Envelope Method for Time- and Space-Dependent Reliability Prediction

Hao Wu

Graduate Research Assistant

School of Mechanical Engineering

Purdue University

West Lafayette, IN, 47907, United States

E-mail: wu1508@purdue.edu

Xiaoping Du

Professor

Department of Mechanical and Energy Engineering

Indiana University - Purdue University Indianapolis,

723 W. Michigan Street

Indianapolis, IN, 46202, United States

E-mail: duxi@iu.edu

ABSTRACT

Reliability can be predicted by a limit-state function, which may vary with time and space. This work extends the envelope method for a time-dependent limit-state function to a time- and space-dependent limit-state function. The proposed method uses the envelope function of time- and space-dependent limit-state function. It at first searches for the most probable point (MPP) of the envelope function using the sequential efficient global optimization in the domain of the space and time under consideration. Then the envelope function is approximated by a quadratic function at the MPP, for which analytic gradient and Hessian matrix of the envelope function are derived. Subsequently, the second-order saddlepoint approximation method is employed to estimate the probability of failure. Three examples demonstrate the effectiveness of the proposed method. The method can efficiently produce an accurate reliability prediction when the MPP is within the domain of the space and time under consideration.

1. INTRODUCTION

Reliability is the probability that a product or component performs its intended function under a specific condition. Reliability can be predicted by a physics-based approach if the state of a component can be predicted by a physical model, which is called a limit-state function. A physics-based reliability problem may be time- and space-independent, time-dependent, space-dependent, or time- and space-dependent.

A time- and space-independent reliability problem involves limit-state functions that do not vary with respect to time and space, and its inputs might involve random variables or random fields at a specific point in space. Many methods are available for this problem. Monte Carlo simulation

(MCS) is one method. It is accurate when the sample size is sufficiently large, but it is computationally expensive [1, 2]. When failure probabilities are small in reliability analysis of engineering systems, subset simulation is an alternative method [3]. Importance sampling methods could be used to reduce the computational cost because they generate more samples in the failure region [4].

The first-order reliability method (FORM) [5-7] is much more efficient because it linearizes the limit-state function. FORM can produce satisfactory accuracy for many engineering applications, but it is less accurate for highly nonlinear limit-state functions. The second-order reliability method (SORM) [8, 9] can produce higher accuracy than FORM due to the second-order approximation but is less efficient than FORM. The accuracy of SORM may be further improved by the second-order saddlepoint approximation (SOSPA) since the saddlepoint approximation may yield a more accurate probability estimation, especially in the tail area of distribution [10-12]. Reliability can also be predicted by regressions, such as the Gaussian process method [13-16] and the support vector machines method [17-19].

The limit-state function may vary over time, which results in a time-dependent reliability problem. The input of the limit-state function may involve time and random processes. Rice's formula-based methods are commonly used [20,21]. They are in general more efficient than other methods but may lead to large errors if up-crossing events are strongly dependent [21]. Regression methods can also be used and may achieve higher accuracy if the surrogate model is well trained [23-26]. Converting a time-dependent problem into a time-independent counterpart is possible by using the extreme value of the limit-state function [27-30]. The methods include the envelope function method [27], extreme value response method [28], and the composite limit-state function method [29,30], The most general problems are those with time- and space-dependent limit-sate functions, which may take input of stochastic processes, random fields, and tempo-spatial variables [31-36]. Hu and Mahadevan developed a surrogate modeling approach for reliability analysis of a multidisciplinary system [31]. Shi et al. presented a method for the moment estimation of the extreme response using two strategies [32]. One strategy is combining the sparse grid technique and the fourth-moment method while the other one is combining the dimensional reduction with the maximum entropy method. Shi and Lu proposed an active learning Kriging method [33]. Wei and Du combined FORM and SORM for the time- and space-dependent reliability analysis [34, 35]. Despite the progress, there is still a need to improve the accuracy and efficiency of time- and space-dependent reliability prediction.

The proposed method is an extension of the time-dependent methodology in Ref. [37]. This method converts a time- and space-dependent problem into a time- and space-independent problem by using the envelope function or the extreme value of a limit-state function over the time and the space span. The MPP of the envelope function is found by combing the sequential efficient global optimization (EGO) with FORM. Then the quadratic envelope function is approximated at the MPP with its gradient and Hessian matrix. Then the probability of failure is estimated by the second-order saddlepoint approximation method.

The rest of the paper is organized as follows. Section 2 reviews FORM for time- and space-dependent reliability. Section 3 discusses the proposed method. Section 4 presents three examples, and Section 5 provides the conclusions and future work.

2. Review of Fundamental Methodologies

2.1 Problem Statement

In this work, we consider a limit-state function given by

$$y = g(\mathbf{X}, \mathbf{z}) \tag{1}$$

in which $\mathbf{X} = [X_1, \dots, X_n]^{\mathrm{T}}$ are n input random variables. The time variable is $z_1 \in [\underline{z}_1, \overline{z}_1]$, and the spatial variables are z_k with the following ranges: $z_k \in [\underline{z}_k, \overline{z}_k], (k = 2, \dots, m)$. Then, $\mathbf{z} = [z_1, z_2, \dots, z_m]^{\mathrm{T}}$ is a vector of the temporal/spatial variables bounded on $\mathbf{\Omega} = [\underline{z}_k, \overline{z}_k]$.

The reliability over the temporal and spatial domain is defined by

$$R = \Pr\{g(\mathbf{X}, \mathbf{z}) > 0, \forall \mathbf{z} \in \mathbf{\Omega}\} \tag{2}$$

where $\forall$ means "for all". The associated probability of failure is given by

$$p_f = \Pr\{g(\mathbf{X}, \mathbf{z}) \le 0, \exists \mathbf{z} \in \mathbf{\Omega}\} \tag{3}$$

where $\exists$ means "there exists at least one".

Note that the spatio-temporal domain in Eq. (1) is rectangular. In reality, the domain may be non-rectangular. This study focuses on only a rectangular domain.

2.2 First Order Reliability Method (FORM)

FORM is the commonly used reliability method. It is originally intended for time- and space-independent reliability analysis. In this work, we at first review the time- and space-independent reliability problem with the FORM method, then the discussion furtherly can be extended to the time- and space-dependent reliability problem.

2.2.1 Time- and space-independent reliability problem

The time- and space-independent reliability is defined by

$$R = \Pr\{y = g(\mathbf{X}) > 0\} \tag{4}$$

where y is response and $\mathbf{X}$ is a random vector. FORM at first searches for the most probable point (MPP) in the standard normal space. At first, random variables $\mathbf{X}$ are transformed into standard and independent normal variables $\mathbf{U}$ [38]. Then, the minimum distance from the origin to the limit-state surface $g(\mathbf{X}) = 0$ is identified. The distance is the reliability index β. The minimum distance point is called the MPP. The model for searching for the MPP is given by

$$\begin{cases} \min \sqrt{\mathbf{u}^{\mathrm{T}}\mathbf{u}} \\ \text{s.t. } g(\mathbf{X}) = g(\mathrm{T}(\mathbf{u})) = 0 \end{cases} \tag{5}$$

where $\mathrm{T}(\cdot)$ is an operator of the transformation from **U** to **X.**

$$\beta = \|\mathbf{u}\| = \sqrt{u_1^2 + u_2^2 + \cdots + u_n^2} \tag{6}$$

The solution from Eq. (5) is the MPP $\mathbf{u}_{\mathrm{MPP}}$.

Lastly, the reliability is calculated by

$$R = \Pr\{y = g(\mathbf{X}) > 0\} \approx \Phi(\beta) = \Phi(\|\mathbf{u}_{\mathrm{MPP}}\|) \tag{7}$$

where $\Phi(\cdot)$ is the cumulative distribution function (CDF) of the standard normal distribution.

2.2.2 Time-dependent reliability problem

When it comes to the limit-state function that varies over time, FORM can still be used to find the MPP. The MPP $\mathbf{u}_{\mathrm{MPP}}$ at the time instant z_1 is identified by the following model:

$$\begin{cases} \min \|\mathbf{u}\| \\ \text{s.t. } g(\mathbf{X}, z_1) = g(\mathrm{T}(\mathbf{u}), z_1) = 0 \end{cases} \tag{8}$$

The limit-state function is linearized at $\mathbf{u}_{\mathrm{MPP}}(z_1)$ by

$$g(\mathrm{T}(\mathbf{u}), z_1) \approx g(\mathbf{u}_{\mathrm{MPP}}, z_1) + \sum_{i=1}^{\mathrm{N}} \left.\frac{\partial g}{\partial U_i}\right|_{\mathbf{u}_{\mathrm{MPP}}} (U_i - \mathrm{u}_{\mathrm{MPP}i}) = \nabla g \times (\mathbf{U} - \mathbf{u}_{\mathrm{MPP}}) \tag{9}$$

where $\nabla g = \left[\left.\frac{\partial g}{\partial U_1}\right|_{\mathbf{u}_{\mathrm{MPP}}}, \ldots, \left.\frac{\partial g}{\partial U_n}\right|_{\mathbf{u}_{\mathrm{MPP}}}\right]$ is the gradient, and the probability of failure is computed by

$$p_f = \Pr\left(g(\mathbf{X}, z_1) \le 0, z_1 \in [\underline{z}_1, \overline{z}_1]\right) \approx \Pr\left(\beta(z_1) + \boldsymbol{\alpha}(z_1)\mathbf{U} \le 0, z_1 \in [\underline{z}_1, \overline{z}_1]\right) \tag{10}$$

where $\beta(z_1)$ is the time-dependent reliability index, given by

$$\beta(z_1) = \|\mathbf{u}_{\mathrm{MPP}}\| \tag{11}$$

and $\boldsymbol{\alpha}(z_1)$ is the time-dependent unit gradient vector given by

$$\boldsymbol{\alpha}(z_1) = \frac{\nabla g(z_1)}{\parallel \nabla g(z_1) \parallel} = [\, \alpha_1(z_1), \alpha_2(z_1), \dots, \alpha_n(z_1)] \tag{12}$$

As indicated in Eq. (9), the limit-state function $g(\mathbf{X}, z_1)$ is approximated as a linear combination of standard normal random variables. Many methodologies are available for solving for the probability of failures, such as Rice's formula-based methods and metamodeling-based methods.

3 Envelope Method for Time- and Space-Dependent Problem

The envelope function is tangent to all the instantaneous limit-state functions with respect to time and space. The envelope function of a limit-state function is in general nonlinear and can be approximated as a quadratic function at its MPP by the second-order approximation method.

It is known that the MPP of the envelope function is the worst-case MPP of the limit-state function [37]. In other words, the MPP is the closest point between the origin and all the instantaneous limit-state functions. The MPP of the envelope function can be efficiently found by the sequential single-loop method [37]. Consequently, the gradient of the envelope function is consistent with the gradient of the worst-case limit-state functions at MPP [37]. However, as the curvature of the envelope function may not be the curvature of the worst-case limit-state function, the analytical Hessian matrix of the envelope function is derived. In this paper, we extend our work in a more general situation. The second derivative of the envelope function with respect to random variables and multiple temporal/spatial variables is analytically derived. As a result, the Hessian matrix of the envelope function can be accurately obtained.

Different from the existing method [37], the new method also covers problems where a single call of a limit-state function returns a complete response with respect to time and space. Hence the method can be used for the following two cases.

Case 1: The input includes a realization of random variables $\mathbf{X}$, as well as a time instance/spatial location $\mathbf{z}$, and the output is a single response. This case requires calling the limit-state function repeatedly so that the worst-case response can be found.

Case 2: The input includes a realization of random variables $\mathbf{X}$ and the temporal/spatial domain $\mathbf{\Omega}$ of $\mathbf{z}$. Calling the limit-function returns a complete time- and space-dependent response with respect to $\mathbf{z}$ in $\mathbf{\Omega}$. In this case, the output is a hypersurface of the response $y(\mathbf{z})$. For example, if we call a computational fluid dynamics (CFD) simulation, we obtain the 4-D pressure and velocity fields with respect to time and space. Since we know $y(\mathbf{z})$, the minimum value $\min_{\mathbf{z}\in\mathbf{\Omega}} y(\mathbf{z})$ is also known.

In Sec. 3.1, we focus our discussions on Case 1 for limit-state function $y = g(\mathbf{X}, \mathbf{z})$. Since Case 2 is much easier than Case 1, we briefly discuss it at the end of Sec. 3.1. We then extend the method into a general problem with input random fields in Sec. 3.2.

3.1 Problems include random variables, and explicit temporal/spatial parameters

We now discuss Case 1 with the limit-state function is given in Eq. (1). For this case we need to search for the worst-case MPP.

3.1.1 Search for worst-case MPP

The time- and space-dependent probability of failure in the time span $[\underline{z}_1, \overline{z}_1]$ and the space span $[\underline{z}_k, \overline{z}_k]$ can be evaluated by the extreme value of the limit-state function.

$$p_f = \Pr(g(\mathbf{X}, \mathbf{z}) < 0, \exists \mathbf{z} \in \mathbf{\Omega}) = \Pr\left(\min_{\mathbf{z}\in\mathbf{\Omega}} g(\mathbf{X}, \mathbf{z}) < 0\right) \tag{13}$$

Eq. (13) indicates that a failure occurs if the minimum response is negative. The function of the extreme response is equivalent to the envelope function or the composite limit-state function [29], which is given by

$$G(\mathbf{X}) = \min_{\mathbf{z}\in\mathbf{\Omega}} g(\mathbf{X}, \mathbf{z}) = g(\mathbf{X}, \tilde{\mathbf{z}}) \tag{14}$$

where the envelope function $G(\mathbf{X})$ is the global minimum of $g(\mathbf{X},\mathbf{z})$ with respect to $\mathbf{z}$, and the global minimum occurs at $\tilde{\mathbf{z}}$.

If FORM is used to linearize $G(\mathbf{X})$, the MPP is obtained by

$$\begin{cases}\min\sqrt{\mathbf{u}^{\mathrm{T}}\mathbf{u}} \\ \text{s.t.} \min_{\mathbf{z}\in\boldsymbol{\Omega}} g(\mathrm{T}(\mathbf{u}),\mathbf{z}) = 0\end{cases} \tag{15}$$

Eq. (15) requires a double loop optimization process because minimization appears in both the objective and constraint functions. The inner loop is for the minimum value of $g(\mathrm{T}(\mathbf{u}),\mathbf{z})$ relative to $\mathbf{z}$ while the outer loop is the MPP search relative to $\mathbf{u}$. In this work, we decouple the double loop into sequential single loops.

The first loop is FORM analysis, the MPP $\mathbf{u}_{\mathrm{MPP}}^{(1)}$ at the initial $\tilde{\mathbf{z}}^{(0)} = [z_1^0, z_2^0, \dots z_m^0]$ is obtained by

$$\begin{cases}\min\sqrt{\mathbf{u}^{\mathrm{T}}\mathbf{u}} \\ \text{s.t. } g(\mathrm{T}(\mathbf{u}),\mathbf{z}_0) = 0\end{cases} \tag{16}$$

Then $\mathbf{z}$ is determined by fixing the random variables on its realization $\mathbf{u}_{\mathrm{MPP}}^{(1)}$, and $\mathbf{z}$ is denoted by $\tilde{\mathbf{z}}^{(1)}$, which is given by

$$\tilde{\mathbf{z}}^{(1)} = \underset{\mathbf{z}\in\boldsymbol{\Omega}}{\operatorname{argmin}}\, g\left(\mathrm{T}\left(\mathbf{u}_{\mathrm{MPP}}^{(1)}\right),\mathbf{z}\right) \tag{17}$$

In the next loop, the new MPP $\mathbf{u}_{\mathrm{MPP}}^{(2)}$ is located at point $\tilde{\mathbf{z}}^{(1)}$ using Eq. (16). And then $\mathbf{z}$ is updated to $\tilde{\mathbf{z}}^{(2)}$.

$$\tilde{\mathbf{z}}^{(2)} = \underset{\mathbf{z}\in\boldsymbol{\Omega}}{\operatorname{argmin}}\, g\left(\mathrm{T}\left(\mathbf{u}_{\mathrm{MPP}}^{(2)}\right),\mathbf{z}\right) \tag{18}$$

The above process is repeated until convergence, and the MPP is found. It is the worst-case MPP of the limit-state function with respect to $\mathbf{z}$.

3.1.2 Find the global minimum value of $G(\mathbf{X})$

The global minimum value of $G(\mathbf{X})$ occurs at $\tilde{\mathbf{z}}^{(1)} = [\tilde{z}_1^{(1)}, \tilde{z}_2^{(1)}, \dots, \tilde{z}_m^{(1)}]$, which is given by

$$\tilde{\mathbf{z}}^{(1)} = \underset{\mathbf{z}\in\boldsymbol{\Omega}}{\operatorname{argmin}}\, g(\mathrm{T}(\mathbf{u}_{\mathrm{MPP}}), \mathbf{z}) \tag{19}$$

Note that finding the optimal point is still in the sequential loops. There are many methods to solve the optimal point $\tilde{\mathbf{z}}^{(1)}$ corresponding to the global minimum value of $G(\mathbf{X})$. The first partial derivative of the limit-state function with respect to z_k at MPP is as below:

$$\begin{cases} \dfrac{\partial g(\mathrm{T}(\mathbf{u}_{\mathrm{MPP}}), z_1, z_2, \dots, z_m)}{\partial z_1} = 0 \\ \vdots \\ \dfrac{\partial g(\mathrm{T}(\mathbf{u}_{\mathrm{MPP}}), z_1, z_2, \dots, z_m)}{\partial z_m} = 0 \end{cases} \tag{20}$$

The optimal point $\tilde{\mathbf{z}}^{(1)} = [\tilde{\mathrm{z}}_1^{(1)}, \tilde{\mathrm{z}}_2^{(1)}, \dots \tilde{\mathrm{z}}_m^{(1)}]$ can be obtained by solving Eq. (20).

We use efficient global optimization (EGO) to find the MPP. EGO has been widely used in various areas because it can efficiently search for the global optimum [39]. Suppose we have called the limit-state function at several initial training points of $\mathbf{z}^{in}$ and the number of initial training points is n_{in}, which denote by as follows

$$\mathbf{z}^{in} = \begin{bmatrix} z_1^1 & \cdots & z_m^1 \\ \vdots & \ddots & \vdots \\ z_1^{n_{in}} & \cdots & z_m^{n_{in}} \end{bmatrix}$$

and the associated responses are $\mathbf{y}^{in} = [g(\mathrm{T}(\mathbf{u}^*), \mathbf{z}^1), g(\mathrm{T}(\mathbf{u}^*), \mathbf{z}^2), \dots, g(\mathrm{T}(\mathbf{u}^*), \mathbf{z}^{n_{in}})]^{\mathrm{T}}$. An initial function is fitted from $(\mathbf{z}^{in}, \mathbf{y}^{in})$ by the following surrogate model [39]:

$$\hat{y} = g(\mathbf{z}) = g(\mathrm{T}(\mathbf{u}^*), \mathbf{z}) = F(\mathbf{z})^{\mathrm{T}}\gamma + e(\mathbf{z}) \tag{21}$$

where $F(\mathbf{z})^{\mathrm{T}}\gamma$ is a deterministic term, $e(\mathbf{z})$ is a vector of regression functions, γ is a vector of regression coefficients, and $e(\mathbf{z})$ is a stationary Gaussian process with zero mean and a covariance given by

$$\text{Cov}\big(e(\mathbf{z}_1), e(\mathbf{z}_2)\big) = \sigma_e^2 C(\mathbf{z}_1, \mathbf{z}_2) \tag{22}$$

where σ_e^2 is process variance, and $C(\cdot,\cdot)$ is the correlation function.

The output of the surrogate model is a Gaussian random variable following

$$\hat{y} = g(\mathbf{z}) \sim N\big(\mu(\mathbf{z}), \sigma^2(\mathbf{z})\big) \tag{23}$$

where $\mu(\mathbf{z})$ and $\sigma(\mathbf{z})$ are the mean and standard deviation of $\hat{y}$, respectively.

The initial model is likely not accurate. The expected improvement (EI) metric [39] is used to identify new training points that will be added to refine the model. The improvement is defined by

$$\text{I} = \max(y^* - y, 0) \tag{24}$$

where $y^* = \min\limits_{i=1,2,\ldots,n_{in}} g(\mathbf{z}^i)$ is the minimum from the sampling training points.

EI is computed by

$$\text{EI}(\mathbf{z}) = \text{E}[\max(y^* - y, 0)] = \big(y^* - \mu(\mathbf{z})\big)\Phi\left(\frac{y^* - \mu(\mathbf{z})}{\sigma(\mathbf{z})}\right) + \sigma(\mathbf{z})\phi\left(\frac{y^* - \mu(\mathbf{z})}{\sigma(\mathbf{z})}\right) \tag{25}$$

where $\phi(\cdot)$ is the probability density function (PDF).

A new training point $\mathbf{z}_{new}$ is identified by minimizing the expected improvement.

$$\mathbf{z}_{new} = \underset{\mathbf{z}}{\text{argmin}}\text{EI}(\mathbf{z}) \tag{26}$$

By combining sequential strategy with EGO, $\mathbf{u}_{\text{MPP}}$ of envelope function $G(\mathbf{X})$ can be obtained efficiently by solving Eq. (15). The probability of failure with FORM is estimated by

$$p_f = \Pr(g(\mathbf{X}, \mathbf{z}) < 0, \exists \mathbf{z} \in \mathbf{\Omega}) \approx \Pr\,(G(\mathbf{X}) < 0) = \Phi(-\beta) \tag{27}$$

where $\beta = \| \mathbf{u}_{\text{MPP}} \|$ is the first-order reliability index.

In general, the envelope function is nonlinear, and FORM may not be accurate enough. Thus, a second-order method is preferred, and it uses the envelope theorem to obtain the second-order information of the extreme limit-state function. Then SOSPA is used to estimate the probability of failure.

3.1.3 Derivatives of the envelope function

The envelope function is generally nonlinear, and we therefore approximate it as a quadratic function, instead of a linear function in FORM. As a result, we need the gradient ∇G and Hessian matrix $\mathbf{H}$ at the MPP of the envelope function. The quadratic function is formed as follows [12]:

$$G(\mathbf{U}) = \mathrm{a} + \mathbf{b}^{\mathrm{T}}\mathbf{U} + \mathbf{U}^{\mathrm{T}}\mathbf{C}\mathbf{U} \tag{28}$$

where

$$\begin{cases} a = \frac{1}{2}(\mathbf{u}_{\mathrm{MPP}})^{\mathrm{T}}\mathbf{H}\mathbf{u}_{\mathrm{MPP}} - \nabla G(\mathbf{u}_{\mathrm{MPP}})^{\mathrm{T}}\mathbf{u}_{\mathrm{MPP}} \\ \mathbf{b} = \nabla G(\mathbf{u}_{\mathrm{MPP}}) - \mathbf{H}\mathbf{u}_{\mathrm{MPP}} = (\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_n) \\ \mathbf{C} = \frac{1}{2}\mathbf{H} = \mathrm{diag}(\tilde{c}_1, \tilde{c}_2, \dots, \tilde{c}_n) \end{cases} \tag{29}$$

$\nabla G(\mathbf{u}^*) = \left[\left.\frac{\partial G}{\partial U_1}\right|_{\mathbf{u}_{\mathrm{MPP}}}, \dots, \left.\frac{\partial G}{\partial U_n}\right|_{\mathbf{u}_{\mathrm{MPP}}}\right]^{\mathrm{T}}$ is the gradient of the envelope function. $\mathbf{H}$ is the Hessian matrix shown below.

$$\mathbf{H} = \left.\begin{bmatrix} \frac{\partial^2 G}{\partial U_1^2} & \cdots & \frac{\partial^2 G}{\partial U_1 \partial U_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 G}{\partial U_n \partial U_1} & \cdots & \frac{\partial^2 G}{\partial U_n^2} \end{bmatrix}\right|_{\mathbf{u}_{\mathrm{MPP}}} \tag{30}$$

The envelope function $G(\mathbf{X})$ at $\mathbf{u}_{\mathrm{MPP}}$ is given by

$$G(\mathbf{U}) = \min_{\mathbf{z}\in\mathbf{\Omega}} g(\mathbf{U}, \mathbf{z}) = g(\mathbf{U}, \tilde{\mathbf{z}})|_{\mathbf{u}_{\mathrm{MPP}}} \tag{31}$$

where $\tilde{\mathbf{z}} = [\tilde{\mathrm{z}}_1, \dots, \tilde{\mathrm{z}}_m]$ is the optimal point where the global minimum value of function $g(\mathbf{U}, \mathbf{z})$ occurs, and it is found by

$$\frac{\partial g(\mathbf{U}, \mathbf{z})}{\partial z_1} = \frac{\partial g(\mathbf{U}, \mathbf{z})}{\partial z_2} = \cdots = \frac{\partial g(\mathbf{U}, \mathbf{z})}{\partial z_m} = 0 \tag{32}$$

The envelope function satisfies the following equation:

$$\begin{cases} \dot{g}(\mathbf{U}, \tilde{z}_1, z_2, \dots, z_m) = 0 \\ \vdots \\ \dot{g}(\mathbf{U}, z_1, z_2, \dots, \tilde{z}_m) = 0 \end{cases} \tag{33}$$

where $\dot{g}$ is the derivative of g with respect to z_i.

Next, the first derivative of $G(\mathbf{U})$ with respect to a random input variable U_i at $\mathbf{u}_{\text{MPP}}$ is

$$\frac{\partial G}{\partial U_i} = \frac{\partial g}{\partial U_i} + \frac{\partial g}{\partial \tilde{z}_1}\frac{\partial \tilde{z}_1}{\partial U_i} + \frac{\partial g}{\partial \tilde{z}_2}\frac{\partial \tilde{z}_2}{\partial U_i} + \cdots + \frac{\partial g}{\partial \tilde{z}_m}\frac{\partial \tilde{z}_m}{\partial U_i} \tag{34}$$

By plugging Eq. (33) into Eq. (34), it becomes

$$\frac{\partial G}{\partial U_i} = \frac{\partial g}{\partial U_i} \tag{35}$$

Eq. (35) indicates that the gradient of the envelope function ∇G is equal to the gradient of the limit-state function ∇g at the MPP. Subsequently, the second derivative of $G(\mathbf{U})$ with respect to the input random variables U_j at $\mathbf{u}^*$ is

$$\frac{\partial^2 G}{\partial U_i \partial U_j} = \frac{\partial}{\partial U_j}\left(\frac{\partial G}{\partial U_i}\right) = \frac{\partial}{\partial U_j}\left(\frac{\partial g}{\partial U_i}\right) = \frac{\partial^2 g}{\partial U_i \partial U_j} + \frac{\partial^2 g}{\partial U_i \partial \tilde{z}_1}\frac{\partial \tilde{z}_1}{\partial U_j} + \cdots + \frac{\partial^2 g}{\partial U_i \partial \tilde{z}_m}\frac{\partial \tilde{z}_m}{\partial U_j} \tag{36}$$

Take the derivative of Eq. (32) with respect to U_j, and it is given by

$$\frac{\partial \dot{g}}{\partial U_j} + \frac{\partial \dot{g}}{\partial \tilde{z}_k}\frac{\partial \tilde{z}_k}{\partial U_j} = 0 \tag{37}$$

$$\frac{\partial \tilde{z}_k}{\partial U_j} = -\frac{\partial \dot{g}}{\partial U_j} \Big/ \frac{\partial \dot{g}}{\partial \tilde{z}_k} = -\frac{\partial^2 g}{\partial \tilde{z}_k \partial U_j} \Big/ \frac{\partial^2 g}{\partial \tilde{z}_k^2} \tag{38}$$

The Hessian matrix $\mathbf{H}$ with respect to random variables and multiple temporal/spatial variables is obtained by plugging Eq. (38) into Eq. (36) at $\mathbf{u}_{\text{MPP}}$, $\tilde{z}_k$.

$$\left.\frac{\partial^2 G}{\partial U_i \partial U_j}\right|_{\mathbf{u}^*, \tilde{z}_k} = \frac{\partial^2 g}{\partial U_i \partial U_j} - \sum_{k=1}^{m} \frac{\partial^2 g}{\partial U_i \partial \tilde{z}_k}\frac{\partial^2 g}{\partial U_j \partial \tilde{z}_k} \Big/ \frac{\partial^2 g}{\partial \tilde{z}_k^2} \tag{39}$$

The forward finite difference method with step size $\delta = \max\left(|u|/1000, \epsilon\right)$, where $\epsilon = 10^{-4}$, is employed to calculate the derivations in Eq. (39).

3.1.4 Saddlepoint approximation

Once the envelope function is approximated by a quadratic function, we use the second order saddlepoint approximation to estimate the probability of failure. The reason we use saddlepoint approximation is due to its high accuracy in the tail area of a distribution; a failure usually occurs in a tail area.

Eq. (28) can be written as the sum of quadratic functions of different standard normal variables

$$G(\mathbf{U}) = \sum_{i=1}^{n} Q_i(\widetilde{\mathbf{U}}) = \sum_{i=1}^{n} \left(\tilde{a}_i + \tilde{b}_i \widetilde{U}_i + \tilde{c}_i \widetilde{U}_i^2\right) \tag{40}$$

The cumulant generating function (CGF) of $G(\mathbf{U})$ is given by

$$K_Q(t_s) = \sum_{i=1}^{n} K_{Q_i}(t_s) \tag{41}$$

After the CGF $K_Q(t_s)$ is obtained, it is straightforward to find the PDF of the limit-state function, and this needs to solve the saddlepoint t_s, which is found by solving the following equation:

$$K_Q^{'}(t_s) = 0 \tag{42}$$

where $K_Q^{'}(t_s)$ is the first derivative of $K_Q(t_s)$. The details of the implementation of SOSPA refer to Ref. [12]. According to Lugannani and Rice's formula,

Then the probability of failure is evaluated by

$$p_f \approx \Pr(G(\mathbf{U}) < 0) = \Phi(w) + \phi(w)\left(\frac{1}{w} - \frac{1}{v}\right) \tag{43}$$

where

$$w = \operatorname{sgn}(t_s)\{2[-K_Q(t_s)]\}^{\frac{1}{2}} \tag{44}$$

$$v = t_s\left[K_Q''(t_s)\right]^{\frac{1}{2}} \tag{45}$$

in which $\operatorname{sgn}(t_s) = +1, -1$, or 0, depending on whether t_s is positive, negative, or zero. $K_Q''(t_s)$ is the second derivative of $K_Q(t_s)$ concerning t_s. Since the above method uses SOSPA and envelope theorem, we denote this method as SOSPA/ENV.

Case 2: Calling the limit-function returns a complete time- and space-dependent response

In this case, the output is a hypersurface of the response $y(\mathbf{z})$. The complete response $y(\mathbf{z})$ is available, so the minimum value $\min_{\mathbf{z}\in\boldsymbol{\Omega}} y(\mathbf{z})$ is also known. We do not need to use the sequential single loops in case 1. Thus, the MPP in Eq. (15) can be obtained from the following model:

$$\begin{cases} \min \sqrt{\mathbf{u}^{\mathrm{T}}\mathbf{u}} \\ \text{s.t.} \min_{\mathbf{z}\in\boldsymbol{\Omega}} y(\mathbf{z}) = 0 \end{cases} \tag{46}$$

where $\min_{\mathbf{z}\in\boldsymbol{\Omega}} y(\mathbf{z})$ is a function of $\mathbf{u}$ and is obtained by calling the limit-state function once at $\mathbf{u}$, where $\mathbf{u}$ is the vector of independent normal variables transformed from $\mathbf{X}$. We just need a single-loop MPP search, which is more efficient than the sequential loop approach.

The model in Eq. (46) may have multiple MPPs [40]. The accuracy of the reliability prediction may be poor if only one MPP is used and if other MPPs also have significant contributions. There are three strategies to deal with multiple MPPs. The first strategy is to repeat the standard MPP search with different starting points and find different solutions if they exist. The second strategy is to use an optimization algorithm that can find multiple local optima. The methods include genetic algorithm [40] and particle swarm optimization [41]. The third strategy is to employ methodologies specifically designed for multiple MPP search [29,42]. Although there is no guarantee to find all possible MPPs, these strategies can significantly increase the chance of finding multiple MPPs [29,40-42]. Once all potential MPPs are identified, the corresponding limit-state surfaces are linearized at these points as

$$Q_i(\mathbf{U}) = -\nabla G(\mathbf{u}_{\mathrm{MPP}i})^{\mathrm{T}}\mathbf{u}_{\mathrm{MPP}i} + \nabla G(\mathbf{u}_{\mathrm{MPP}i})\mathbf{U} \tag{47}$$

where $i = 1,2,\ldots,m$, in which m is the number of MPPs. The reliability is calculated as the reliability of a series system.

$$R = \Pr\left(\bigcap_{i=1}^{m} Q_i(\mathbf{U}) > 0\right) = \left(\bigcap_{i=1}^{m} = -\nabla G(\mathbf{u}_{\mathrm{MPP}i})^{\mathrm{T}}\mathbf{u}_{\mathrm{MPP}i} + \nabla G(\mathbf{u}_{\mathrm{MPP}i})\mathbf{U} > 0\right) \tag{48}$$

Since $Q_i(\mathbf{U})$ follows a normal distribution, all the responses at their MPPs follow a multivariate normal distribution, whose joint probability density is integrated in the safe region, resulting the reliability. The second order method is used for higher accuracy. The method still uses a multivariate normal distribution, whose mean vector is obtained by the second order saddlepoint approximation and whose covariance matric is estimated by the first order approximation [12].

3.2 Extension to problems with random variable, random fields, and temporal/spatial variables z

We have discussed limit-state functions with random variables **X** and temporal/spatial variables **z**. In this subsection, we discuss how to extend the method to limit-state functions with random variable **X**, random fields $\boldsymbol{\mathcal{F}}(\mathbf{z})$ and temporal/spatial variables **z.** A limit-state function is given by $y(\mathbf{z}) = \boldsymbol{g}(\mathbf{X}, \boldsymbol{\mathcal{F}}(\mathbf{z}), \mathbf{z})$. The time- and space-dependent probability of failure is calculated by

$$p_f = \Pr(g(\mathbf{X}, \boldsymbol{\mathcal{F}}(\mathbf{z}), \mathbf{z}) < 0, \exists \mathbf{z} \in \boldsymbol{\Omega}) = \Pr\left(\min_{\mathbf{z}\in\boldsymbol{\Omega}} y(\mathbf{z}) < 0\right) \tag{49}$$

Eq. (49) indicates that failure happens when the minimum value of the limit-state function g(**X**, $\boldsymbol{\mathcal{F}}$(**z**),**z**) is negative. There are still two cases: a single call of a limit-state function does not return a time- and space-dependent response and a single call of a limit-state function returns a complete response with respect to time and space.

Case 1 requires calling the limit-state function repeatedly to obtain the worst-case response in $\mathbf{\Omega}$. We need to convert random fields into time- and space-dependent random variables so that the proposed method can be used. The expansion optimal linear estimation method (EOLE) [43] can be used to convert the random fields $\boldsymbol{\mathcal{F}}(\mathbf{z})$ into independent standard Gaussian random variables $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_r)$, where r is the dimension of $\boldsymbol{\xi}$. Take a two-dimensional random field $\mathcal{F}(\boldsymbol{z})$, where $\boldsymbol{z} = (z_1, z_2)$, as an example. z_1 and z_2 are discretized into n_{z_1} and n_{z_2} points, respectively. The autocorrelation coefficient matrix is given by

$$\mathbf{\Sigma} = \left[\rho\left(\mathbf{z}_i,\ \mathbf{z}_j\right)\right]_{n_{z_1}n_{z_2}\times n_{z_1}n_{z_2}} \tag{50}$$

where $\rho\left(\mathbf{z}_i,\ \mathbf{z}_j\right)$ is the correlation between two points $\mathbf{z}_i$ $(i = 1,2, \dots, n_{z_1}n_{z_2})$ and $\mathbf{z}_j$ $(j = 1,2, \dots, n_{z_1}n_{z_2})$ in the domain of $\mathcal{F}(\mathbf{z})$. Then $\mathcal{F}(\mathbf{z})$ is expanded by

$$\mathcal{F}(\boldsymbol{\xi}, \mathbf{z}) \approx \mu(\mathbf{z}) + \sigma(\mathbf{z}) \sum_{k=1}^{r} \frac{\xi_k}{\sqrt{\lambda_k}} \boldsymbol{\phi}_k^{\mathrm{T}} \mathbf{\Sigma}(:, \mathbf{z}), k = 1,2, \dots, r \tag{51}$$

where $\mu(\mathbf{z})$ is the mean of $\mathcal{F}(\mathbf{z})$, and $\sigma(\mathbf{z})$ is the standard deviation of $\mathcal{F}(\mathbf{z})$. ξ_k $(k = 1,2, \dots r)$ are independent standard normal variables, $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \dots, \lambda_r)$ is the eigenvalue vector, and $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \dots, \boldsymbol{\phi}_r$ are eigenvectors of $\mathbf{\Sigma}$. Note that r is determined as the smallest integer that meets the following criterion:

$$\frac{\sum_{j=1}^{r} \lambda_k}{\sum_{j=1}^{n_{z_1}n_{z_2}} \lambda_k} \geq \eta \tag{52}$$

where η is a hyperparameter determining the accuracy of the expansion. It takes a value close to, but not larger than 1. The smaller is η, the less accurate is the expansion. If $\eta = 1$, the expansion is exact at the points of discretization. Normally, η is set to 0.9999.

Then the limit-state function becomes $y = g(\widetilde{\mathbf{X}}, \mathbf{z})$, where $\widetilde{\mathbf{X}} = (\mathbf{X}, \boldsymbol{\xi})$. It is a function given in Eq. (1) and the proposed method in Sec. 3.1 of case 1 can be used.

For Case 2, a single call of a limit-state function returns a complete response with respect to time and space. After random fields are expanded with respect to random variables, the problem becomes the one discussed in Sec. 3.1 for Case 2. The same method in Sec. 3.1 can then be used.

3.3 Implementation

The detailed steps of solving time- and space-dependent reliability problems using SOSPA are summarized below.

Step 1: Transform random variables $\mathbf{X}$ into $\mathbf{U}$ in the standard normal space.

Step 2: Set $k = 1$. Generate a random point $\mathbf{z} \in \Omega$ as the initial optimal point $\tilde{\mathbf{z}}^{(0)}$ and use a unit vector as the initial MPP $\mathbf{u}_{\mathrm{MPP}}^{(1)} = \mathbf{u}_0$.

Step 3: Perform the MPP search at the point $\tilde{\mathbf{z}}^{(k-1)}$, and obtain the MPP $\mathbf{u}_{\mathrm{MPP}}^{(k)}$ and the corresponding $\beta^{(k)}$ by solving the following optimization model:

$$\begin{cases} \min \sqrt{\mathbf{u}^{\mathrm{T}}\mathbf{u}} \\ \mathrm{s.t.}\ \ g\left(\mathrm{T}(\mathbf{u}), \tilde{\mathbf{z}}^{(k-1)}\right) = 0 \end{cases}$$

Step 4: Determine the optimal point $\tilde{\mathbf{z}}^{(k)}$ implementing EGO method at $\mathbf{u}_{\mathrm{MPP}}^{(k)}$. The optimal point $\tilde{\mathbf{z}}^{(k)}$ makes the limit-state function minimized. The initial number of training points to determine the time and spatial parameters is n_{in}=2.

$$\tilde{\mathbf{z}}^{(k)} = \underset{\mathbf{z} \in \boldsymbol{\Omega}}{\operatorname{argmin}}\, g\left(\mathrm{T}\left(\mathbf{u}_{\mathrm{MPP}}^{(k)}\right), \mathbf{z}\right)$$

Step 5: Repeat step 3 and step 4 until convergence. The convergence criterion is defined as

$$|\beta^{(k)} - \beta^{(k-1)}| \leq \varepsilon$$

The tolerance ε can take a small positive value, for example, 10^{-4}. If $\left|\beta^{(k)} - \beta^{(k-1)}\right| \leq 10^{-4}$, terminate the iteration. Otherwise, set $k = k + 1$, and return to step 3. Note that the method of a single-loop MPP search can be used if calling the limit-state function returns a complete time- and space-dependent response

Step 6: Calculate the gradient ∇G and Hessian matrix $\mathbf{H}$ of the envelope function.

Step 7: Calculate the probability of failure using SOSPA/ENV from the above information $\mathbf{u}_{\mathrm{MPP}}^{(k)}$, gradient ∇G, and Hessian matrix $\mathbf{H}$.

4 EXAMPLES

In this section, three examples are used to demonstrate the proposed method. Example 1 is a mathematical problem that is used to show the details of the proposed method. The remaining examples are engineering problems. MCS is employed to provide accurate solutions for the accuracy comparison. SOSPA/ENV is compared with the FORM-based envelope method (FORM/ENV). The errors of SOSPA/ENV and FORM/ENV are calculated by

$$\varepsilon = \frac{\left|p_f - p_f^{\mathrm{MCS}}\right|}{p_f^{\mathrm{MCS}}} \times 100\% \tag{53}$$

where p_f is the result from SOSPA/ENV or FORM/ENV, and p_f^{MCS} is the result from MCS. We also use the number of function calls as a measure of efficiency.

4.1 Example 1: A math problem

This example is a math problem, which belongs to Case 1 without any random field input. The limit-state function $g(\mathbf{X}, s, t)$ regarding random variables and explicit temporal/spatial parameter is defined by

$$g(\mathbf{X}, s, t) = X_1^2 X_2 - 5X_1 t + (X_2 + 1)t^2 - 2X_2 s + X_1 s^2 - 8 \tag{54}$$

where $\mathbf{X} = (X_1, X_2)$, X_i $(i = 1,2)$ are normally distributed with parameters $\mu_i = 3.5$ and $\sigma_i = 0.25$. The temporal parameter is $t \in [0,5]$ and the spatial parameter is $s \in [0,5]$. Therefore, $\mathbf{z} = (s, t)$, and $\Omega = \{[0,5] \times [0,5]\}$. X_1 and X_2 are independent.

We can easily plot the envelope function for this problem since an analytic envelope function $G(\mathbf{X})$ is available for this problem. From the partial derivatives of the limit-state function with respective to t and s

$$\begin{cases} \dfrac{\partial g(\mathbf{X}, s, t)}{\partial t} = 0 \\ \dfrac{\partial g(\mathbf{X}, s, t)}{\partial s} = 0 \end{cases} \tag{55}$$

we have

$$\begin{cases} t = \dfrac{5X_1}{2(X_2 + 1)} \\ s = \dfrac{X_2}{X_1} \end{cases} \tag{56}$$

Plugging Eq. (56) into Eq. (54) yields the envelope function.

$$G(\mathbf{X}) = X_1^2 X_2 - \frac{25X_1^2}{4(X_2 + 1)} - \frac{X_2^2}{X_1} - 8 \tag{57}$$

The envelope function at the limit state $G(\mathbf{X}) = 0$ is plotted in Fig. 1, and the failure region is colored grey. The figure shows that the envelope function is nonlinear.

Place Fig. 1here

Fig. 1 The envelope function

Even though the envelope function has an explicit function, we treat it as a black box by following the numerical procedure discussed in Sec. 3. SOSPA/ENV searches for the worst-case MPP with the sequential EGO. Table 1 shows the iteration history of the MPP search. The worst-case MPP is found at $\mathbf{u}_{\text{MPP}} = (-2.1702, -2.6185)$ with $\tilde{t} = 1.8150$ and $\tilde{s} = 0.8763$. Fig. 2 displays the convergence history of first-order reliability index β. With FORM/ENV, the probability of failure is $p_f = 3.3575 \times 10^{-4}$.

Once the worst-case MPP is available, the gradient and Hessian matrix are computed at the MPP. The latter is given by

$$\nabla^2 G(\mathbf{u}_{\mathrm{MPP}}) = \begin{bmatrix} 0.1200 & 0.5542 \\ 0.5542 & -0.1494 \end{bmatrix}$$

Table 1 Iteration history of searching for the worst-case MPP

Place Table 1 here

Place Fig. 2 here

Fig. 2 Convergence history of reliability index β

Then SOSPA/ENV produces $p_f = 4.9022 \times 10^{-4}$. The number of simulations for MCS is $N_C = 10^7$. The time and space intervals are discretized evenly into 20 points, yielding 400 points. Accordingly, the number of function calls of MCS is 4×10^9.

All the results are shown in Table 2. SOSPA/ENV is much more accurate than FORM/ENV as the error of the former is 3.5% while that of the latter is 33.9%. SOSPA/ENV, however, is less efficient than FORM/ENV.

Table 2 Results of Example 1

Place Table 2 here

4.2 Example 2: A Truss Structure

A truss structure is shown in Fig. 3. This example belongs to Case 1 without any random field input. The inputs of this truss structure are random variables, temporal parameter t and spatial parameter h. Each bar of the system has its cross-sectional area A_i and the modulus of elasticity E_i, $i = 1,2,3$. The coefficient of thermal expansion of all bars is $\alpha = 12 \times 10^{-6}\text{°C}^{-1}$. The temperature change is related to the installation height of the truss structure and is given by $\Delta T = Te^{-0.01(\Delta h^2+2\Delta h+1)^2}$, where $\Delta h \in [2,5]$ m is the difference of two different installation heights. A downward force $P = P_0(0.9 + 0.1\cos(0.2t))$ is applied at joint *A*, where $t \in [0,10]$ years. The domain Ω of $\mathbf{z} = [\Delta h, t]$ is $\{[2,5] \times [0,10]\}$. All the random variables are given in Table 3.

Place Fig. 3 here

Fig. 3 A truss structure

The perpendicular displacement of joint *A* is calculated by

$$\Delta\delta = \frac{A}{B} \tag{58}$$

where

$$A = L_{\text{AD}}(PA_1E_1L_{\text{AC}}\cos\theta_1^2 + PA_2E_2L_{\text{AB}}\cos\theta_2^2 + A_1A_3E_1E_3L_{\text{AC}}T\alpha\cos\theta_1^2 + A_2A_3E_2E_3L_{\text{AB}}T\alpha\cos\theta_2^2 + A_1A_2E_1E_2T\alpha(L_{\text{AB}}\sin\theta_1\cos\theta_2^2 + L_{\text{AC}}\sin\theta_2\cos\theta_1^2 + L_{\text{AC}}\sin\theta_1\cos\theta_2\cos\theta_1 + L_{\text{AB}}\sin\theta_2\cos\theta_2\cos\theta_1))$$

$$B = A_1A_3E_1E_3L_{\text{AC}}\cos\theta_1^2 + A_2A_3E_2E_3L_{\text{AB}}\cos\theta_2^2 + A_1A_2E_1E_2L_{\text{AD}}(\sin\theta_2^2\cos\theta_1^2 + \sin\theta_1^2\cos\theta_2^2 + 2\sin\theta_1\sin\theta_2\cos\theta_1\cos\theta_2)$$

$$\theta_1 = \arctan\left(\frac{L_{AD}}{\sqrt{L_{AB}^2 - L_{AD}^2}}\right)$$

$$\theta_2 = \arctan\left(\frac{L_{AD}}{\sqrt{L_{AC}^2 - L_{AD}^2}}\right)$$

A failure occurs when $\Delta\delta > 0.65$ mm. Thus, the limit-state function is defined by

$$g(\mathrm{X}, s, t) = 0.65 - \Delta\delta \tag{59}$$

Table 3 Random variables of Example 2

Place Table 3 here

10^7 samples are used for MCS and the domain of $\mathbf{Z} = (\Delta h, t)$ is discretized evenly into $10 \times 10 = 100$ points. FORM/ENV and SOSPA/ENV are used to calculate the probability of failure. Table 4 shows the results. Even though FORM/ENV is more efficient than SOSPA/ENV, it produces a large error. SOSPA/ENV achieves higher accuracy than FORM/ENV although it needs more function calls.

Table 4 Results of Example 2

Place Table 4 here

4.3 Example 3: An Electron Accelerator

Fig. 4 shows an electron accelerator that accelerates electrons. The inputs of this example are random variable L and random field $V(w, h, t)$. Calling the limit-state function can return a complete time-and space-dependent responses by sampling the random field $V(w, h, t)$. This problem belongs to Case 2 with an input random field, and it therefore requires single-loop MPP search. The device is placed horizontally. Electrons are emitted from the electrode and then enter the electric field E in the accelerator, and finally fly out. The initial velocity of the electrons is a non-stationary Gaussian random field $V_0(w, h, t)$, whose mean is $\mu_{V_0} = 10^5 e^{-0.001(w^2+h^2+(t-6)^2)}$ m/s and standard deviation is $\sigma_{V_0} = 10000$ m/s. The spatial variable $w \in [-0.05, 0.05]$ m is the width of the electrode, and $h \in [-0.05, 0.05]$ m is the height of the electrode. The temporal variable is $t \in [0,10]$ s. The autocorrelation coefficient function of the Gaussian field is given by

$$\rho_{V_0}(w_1, h_1, t_1; w_2, h_2, t_2) = \exp\left[-\left(\frac{w_1 - w_2}{5}\right)^2 - \left(\frac{h_1 - h_2}{5}\right)^2 - \left(\frac{t_1 - t_2}{10}\right)^2\right] \tag{60}$$

The length of the accelerator L is normally distributed with $N(1, 0.01^2)$ m. The electric field $E(w, h)$ is a two-dimensional stationary Gaussian random field, whose mean μ_E and standard deviation σ_E are 10 N/C and 1 N/C, respectively. Its autocorrelation coefficient function is given by

----
Place Fig. 4 here

Fig.4 An electron accelerator

$$\rho_E(w_1, h_1; w_2, h_2) = \exp\left[-\left(\frac{w_1 - w_2}{5}\right)^2 - \left(\frac{h_1 - h_2}{5}\right)^2\right] \tag{61}$$

If the acceleration time and the interaction among the electrons are negligible, the velocity $V(w, h, t)$ of the electrons after acceleration is

$$V(w, h, t) = \sqrt{\frac{2qE(w, h)L}{m} + V_0^2(w, h, t)} \tag{62}$$

where $q = 1.6 \times 10^{-19}$C and $m = 9.109 \times 10^{-31}$kg are the electric quantity and mass of an electron, respectively. The target velocity is $V_t = 1.4519 \times 10^6$ m/s. The domain Ω of $\mathbf{z} = [w, h, t]$ is $\{[-0.05, 0.05] \times [-0.05, 0.05] \times [0,10]\}$. The limit-state function is defined by

$$g\big(\mathbf{X}, V(w, h, t)\big) = V(w, h, t) - V_t \tag{63}$$

in which a failure occurs if the velocity after acceleration is smaller than the target velocity.

The EOLE method is used to generate the series expansion of the nonstationary Gaussian field $V_0(w,h,t)$. w,h, and t are evenly discretized into 10 points, so there are a total of 1000 discretization points. The 1000×1000 autocorrelation coefficient matrix $\mathbf{\Sigma}_{V_0}$ of the random field is obtained. The three most significant eigenvalues of $\mathbf{\Sigma}_{V_0}$ are 841, 146, and 12, and therefore $V_0(w,h,t)$ can be expanded with three standard independent normal variables $\xi_k, k = 1,2,3$. Similarly, we use EOLE to generate the series expansion of $E(w,h)$ and keep only the first two orders. With 1000 discretization expansions points of $V(w,h,t)$, the minimal value of $g_{\min}(w,h,t)$ can be found. Then the standard FORM method is employed to find the worst-case MPP $\mathbf{u}_{\text{MPP}} = (-2.2726, -0.0164, -0.0038, 0.0014, -2.2726, -0.0050)$ and the reliability index $\beta = 3.2140$. To check if the worst-case $\mathbf{u}_{\text{MPP}}$ is the global solution of Eq. (46) and detect if there are multiple MPPs, we use the first strategy discussed in Sec. 3.1.4. We repeat the standard MPP search using different starting point (0, 0, 0, 0, 0, 0), (1.5, 1.5, 1.5, 1.5, 1.5, 1.5), (-2, -2, -2, -2, -2, -2), and (3, 3, 3, -3, -3, -3). The final solutions from different starting points converge to the same solution. Thus, it is likely that the worst-case MPP $\mathbf{u}_{\text{MPP}}$ is the global MPP of Eq. (46) and that there is only one MPP. Then FORM/ENV produces $p_f = 6.5558 \times 10^{-4}$ with only 28 function calls which leading tremendous efficiency improvement instead of using sequential loops to find the worst-case MPP. SOSPA/ENV produces $p_f = 7.8862 \times 10^{-4}$ with 87 function calls. MCS uses 10^7 samples of all random variables at each of the 1000 discretization points of the temporal/spatial variables. The results are provided in Table 5. By using the sing-loop MPP search method, the function calls of both FORM/ENV and SOSPA/ENV methods are reduced tremendously. SOSPA/ENV is more accurate than FORM/ENV but less efficient.

Table 5 Results of Example 3

Place Table 5 here

Conclusions

In this work, the envelope method for time-dependent reliability is extended to time- and space-dependent reliability analysis for limit-state functions with input of random variables, random fields, and temporal and spatial parameters. The envelope function is obtained with respect to temporal/spatial variables. Then the time- and space-dependent problem is converted into a time- and space-independent counterpart, and the second order saddlepoint approximation method is used to estimate the reliability. Equations of the second derivatives of the envelope function are derived for the second order approximation. The major computational cost is the MPP search and second derivative calculations. In this case, efficient global optimization is used for the MPP search, and other global optimization methods can also be used. The first and second derivatives are evaluated by the finite difference method. The results show that the proposed method is much more accurate than the first-order approximation method since the envelope function is in general nonlinear. The new method, however, is less efficient than the first-order approximation method because it requires second derivatives of the envelope function.

The new method shares the same drawbacks of the MPP-based reliability methods. If only one MPP is found but multiple MPPs exist or if the global MPP is not found, the accuracy of the reliability prediction will be low. If the MPP occurs on the boundary of the time and space domain, the derivatives of the envelope function may not exist, and the proposed method may not work. How to address these problems needs further investigations.

Acknowledgments

We would like to acknowledge the support from the National Science Foundation under Grants No. 1923799.

Reference

[1] Binder, K., Heermann, D., Roelofs, L., Mallinckrodt, A. J., and McKay, S. J. C. i. P., 1993, "Monte Carlo simulation in statistical physics," 7(2), pp. 156-157.
[2] Zhang, X., Lu, Z., Cheng, K., and Wang, Y., 2020, "A novel reliability sensitivity analysis method based on directional sampling and Monte Carlo simulation," Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, p. 1748006X19899504.
[3] Au, S.-K., and Beck, J. L., 2001, "Estimation of small failure probabilities in high dimensions by subset simulation," Probabilistic engineering mechanics, 16(4), pp. 263-277.
[4] Kanjilal, O., Papaioannou, I. and Straub, D., 2021. "Cross entropy-based importance sampling for first-passage probability estimation of randomly excited linear structures with parameter uncertainty, " Structural Safety, 91, p.102090.
[5] Hohenbichler, M., & Rackwitz, R., 1981, "Non-normal dependent vectors in structural safety. Journal of the Engineering Mechanics Division," 107(6), 1227-1238.
[6] Du, X., 2008, "Unified uncertainty analysis by the first order reliability method," Journal of mechanical design, 130(9).
[7] Hohenbichler, M., and Rackwitz, R., 1982, "First-order concepts in system reliability," Structural Safety, 1(3), pp. 177-188.
[8] Du, X., and Zhang, J., 2010, "A Second-Order Reliability Method With First-Order Efficiency," Journal of Mechanical Design, 132(10).
[9] Lim, J., Lee, B., and Lee, I., 2014, "Second-order reliability method-based inverse reliability analysis using Hessian update for accurate and efficient reliability-based design optimization," International Journal for Numerical Methods in Engineering, 100(10), pp. 773-792.
[10] Daniels, H. E., 1954, "Saddlepoint Approximations in Statistics," The Annals of Mathematical Statistics, 25(4), pp. 631-650.
[11] Du, X., 2008, "Saddlepoint Approximation for Sequential Optimization and Reliability Analysis," Journal of Mechanical Design, 130(1), pp. 842-849.
[12] Wu, H., and Du, X., 2020, "System Reliability Analysis With Second-Order Saddlepoint Approximation," ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg, 6(4).
[13] Wu, H., Zhu, Z., and Du, X., 2020, "System Reliability Analysis With Autocorrelated Kriging Predictions," Journal of Mechanical Design, 142(10).
[14] Echard, B., Gayton, N., and Lemaire, M., 2011, "AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation," Structural Safety, 33(2), pp. 145-154.
[15] Jeong, S., Murayama, M., and Yamamoto, K., 2005, "Efficient optimization design method using kriging model," Journal of aircraft, 42(2), pp. 413-420.
[16] Zhou, C., Xiao, N.-C., Zuo, M. J., and Huang, X., 2020, "AK-PDF: An active learning method combining kriging and probability density function for efficient reliability analysis," Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, 234(3), pp. 536-549.
[17] Basudhar, A., Missoum, S., and Sanchez, A. H., 2008, "Limit state function identification using support vector machines for discontinuous responses and disjoint failure domains," Probabilistic Engineering Mechanics, 23(1), pp. 1-11.
[18] Moustapha, M., Bourinet, J.-M., Guillaume, B., and Sudret, B., 2018, "Comparative Study of Kriging and Support Vector Regression for Structural Engineering Applications," ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering, 4(2), p. 04018005.

[19] Xiao, N.-C., Duan, L., and Tang, Z., 2017, "Surrogate-model-based reliability method for structural systems with dependent truncated random variables," Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, 231(3), pp. 265-274.
[20] Andrieu-Renaud, C., Sudret, B., and Lemaire, M., 2004, "The PHI2 method: a way to compute time-variant reliability," Reliability Engineering & System Safety, 84(1), pp. 75-86.
[21] Hu, Z., and Du, X., 2013, "Time-dependent reliability analysis with joint upcrossing rates," Structural and Multidisciplinary Optimization, 48(5), pp. 893-907.
[22] Fujimura, K. and Der Kiureghian, A., 2007, "Tail-equivalent linearization method for nonlinear random vibration. Probabilistic Engineering Mechanics, " 22(1), pp.63-76.
[23] Hu, Z., and Du, X., "Efficient Global Optimization Reliability Analysis (EGORA) for Time-Dependent Limit-State Functions," Proc. ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, pp. V02BT03A044-V002BT003A044.
[24] Zhang, D., Han, X., Jiang, C., Liu, J., and Li, Q., 2017, "Time-dependent reliability analysis through response surface method," Journal of Mechanical Design, 139(4).
[25] Hu, Z., and Du, X., 2013, "A sampling approach to extreme value distribution for time-dependent reliability analysis," Journal of Mechanical Design, 135(7), p. 071003.
[26] Shi, Y., Lu, Z., and He, R., 2020, "Advanced time-dependent reliability analysis based on adaptive sampling region with Kriging model," Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, p. 1748006X20901981.
[27] Du, X., 2014, "Time-dependent mechanism reliability analysis with envelope functions and first-order approximation," Journal of Mechanical Design, 136(8), p. 081010.
[28] Wang, Z., and Wang, P., 2012, "A Nested Extreme Response Surface Approach for Time-Dependent Reliability-Based Design Optimization," Journal of Mechanical Design, 134(12), pp. 121007-121014.
[29] Singh, A., Mourelatos, Z. P., and Li, J., 2010, "Design for lifecycle cost using time-dependent reliability," Journal of Mechanical Design, 132(9).
[30] Li, J., Chen, J.-b., and Fan, W.-l., 2007, "The equivalent extreme-value event and evaluation of the structural system reliability," Structural safety, 29(2), pp. 112-131.
[31] Hu, Z., and Mahadevan, S., 2017, "A surrogate modeling approach for reliability analysis of a multidisciplinary system with spatio-temporal output," Structural and Multidisciplinary Optimization, 56(3), pp. 553-569.
[32] Shi, Y., Lu, Z., Zhang, K., and Wei, Y., 2017, "Reliability analysis for structures with multiple temporal and spatial parameters based on the effective first-crossing point," Journal of Mechanical Design, 139(12), pp. 121403-121403.
[33] Shi, Y., and Lu, Z., 2019, "Dynamic reliability analysis for structure with temporal and spatial multi-parameter," Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, 233(6), pp. 1002-1013.
[34] Wei, X., and Du, X., 2019, "Uncertainty Analysis for Time-and Space-Dependent Responses With Random Variables," Journal of Mechanical Design, 141(2), p. 021402.
[35] Wei, X., and Du, X., 2019, "Robustness Metric for Robust Design Optimization Under Time- and Space-Dependent Uncertainty Through Metamodeling," Journal of Mechanical Design, 142(3).
[36] Der Kiureghian, A. and Zhang, Y., 1999, "Space-variant finite element reliability analysis. Computer methods in applied mechanics and engineering, " 168(1-4), pp.173-183.
[37] Wu, H., Hu, Z. and Du, X., 2021. Time-Dependent System Reliability Analysis With Second-Order Reliability Method. Journal of Mechanical Design, 143(3), p.031101.
[38] Haldar, A. and Mahadevan, S., 1995, "First-order and second-order reliability methods, " Probabilistic structural mechanics handbook. Springer, Boston, MA, pp. 27-52.
[39] Jones, D.R., Schonlau, M. and Welch, W.J., 1998. Efficient global optimization of expensive black-box functions. Journal of Global optimization, 13(4), pp.455-492.
[40] Li, J., and Mourelatos, Z. P.,2009. "Time-Dependent Reliability Estimation for Dynamic Problems Using a Niching Genetic Algorithm." ASME. J. Mech. Des. July 2009; 131(7): 071009.

[41] Venter, G. and Sobieszczanski-Sobieski, J., 2003. "Particle swarm optimization. AIAA journal, " 41(8), pp.1583-1589.
[42] Der Kiureghian, A. and Dakessian, T., 1998. “Multiple design points in first and second-order reliability.” Structural Safety, 20(1), pp.37-49.
[43] Li, C.-C., and Der Kiureghian, A., 1993, "Optimal discretization of random fields," Journal of engineering mechanics, 119(6), pp. 1136-1154.

List of Table Captions

List of Figure Captions

Table 1 Iteration history of searching for the worst-case MPP

Iterations	u^*	$\tilde{t}$	$\tilde{s}$
1	(-7.4573, -2.0392)	0.9157	1.2272
2	(-3.9028, -1.4544)	1.2886	1.0077
3	(-3.1172, -2.0203)	1.4821	0.8722
4	(-2.7126, -2.3219)	1.5695	0.8059
5	(-2.5333, -2.4574)	1.7458	0.9219
6	(-2.3025, -2.5225)	1.7859	0.8956
7	(-2.2254, -2.5784)	1.8030	0.8843
8	(-2.1928, -2.6021)	1.8101	0.8795
9	(-2.1928, -2.6120)	1.8131	0.8776
10	(-2.1735, -2.6161)	1.8143	0.8767
11	(-2.1712, -2.6178)	1.8148	0.8764
12	(-2.1702, -2.6185)	1.8150	0.8763

Table 2 Results of Example 1

Method	Probability of failure	Error	Number of function calls
MCS	5.080×10^{-4}	-	4×10^{9}
FORM/ENV	3.3575×10^{-4}	33.9%	314
SOSPA/ENV	4.9022×10^{-4}	3.5%	333

Table 3 Random variables of Example 2

Variable (Unit)	Mean	Standard deviation	Distribution
$A_1(\mathrm{mm}^2)$	60	0.6	Normal
$A_2(\mathrm{mm}^2)$	60	0.6	Normal
$A_3(\mathrm{mm}^2)$	60	0.6	Normal
$E_1(\mathrm{GPa})$	200	20	Lognormal
$E_2(\mathrm{GPa})$	200	20	Lognormal
$E_3(\mathrm{GPa})$	200	20	Lognormal
$P_0(\mathrm{KN})$	40	6	Normal
$L_{\mathrm{AB}}(\mathrm{mm})$	200	2	Normal
$L_{\mathrm{AD}}(\mathrm{mm})$	231	2.31	Normal
$L_{\mathrm{AC}}(\mathrm{mm})$	283	2.83	Normal
$T(°\mathrm{C})$	35	7	Lognormal

Table 4 Results of Example 2

Method	Probability of failure	Error (%)	Number of function calls
MCS	3.0270×10^{-4}	-	10^9
FORM/ENV	2.7654×10^{-4}	8.64%	189
SOSPA/ENV	2.9958×10^{-4}	1.03%	305

Table 5 Results of Example 3

Method	Probability of failure	Error (%)	Number of function calls
MCS	$8,1360 \times 10^{-4}$	-	10^7
FORM/ENV	6.5558×10^{-4}	19.4%	28

SOSPA/ENV	7.8862×10^{-4}	3.1%	87