.32) are essentially different from Eqs. (10.10) and (10.11) 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.

10.3.1.4 Estimating the approximate posterior mean and variance of the failure probability

The approximate posterior mean and variances of the failure probability defined in (10.31) and (10.32) 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:

$$\hat{\boxed{m}}_{P_{f,n}} = \frac{1}{N}\sum_{q=1}^{N}\Phi\left(\frac{-\boxed{m}_{\beta_n}\left(\boldsymbol{u}^{\perp,(q)}\right)}{\sqrt{1+\boxed{\sigma}^2_{\beta_n}\left(\boldsymbol{u}^{\perp,(q)}\right)}}\right)\frac{\varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp,(q)})}{\varphi_{\boldsymbol{U}^{\perp},\lambda}(\boldsymbol{u}^{\perp,(q)})}, \tag{10.33}$$

$$\hat{\boxed{\sigma}}^2_{P_{f,n}} = \frac{1}{N}\sum_{i=1}^{N} \boxed{k}_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp,(q)}, \boldsymbol{u}^{\perp\prime,(q)}) \frac{\varphi_{\boldsymbol{U}^\perp}(\boldsymbol{u}^{\perp,(q)})\varphi_{\boldsymbol{U}^\perp}(\boldsymbol{u}^{\perp\prime,(q)})}{\varphi_{\boldsymbol{U}^\perp,\lambda}(\boldsymbol{u}^{\perp,(q)})\varphi_{\boldsymbol{U}^\perp,\lambda}(\boldsymbol{u}^{\perp\prime,(q)})}, \tag{10.34}$$

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 $\varphi_{\boldsymbol{U}^\perp,\lambda}(\boldsymbol{u}^\perp)$ and $\varphi_{\boldsymbol{U}^\perp,\lambda}(\boldsymbol{u}^{\perp\prime})$, respectively; $\varphi_{\boldsymbol{U}^\perp,\lambda}(\boldsymbol{u}^\perp)$ is the SDA-IS density of the form $\varphi_{\boldsymbol{U}^\perp,\lambda}(\boldsymbol{u}^\perp) = \prod_{i=1}^{d-1}\varphi_{U_i^\perp,\lambda}(u_i^\perp)$, in which

$$\varphi_{U_i^\perp,\lambda}(u_i^\perp) = \frac{1}{\lambda\sqrt{2\pi}}\exp\left(-\frac{u_i^{\perp,2}}{2\lambda^2}\right), \tag{10.35}$$

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

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

$$\mathbb{V}\left[\hat{\boxed{m}}_{P_{f,n}}\right] = \frac{1}{N(N-1)}\sum_{q=1}^{N}\left[\Phi\left(\frac{-\boxed{m}_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(q)}\right)}{\sqrt{1+\boxed{\sigma}^2_{\hat{\beta}_n}\left(\boldsymbol{u}^{\perp,(q)}\right)}}\right)\frac{\varphi_{\boldsymbol{U}^\perp}(\boldsymbol{u}^{\perp,(q)})}{\varphi_{\boldsymbol{U}^\perp,\lambda}(\boldsymbol{u}^{\perp,(q)})} - \hat{\boxed{m}}_{P_{f,n}}\right]^2, \tag{10.36}$$

$$\mathbb{V}\left[\hat{\boxed{\sigma}}^2_{P_{f,n}}\right] = \frac{1}{N(N-1)}\sum_{q=1}^{N}\left[\boxed{k}_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp,(q)}, \boldsymbol{u}^{\perp\prime,(q)})\frac{\varphi_{\boldsymbol{U}^\perp}(\boldsymbol{u}^{\perp,(q)})\varphi_{\boldsymbol{U}^\perp}(\boldsymbol{u}^{\perp\prime,(q)})}{\varphi_{\boldsymbol{U}^\perp,\lambda}(\boldsymbol{u}^{\perp,(q)})\varphi_{\boldsymbol{U}^\perp,\lambda}(\boldsymbol{u}^{\perp\prime,(q)})} - \hat{\boxed{\sigma}}^2_{P_{f,n}}\right]^2. \tag{10.37}$$

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.

10.3.1.5 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:

$$\frac{\hat{\boxed{\sigma}}_{P_{f,n}}}{\hat{\boxed{m}}_{P_{f,n}}} < \eta, \tag{10.38}$$

where η is a tolerance value. The stopping criterion 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:

$$\mathrm{APSDC}\left(\boldsymbol{u}^{\perp}\right) = \varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}) \times \int_{\mathbb{R}^{d-1}} \boxed{k}_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime}) \varphi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime}) \mathrm{d}\boldsymbol{u}^{\perp\prime}. \tag{10.39}$$

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:

$$\widehat{\mathrm{APSDC}}\left(\boldsymbol{u}^{\perp}\right) = \varphi_{\boldsymbol{U}^{\perp}}(\boldsymbol{u}^{\perp}) \frac{1}{M} \sum_{p=1}^{M} \boxed{k}_{\Phi_n(-\beta)}(\boldsymbol{u}^{\perp}, \boldsymbol{u}^{\perp\prime,(p)}), \tag{10.40}$$

where $\left\{\boldsymbol{u}^{\perp\prime,(p)}\right\}_{p=1}^{M}$ is a set of M integration points, which are generated according to $\varphi_{\boldsymbol{U}^{\perp\prime}}(\boldsymbol{u}^{\perp\prime})$ using Sobol sequence in this study.

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

$$\boldsymbol{u}^{\perp,(n+1)} = \underset{\boldsymbol{u}^{\perp} \in \mathbb{R}^{d-1}}{\arg\max}\, \widehat{\mathrm{APSDC}}\left(\boldsymbol{u}^{\perp}\right), \tag{10.41}$$

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.

10.3.2 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. 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. 10.1.

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:

$$\boldsymbol{\alpha}^{(0)} = -\frac{\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0})}{||\nabla_{\boldsymbol{u}}\mathcal{G}(\mathbf{0})||}, \tag{10.42}$$

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. (10.23). 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 trivial to obtain the LP posterior of the β-function, as given by Eq. (10.26). 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. (10.31) and (10.32)). 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 10.3.1.4. 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. (10.40)), where $M = 20$ is adopted. Second, the approximate distance $\underline{\tilde{y}}^{(n+1)}$ 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 $\boldsymbol{\alpha}^{(q)}\underline{\tilde{y}}^{(n+1)} + \boldsymbol{B}^{(q)}\underline{\boldsymbol{u}}^{\perp,(n+1)}$. 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 $\left\{\underline{\boldsymbol{u}}^{\perp,(n+1)}, \log \underline{\tilde{y}}^{(n+1)}\right\}$. Otherwise, the previous important direction is then updated to a new one, i.e., $\boldsymbol{\alpha}^{(q+1)} = \frac{\boldsymbol{\alpha}^{(0)}\underline{\tilde{y}}^{(n+1)} + \boldsymbol{B}^{(q)}\underline{\boldsymbol{u}}^{\perp,(n+1)}}{||\boldsymbol{\alpha}^{(0)}\underline{\tilde{y}}^{(n+1)} + \boldsymbol{B}^{(q)}\underline{\boldsymbol{u}}^{\perp,(n+1)}||}$. 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.

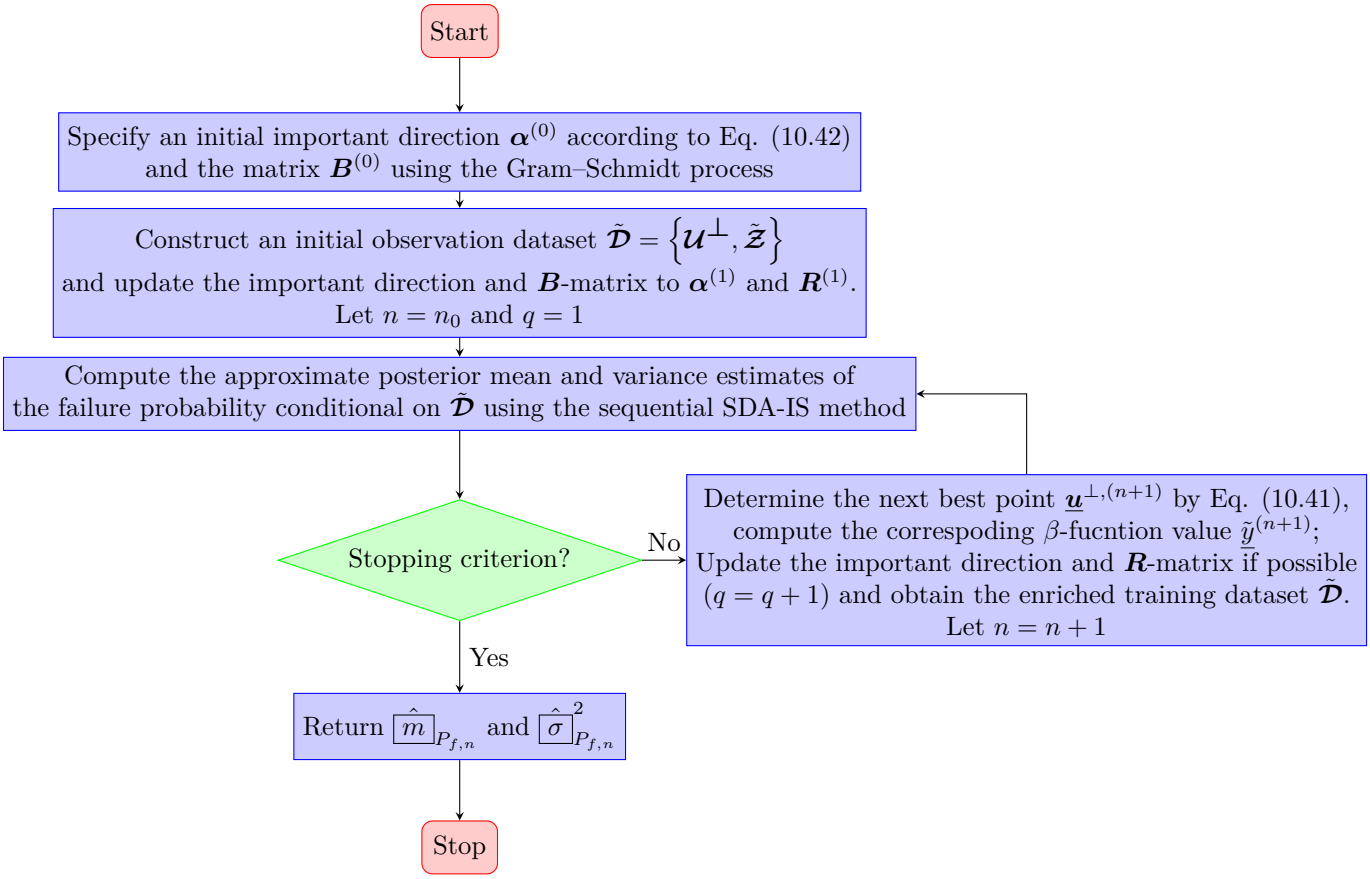


Figure 10.1: Flowchart of the proposed BAL-LS-LP method.

10.4 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) [37], traditional LS [8], combination line sampling (CLS) [14], active learning reliability method in UQLab version 2.0 (denoted as ALR in UQLab) [38] 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. (10.42)). The ALR in UQLab employs the Kriging model with Gaussian 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.

10.4.1 Example 1: A test function

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

$$Y = g(\boldsymbol{X}) = a - X_2 + bX_1^3 + c\sin\left(dX_1\right), \tag{10.43}$$

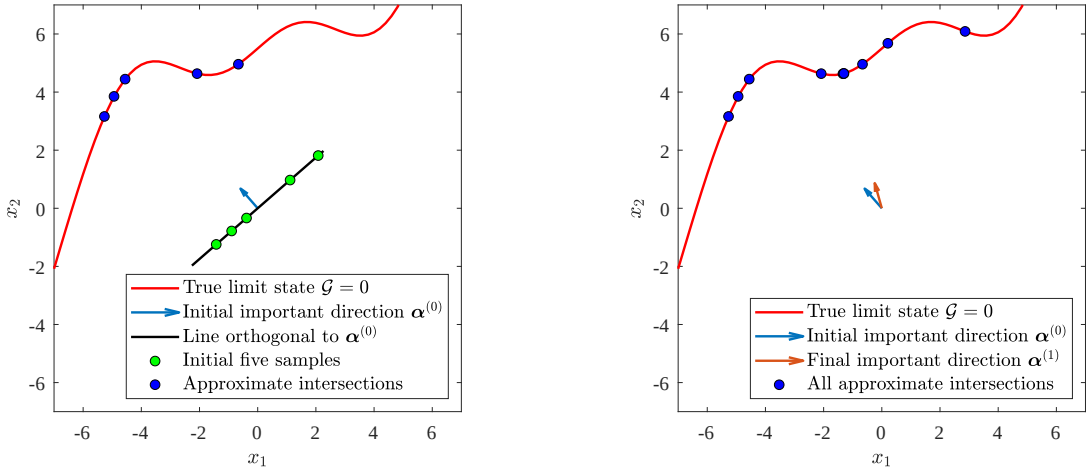
where a, b, c and d are four parameters that can influence the non-linearity of the problem and the level of failure probability, which are specified as: $a = 5.5$, $b = 0.02$, $c = \frac{5}{6}$, $d = \frac{\pi}{3}$; X_1 and X_2 are two standard normal variables.

The results of the proposed BAL-LS-LP method and several existing methods are summarized in Table 10.1. The reference failure probability is taken as 3.54×10^{-7}, which is provided by MCS with 10^{11} samples. The failure probability estimate from FORM-SQP (say 7.19×10^{-7}) differs significantly from the reference one. To obtain a failure probability estimate with a reasonable COV, both traditional LS and CLS require a large number of performance function calls. ALR in UQLab only needs a dozen performance function evaluations on average, but it results in an 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 revelations 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 with a small variability.

To provide a schematic illustration of the proposed method, Fig. 10.2 shows some of the results obtained from an exemplary run. It can be observed from Fig. 10.2(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. 10.2(b), the final important direction is almost optimal.

Table 10.1: Results of Example 1 by several methods.

Method	N_{line}	N_{call}	$\hat{P}_f$	COV $\hat{P}_f$
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%



(a) Initial stage (b) Final stage

Figure 10.2: Schematic illustration of the proposed BAL-LS-LP method for Example 1.

10.4.2 Example 2: A non-linear oscillator

The second example consists of a non-linear oscillator subject to a rectangular-pulse load [39], as shown in Fig. 10.3. The performance function is defined by:

$$Z = g\left(m, k_1, k_2, r, F_1, t_1\right) = 3r - \left| \frac{2F_1}{k_1 + k_2} \sin\left(\frac{t_1}{2} \sqrt{\frac{k_1 + k_2}{m}} \right) \right|, \tag{10.44}$$

where m, k_1, k_2, r, F_1 and t_1 are six random variables, as detailed in Table 10.2.

In Table 10.3, 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

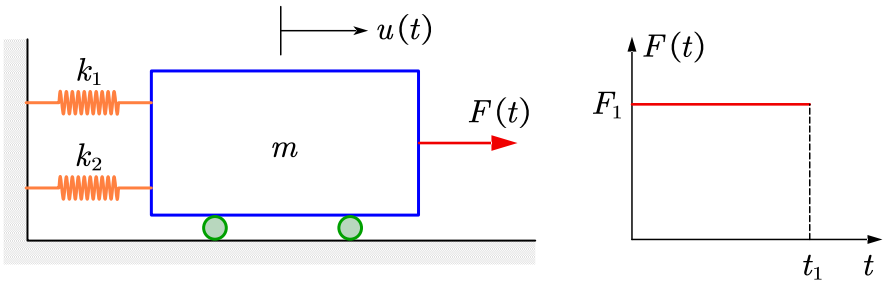


Figure 10.3: A nonlinear SDOF oscillator driven by a rectangular pulse load.

Table 10.2: 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

probability is 4.01×10^{-8} 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 a small number of extra lines (e.g., 50), 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. BAL-LS-LP requires on average slightly more lines and $\mathcal{G}$-function calls than BAL-LS, but produces a almost unbiased result with a significantly smaller COV.

10.4.3 Example 3: An I beam

As a third example, we consider a simply-supported I beam subject to a concentrated force [40], as depicted in Fig. 10.4. The performance function is expressed as:

$$Y = g(\boldsymbol{X}) = S - \sigma_{\max}, \tag{10.45}$$

Table 10.3: Results of Example 2 by several methods.

Method	N_{line}	N_{call}	$\hat{P}_f$	COV $\boxed{\hat{P}_f}$
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%

in which

$$\sigma_{\max} = \frac{Pa(L-a)d}{2LI}, \tag{10.46}$$

with

$$I = \frac{b_f d^3 - (b_f - t_w)(d - 2t_f)^3}{12}. \tag{10.47}$$

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 10.4.

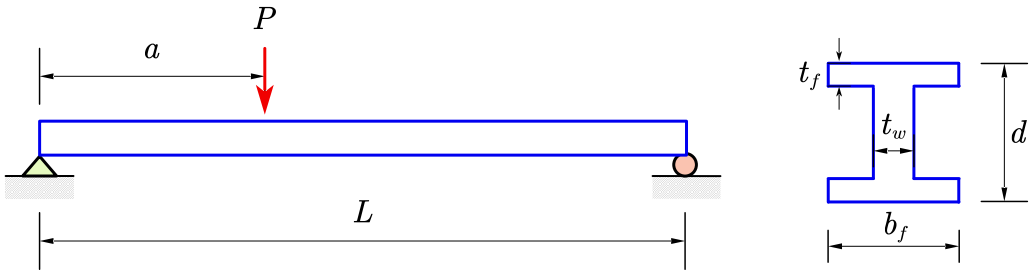


Figure 10.4: A simply-supported I beam.

Table 10.4: 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

The results obtained from several methods are reported in Table 10.5. 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 requires a large number (i.e., 1511) of performance function evaluations. In order to achieve a good result, traditional LS needs many additional $\mathcal{G}$-function calls. 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 give a failure probability mean that is closed to the reference one and with a smaller COV.

Table 10.5: 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%

10.4.4 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. 10.5. 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. 10.5) are denoted as $P_0 \sim P_{12}$. The performance function is defined as:

$$Y = g(\boldsymbol{X}) = \Delta - V_0(A, E, P_0 \sim P_{12}), \tag{10.48}$$

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 10.6.

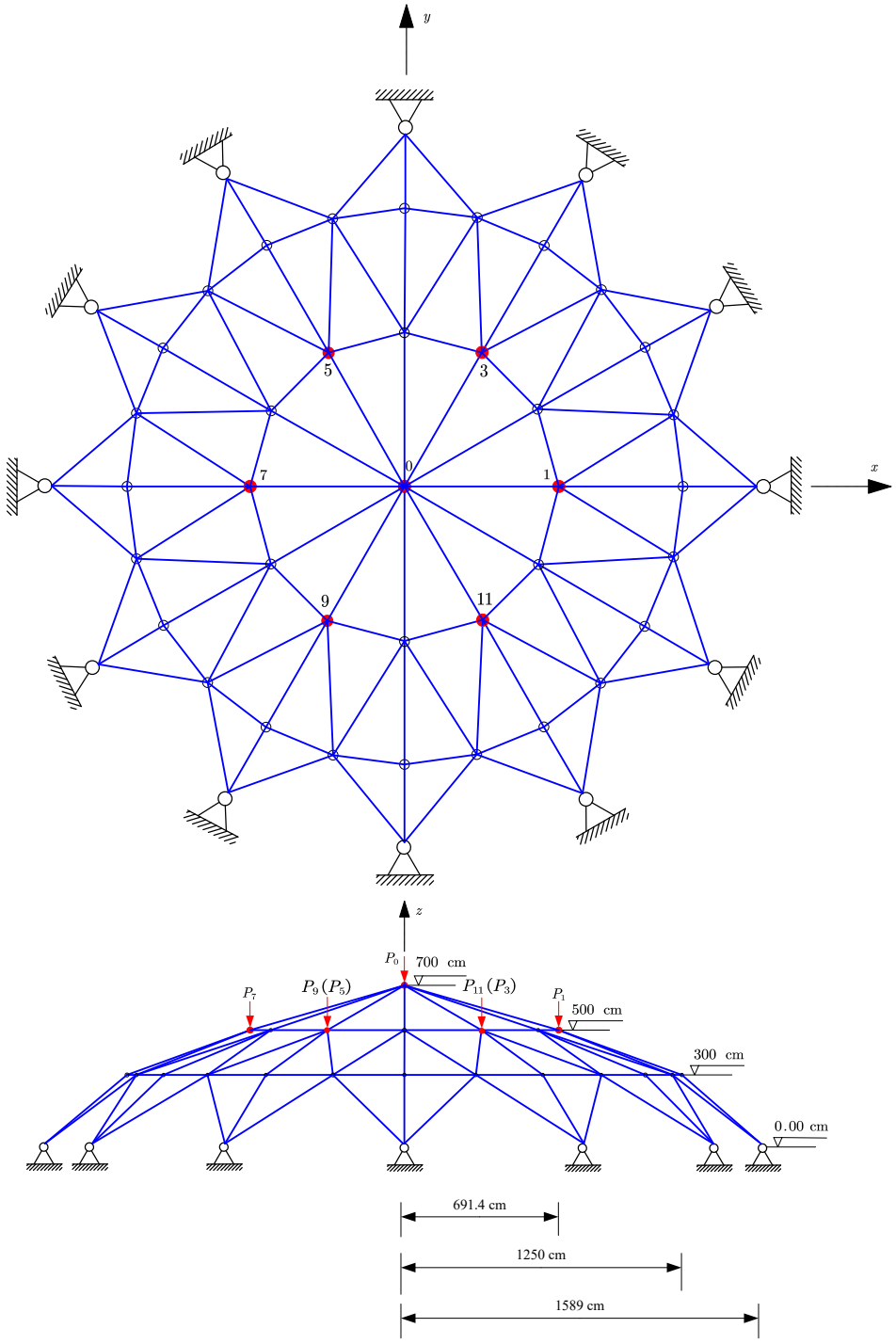


Figure 10.5: A 120-bar space truss structure subject to seven vertical loads.

Table 10.6: 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

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 [38] is then implemented as an alternative. 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 10.7. 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 stops when only several additional points are added, but the result is completely wrong. 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 10.7: Results of Example 4 by several methods.

Method	N_{line}	N_{call}	$\hat{P}_f$	COV $\boxed{\hat{P}_f}$
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%

10.5 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 a 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.

Optimizing the learning function using a nature-inspired global optimization algorithm can be time consuming as the dimensions increase. This degenerates the efficiency of the proposed method in higher dimensions. The problem may be solved by simplifying the learning function or employing a more efficient optimization algorithm (e.g., Bayesian global optimization). Besides, it is also somewhat demanding to approximate the approximate posterior variance of the failure probability by the SDA-IS method. One possible solution is to simplify the approximate posterior variance of the failure probability, or to develop a more efficient numerical integrator. Future research efforts can be devoted to those directions.

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.

Bibliography

[1] RE Melchers. Importance sampling in structural systems. *Structural Safety*, 6(1):3–10, 1989.

[2] Siu-Kui Au and James L Beck. A new adaptive importance sampling scheme for reliability calculations. *Structural Safety*, 21(2):135–158, 1999.

[3] Nolan Kurtz and Junho Song. Cross-entropy-based adaptive importance sampling using gaussian mixture. *Structural Safety*, 42:35–44, 2013.

[4] Siu-Kui Au and James L Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, 16(4):263–277, 2001.

[5] Siu-Kui Au and Yu Wang. *Engineering risk assessment with subset simulation*. John Wiley & Sons, 2014.

[6] Ove Ditlevsen, Robert E Melchers, and H Gluver. General multi-dimensional probability integration by directional simulation. *Computers & Structures*, 36(2):355–368, 1990.

[7] Jinsuo Nie and Bruce R Ellingwood. Directional methods for structural reliability analysis. *Structural Safety*, 22(3):233–249, 2000.

[8] Phaedon-Stelios Koutsourelakis, Helmuth J Pradlwarter, and Gerhart Iwo Schueller. Reliability

of structures in high dimensions, part i: algorithms and applications. *Probabilistic Engineering Mechanics*, 19(4):409–417, 2004.

[9] HJ Pradlwarter, Gerhart Iwo Schueller, Phaedon-Stelios Koutsourelakis, and Dimos C Charmpis. Application of line sampling simulation method to reliability benchmark problems. *Structural Safety*, 29(3):208–221, 2007.

[10] Phaedon-Stelios Koutsourelakis. Reliability of structures in high dimensions. part ii. theoretical validation. *Probabilistic engineering mechanics*, 19(4):419–423, 2004.

[11] Michael Hohenbichler and Ruediger Rackwitz. Improvement of second-order reliability estimates by importance sampling. *Journal of Engineering Mechanics*, 114(12):2195–2199, 1988.

[12] Marco de Angelis, Edoardo Patelli, and Michael Beer. Advanced line sampling for efficient robust reliability analysis. *Structural Safety*, 52:170–182, 2015.

[13] Mohsen Ali Shayanfar, Mohammad Ali Barkhordari, Moien Barkhori, and Mehrollah Rakhshanimehr. An adaptive line sampling method for reliability analysis. *Iranian Journal of Science and Technology, Transactions of Civil Engineering*, 41:275–282, 2017.

[14] Iason Papaioannou and Daniel Straub. Combination line sampling for structural reliability analysis. *Structural Safety*, 88:102025, 2021.

[15] Ivan Depina, Thi Minh Hue Le, Gordon Fenton, and Gudmund Eiksund. Reliability analysis with metamodel line sampling. *Structural Safety*, 60:1–15, 2016.

[16] Jingwen Song, Pengfei Wei, Marcos Valdebenito, and Michael Beer. Active learning line sampling for rare event analysis. *Mechanical Systems and Signal Processing*, 147:107113, 2021.

[17] Zhenzhou Lu, Shufang Song, Zhufeng Yue, and Jian Wang. Reliability sensitivity method by line sampling. *Structural Safety*, 30(6):517–532, 2008.

[18] Marcos A Valdebenito, Hector A Jensen, HB Hernández, and L Mehrez. Sensitivity estimation of failure probability applying line sampling. *Reliability Engineering & System Safety*, 171:99–111, 2018.

[19] Marcos A Valdebenito, Herman B Hernández, and Héctor A Jensen. Probability sensitivity estimation of linear stochastic finite element models applying line sampling. *Structural Safety*, 81:101868, 2019.

[20] Jingwen Song, Marcos Valdebenito, Pengfei Wei, Michael Beer, and Zhenzhou Lu. Non-intrusive imprecise stochastic simulation by line sampling. *Structural Safety*, 84:101936, 2020.

[21] Jingwen Song, Pengfei Wei, Marcos Valdebenito, and Michael Beer. Adaptive reliability analysis for rare events evaluation with global imprecise line sampling. *Computer Methods in Applied Mechanics and Engineering*, 372:113344, 2020.

[22] Jiaqi Wang, Zhenzhou Lu, and Lu 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):308–333, 2023.

[23] Jiaqi Wang, Zhenzhou Lu, Yulong Cheng, and Lu Wang. An efficient method for estimating failure probability bound functions of composite structure under the random-interval mixed uncertainties. *Composite Structures*, 298:116011, 2022.

[24] Xiaobo Zhang, Zhenzhou Lu, Wanying Yun, Kaixuan Feng, and Yanping Wang. Line sampling-based local and global reliability sensitivity analysis. *Structural and Multidisciplinary Optimization*, 61:267–281, 2020.

[25] Xiukai Yuan, Zhenxuan Zheng, and Baoqiang Zhang. Augmented line sampling for approximation of failure probability function in reliability-based analysis. *Applied Mathematical Modelling*, 80:895–910, 2020.

[26] Marcos A Valdebenito, Pengfei Wei, Jingwen Song, Michael Beer, and Matteo Broggi. Failure probability estimation of a class of series systems by multidomain line sampling. *Reliability Engineering & System Safety*, 213:107673, 2021.

[27] Chao Dang, Pengfei Wei, Jingwen Song, and Michael 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):04021054, 2021.

[28] Chao Dang, Pengfei Wei, Matthias GR Faes, Marcos A Valdebenito, and Michael Beer. Parallel adaptive Bayesian quadrature for rare event estimation. *Reliability Engineering & System Safety*, 225:108621, 2022.

[29] Chao Dang, Marcos A Valdebenito, Matthias GR Faes, Pengfei Wei, and Michael Beer. Structural reliability analysis: A Bayesian perspective. *Structural Safety*, 99:102259, 2022.

[30] Chao Dang, Marcos A Valdebenito, Jingwen Song, Pengfei Wei, and Michael Beer. Estimation of small failure probabilities by partially Bayesian active learning line sampling: Theory and algorithm. *Computer Methods in Applied Mechanics and Engineering*, Under Review.

[31] Chao Dang, Marcos A Valdebenito, Matthias G.R. Faes, Jingwen Song, Pengfei Wei, and Michael Beer. Structural reliability analysis by line sampling: A Bayesian active learning treatment. *Structural Safety*, Under Review.

[32] Gerhart Iwo Schuëller, Helmuth J Pradlwarter, and Phaedon-Stelios Koutsourelakis. A critical appraisal of reliability estimation procedures for high dimensions. *Probabilistic Engineering Mechanics*, 19(4):463–474, 2004.

[33] Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

[34] Tom Gunter, Michael A Osborne, Roman Garnett, Philipp Hennig, and Stephen J Roberts. Sampling for inference in probabilistic models with fast bayesian quadrature. *Advances in neural information processing systems*, 27, 2014.

[35] Henry R Chai and Roman Garnett. Improving quadrature for constrained integrands. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2751–2759. PMLR, 2019.

[36] Simon Julier, Jeffrey Uhlmann, and Hugh F Durrant-Whyte. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control*, 45(3):477–482, 2000.

[37] Pei-Ling Liu and Armen Der Kiureghian. Optimization algorithms for structural reliability. *Structural Safety*, 9(3):161–177, 1991.

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

[39] Christian G Bucher and Ulrich Bourgund. A fast and efficient response surface approach for structural reliability problems. *Structural Safety*, 7(1):57–66, 1990.

[40] Beiqing Huang and Xiaoping Du. Uncertainty Analysis by Dimension Reduction Integration and Saddlepoint Approximations. *Journal of Mechanical Design*, 128(1):26–33, 03 2005.

Chapter 11

Conclusions and outlook

11.1 Concluding remarks

Forward UQ is a branch of UQ that involves quantifying the uncertainty in the outputs of a computational model given uncertain inputs. This in turn enables more informed decision-making, risk management and improved predictions. Due to its increasing importance in many areas of science and engineering, the study of forward UQ has attracted a lot of attention from researchers and practitioners over the past several decades. This has led to the development of a variety of methods, as well as interdisciplinary research. However, there is a great need for more advanced methods in order to address more challenging problems as our computational models tend to be more and more complex. This thesis aims at developing a set of cutting-edge methods for forward UQ analysis, and offering some new insights into the problems under consideration, especially from a Bayesian active learning perspective. The main results of the research are summarized and the most important findings are highlighted in the following:

Chapter 2 presents the 'moment-generating function based mixture distribution' (MGF-MD) method for dynamic reliability analysis of nonlinear structures with uncertain structural parameters under stochastic seismic excitations. Such problems can be rather challenging because strong nonlinearity and high-dimensional randomness are likely to coexist. The proposed method tackles the challenge by estimating the extreme value distribution (EVD) from a small number of simulations. The MGF as a descriptor for the EVD of interest is evaluated by Latinized partially stratified sampling method. Then, the EVD is reconstructed by the proposed mixture distribution from the knowledge of the estimated MGF. It is shown that the developed MGF-MD method is able to provide reasonable accuracy and efficiency in three numerical examples.

Chapter 3 is devoted to the development of the 'triple-engine parallel Bayesian global optimization' (T-PBGO) method for interval uncertainty propagation. The main contribution lies in developing a novel infill sampling criterion, i.e., triple-engine pseudo expected improvement strategy, to identify multiple promising points for minimization and/or maximization based on the past observations at each iteration. Two salient features of T-PBGO are that: (1) it can produce the lower and upper bounds of a model output of interest in a single run; and (2) it enables the use of ever-increasing parallel computing facilities.

Chapter 4 reports the Bayesian active learning method, 'parallel Bayesian quadrature optimization' (PBQO), for propagating hybrid uncertainties in the form of probabilistic models, parameter-

ized probability-box models and interval models. In this method, we first treat the estimation of response expectation function (REF) as a Bayesian inference problem. This also allows to quantify the discretization error by a Bayesian approach. Based on the uncertainty representation of the REF, learning functions and stopping criteria are developed for active learning proposes. Three key advantages of PBQO are that: (1) it is fully decoupled in nature; (2) it supports parallel computing; and (3) it is capable of yielding the REF, its variable importance and bounds simultaneously in a single run.

Chapter 5 is about the 'active learning augmented probabilistic integration' (ALAPI) method, which is designed for the propagation of parameterized probability-box models and the failure probability function is of interest. To do so, we first develop a partially Bayesian active learning method, 'active learning probabilistic integration' (ALPI), for probabilistic reliability analysis. In this method, an upper-bound of the posterior variance for the failure probability is derived in analytic form, based on which a learning function is also put forward. The ALAPI method is formed by a combination of ALPI and high-dimensional model representation in the augmented uncertainty space.

Chapter 6 develops a parallel partially Bayesian active learning method, 'parallel adaptive Bayesian quadrature' (PABQ), for probabilistic reliability analysis with small failure probabilities. The theoretical basis of PABQ is rooted in our previously developed ALPI. The main contribution lies in two aspects: (1) enabling the estimation of small failure probabilities by proposing an importance ball sampling; and (2) facilitating parallel distributed processing by devising a multi-point selection strategy.

Chapter 7 offers a Bayesian perspective on the failure probability integral estimation, distinguished from the classical frequentist perspective. We develop a principled 'Bayesian failure probability inference' (BFPI) framework, and it allows to account for the numerical uncertainty behind the failure probability by a Bayesian approach. In addition, a 'parallel adaptive-Bayesian failure probability learning' (PA-BFPL) method is proposed within the BFPI framework. The method can identity a batch of points at each iteration, and hence enables parallel computing.

Chapter 8 proposes a partially Bayesian active learning method based on line sampling, termed 'partially Bayesian active learning line sampling' (PBAL-LS), for probabilistic reliability analysis, especially when involving small failure probabilities. In this method, we derive the posterior mean of the failure probability, as well as an upper bound of the posterior variance. On the basis of

the posterior statistics, two crucial components (i.e., learning function and stopping criterion) are devised for active learning proposes. Besides, the important direction can be adapted on the fly throughout the simulation. The developed method is called 'partially' because we only arrive at an upper bound of the posterior variance of the failure probability.

Chapter 9 provides a more complete 'Bayesian active learning line sampling' (BAL-LS) method for estimating small failure probabilities. Compared to our previously developed PBAL-LS, the main advances in BAL-LS are: (1) an exact expression of the posterior variance for the failure probability is derived; (2) based on which a new learning function and stopping criterion are put forward. The proposed method outperforms several existing methods in terms of the number of performance function evaluations on four test examples, including the PBAL-LS method.

Chapter 10 introduces a more advanced Bayesian active learning alternative, 'Bayesian active learning line sampling with log-normal process' (BAL-LS-LP), to the traditional LS. Different from using a Gaussian process as in both PBAL-LS and BAL-LS, we assign a LP prior over the beta function, which can explicitly account for the discretization error and non-negativity constraint. Additionally, the approximation error of the beta function is also modeled. Numerical studies indicate that the proposed BAL-LS-LP can assess extremely small failure probabilities with desired accuracy and efficiency.

Overall, this thesis develops a set of advanced methods that can be applied to a wide class of problems in the field of forward UQ. Along the development of these methods, we also gain some new insights into the problems of interest, among which a Bayesian active learning perspective is particularly emphasized. The developed methods and gained insights are expected to benefit both research progress and practical applications of forward UQ.

11.2 Future work

The results of this thesis can provide a promising starting point for a number of further studies. Among them, some possible directions are especially suggested in the following:

(i) Bayesian active learning with fully Bayesian Gaussian processes. In most studies of this thesis, we make use of the Gaussian processes with point estimate hyperparameters for Bayesian active learning. However, this kind of Gaussian processes has known problems, e.g., over-fitting and underestimating prediction uncertainty. Alternatively, some fully Bayesian Gaussian pro-

cesses have been developed in the literature, where a Bayesian treatment of the hyperparameters is advocated. The future study can use fully Bayesian Gaussian processes instead for developing Bayesian active leaning methods for forward UQ;

(ii) Bayesian active learning with student-t processes. In addition to the Gaussian processes, there are many different types of Bayesian models. Some of these models might be usable for our Bayesian active learning purposes, e.g., student-t processes. One advantage of student-t processes over Gaussian processes are that they can model heavy tailed behaviour. In the literature, student-t processes have shown to outperform Gaussian processes for Bayesian optimization in many cases. However, the applications of student-t processes to uncertainty propagation still remain open;

(iii) Bayesian active learning with dimension reduction. Whether using Gaussian processes or student-t processes, a common issue is that they are not suitable for modelling high-dimensional problems. A potential solution to this issue is to employ dimension reduction techniques, such as principal component analysis, active subspace and manifold learning, etc.