Submitted to Structural Safety V.3

Active Learning with Generalized Sliced Inverse Regression for High-Dimensional Reliability Analysis

Jianhua Yin[1,2], Xiaoping Du[1]

Department of Mechanical and Energy Engineering, Indiana University - Purdue University Indianapolis,
723 W. Michigan Street, Indianapolis, IN 46202-5195
[2]School of Mechanical Engineering, Purdue University, West Lafayette, IN, 47907, United States

Abstract

It is computationally expensive to predict reliability using physical models at the design stage if many random input variables exist. This work introduces a dimension reduction technique based on generalized sliced inverse regression (GSIR) to mitigate the curse of dimensionality. The proposed high dimensional reliability method enables active learning to integrate GSIR, Gaussian Process (GP) modeling, and Importance Sampling (IS), resulting in an accurate reliability prediction at a reduced computational cost. The new method consists of three core steps, 1) identification of the importance sampling region, 2) dimension reduction by GSIR to produce a sufficient predictor, and 3) construction of a GP model for the true response with respect to the sufficient predictor in the reduced-dimension space. High accuracy and efficiency are achieved with active learning that is iteratively executed with the above three steps by adding new training points one by one in the region with a high chance of failure.

Keywords: dimension reduction, reliability analysis, Gaussian process, active learning

1. Introduction

Reliability is measured by the probability that a system performs its intended function without failure. Reliability analysis is a core task in engineering design, where the probability of failure is predicted for a given design. If the probability of failure exceeds the design requirement, the design is updated, and the reliability analysis is performed again. This process repeats until the reliability target is achieved. The probability of failure can be predicted by physical models derived from physical principles or data-driven models. It is given by

$$p_f = \Pr\{Y = g(\mathbf{X}) < 0\} \tag{1}$$

where $\mathbf{X} = \left(X_1, X_2, \dots, X_p\right)^{\mathrm{T}}$ is a vector of input random variables, $g(\mathbf{X})$ is a performance function that could be a physical model derived from physical principles or a regression model based on data, and Y is a response that indicates the state of the product. Conventionally, when $Y < 0$, a failure occurs. In this study, we assume the input random variables $\mathbf{X}$ are independent. If they are not independent, they could be transformed into independent ones [1].

There are three types of reliability analysis methods: 1) approximation methods [2-4], 2) meta-modeling methods [5-12], and 3) sampling methods [13-17]. Commonly used approximation methods include the first order reliability method (FORM) [2] and the second order reliability method (SORM) [3]. They approximate the performance function by making use of Taylor expansion. Meta-modeling methods construct a surrogate model to replace the performance function using regression or interpolation methods. Design of Experiments (DoE) [18] is a commonly used tool to generate optimal training points to build the surrogate model. The efficiency of mate-modeling based reliability analysis methods can be improved by active learning [19]. Sampling methods, such as Monte Carlo Simulation (MCS) [20], importance sampling (IS)

[21], and subset simulation (SS) [22], are not affected by the dimensionality. However, their computational effort is still very high regardless of the dimension, especially when the probability of failure is low. Although meta-modeling approaches may be more efficient, a dimension reduction is still needed to handle high-dimension problems.

A commonly used dimension reduction approach is the principal component analysis (PCA) [23-25]. PCA reduces the dimension of random variables by exploiting their correlation structure. If the random variables are strongly correlated, PCA can effectively reduce the dimension by linear combinations of the random variables, resulting in the so-called principal components. It does not work well for independent random variables. PCA is an unsupervised method that does not use the information of the response Y. High-dimensional model representation (HDMR) [26-28] is another high-dimensional reliability method, which decomposes $g(\mathbf{X})$ into the sum of several low-dimensional functions. However, when the interaction terms dominate the performance function, the accuracy is poor.

Machine learning and regression methods have recently been used in high dimensional reliability analysis. Several studies [29-34] combine meta-modeling and dimension reduction techniques. Two steps are typically involved. A low dimensional latent subspace is identified by the sliced inverse regression (SIR) [35], which is a linear sufficient dimension reduction (SDR) technique, or other dimension reduction methods [32, 34, 36] using training points generated by DoE. A surrogate model of the performance function is then constructed in the low dimensional latent subspace and is refined by cross validation. Since the training points are pre-defined by DoE in the first step, there is no guarantee that 1) a suitable latent subspace exists, and 2) the accuracy of the surrogate model is satisfactory. An active learning based meta-modeling approach combined with dimension reduction is reported in [33]. It combines AK-MCS [8] with a dimension reduction technique called active subspace (AS) [36] to iteratively select the optimum training points in the original high dimensional space, and good accuracy and efficiency are achieved. SIR and AS, however, are both linear dimension reduction techniques, and they may not work well for problems that need nonlinear dimension reduction.

It is desirable to use nonlinear dimension reduction approaches for high dimensional reliability analysis. Nonlinear dimension reduction techniques can be classified into two groups, supervised nonlinear dimension reduction [37, 38] and unsupervised nonlinear dimension reduction [39, 40]. Similar to PCA, unsupervised nonlinear dimension reduction, such as Kernel PCA [39], autoencoder [31], and diffusion maps [41], do not make use of the information of the model response or labels in the dimension reduction process. For the supervised dimension reduction methods, studies in [42-44] combine the so-called kernel trick [45, 46] with SDR to overcome the limitation of linear SDR, making supervised nonlinear sufficient dimension reduction feasible. The approaches include the kernel canonical correlation analysis (KCCA) [44], kernel SIR (KSIR) [43], and generalized SIR (GSIR) [42]. GSIR not only relaxes the stringent conditions required by linear SDR where the reduced subspace is the linear combination of the original random variables, but also relieves the assumption of KSIR that the subspace is the linear combination of a set of nonlinear functions. Given the advantages of GSIR, it is worth investigating its use in high dimensional reliability analysis.

This work develops a high dimensional reliability method that combines GSIR with GP, IS, and active learning. The proposed method inherits the advantage of GSIR, which is more general and robust no matter if the linear or nonlinear dimension reduction is required. The computational cost of constructing the surrogate model used for reliability analysis is decreased drastically due to the dimension reduction by GSIR. Since the use of IS requires less computational effort than MCS based methods, the proposed method can also handle small probabilities of failure.

The rest of the paper is organized as follows. Section 2 reviews the related methodologies used in this paper. Section 3 presents the details of the proposed method followed by four examples in Section 4. Concluding remarks are provided in Section 5.

2. Literature Review

2.1 Generalized Sliced Inverse Regression (GSIR)

GSIR is an approach belonging to sufficient dimension reduction (SDR). Given input variables $\mathbf{X} \in \mathbb{R}^{n\times p}$ and the response $Y \in \mathbb{R}^{n\times 1}$ that depends on $\mathbf{X}$, SDR seeks a function $R(\mathbf{X})$ to map $\mathbf{X}$ to a subspace so that the distribution of Y given $\mathbf{X}$ is the same as that given $R(\mathbf{X})$, where n is the number of training points, and p is the dimension of $\mathbf{X}$. For linear SDR, $R(\mathbf{X})$ contains one or more linear combinations of $\mathbf{X}$, and the task is to find a matrix $\beta \in \mathbb{R}^{p\times d}$ such that

$$Y \perp\!\!\!\perp \mathbf{X}|\beta^{\mathrm{T}}\mathbf{X} \tag{2}$$

where d is the dimension of the subspace, and $d < p$; $\perp\!\!\!\perp$ denotes independence, meaning that the distribution of Y is conditionally independent of $\mathbf{X}$ given $\beta^{\mathrm{T}}\mathbf{X}$. Different from SDR, nonlinear SDR searches for a set of nonlinear functions $f_1(\mathbf{X}), \ldots, f_d(\mathbf{X})$ such that

$$Y \perp\!\!\!\perp \mathbf{X}|f_1(\mathbf{X}), \ldots, f_d(\mathbf{X}) \tag{3}$$

Since $d < p$, the dimension is reduced from p to d.

The nonlinear functions may be hard to define in practice. But the use of the kernel trick could allow dimension reduction to proceed without defining the nonlinear function. This is done by projecting $\mathbf{X}$ and Y to the kernel space.

GSIR [42] is a nonlinear dimension reduction method that stems from the nonlinear SDR theory. The conditional expectation of $\mathbf{X}$ given Y is denoted by

$$E_{\mathbf{X}|Y} = \Sigma_{YY}^{-1/2} R_{Y\mathbf{X}} \Sigma_{\mathbf{XX}}^{1/2} \quad (4)$$

where $R_{Y\mathbf{X}}$ is called the correlation operator denoted by

$$R_{Y\mathbf{X}} = \Sigma_{YY}^{-1/2} \Sigma_{Y\mathbf{X}} \Sigma_{\mathbf{XX}}^{-1/2} \quad (5)$$

and $\sum$ is the covariance operator.

If a data set of training points are available with $(\mathbf{x}_1, \dots, \mathbf{x}_n)$ and $(y_1, \dots, y_n)$, then

$$\Sigma_{\mathbf{XX}} = \Sigma_{Y\mathbf{X}} = \frac{1}{n} G_{\mathbf{X}} = \frac{1}{n} Q K_{\mathbf{X}} Q = \frac{1}{n} Q \begin{bmatrix} K(\mathbf{x}_1, \mathbf{x}_1) & \cdots & K(\mathbf{x}_1, \mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ K(\mathbf{x}_n, \mathbf{x}_1) & \cdots & K(\mathbf{x}_n, \mathbf{x}_n) \end{bmatrix} Q \quad (6)$$

$$\Sigma_{YY} = \frac{1}{n} G_Y = \frac{1}{n} Q K_Y Q = \frac{1}{n} Q \begin{bmatrix} K(y_1, y_1) & \cdots & K(y_1, y_n) \\ \vdots & \ddots & \vdots \\ K(y_n, y_1) & \cdots & K(y_n, y_n) \end{bmatrix} Q \quad (7)$$

where $Q = I_n - 1_n 1_n^T / n$; $G_{\mathbf{X}}$ and G_Y are the centered versions of the kernel matrixes $K_{\mathbf{X}}$ and K_Y; I_n is an $n \times n$ identity matrix; and 1_n is an $n \times 1$ vector with all elements being 1. The kernel function used in this paper is the anisotropic squared-exponential function and is defined by

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\theta_{\mathbf{X}} (\mathbf{x}_i - \mathbf{x}_j)^2\right), i.j = 1, \dots, n \quad (8)$$

$K(y_i, y_j)$ is obtained by replacing $\theta_{\mathbf{X}}(\mathbf{x}_i - \mathbf{x}_j)^2$ with $\theta_Y (y_i - y_j)^2$ in Eq. (8). $\theta_{\mathbf{X}}$ is computed by

$$\frac{1}{\theta_{\mathbf{X}}} = \binom{n}{2}^{-1} \sum_{i<j} |\mathbf{x}_i - \mathbf{x}_j|^2 \quad (9)$$

θ_Y is obtained by replacing $\left|\mathbf{x}_i - \mathbf{x}_j\right|$ with $\left|y_i - y_j\right|$ in Eq. (9). Similar to other kernel-based methods, such as GP and SVM, we can choose a kernel from several well established options [47]. Anisotropic squared-exponential function or squared-exponential function is a good starting point when we are short of knowledge about a problem.

Substituting $\sum_{YY}$, $\sum_{\mathbf{XX}}$, and $\sum_{Y\mathbf{X}}$ into Eqs. (12) and (13) yields the correlation operator and conditional expectation.

$$R_{Y\mathbf{X}} = G_Y^{+1/2} G_{\mathbf{X}} G_{\mathbf{X}}^{+1/2} \tag{10}$$

$$E_{\mathbf{X}|Y} = G_Y^{+} G_{\mathbf{X}} G_{\mathbf{X}}^{+1/2} G_{\mathbf{X}}^{1/2} \tag{11}$$

where $+$ means the Moore–Penrose inverse [48] of a matrix in a general sense. In the numerical computation, the Moore-Penrose inverses $G_{\mathbf{X}}^{+}$ and G_Y^{+} are replaced by the ridge-regression-type regularized inverses $(G_{\mathbf{X}} + \epsilon_{\mathbf{X}} I_n)^{-1}$ and $(G_Y + \epsilon_Y I_n)^{-1}$, respectively, where $\epsilon_{\mathbf{X}}$ and ϵ_Y are the penalty terms. The first d eigenvectors $\xi_1, \xi_2, \dots, \xi_d$ used to form the sufficient predictors are obtained by performing the eigen-analysis of the matrix in Eq. (12).

$$G_X^{+}\left[E_{X|Y}\right]^T G_Y^{2}\left[E_{X|Y}\right] G_X^{+} =$$

$$(G_X + \epsilon_X I_n)^{-3/2} G_X^{3/2} (G_Y + \epsilon_Y I_n)^{-1} G_Y^{2} (G_Y + \epsilon_Y I_n)^{-1} G_X^{3/2} (G_X + \epsilon_X I_n)^{-3/2} \tag{12}$$

After the dimension reduction (training) is complete, predictions of new input variables can be made. Given a new set of input variables $(\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_m)$, denote their responses by $\hat{Y} = (\hat{y}_1, \dots, \hat{y}_m)$, and their predictors can be obtained as follow.

The kernel matrix of the training points and new points are obtained by

$$K_{\mathbf{X}\hat{\mathbf{X}}} = \begin{bmatrix} K(\mathbf{x}_1, \hat{\mathbf{x}}_1) & \cdots & K(\mathbf{x}_1, \hat{\mathbf{x}}_m) \\ \vdots & \ddots & \vdots \\ K(\mathbf{x}_n, \hat{\mathbf{x}}_1) & \cdots & K(\mathbf{x}_n, \hat{\mathbf{x}}_m) \end{bmatrix} \tag{13}$$

Then, the sufficient predictor $\hat{f}_i$ is given by

$$\hat{f}_i = \xi_i^T Q K_{\mathbf{X}\hat{\mathbf{X}}}, i = 1, \dots, d \tag{14}$$

The corresponding eigenvalues of the eigenvectors $(\xi_1, \xi_2, \dots, \xi_d)$ are sorted in a descending order, as is the importance of the corresponding eigenvectors. The first sufficient predictor $\hat{f}_1$ is therefore the most important predictor. As indicated in [42], the relationship between $\hat{f}_1$ and the response $\hat{Y}$ is usually monotonic, and Spearman's correlation is used to measure the monotonic relationship. The monotonicity is an advantage of GSIR over many other dimension reduction methods [42] since the monotonic relationship can clearly classify a training point into either the safe region or the failure region for the reliability prediction. The advantage is also demonstrated in this study as will be shown in Sec. 4. It is therefore possible to reduce the original dimension p to 1 because $\hat{f}_1$ is in a one-dimensional space.

The GSIR algorithm is summarized as follows.

Algorithm 1 Generalized sliced inverse regression [42]

```
1. Collect training points $(\mathbf{x}_1, \dots, \mathbf{x}_n)$ and $(y_1, \dots, y_n)$.
2. Select the ridge parameters $\epsilon_X$ and $\epsilon_Y$ and compute $\theta_X$, $\theta_Y$ by Eq. (9).
3. Solve for the first $d$ eigenvectors $\xi_1, \xi_2, \dots, \xi_d$ of the matrix in Eq. (12).
4. Form the sufficient predictors by Eq. (14).
```

2.2 Importance Sampling (IS)

Importance sampling (IS) is a sampling method that approximates a mathematical expectation with respect to a target distribution by a weighted average of random draws from another distribution (called an importance distribution). For high reliability problem, if samples are drawn from the joint (target) distribution of the original random variables, the chance of getting samples in the failure region is low. Such a chance will be much higher if the samples are drawn from a suitable importance distribution, thereby increasing the computational efficiency. Therefore, it is desirable to use an importance distribution that is centered in the region where the failure is most likely. In risk analysis literature, the Most Probable Point (MPP) [19] is usually used as the center of the importance distribution. The MPP belongs to the limit state surface, and this point has the highest probability density in the standard normal space (U-space).

To solve for MPP, we first transformed $\mathbf{X}$ to $\mathbf{U}$, whose components are independent standard normal variables [1]. The transformation is denoted by $\mathbf{X} = T(\mathbf{U})$. The performance function then becomes $Y = g(\boldsymbol{T}(\mathbf{U})) = G(\mathbf{U})$. The next step is to obtain the IS center. There is no need to search for the true MPP in practice. We can use the point from the first iteration of the MPP search as the IS center to reduce the computation time. Although it may not be close to the true MPP, the one-iteration MPP allows the IS samples centered around it to cover a sufficiently large area of failure region if a proper sample size is used. The one-iteration MPP is obtained by

$$\mathbf{u}^* = -\frac{G(\mathbf{u}_0)\nabla G(\mathbf{u}_0)}{\|\nabla G(\mathbf{u}_0)\|^2} \tag{15}$$

where $\mathbf{u}_0 = (0, \dots, 0)^T$ is the origin of the U-space. For the highly nonlinear problems, more iterations of the MPP search may be needed to approach the failure boundary.

With the approximate MPP $\mathbf{u}^*$, we shift the center of the probability density to $\mathbf{u}^*$, resulting in importance probability density $\varphi_{\mathbf{U}}(\cdot)$, represented by the new distribution $\widehat{U}_i \sim N(u_i^*, 1^2)$, where u_i^* is the i-th component of $\mathbf{u}^*$. In this paper, we use the same standard deviations of standard normal variables. The probability of failure is estimated with density $\phi_{\mathbf{U}}(\cdot)$ of $\mathbf{U}$ and the importance density $\varphi_{\mathbf{U}}(\mathbf{u})$.

$$p_f = \int I_F(\mathbf{u}) \frac{\phi_{\mathbf{U}}(\mathbf{u})}{\varphi_{\mathbf{U}}(\mathbf{u})} \varphi_{\mathbf{U}}(\mathbf{u}) d\mathbf{u} \tag{16}$$

where $I_F(\cdot)$ is an indicator function and is defined as

$$I_F(\mathbf{u}) = \begin{cases} 0, G(\mathbf{U}) > 0 \\ 1, G(\mathbf{U}) \leq 0 \end{cases} \tag{17}$$

With the samples drawn from the importance density $\phi_{\mathbf{U}}(\mathbf{u})$, p_f in Eq. (16) is estimated by

$$p_f \approx \hat{p}_f = \frac{1}{N_{IS}} \sum_{i=1}^{N_{IS}} I_F(\hat{\mathbf{u}}_i) \frac{\phi_{\mathbf{U}}(\hat{\mathbf{u}}_i)}{\varphi_{\mathbf{U}}(\hat{\mathbf{u}}_i)} \tag{18}$$

where $\hat{\mathbf{u}}_i$, $i = 1, \ldots, N_{IS}$, are the samples generated from $\phi_{\mathbf{U}}(\mathbf{u})$. The variance of the probability of failure is estimated by

$$\mathrm{Var}(\hat{p}_f) = \frac{1}{N_{IS}} \left(\frac{1}{N_{IS}} \sum_{i=1}^{N_{IS}} \left(I_F(\hat{\mathbf{u}}_i) \left(\frac{\phi_{\mathbf{U}}(\hat{\mathbf{u}}_i)}{\varphi_{\mathbf{U}}(\hat{\mathbf{u}}_i)} \right)^2 \right) - \hat{p}_f^2 \right) \tag{19}$$

If a proper importance distribution is chosen, $\mathrm{Var}(\hat{p}_f)$ is less than the variance of MCS, therefore increasing the computational efficiency.

The coefficient of variation δ_{IS} of $\hat{p}_f$ is calculated by

$$\delta_{IS} = \frac{\sqrt{\mathrm{Var}(\hat{p}_f)}}{\hat{p}_f} \tag{20}$$

2.3 Gaussian Process (GP) Modeling

GP modeling [49] views a function $G(\mathbf{U})$ as a realization of a Gaussian process. Given a set of training points, a GP model is obtained by

$$\hat{G}(\mathbf{u}) = \mathbf{f}(\mathbf{u})^{\mathrm{T}}\boldsymbol{\beta} + Z(\mathbf{u}) \tag{21}$$

where $\mathbf{f}(\mathbf{u})^{\mathrm{T}}\boldsymbol{\beta}$ is a deterministic term, providing the trend and the mean response; $\mathbf{f}(\mathbf{u}) = \left(f_1(\mathbf{u}), f_2(\mathbf{u}), \dots, f_p(\mathbf{u})\right)^T$ is a vector of regression functions; $\boldsymbol{\beta} = \left(\beta_1, \beta_2, \dots, \beta_p\right)^T$ is a vector of regression coefficients; $Z(\cdot)$ is a stationary Gaussian process with zero mean and covariance. The covariance is denoted by

$$\mathrm{Cov}\left(Z(\mathbf{u}_i), Z(\mathbf{u}_j)\right) = \sigma_Z^2 R(\mathbf{u}_i, \mathbf{u}_j) \tag{22}$$

where σ_Z^2 is the process variance, $R(\cdot,\ \cdot)$ is the correlation function, specifically the squared-exponential kernel used in this work. GP can also provide the variance of the prediction as the GP predictor $\hat{G}(\mathbf{u})$ follows a normal distribution, denoted by

$$\hat{G}(\mathbf{u}) \sim N\left(\mu_G(\mathbf{u}), \sigma_G^2(\mathbf{u})\right) \tag{23}$$

where $\mu_G(\cdot)$ and $\sigma_G^2(\cdot)$ represent the mean GP prediction and GP variance, respectively.

2.4 AK-IS

AK-IS [19] is an active learning method combining Kriging (GP modeling) with IS for reliability analysis. AK-IS at first uses the MPP-centered importance distribution to generate samples, called the IS population. It then constructs the GP model by the point used for solving the MPP and refines the model by adding training points selected from the IS population. A new training point is selected by a learning function and is added to the set of training points, which allows for an update of the GP model. The process stops once the desired accuracy is achieved. The size of the IS population will be increased if a target coefficient of variation is unsatisfied. The learning function is defined by

$$\mathbb{U}(\hat{\mathbf{u}}) = \frac{|\mu_G(\hat{\mathbf{u}})|}{\sigma_G^2(\hat{\mathbf{u}})} \tag{24}$$

where $\hat{\mathbf{u}}$ is a point in the IS population, and $\mu_G(\cdot)$ and $\sigma_G^2(\cdot)$ are given in Eq. (23). A lower $\mathbb{U}(\hat{\mathbf{u}})$ means a higher probability that the point is misclassified. Then the point with the minimum $\mathbb{U}(\hat{\mathbf{u}})$ in the IS population is selected as the new training point. The learning process stops when $\min \mathbb{U}(\hat{\mathbf{u}}) \geq 2$.

3. Methodology

The purpose of this study is to explore the use of GSIR in high dimensional reliability analysis to reduce computational efforts. The central strategy is reducing the dimension of input variables by GSIR so that a GP model can be constructed in a low dimensional subspace. The following three steps are involved and are illustrated in Fig. 1, where TP stands for training point.

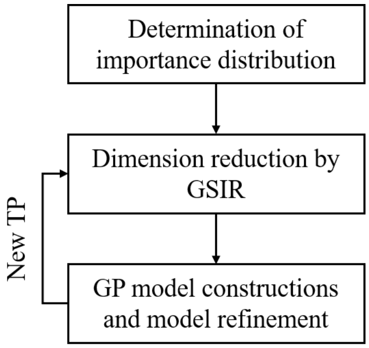

Determination of importance distribution
Dimension reduction by GSIR
GP model constructions and model refinement
New TP

Fig. 1 Schematic of the proposed method

1) Determine the importance distribution: we first obtain the one-iteration MPP which is the importance distribution center. Samples are generated from the importance distribution as described in Section 2.2 to form the importance population.
2) Initialization and dimension reduction by GSIR. Initially, training points $\mathbf{U}$ are generated by Latin Hypercube sampling, centered at the origin of the U-space, and the corresponding responses are obtained by calling the performance function $Y = G(\mathbf{U})$. GSIR then trains sufficient predictors using the set of training points and associated responses. In subsequent iterations, new training points are selected by active learning from the IS population.
3) Surrogate model creation in subspace: the GP model is constructed in the one-dimensional space of the sufficient predictor (Eq. (14)). The input of the model is the first sufficient predictor from step 2), and the output is the prediction of the response Y for the GP model.

Since the sufficient predictor and GP model may not be accurate, steps 2 and 3 are performed iteratively to refine the GP model by selecting new training points from the IS population through an active learning strategy. In each iteration, only one new training point is added. The model update completes once the convergence criterion is met. When the algorithm converges, the probability of failure is obtained by the IS estimation method discussed in Section 2.2. Next, we provide detailed descriptions of the three major steps.

3.1 Importance Distribution

The first step of the proposed method is to generate a sample population that supplies candidate training points during active learning. As discussed in Sec. 2.2, if the sample population covers both safe and failure regions where the probability density is high, the variance of the estimated probability of failure will be reduced, thereby increasing computational efficiency.

We first transform random variables $\mathbf{X}$ in the X-space to $\mathbf{U}$ in the U-space. The performance function then becomes $Y = G(\mathbf{U})$. Then all the derivations will be performed with respect to $\mathbf{U}$. After the transformation, the one-iteration MPP $\mathbf{u}^*$ is obtained by Eq. (15) and serves as the IS center [19]. The computational cost for the one-iteration MPP is $n + 1$ evaluations of the performance function.

As mentioned in Section 2.2, the importance probability density $\varphi_{\mathbf{U}}(\cdot)$ results in new distributions $\widehat{U}_i \sim N(u_i^*, 1^2)$, $i = 1, \dots, n$, where u_i^* is the i-th component of $\mathbf{u}^*$. We then draw samples $\widehat{\mathbf{U}} = \left(\widehat{\mathbf{u}}_1, \dots, \widehat{\mathbf{u}}_{N_{IS}}\right)$ from $\varphi_{\mathbf{U}}(\cdot)$ to establish an IS population denoted by $\mathbb{P}_{IS}$. The IS population can cover both safety and failure regions with balanced samples in both regions. It is recommended that the size of IS population should be sufficiently large (e.g., 10^4), especially for high dimensional problems. If the coefficient of variation in Eq. (20) is large, the population size should be increased accordingly. If the one-iteration MPP is far away from the true MPP, we can also increase the importance sampling size or increase the standard deviations of $\mathbf{U}$ to cover the critical failure region. The added training points during the active learning stage are selected from the IS population.

3.2 Initialization and Dimension Reduction by GSIR

The initial training points are generated by Latin Hypercube sampling and are centered at the origin of the U-space, which is denoted by $\mathbf{U}_t = (\mathbf{u}_{t1}, \dots, \mathbf{u}_{tn}), \mathbf{u}_{t1} \in \mathbb{R}^{p\times 1}$. Then, the corresponding responses $Y_t = (y_{t1}, \dots, y_{tn}), y_{t1} \in \mathbb{R}^{1\times 1}$ are obtained by calling the performance function at $\mathbf{U}_t$. It is recommended that the sample size of initial training points is three to five times of the input dimension. This number of initial training points can help create an accurate initial model and can therefore reduce the number of new training points in the subsequent iterations. It is also possible to use fewer initial training points, and the number of new training points will be likely increase.

Once the training points $(\mathbf{U}_t, Y_t)$ are available, GSIR is used to reduce the dimension of input variables such that the GP model can be constructed in a low dimensional space. We first obtained the kernel matrices $K_{\mathbf{U}_t}$ and K_{Y_t} by

$$K_{\mathbf{U}_t} = \begin{bmatrix} K(\mathbf{u}_{t1}, \mathbf{u}_{t1}) & \cdots & K(\mathbf{u}_{t1}, \mathbf{u}_{tn}) \\ \vdots & \ddots & \vdots \\ K(\mathbf{u}_{tn}, \mathbf{u}_{t1}) & \cdots & K(\mathbf{u}_{tn}, \mathbf{u}_{tn}) \end{bmatrix} \tag{25}$$

$$K_{Y_t} = \begin{bmatrix} K(y_{t1}, y_{t1}) & \cdots & K(y_{t1}, y_{tn}) \\ \vdots & \ddots & \vdots \\ K(y_{tn}, y_{t1}) & \cdots & K(y_{tn}, y_{tn}) \end{bmatrix} \tag{26}$$

where $K(\cdot,\cdot)$ is the kernel function defined in Eq. (8). The centered kernel matrices of $K_{\mathbf{U}_t}$ and K_{Y_t} are obtained by $G_{\mathbf{U}_t} = QK_{\mathbf{U}_t}Q$ and $G_{Y_t} = QK_{Y_t}Q$. Then the correlation operator and conditional expectation are obtained by

$$R_{Y_t\mathbf{U}_t} = G_{Y_t}^{+1/2} G_{\mathbf{U}_t} G_{\mathbf{U}_t}^{+1/2} \tag{27}$$

$$E_{\mathbf{U}_t|Y_t} = G_{Y_t}^{+} G_{\mathbf{U}_t} G_{\mathbf{U}_t}^{+1/2} G_{\mathbf{U}_t}^{1/2} \tag{28}$$

The first d eigenvectors $\xi_1, \xi_2, \dots, \xi_d$, which are used to form sufficient predictors, are calculated by performing eigen-analysis to the following matrix:

$$G_{\mathbf{U}_t}^{+} \left[E_{\mathbf{U}_t|Y_t}\right]^T G_{Y_t}^{2} \left[E_{\mathbf{U}_t|Y_t}\right] G_{\mathbf{U}_t}^{+} =$$

$$\left(G_{\mathbf{U}_t} + \epsilon_{\mathbf{U}_t} I_n\right)^{-3/2} G_{\mathbf{U}_t}^{3/2} \left(G_{Y_t} + \epsilon_{Y_t} I_n\right)^{-1} G_{Y_t}^{2} \left(G_{Y_t} + \epsilon_{Y_t} I_n\right)^{-1} G_{\mathbf{U}_t}^{3/2} \left(G_{\mathbf{U}_t} + \epsilon_{\mathbf{U}_t} I_n\right)^{-3/2} \tag{29}$$

As mentioned in Section 2.1, the first sufficient predictor ($\hat{f}_1$) and the real response Y_t are in a good monotonic relationship. We now denote $\hat{f}_1$ by f_{GSIR}. Then the training points $\mathbf{U}_t$ are mapped to a one-dimensional space through the sufficient predictor

$$f_{GSIR}(\mathbf{U}_t) = \xi_1^T Q K_{\mathbf{U}_t} \tag{30}$$

Predictions of the sufficient predictor can be made for new untried points $\mathbf{U} = (\mathbf{u}_1, \dots, \mathbf{u}_l\,)$ by

$$f_{GSIR}(\mathbf{U}) = \xi_1^T Q K_{\mathbf{U}_t\mathbf{U}} \tag{31}$$

where

$$K_{\mathbf{U}_t\mathbf{U}} = \begin{bmatrix} K(\mathbf{u}_{t1}, \mathbf{u}_1) & \cdots & K(\mathbf{u}_{t1}, \mathbf{u}_l) \\ \vdots & \ddots & \vdots \\ K(\mathbf{u}_{tn}, \mathbf{u}_1) & \cdots & K(\mathbf{u}_{tn}, \mathbf{u}_l) \end{bmatrix} \tag{32}$$

The next step is to construct a functional relationship between the real response Y and the sufficient predictor f_{GSIR}.

3.3 GP Modeling in Subspace

As discussed in Sec. 3.3, the sufficient predictor f_{GSIR} does not provide the prediction of the true response Y, but both have a monotonic relationship. The task now is to transform the sufficient predictor f_{GSIR} to the response Y. Despite the monotonicity feature, the relationship may be nonlinear. Many regression techniques can be used for this task, such as GP, support vector machine (SVM), polynomial chaos expansions (PCEs), and Neural Networks (NNs). In this study, we use GP modeling as an example to illustrate the process. The GP method can not only handle nonlinearity well but also provide the uncertainty estimate of the prediction, enabling an active learning process.

Given a set of training points $(\mathbf{U}_t, Y_t)$, the sufficient predictor $f_{GSIR}(\mathbf{U}_t)$, which is a one-dimensional variable, is obtained by Eq. (30) as discussed in Sec. 3.2. Then we construct a one-dimensional GP model using the low dimensional training points $(f_{GSIR}(\mathbf{U}_t), Y_t)$ denoted by

$$Y = \hat{G}(f_{GSIR}) = \mathbf{f}(f_{GSIR})^{\mathrm{T}}\boldsymbol{\beta} + Z(f_{GSIR}) \tag{33}$$

Since f_{GSIR} is obtained from training points $(\mathbf{U}_t, Y_t)$, it is a function of $\mathbf{U}$. Eq. (33) can be rewritten as

$$Y = \hat{G}\big(f_{GSIR}(\mathbf{U})\big) = \mathbf{f}\big(f_{GSIR}(\mathbf{U})\big)^{\mathrm{T}}\boldsymbol{\beta} + Z\big(f_{GSIR}(\mathbf{U})\big) \tag{34}$$

For an untried point $\mathbf{u}$, $f_{GSIR}(\mathbf{u})$ is obtained by Eq. (31), and the Gaussian predictor $\hat{G}\big(f_{GSIR}(\mathbf{u})\big)$ follows a normal distribution as follow.

$$Y = \hat{G}\big(f_{GSIR}(\mathbf{u})\big) \sim N\left(\mu_G\big(f_{GSIR}(\mathbf{u})\big), \sigma_G^2\big(f_{GSIR}(\mathbf{u})\big)\right) \tag{35}$$

where $\mu_G\big(f_{GSIR}(\mathbf{u})\big)$ is the prediction of the mean value of Y at $\mathbf{u}$, and $\sigma_G\big(f_{GSIR}(\mathbf{u})\big)$ measures the uncertainty in the prediction. The accuracy of the prediction will be gradually improved during the learning process discussed in Sec. 3.4.

3.4 Active Learning

The dimension reduction and regression discussed above are executed iteratively to improve the accuracy of the GP model. The accuracy depends on the size and location of the training points, which can be hard to be determined beforehand. To have the best balance between accuracy and efficiency, we gradually improve the accuracy of the regression model by an active learning strategy that adds new training points one by one selected from the IS population. The GSIR dimension reduction and the GP model are updated and refined until the convergence criterion is met. Next, we discuss how to select a new training point and how to measure the accuracy.

We adopt the $\mathbb{U}$-learning function [8] to select the next training point at each iteration. Since the GP model is created in the space of sufficient predictor $f_{GSIR}(\mathbf{U})$, given the IS population $\widehat{\mathbf{U}} = (\widehat{\mathbf{u}}_1, \dots, \widehat{\mathbf{u}}_{N_{IS}})$, the sufficient predictor $f_{GSIR}(\widehat{\mathbf{U}}) = (f_{GSIR}(\widehat{\mathbf{u}}_1), \dots, f_{GSIR}(\widehat{\mathbf{u}}_{N_{IS}}))$ is obtained by Eq. (31). Then the learning function is denoted by

$$\mathbb{U}(f_{GSIR}(\widehat{\mathbf{u}}_i)) = \frac{|\mu_G(f_{GSIR}(\widehat{\mathbf{u}}_i))|}{\sigma_G^2(f_{GSIR}(\widehat{\mathbf{u}}_i))} \tag{36}$$

As discussed in Sec. 2.4, the value of the learning function indicates the probability of misclassification of the GP model. The smaller is $\mathbb{U}(f_{GSIR}(\widehat{\mathbf{u}}_i))$, the higher chance the point is misclassified. Therefore, the next training point is the point that $\mathbb{U}\left(\mu_G(f_{GSIR}(\widehat{\mathbf{u}}_i))\right)$ is the smallest and is therefore found by

$$f_{GSIR}(\mathbf{u}_{new}) = \min_{\widehat{\mathbf{u}}_i \in \mathbb{P}_{IS}} \mathbb{U}(f_{GSIR}(\widehat{\mathbf{u}}_i)) \tag{37}$$

Then, the corresponding response Y_{new} is available by evaluating the performance function $Y_{new} = G(\mathbf{u}_{new})$. $(\mathbf{u}_{new}, Y_{new})$ is then added to the existing training points. The indicator function in the low dimensional space is $I_F(f_{GSIR}(\mathbf{u})) = 0$ if $\hat{G}(f_{GSIR}(\mathbf{u})) > 0$ or 1 if $\hat{G}(f_{GSIR}(\mathbf{u})) < 0$. The probability of failure (p_f) is obtained by Eq. (18). Here we use the original joint PDF $\phi_{\mathbf{U}}(\cdot)$ of $\mathbf{U}$ and the importance density $\varphi_{\mathbf{U}}(\mathbf{u})$ to estimate the probability of failure instead of using the joint PDF in the subspace after dimension reduction. First, it is difficult or almost impossible to estimate the joint PDF of the variables in the subspace. Second, the sufficiency maintained by the sufficient dimension reduction means that the information in the original space is preserved after dimension reduction. Based on the two reasons, we use the original joint PDF $\phi_{\mathbf{U}}(\cdot)$ of $\mathbf{U}$ and the importance density $\varphi_{\mathbf{U}}(\mathbf{u})$.

The $\mathbb{U}$-learning function is adapted from the lower confidence bounding (lcb) function [50]. The value of $\mathbb{U}\left(f_{GSIR}(\widehat{\mathbf{u}}_i)\right)$ reflects the least confidence level that the indicator function $I_F\left(f_{GSIR}(\widehat{\mathbf{u}}_i)\right)$ is classified into the correct group (safe or failure). Thus, the stopping criterion is set to be $\min \mathbb{U}\left(f_{GSIR}(\widehat{\mathbf{u}}_i)\right) \geq 2$, which means that, at the lowest confidence level, the probability of $I_F\left(f_{GSIR}(\widehat{\mathbf{u}}_i)\right)$ being accurately classified is $\Phi(2) = 97.7\%$, where $\Phi(\cdot)$ is the cumulative density function (CDF) of a standard normal variable. The iterative process terminates until the stopping criterion is satisfied.

Since the probability of failure is calculated in every iteration with the updated GP model, the final probability of failure is obtained from the last iteration. It is recommended that if the coefficient of variation in Eq. (20) is high, for example, 5%, the IS population size should be increased.

3.5 Numerical procedure

The numerical procedures are summarized below.

```
Algorithm 2: GSIR-GP-IS
Initialization
Determine the approximate MPP and importance distribution.
Generate IS population $\widehat{\mathbf{U}} = (\widehat{\mathbf{u}}_i)_1^{N_{IS}}$.
Select ridge parameters $\epsilon_{\mathbf{X}}$, $\epsilon_{Y}$.
Define initial TPs $(\mathbf{U}_t, Y_t)$ by Latin hypercube sampling.
while convergence is false do
    Perform dimension reduction GSIR, and obtain the first eigenvector ($\xi_1$) of the sufficient
    predictor in Eq. (30) and $f_{GSIR}(\mathbf{U}_t) = \xi_1^T Q K_{\mathbf{U}_t}$.
    Construct the GP model $\hat{G}(f_{GSIR}(\mathbf{U}))$ using the low-dimensional training points $(f_{GSIR}(\mathbf{U}_t), Y_t)$.
    Obtain the sufficient predictor at $\widehat{\mathbf{U}}$: $f_{GSIR}(\widehat{\mathbf{U}}) = \xi_1^T Q K_{\mathbf{U}_t\widehat{\mathbf{U}}}$.
    Run the GP model at $f_{GSIR}(\widehat{\mathbf{U}})$ to have $\hat{G}\left(f_{GSIR}(\widehat{\mathbf{U}})\right)$, the probability of failure $p_f$ is obtained by Eq. (18).
    **If** min $U(f_{GSIR}(\widehat{\mathbf{u}}_i)) \geq 2$ is false
        Find the next training point ($\mathbf{u}_{new}$) using Eqs. (36) and (37); obtain $Y_{new} = G(\mathbf{u}_{new})$.
        Add $(\mathbf{u}_{new}, Y_{new})$ to TPs.
    **Elseif** min $\mathbb{U}(f_{GSIR}(\widehat{\mathbf{u}}_i)) \geq 2$ is true
        Stop.
    **End if**
**End while**
**If** $\delta_{IS}$ (Eq. (20)) and accuracy of $p_f$ is satisfied
    Stop.
**Else**
    Go to **Initialization** and increase $N_{IS}$ (the size of IS population).
**End**
**Output:** $p_f$ and associated error.
```

4. Examples

In this section, four examples are provided to demonstrate the proposed method. The first example is a mathematical problem followed by three engineering examples. We compare the proposed method with MCS, FORM, the second order saddlepoint approximation (SOSPA), and SIR-GP-IS. SOSPA [48] is a second-order approximation method based on SORM and saddlepoint approximation (SPA). SIR-GP-IS uses the same settings as the proposed method but using the linear dimension reduction method. Since the first example has been studied by SS-SVM [49] and SIR-SPCE [29], we also compare the two methods for the first example. Since constructing a GP model in the high dimensional space is more expansive than in a subspace, we only use GP-IS, the algorithm without dimension reduction, to evaluate the first case of example 1 to show the necessity of dimension reduction for high-dimensional problems.

A weak penalty is applied for the four examples, and the ridge parameters $\epsilon_{\mathbf{X}}$ and ϵ_Y of GSIR are set to be 10^{-5}. Since the proposed method is a sampling-based meta-modeling method, we run the method 20 times to assess its performance. We then report the medians of the results, including the probability of failure, the error, and the number of function calls. The accuracy of different methods is assessed by the error relative to MCS. The relative error is defined by

$$\varepsilon = \left| \frac{p_f - p_{f,\mathrm{MCS}}}{p_{f,\mathrm{MCS}}} \right| \times 100\% \tag{38}$$

where p_f is the probability obtained by a non-MCS method. The efficiency is measured by the number of function calls and the coefficient of efficiency (CoE). CoE describes the efficiency with respect to the dimension and is defined by

$$\mathrm{CoE} = \frac{\text{The number of performance function calls}}{\text{The dimension of input random variables}} \tag{39}$$

4.1 A mathematical problem

The mathematical example is given in [51] and is further studied in [29, 52]. The performance function is defined by

$$g(\mathbf{X}) = n + 3\sigma\sqrt{n} - \sum_{i=1}^{n} x_i \tag{40}$$

in which $x_i, i = 1, \dots, n$, are independent and lognormally distributed with means and standard deviations being 1 and 0.2, respectively. We study three cases that n is equal to 40, 100, and 250. The corresponding initial DoE sample sizes are 200, 400, and 1000. The IS population sizes N_{IS} are 10^4, 10^4, and 3×10^4 for the three cases. The IS population sizes are large enough to cover the major failure boundaries. If this condition was not satisfied, a larger sample size ($\mathrm{e.g.,}\ 10^5$ or 10^6) would be needed.

The results of dimension reduction are presented in Figs. 2, 3, and 4, in which TPs means training points. The sufficient predictor, which is obtained by GSIR using Eq. (31), is in the one-dimensional space. The three cases show that the sufficient predictor and the real response have perfect monotonicity, for which Spearman's correlations are 0.9993, 0.9999, and 1.0 for the three cases. It is found that the added learning points are concentrated on the failure boundary or the limit state, which means that the GSIR-aided dimension reduction method can identify points in the vicinity of the failure boundary and alleviate the curse of dimensionality.

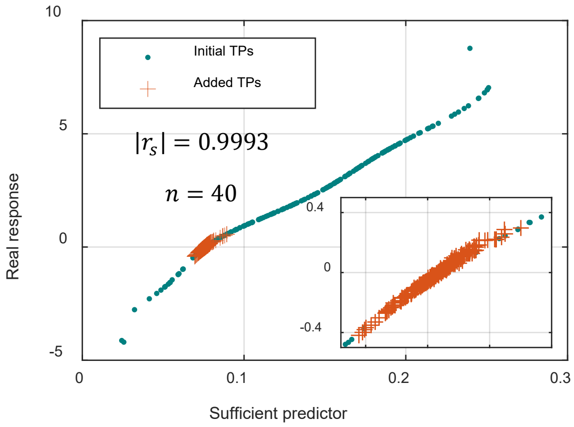


Fig. 2 Sufficient predictor versus real response of Case 1

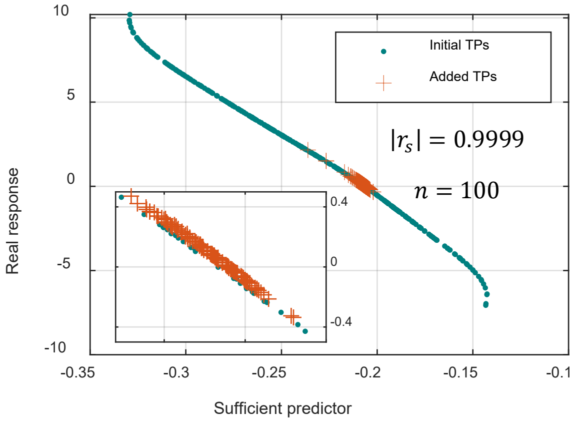


Fig. 3 Sufficient predictor versus real response of Case 2

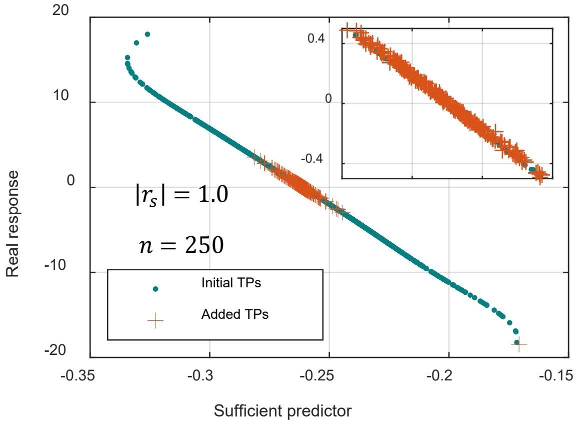


Fig. 4 Sufficient predictor versus real response of Case 3

In addition to the proposed method (GSIR-GP-IS), other methods are also performed, including MCS, FORM, and SOSPA, GP-IS, SIR-GP-IS. We only run MCS for one time with sufficient samples. FORM and SOSPA are also run for one time since these approximation methods are not influenced by randomness. The results of SS-SVM and SIR-SPCE are directly from the literatures as mentioned previously. The results are summarized in Table 1.

Table 1 Results of different methods for Example 1

n	Methods	p_f	Error (%)	FCs	CoE
40	MCS	1.97×10^{-3}	-	10^7	2.5×10^5
	FORM	2.152×10^{-4}	89.1	164	4.1
	SOSPA	2.028×10^{-3}	2.96	1,025	25.6
	GP-IS	7.41×10^{-2}	3600	1241	31.0
	SIR-GP-IS	1.14×10^{-3}	42.04	1241	31.0
	SS-SVM	1.95×10^{-3}	1.5	3,729	93.2
	SIR-SPCE	1.88×10^{-3}	3.3	1,200	30
	GSIR-GP-IS	1.98×10^{-3}	0.37	400.65	10.0
100	MCS	1.72×10^{-3}	-	10^7	10^5
	FORM	4.204×10^{-5}	97.55	404	4.0
	SOSPA	1.796×10^{-3}	4.42	5,555	5.6
	SIR-GP-IS	0.1614	9283	559	5.59
	SS-SVM	1.74×10^{-3}	0.58	6036	60.4
	SIR-SPCE	1.63×10^{-3}	5.6	3,000	30
	GSIR- GP-IS	1.74×10^{-3}	1.03	715.4	7.2
250	MCS	1.56×10^{-3}	-	10^7	4×10^4
	FORM	2.82×10^{-6}	99.82	1004	4.0
	SOSPA	1.673×10^{-3}	7.24	32,630	130.5
	SIR-GP-IS	0.1487	9431	1281	5.1
	SS-SVM	1.61×10^{-3}	1.26	10,707	42.8
	SIR-SPCE	1.59×10^{-3}	0.6	10,000	40
	GSIR-GP-IS	1.60×10^{-3}	2.25	1,548.6	6.2

FORM is the most efficient, but least accurate method. When $n = 40$, the proposed method outperforms SOSPA, SS-SVM, and SIR-SPCE with respect to both accuracy and efficiency. Its error and number of function calls are 0.37% and 401.7, respectively. For the case $n = 100$, GSIR-GP-IS obtains an error of 1.03% with 715.4 function calls. Although SS-SVM has a slightly smaller error (0.58%), it calls the performance function 3,729 times. The proposed method performs well for the high dimensional case ($n = 250$). Although SS-SVM and SIR-SPCE are more accurate than the proposed method, their efficiency is much poorer with CoEs of 40.8 and 40, respectively. The proposed method produces an accurate solution (2.25% error) with much fewer function calls ($\text{CoE} = 6.2$). GP-IS, the algorithm without dimension reduction, cannot converge within the prespecified number of iterations (1000) for the case with the lowest dimension ($n = 40$). SIR-GP-IS, the algorithm with the linear dimension reduction, cannot converge either within 1000 iterations for the same case. The results reported in Table 1 are from the last iteration of the two methods. Although SIR-GP-IS displays a fast convergence rate for the other two cases, the algorithm does not work as the errors are 9283% and 9341%, respectively. Since the surrogate model construction in a high dimensional space is time-consuming, we only run GP-IS for $n = 40$ to demonstrate the necessity of dimension reduction.

To analyze the uncertainty of the result from the proposed method, we also provide box plots in Fig. 5 for the probabilities of failure and errors from the 20 runs. The medians of the probability of failure and the corresponding errors are $(1.98 \times 10^{-3}, 0.37\%)$, $(1.74 \times 10^{-3}, 1.03\%)$, and $(1.60 \times 10^{-3}, 2.25\%)$ for the three cases, where the errors here are obtained by comparing the median probability of failure with the MCS by Eq. (38). The error plot is from the 20 simulations whose median errors are 1.06%, 2.98%, and 2.41%. The standard deviations of errors are 1.77%, 2.80%, and 3.01%. The highest error is smaller than 6%.

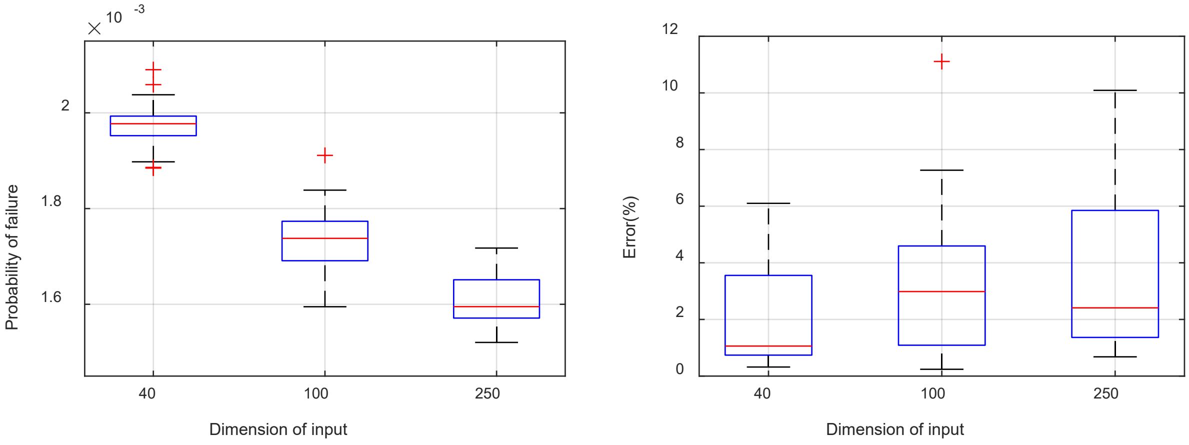


Fig. 5 Box plot of 20 simulations

As mentioned in Sec. 2.1, the nonlinearity between the sufficient predictors and real response is ascending. Fig. 6 shows the first ten sufficient predictors versus the real response based on the result of case 1 ($n = 40$). Since the other cases and the other three examples also have the same pattern, we provide the figure for illustration for only case 1 in this example. For GSIR, the first d sufficient predictors are obtained, and we use the first one for GP modeling and active learning. After the algorithm converges, we plot the first 10 sufficient predictors versus the real response. It is clear that the added training points by active learning cluster at the failure boundary for all sufficient predictors. Therefore, we need to use only the first sufficient predictor.

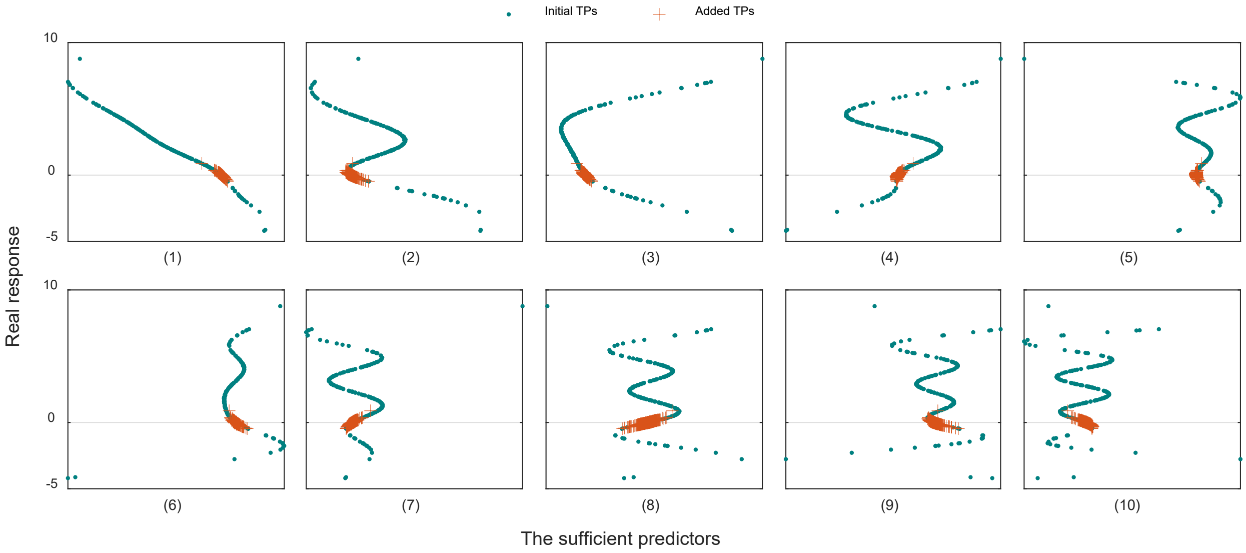


Fig. 6 The relationship between sufficient predictors and real response

4.2 A cantilever beam

The second example is a beam (Fig. 7) that is subjected to 106 random forces on the top. Six forces ($F_1, \ldots, F_6$) are lognormally distributed, and the rest of the forces ($F_7, \ldots, F_{106}$) are normally distributed. The locations ($l_{F_1}, \ldots, l_{F_{106}}$) that the forces, the width (w), height (h), and the yield strength (S_y) are also normally distributed. All the 215 random variables are independent. Their distributions are given in Table 2.

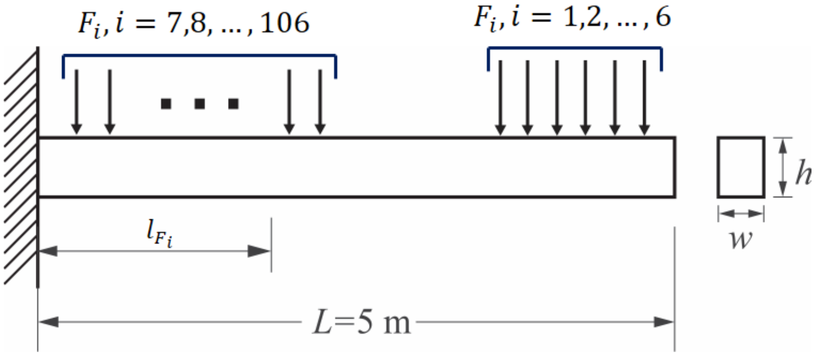


Fig. 7 A cantilever beam

Table 2 Distributions of random variables in Example 2

Random variables	Distribution	Mean	Standard deviation
S_y (MPa)	Normal	720	60
w (m)	Normal	0.2	0.001
h (m)	Normal	0.4	0.001
$F_i, i = 1,2, \dots ,6$ (kN)	Lognormal	$30 + 5i$	$2.4 + 0.4i$
$l_{F_i}, i = 1,2, \dots ,6$ (m)	Normal	$4.3 + 0.1i$	0.01
$F_i, i = 7,8, \dots ,106$ (kN)	Normal	10	1
$l_{F_i}, i = 7,8, \dots ,106$ (m)	Normal	$0.02i$	0.01

A failure would occur if the yield strength S_y is smaller than the maximum stress, and the performance function is therefore given by

$$g(\mathbf{X}) = S_y - \frac{6\sum_{i=1}^{106} F_i l_{F_i}}{wh^2} \tag{41}$$

There are 600 initial training points in this example. We project the 215 input random variables in the high dimensional space to the sufficient predictor, and the relationship between the sufficient predictor and the real response is shown in Fig. 8. A good monotonic relationship is obtained with the Spearman's correlation being 0.9991. The added training points are concentrated on the failure boundary.

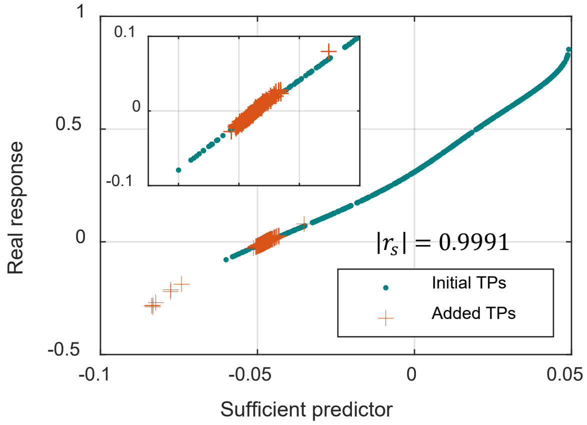


Fig. 8 Sufficient predictor versus real response of Example 2

The results of the proposed method with 20 runs are provided in Fig. 9. The probability of failure and the corresponding error are within the intervals $[1.85 \times 10^{-6}, 2.08 \times 10^{-6}]$ and $[0.02\%, 8.96\%]$, respectively. And the median failure probability is 1.9448×10^{-6} with the corresponding error of 2.83%.

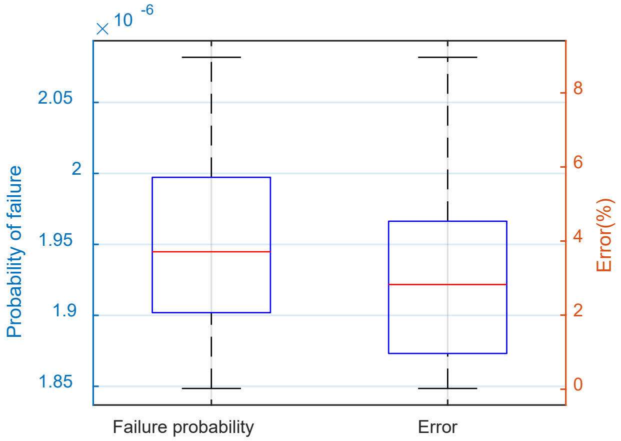


Fig. 9 Statistical results of Example 2

In addition to the proposed method, MCS, FORM, SOSPA, and SIR-GP-IS are also used. The results are listed in Table 3. Although FORM calls the performance function only 648 times, its error is 5.9%. SOSPA is the most accurate method, but its computational cost is extremely high, with 24,084 function calls and a CoE of 112. SIR-GP-IS cannot converge within the maximum of 1,000 iterations. The results shown in Table 3 are from the last iteration. GSIR-GP-IS maintains a good balance between accuracy and efficiency with an error of 2.83% and a CoE of 4.84.

Table 3 Results of different methods for Example 2

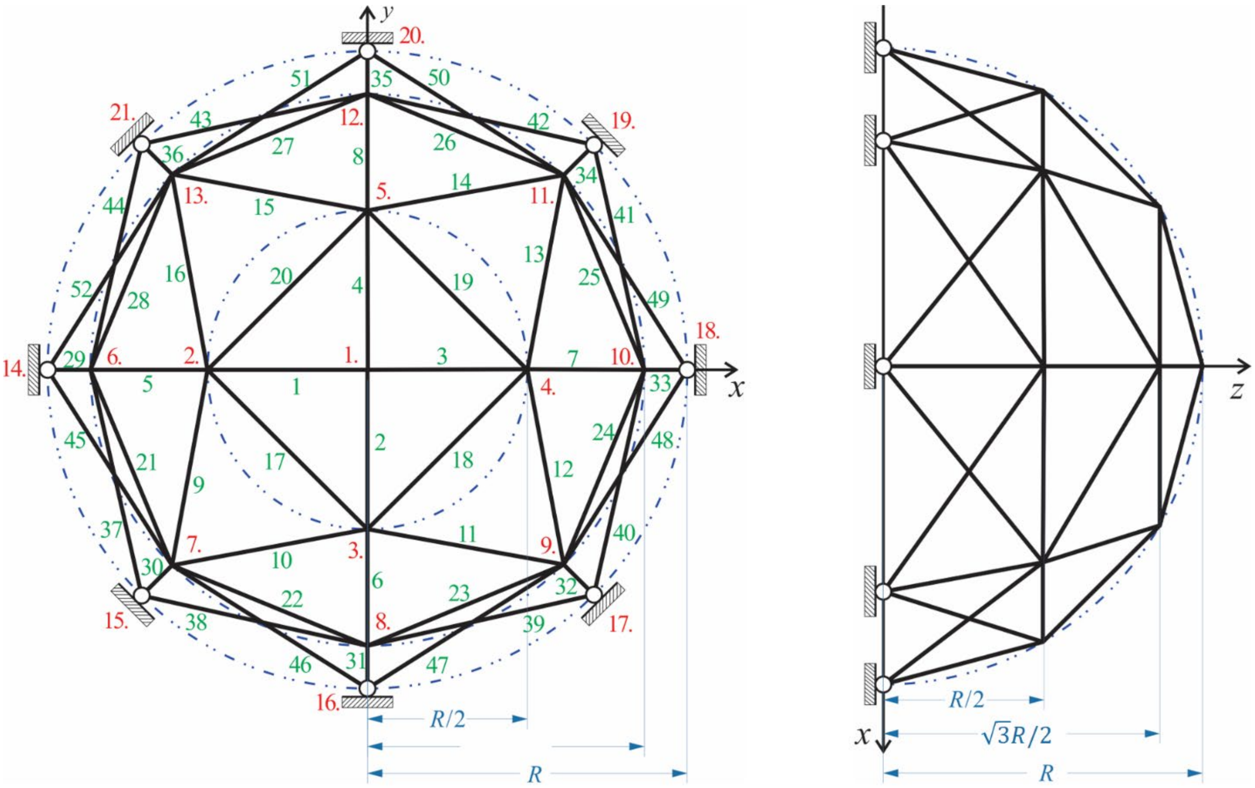
Methods	p_f	Error (%)	FC	CoE
MCS	1.9106×10^{-6}	-	1.6×10^{9}	7.4×10^{6}
FORM	1.7964×10^{-6}	5.9	648	3.0
SOSPA	1.9200×10^{-6}	0.5	24,084	112.0
SIR-GP-IS	2.3097×10^{-6}	20.89	1816	8.45
GSIR-GP-IS	1.9448×10^{-6}	2.83	1040.4	4.84

4.3 A truss system

A dome-truss [53] consists of 52 bars with 21 nodes as shown in Fig. 10, where numbers without dots represent bars and the others with dots mean nodes. All the nodes lie on an imaginary hemisphere with a radius of 240 in. The cross-section areas and the young's moduli of the bars are normally distributed. Six random forces $(F_1, \ldots, F_6)$ that point to the center of the imaginary hemisphere are applied to nodes 1-13. The forces are applied as follows: F_1 to node 1, F_2 to nodes 2, 4, F_3 to nodes 3, 5, F_4 to nodes 6, 10, F_5 to nodes 8, 12, and F_6 to nodes 7, 9, 11, and 13. The random variables are independent and their distributions are summarized in Table 4.

Table 4 Distributions of random variables in Example 3

Random variables	Distribution	Mean	Standard deviation
$E_i, i = 1\sim50$ (ksi)	Normal	2.5×10^4	1000
$A_i, i = 1\sim8,$ and $29\sim36$ (in^2)	Normal	2	0.001
$A_i, i = 9\sim16$ (in^2)	Normal	1.2	0.0006
$A_i, i = 17\sim28,$ and $37\sim52$ (in^2)	Normal	0.6	0.0003
F_1 (kip)	Normal	45	3.6
F_2 (kip)	Extreme	40	6.0
F_3 (kip)	Extreme	35	5.25
F_4 (kip)	Normal	30	2.4
F_5 (kip)	Normal	25	2.0
F_6 (kip)	Normal	20	1.6



(a) Top view

(b) Side view

Fig. 10 A 52-bars truss system

The performance function is given in Eq. (42).

$$Y = g(\mathbf{X}) = \delta_0 - \delta(\mathbf{E}; \mathbf{A}; \mathbf{F}) \tag{42}$$

where $\delta_0 = 0.7$ in is the allowed maximum displacement of node 1, and δ is the actual displacement of the same node, which is obtained by the finite element method (FEM). $\mathbf{E} = [E_1, E_2, \dots, E_{52}]^{\mathrm{T}}$ and $\mathbf{A} = [A_1, A_2, \dots, A_{52}]^{\mathrm{T}}$ are vectors of the young's moduli and cross-sectional areas, respectively, $\mathbf{F} = [F_1, F_2, \dots, F_6]^{\mathrm{T}}$ is a force vector.

We have Fig. 11 shows that the sufficient predictor is monotonic to the real response that the Spearman's correlation is 0.9996. The failure boundary is identified by making use of the monotonic relationship through active learning.

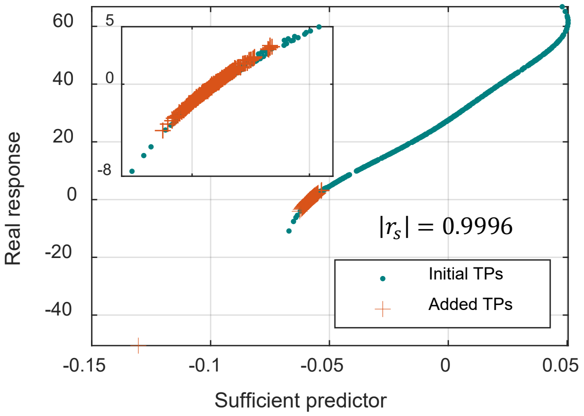


Fig. 11 Sufficient predictor versus real response of Example 3

The statistical results of the proposed method are given in Fig. 12. For the 20 runs, most of the errors are smaller than 6%, and the maximum error is about 8.5%.

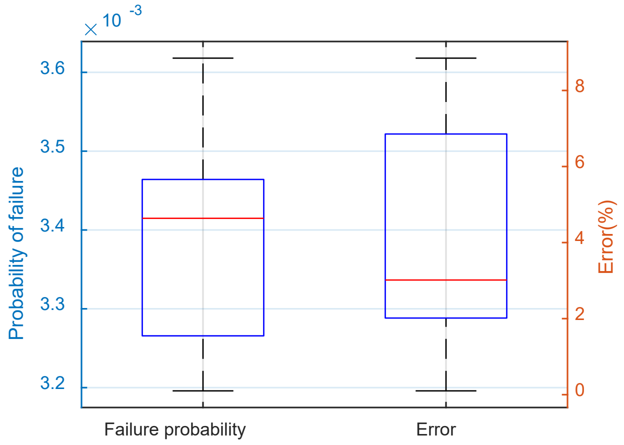


Fig. 12 Statistical results of Example 3

The results of all the methods are given in Table 5. GSIR-GP-IS is less accurate than SOSPA but is far more efficient than SOSPA. GSIR-GP-IS has only 764.4 function calls and a CoE of 6.95 while SOSPA has 6,771 function calls with a CoE of 61.55. The accuracy of FORM is poor, and its error is 15.77%. SIR-GP-IS cannot converge with a subspace of 1 in 1000 iterations and the results reported are from the last iteration.

Table 5 Results of different methods for Example 3

Methods	p_f	Error (%)	FC	CoE
MCS	3.506×10^{-3}	-	10^7	9.09×10^4
FORM	4.059×10^{-3}	15.77	555	5.05
SOSPA	3.529×10^{-3}	0.65	6,771	61.55
SIR-GP-IS	3.046×10^{-3}	13.11	1511	13.7
GSIR-GP-IS	3.4148×10^{-3}	3.02	764.4	6.95

4.4 Nonlinear seismic dynamic analysis of a shear Frame

This example involves a 25-story shear frame structure (Fig. 13) under stochastic seismic excitation. The masses $(m_1, \dots, m_{25})$ of all stories are normal variables with means of 3×10^5 kg and coefficients of variation (C.O.V) of 0.05. The inter-story stiffnesses $(k_1, \dots, k_{25})$ follow lognormal distributions with means of 1.2×10^8 N/m and C.O.V. of 0.1. The motion of the shear frame under seismic ground motions is characterized by the extended Bouc-Wen model [54] given by

$$\boldsymbol{M}\ddot{\boldsymbol{X}} + \boldsymbol{C}\dot{\boldsymbol{X}} + \alpha_h \boldsymbol{K}\boldsymbol{X} + (1-\alpha_h)\boldsymbol{K}\boldsymbol{Z} = -\boldsymbol{M}\ddot{u}_g(t) \tag{43}$$

where $\ddot{\boldsymbol{X}}$, $\dot{\boldsymbol{X}}$, and $\boldsymbol{X}$ are vectors of acceleration, velocity, and displacement, respectively; $\boldsymbol{M}$, $\boldsymbol{C}$ and $\boldsymbol{K}$ are the mass matrix, damping matrix, and stiffness matrix, respectively; α_h is a weighting parameter regarding hysteresis; $\boldsymbol{Z}$ is a vector of hysteretic displacement; $\ddot{u}_g(t)$ denotes the random ground motion and is given by

$$\ddot{u}_g(t) = \xi_{NS}\ddot{u}_{NS}(t) + \xi_{WE}\ddot{u}_{WE}(t) \tag{44}$$

where ξ_{NS} and ξ_{WE} are independent extreme-value variables whose means and C.O.V both are 1 and 0.1, respectively; $\ddot{u}_{NS}(t)$ and $\ddot{u}_{WE}(t)$ are accelerations in the N-S and W-E directions, respectively, obtained from the EI Centro Earthquake [55]. There are 52 random variables in this example. The 13 parameters of the extended Bouc-Wen model used in this paper are $A = 1$, $\alpha_h = 0.04$, $\beta_h = 30$, $\gamma_h = 10$, $n = 1$, $\delta_\nu = 2000$, $\delta_\eta = 2000$, $\zeta_s = 0.99$, $q = 0.25$, $p = 1000$, $\psi = 0.05$, $\delta_\psi = 5$, $\lambda = 0.5$.

The damping matrix is $\boldsymbol{C} = \alpha\boldsymbol{M} + \beta\boldsymbol{K}$, where α and β are the damping coefficients, which are given by $\alpha = 0.02$ and $\beta = 0.01$. The maximum displacement of the first floor d_{max} is obtained by solving the nonlinear Ordinary Differential Equations system in Eq. (43).

$$d_{max} = \max_{t\in[0,T]} \psi_1(m_1, \dots, m_{25}, k_1, \dots, k_{25}, \xi_{NS}, \xi_{WE}) \tag{45}$$

where $\psi_1(\cdot)$ denotes the function of the displacement of the first floor over time. When the maximum displacement exceeds a threshold ($d_{thres} = 32\ \mathrm{mm}$), the shear frame fails. The performance function of the shear frame is defined by

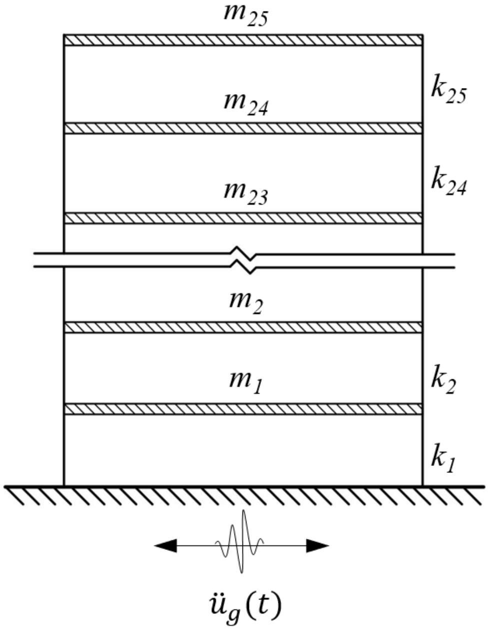


Fig. 13 Schematic of a shear frame

$$Y = d_{thres} - d_{max} \tag{46}$$

For this example, we have 200 initial training points and the IS population is 1×10^4. As shown in Fig. 14, the proposed method can successfully identify the failure boundary for the nonlinear system, although some points are not at the failure boundary at the beginning stage of active learning. Fig. 15 shows that the proposed method maintains a good accuracy for most of the 20 simulations.

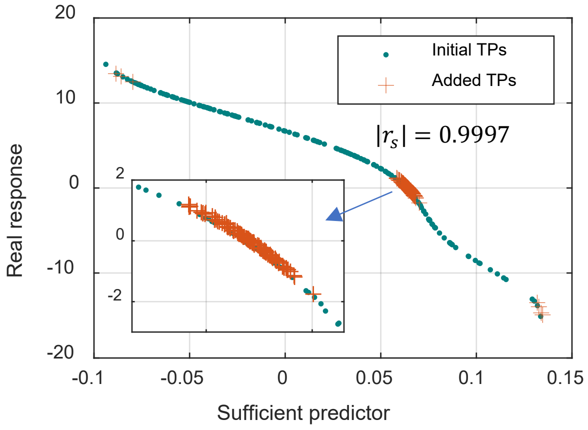


Fig. 14 Sufficient predictor versus real response of Example 4

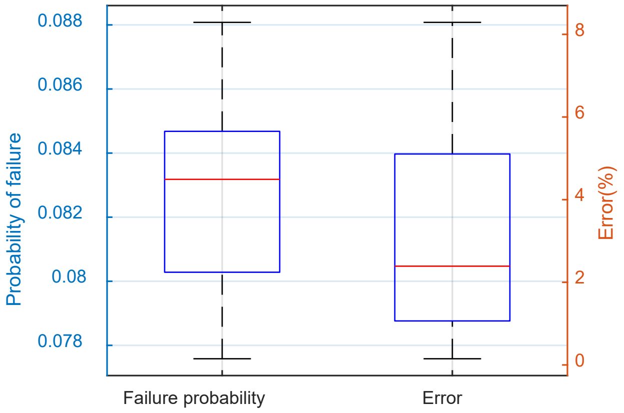


Fig. 15 Statistical results of Example 4

The results of all the methods are provided in Table 7. GSIR-GP-IS outperforms other methods with an error of 2.39%, and 367.25 average function calls, and CoE of 6.93. SIR-GP-IS has a slightly larger average error of 6.02%, but its efficiency is much worse since it needs 938 function calls. The MPP search of FORM cannot converge in 50 iterations with 2650 function calls. The reported p_f for FORM is from the last iteration. Since SOSPA is based on the result of the MPP from FORM, SOSPA also has a large error for this example.

Table 7 Results of different methods for Example 4

Methods	p_f	Error (%)	FC	CoE
MCS	8.46×10^{-2}	-	10^5	9.09×10^4
FORM*	0.4869	476	2650	50
SOSPA*	0.8878	949	4081	78.48
SIR-GP-IS	8.97×10^{-2}	6.02	938	17.7
GSIR-GP-IS	8.32×10^{-2}	2.39	367.25	6.93

*The MPP search does not converge in 50 iterations. Results are reported based on the MPP obtained at the 50th iteration.

5. Discussion and Conclusions

The proposed method combines the generalized sliced inverse regression (GSIR), importance sampling (IS), Gaussian process (GP), and active learning to relieve the curse of dimensionality of high dimensional reliability analysis. A GP model is constructed in a subspace after dimension reduction by GSIR. Then, active learning is used to refine the GP model. By iteratively adding new training points to the training set, the failure boundary is identified, which results in an accurate probability of failure. The four examples demonstrate that GSIR can successfully relieve the curse of dimensionality. The proposed method has a good potential to predict the reliability of high dimensional problems accurately and efficiently.

The proposed method has some limitations. It requires a sufficient number of initial training points, and this may not be computationally efficient for large-scale problems. It is possible that the use of a univariate subspace (the first sufficient predictor) may not be accurate enough for highly nonlinear problems. The proposed method may also produce a large error if multiple failure regions exist. To address the first two limitations, we will study the optimal balance between the number of initial training points and the number of added training points; we will also investigate the use of multiple predictors. For the third limitation, we will explore the possibility of using importance sampling centered at the most probable points of the multiple failure regions.

Acknowledgements

The support from the National Science Foundation under Grant No 1923799 is acknowledged.

Conflict of interest statement

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

References

[1] Rosenblatt M. Remarks on a multivariate transformation. The annals of mathematical statistics. 1952;23:470-2.
[2] Hasofer AM, Lind NC. Exact and invariant second-moment code format. Journal of the Engineering Mechanics division. 1974;100:111-21.
[3] Hohenbichler M, Gollwitzer S, Kruse W, Rackwitz R. New light on first-and second-order reliability methods. Structural safety. 1987;4:267-84.
[4] Yin J, Du X. High-Dimensional Reliability Method Accounting for Important and Unimportant Input Variables. Journal of Mechanical Design. 2021:1-28.
[5] Jin R, Du X, Chen W. The use of metamodeling techniques for optimization under uncertainty. Structural and Multidisciplinary Optimization. 2003;25:99-116.
[6] Isukapalli S, Roy A, Georgopoulos P. Stochastic response surface methods (SRSMs) for uncertainty propagation: application to environmental and biological systems. Risk analysis. 1998;18:351-63.
[7] Hu Z, Du X. Mixed efficient global optimization for time-dependent reliability analysis. Journal of Mechanical Design. 2015;137.
[8] Echard B, Gayton N, Lemaire M. AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation. Structural Safety. 2011;33:145-54.
[9] Huang X, Chen J, Zhu H. Assessing small failure probabilities by AK–SS: an active learning method combining Kriging and subset simulation. Structural Safety. 2016;59:86-95.
[10] Pan Q, Dias D. An efficient reliability method combining adaptive support vector machine and Monte Carlo simulation. Structural Safety. 2017;67:85-95.
[11] Lelièvre N, Beaurepaire P, Mattrand C, Gayton N. AK-MCSi: A Kriging-based method to deal with small failure probabilities and time-consuming models. Structural Safety. 2018;73:1-11.
[12] Teixeira R, Nogal M, O'Connor A. Adaptive approaches in metamodel-based reliability analysis: A review. Structural Safety. 2021;89:102019.
[13] Ramirez-Marquez JE, Coit DW. A Monte-Carlo simulation approach for approximating multi-state two-terminal reliability. Reliability Engineering & System Safety. 2005;87:253-64.
[14] Engelund S, Rackwitz R. A benchmark study on importance sampling techniques in structural reliability. Structural safety. 1993;12:255-76.
[15] Dubourg V, Sudret B, Bourinet J-M. Reliability-based design optimization using kriging surrogates and subset simulation. Structural and Multidisciplinary Optimization. 2011;44:673-90.
[16] Au S-K, Beck JL. A new adaptive importance sampling scheme for reliability calculations. Structural safety. 1999;21:135-58.
[17] Dey A, Mahadevan S. Ductile structural system reliability analysis using adaptive importance sampling. Structural safety. 1998;20:137-54.
[18] Condra L. Reliability improvement with design of experiment: Crc Press; 2001.

[19] Echard B, Gayton N, Lemaire M, Relun N. A combined importance sampling and kriging reliability method for small failure probabilities with time-demanding numerical models. Reliability Engineering & System Safety. 2013;111:232-40.

[20] Binder K, Heermann D, Roelofs L, Mallinckrodt AJ, McKay S. Monte Carlo simulation in statistical physics. Computers in Physics. 1993;7:156-7.

[21] Glynn PW, Iglehart DL. Importance sampling for stochastic simulations. Management science. 1989;35:1367-92.

[22] Au S-K, Beck JL. Estimation of small failure probabilities in high dimensions by subset simulation. Probabilistic engineering mechanics. 2001;16:263-77.

[23] Dunteman GH. Principal components analysis: Sage; 1989.

[24] Hawchar L, El Soueidy C-P, Schoefs F. Principal component analysis and polynomial chaos expansion for time-variant reliability problems. Reliability Engineering & System Safety. 2017;167:406-16.

[25] Bryant FB, Yarnold PR. Principal-components analysis and exploratory and confirmatory factor analysis. 1995.

[26] Rahman S. Global sensitivity analysis by polynomial dimensional decomposition. Reliability Engineering & System Safety. 2011;96:825-37.

[27] Rahman S, Xu H. A univariate dimension-reduction method for multi-dimensional integration in stochastic mechanics. Probabilistic Engineering Mechanics. 2004;19:393-408.

[28] Xie S, Pan B, Du X. High dimensional model representation for hybrid reliability analysis with dependent interval variables constrained within ellipsoids. Structural and Multidisciplinary Optimization. 2017;56:1493-505.

[29] Pan Q, Dias D. Sliced inverse regression-based sparse polynomial chaos expansions for reliability analysis in high dimensions. Reliability Engineering & System Safety. 2017;167:484-93.

[30] Li W, Lin G, Li B. Inverse regression-based uncertainty quantification algorithms for high-dimensional models: Theory and practice. Journal of Computational Physics. 2016;321:259-78.

[31] Li M, Wang Z. Deep learning for high-dimensional reliability analysis. Mechanical Systems and Signal Processing. 2020;139:106399.

[32] Tripathy R, Bilionis I. Deep Active Subspaces: A Scalable Method for High-Dimensional Uncertainty Propagation. International Design Engineering Technical Conferences and Computers and Information in Engineering Conference: American Society of Mechanical Engineers; 2019. p. V001T02A74.

[33] Zhou T, Peng Y. Structural reliability analysis via dimension reduction, adaptive sampling, and Monte Carlo simulation. Structural and Multidisciplinary Optimization. 2020:1-23.

[34] Xu J, Wang D. Structural reliability analysis based on polynomial chaos, Voronoi cells and dimension reduction technique. Reliability Engineering & System Safety. 2019;185:329-40.

[35] Li K-C. Sliced inverse regression for dimension reduction. Journal of the American Statistical Association. 1991;86:316-27.

[36] Constantine PG, Dow E, Wang Q. Active subspace methods in theory and practice: applications to kriging surfaces. SIAM Journal on Scientific Computing. 2014;36:A1500-A24.

[37] Chaudhuri A, Kramer B, Willcox KE. Information Reuse for Importance Sampling in Reliability-Based Design Optimization. Reliability Engineering & System Safety. 2020:106853.

[38] Lataniotis C, Marelli S, Sudret B. Extending classical surrogate modeling to high dimensions through supervised dimensionality reduction: a data-driven approach. International Journal for Uncertainty Quantification. 2020;10.

[39] Schölkopf B, Smola A, Müller K-R. Kernel principal component analysis. International conference on artificial neural networks: Springer; 1997. p. 583-8.

[40] Peng Y, Zhou T, Li J. Surrogate modeling immersed probability density evolution method for structural reliability analysis in high dimensions. Mechanical Systems and Signal Processing. 2021;152:107366.

[41] Coifman RR, Lafon S. Diffusion maps. Applied and computational harmonic analysis. 2006;21:5-30.

[42] Lee K-Y, Li B, Chiaromonte F. A general theory for nonlinear sufficient dimension reduction: Formulation and estimation. The Annals of Statistics. 2013;41:221-49.

[43] Wu H-M. Kernel sliced inverse regression with applications to classification. Journal of Computational and Graphical Statistics. 2008;17:590-610.
[44] Akaho S. A kernel method for canonical correlation analysis. arXiv preprint cs/0609071. 2006.
[45] Aiserman M, Braverman EM, Rozonoer L. Theoretical foundations of the potential function method in pattern recognition. Avtomat i Telemeh. 1964;25:917-36.
[46] Vapnik V. The nature of statistical learning theory: Springer science & business media; 2013.
[47] Rasmussen CE. Gaussian Processes in Machine Learning. In: Bousquet O, von Luxburg U, Rätsch G, editors. Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2 - 14, 2003, Tübingen, Germany, August 4 - 16, 2003, Revised Lectures. Berlin, Heidelberg: Springer Berlin Heidelberg; 2004. p. 63-71.
[48] Prasad KM, Bapat R. The generalized moore-penrose inverse. Linear Algebra and its Applications. 1992;165:59-69.
[49] Lataniotis C, Wicaksono D, Marelli S, Sudret B. UQLab user manual – Kriging (Gaussian process modeling). ETH Zurich, Switzerland: Chair of Risk, Safety and Uncertainty Quantification; 2021.
[50] Cox DD, John S. A statistical method for global optimization. [Proceedings] 1992 IEEE International Conference on Systems, Man, and Cybernetics: IEEE; 1992. p. 1241-6.
[51] Rackwitz R. Reliability analysis—a review and some perspectives. Structural safety. 2001;23:365-95.
[52] Bourinet J-M, Deheeger F, Lemaire M. Assessing small failure probabilities by combined subset simulation and support vector machines. Structural Safety. 2011;33:343-53.
[53] Gomes HM. Truss optimization with dynamic constraints using a particle swarm algorithm. Expert Systems with Applications. 2011;38:957-68.
[54] Ma F, Zhang H, Bockstedte A, Foliente GC, Paevere P. Parameter analysis of the differential model of hysteresis. J Appl Mech. 2004;71:342-9.
[55] Pacific Earthquake Engineering Research Center. Time History Data Files from El Centro Site Imperial Valley Irrigation District, http://www.vibrationdata.com/elcentro.htm; [accessed 1 September 2021].