

Research Article

The Method of Solving Structural Reliability with Multiparameter Correlation Problem

Juan Du, Haibin Li, and Yun He

College of Science, Inner Mongolia University of Technology, Hohhot 010051, China

Correspondence should be addressed to Haibin Li; lhb2003@126.com

Received 1 July 2017; Revised 12 November 2017; Accepted 16 November 2017; Published 11 December 2017

Academic Editor: Fiorenzo A. Fazzolari

Copyright © 2017 Juan Du et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Correlation among variables must be considered to accurately reflect the level of structure reliability. This problem has referential value to engineering practice and has attracted attention from relevant scholars and industries. In this paper, Copula function was used to build the joint probability density function among all variables. The key is to describe the correlation among variables, solve the correlation parameter θ of Copula function, and select the type of correlation structure among variables. The correlation parameter θ of Copula function was solved using Pearson linear correlation coefficient and maximum likelihood estimation. Based on the Akaike information criteria (AIC) and Bayesian information criteria (BIC), the optimal Copula function was selected, and the correlation structure among variables was determined. Monte Carlo method, which is based on Nataf inverse transformation, was introduced and used to evaluate the reliability of the correlated variable. Finally, this paper proposed the reliability calculation method based on dual neural network and direct integration by establishing the dual neural network of original and integrand functions. Compared with the Monte Carlo method, the proposed method can be utilized to efficiently and precisely calculate the structure reliability of multiple correlated random variables.

1. Introduction

In structural reliability analysis, variables are often correlated, for instance, the positive correlation between the seismic peak displacement and permanent displacement [1] and among the stresses of the weld fatigue damage [2] and the significant negative correlation between the shear strength parameters of rock and soil and the parameters of pile-load-displacement curve [3, 4]. Therefore, the correlation among variables must be considered to rationally analyze the structural reliability. The traditional calculation method for structural reliability includes first-order [5] and second-order moments [6], which are only limited to the linear correlation of variables. A 2D or multidimensional distribution model is often used to characterize the correlation among variables, such as 2D lognormal distribution [7] and multidimensional normal distribution [8]. However, one of the drawbacks of these models is the requirement of same-edge distribution variables, which greatly limit their application in reliability analysis.

The emergence and development of Copula theory solve the above problems and provide a new way to construct joint probability density function of correlated variables [9]. Copula theory was first proposed by Sklar in 1959 [10]. Under the framework of Copula function, variables could follow arbitrary edge distributions, and the linear and nonlinear correlations among variables could be defined [11]. Copula functions were introduced in computational structural reliability because of its capacity to handle the arbitrary correlations of variables. Li et al. used the 2D Frank Copula function to construct the joint probability distribution function of two seismic attenuation models [12]. Based on the Copula function, Tang et al. presented the load-displacement hyperbolic probabilistic analysis of pile foundation and used the Copula function to establish the joint probability distribution function of hyperbolic parameters [13]. Huang et al. established a Copula function model of rock mass shear strength parameters based on measured data [14]. Liu et al. used Copulas to develop a reliability model for systems with s-dependent degradation processes.

The proposed model accommodates the assumptions of s -dependence among the degradation processes and allows for different marginal distributions [15]. For two-component and multiple-component systems with multiple failure modes, Liu and Fan established the mixed Copula models for time-independent reliability analysis of series systems, parallel systems, series-parallel systems, and parallel-series systems [16]. Based on the Copula function, Xu et al. constructed a joint probability density function, which was integrated on the failure field to calculate the failure probability of the structure [17]. In reliability-based design optimization (RBDO) of the structure, the exact joint probability density function of the input relevant variables is necessary to obtain the optimal design [18]. Lee et al. used the Copula function to consider the correlation among variables and solve the problem of RBDO [19, 20] and found that the correlation significantly affected RBDO results [21–25]. They also analyzed the influence of the confidence level for Copula function on RBDO results [26] and used the Bayesian and Markov chain Monte Carlo methods to select the Copula function [27]. For RBDO with varying standard deviations (STDs), Cho et al. used the Copula function for the design of sensitivity and then improved the efficiency of the calculation [28].

The calculation methods for structural reliability are classified in three categories. The first method solves the structural response or the probability feature quantity of performance function [29–31]; however, the computational accuracy in this method is heavily dependent on the form of performance functions. The second method was the Monte Carlo method, which directly applies sampling statistics [32–34]. The disadvantage of this method is the large computational complexity. The third method was direct integration [35], which has encountered mathematical difficulties in the multiple integral and quantitative description of computational accuracy.

Neural networks can approximate any functions and thus could be properly applied to structural reliability. Papadrakakis et al. combined neural network with Monte Carlo simulation to analyze the reliability of elastoplastic structure; the time-consuming calculation of the Monte Carlo simulation was reduced, and the efficiency of the calculation was improved [36]. Lopes et al. analyzed reliability using neural network instead of finite element analysis. The neural network had advantages in computational efficiency when compared with Monte Carlo method [37]. Zuo et al. used neural network to fit the performance function of the structure. The values of performance function and partial derivatives were obtained at the point of mean values. Hence, the moments of performance function were calculated based on the moments of random variables [38]. Cheng and Li used the neural network to simulate the limit state equation of long-span bridge. The genetic algorithm was used to train the network, and the failure probability of the structure was obtained [39]. Meng et al. utilized BP neural network for non-linear mapping function trains to obtain the explicit expression of stress in response to the random variable. A study analyzed reliability and the sensitivity of metal structure by combining random perturbation theory and first-order

second-moment method [40]. Elhewy et al. proposed a response surface method based on artificial neural networks and consequently reduced the computational complexity for reliability analysis [41]. Li et al. proposed a structural reliability method, which is based on neural network and direct integration, to solve the reliability of independent variables [42]. Dai and Cao developed a new neural network model based on wavelet support vector machine for reliability analysis; this model has extended the application of wavelet neural network to a high dimension [43].

This paper proposed a direct integration method that is based on dual neural network and could be applied for reliability calculation. The proposed method is composed of two neural networks with similar structure, multiple inputs, single output, and single hidden. By designing the function relation between the weights of two neural networks, one neural network is able to approximate integrands, whereas the other approximates original function. Therefore, the above networks were called integrand and original function neural networks. We only need to train the integrand neural network. Thus, the weights of original function neural network were provided directly by the function relation between the weights of the two neural networks. We subsequently used the original function neural network to calculate the multiple integral. In the proposed method, the integrand could easily obtain the sample data that were directly trained, and the integral computational accuracy would be greatly improved. Therefore, the proposed method can efficiently and accurately solve structural reliability problems.

First, we introduced the method by constructing the joint probability density function of correlation variable based on Copula function. This method includes the solution of correlation parameter θ and the selection of the structural type of the Copula function. Second, we introduced Nataf-Monte Carlo method. Third, we proposed direct integration based on dual neural network, which is in turn based on the integral form of reliability computation and the normalization method of the integral area. Fourth, we compared the proposed technique with Monte Carlo method and verified its effectiveness in simulation. Finally, the full-text conclusion and future prospects were presented.

2. Construction of Joint Probability Density Function Based on Copula Function for Correlated Variables

2.1. Copula Function. Copula theory was first proposed by Sklar in 1959. Sklar pointed out that any multidimensional joint distribution function could be decomposed into a corresponding edge distribution function and a Copula function. The Copula function determines the correlation among variables, including the size of the correlation coefficient and the type of the correlation structure [10]. According to its strict definition stated by Nelsen, the Copula function is a function that associates the joint distribution function of the variable with its edge distribution function. In essence, Copula function is also a joint distribution function. For n -dimensional cases, the Copula function was defined as

TABLE 1: Copula function types.

Copula types	Copula distribution function $C(u_1, u_2, \dots, u_n; \theta)$	θ range of values
Gaussian	$\phi_n(\phi^{-1}(u_1), \phi^{-1}(u_2), \dots, \phi^{-1}(u_n); \theta)$	$[-1, 1]$
Gumbel	$\exp\left\{-\left[\sum_{i=1}^n (-\ln u_i)^\theta\right]^{1/\theta}\right\}$	$[1, \infty)$
Clayton	$\left(\sum_{i=1}^n u_i^{-\theta} - n + 1\right)^{-1/\theta}$	$(0, \infty)$
Frank	$-\frac{1}{\theta} \ln \left[1 + \frac{\prod_{i=1}^n (e^{-\theta u_i} - 1)}{(e^{-\theta} - 1)^{n-1}}\right]$	$n = 2, \theta \in (-\infty, \infty) \cap (\theta \neq 0)$ $n \geq 3, \theta \in (0, \infty)$
Number 16	$\frac{1}{2}(A + \sqrt{A^2 + 4\theta})$ $A = \sum_{i=1}^n u_i - n + 1 - \theta \left(\sum_{i=1}^n \frac{1}{u_i} - n + 1\right)$	$n = 2, \theta \in [0, \infty)$ $n \geq 3, \theta \in (0.1863, \infty)$

the n -dimensional joint distribution function. The edge distribution was $[0, 1]$ in the $[0, 1]^n$ space [9].

According to Sklar theorem [10], the joint distribution function $F(x_1, x_2, \dots, x_n)$ of variables X_1, X_2, \dots, X_n could be expressed as

$$\begin{aligned} F(x_1, x_2, \dots, x_n) &= C(F_1(x_1), F_2(x_2), \dots, F_n(x_n); \theta) \\ &= C(u_1, u_2, \dots, u_n; \theta), \end{aligned} \quad (1)$$

where $u_i = F_i(x_i)$ is the edge distribution function of the variable X_i ; $i = 1, 2, \dots, n$. $C(u_1, u_2, \dots, u_n; \theta)$ is the Copula function, and θ is the correlation parameter of the Copula function.

From (1), the joint probability density function $f(x_1, x_2, \dots, x_n)$ could be obtained as

$$\begin{aligned} f(x_1, x_2, \dots, x_n) &= f_1(x_1) f_2(x_2) \cdots f_n(x_n) \\ &\cdot c(F_1(x_1), F_2(x_2), \dots, F_n(x_n); \theta) \\ &= c(u_1, u_2, \dots, u_n; \theta) \prod_{i=1}^n f_i(x_i), \end{aligned} \quad (2)$$

where $f_i(x_i)$ is the edge probability density function of variables X_1, X_2, \dots, X_n and $c(u_1, u_2, \dots, u_n; \theta) = \partial^n C(u_1, u_2, \dots, u_n; \theta) / \partial u_1 \partial u_2 \cdots \partial u_n$ is the density function for $C(u_1, u_2, \dots, u_n; \theta)$.

If the edge distribution function of variables X_1, X_2, \dots, X_n and the Copula function were known, then the multidimensional distribution model of the variables could be established using (1) and (2).

2.2. Correlation Parameter θ of Copula Function. The correlation parameter θ of the Copula function characterizes the correlation among variables. The method of solving the correlation parameter θ is different for different types of Copula functions.

When the Copula function of multidimensional variable X_1, X_2, \dots, X_n was an Ellipse Copula function, the number of correlation parameters θ_{ij} was the same as that of the

correlation coefficients among variables and had one-to-one correspondence. The correlation parameter of any two variables X_i and X_j was θ_{ij} . The relationship between θ_{ij} and Pearson linear correlation coefficient ρ_{ij} was [9]

$$\begin{aligned} \rho_{ij} &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{x_i - \mu_i}{\sigma_i} \frac{x_j - \mu_j}{\sigma_j} f_i(x_i) f_j(x_j) \\ &\cdot c(u_i, u_j; \theta_{ij}) dx_i dx_j. \end{aligned} \quad (3)$$

When the Copula function of the multidimensional variable X_1, X_2, \dots, X_n was the Archimedean Copula function, the multidimensional Archimedean Copula function had only one correlation parameter θ because only one generator existed. This parameter describes the overall correlation between the multidimensional variables X_1, X_2, \dots, X_n . Maximum likelihood estimation is generally used to solve the correlation parameter θ [44].

$$L(\theta) = \sum_{i=1}^n \ln c(u_1, u_2, \dots, u_n; \theta). \quad (4)$$

2.3. Selection of the Optimal Copula Function. Different Copula functions could describe different correlations among variables. Table 1 lists various Copula function types [9]. Therefore, selecting the optimal Copula function, which could best fit the correlation among variables, is necessary when constructing the joint probability density function among variables. In this paper, we used Akaike information criterion (AIC) and Bayesian information criterion (BIC) to select the optimal Copula function. The best Copula function has the smallest AIC or BIC values and is well fitted for the correlated structure among variables. For multidimensional distribution models, the AIC and BIC values were expressed as follows [45, 46]:

$$\begin{aligned} \text{AIC} &= 2k - 2 \sum_{i=1}^n \ln(c(u_1, u_2, \dots, u_n; \theta)) \\ \text{BIC} &= k \ln n - 2 \sum_{i=1}^n \ln(c(u_1, u_2, \dots, u_n; \theta)), \end{aligned} \quad (5)$$

where k is the number of correlation parameters θ . In the Ellipse Copula function, $k = 0.5n(n - 1)$, whereas in the Archimedean Copula function, $k = 1$.

3. Monte Carlo Method Based on Nataf Inverse Transformation

Monte Carlo is a numerical simulation method that is widely used in structural reliability engineering [47, 48]. Based on statistical sampling theory, Monte Carlo method solves structural reliability by statistical sampling or random simulation of random variables. This method could obtain the true failure probability if the distribution type of the variable is known, and the number of sampling times is sufficient. In reliability calculation, Monte Carlo method is considered a precise calculation method [49]. In this section, the Nataf-Monte Carlo method was introduced to calculate the structural reliability of correlated variables. This method was defined as Nataf-MCS.

The Nataf distribution model was proposed by Nataf in 1962 [50]. The famous Nataf transform is a representation form of the Nataf distribution model, which includes positive and inverse transforms. Nataf transforms could convert correlated variables into independent variables. Nataf inverse transforms convert the independent variables into correlated variables.

If the cumulative distribution function of the variables $\mathbf{X} = (X_1, X_2, \dots, X_n)$ were $F_1(x_1), F_2(x_2), \dots, F_n(x_n)$, then the correlation coefficient matrix was ρ . The concrete steps were given by the Nataf-Monte Carlo method to calculate the structural reliability of the relevant variables as follows [49–51].

(1) The correlation coefficient matrix ρ_0 of the standard normal distribution variables $\mathbf{Z} = (Z_1, Z_2, \dots, Z_n)$ was solved.

$$\rho_{ij} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{x_i - \mu_i}{\sigma_i} \frac{x_j - \mu_j}{\sigma_j} \frac{f_i(x_i) f_j(x_j)}{\varphi(z_i) \varphi(z_j)} \varphi(z_i, z_j; \rho_{0ij}) dx_i dx_j \quad (6)$$

Ang et al. established the following empirical equation [49] for various distribution types under normal circumstances $|\rho_{ij}| \leq |\rho_{0ij}|$:

$$\rho_{0ij} = \xi \rho_{ij}, \quad (7)$$

where the coefficient ξ is a function of the correlation coefficient ρ_{ij} and the edge distribution $F_i(x_i)$. The equation for the coefficient ξ was provided in literature [51].

(2) The standard normal distribution variables $\mathbf{Z} = (Z_1, Z_2, \dots, Z_n)$ were correlated, and their correlation coefficient matrices were ρ_0 . ρ_0 was decomposed by Gorensky to obtain the lower triangular matrix \mathbf{L} .

$$\rho_0 = \mathbf{L}\mathbf{L}^T. \quad (8)$$

(3) The traditional random number generation method was applied to produce independent variables $\mathbf{Y} =$

(Y_1, Y_2, \dots, Y_n) , which all followed the standard normal distribution.

(4) Based on the lower triangular matrix \mathbf{L} , the independent normal random variables $\mathbf{Y} = (Y_1, Y_2, \dots, Y_n)$ were transformed into standard normal distribution variables $\mathbf{Z} = (Z_1, Z_2, \dots, Z_n)$.

$$\mathbf{Z} = \mathbf{L}\mathbf{Y}. \quad (9)$$

(5) The relevant standard normal distribution variables $\mathbf{Z} = (Z_1, Z_2, \dots, Z_n)$ were mapped to the original spatial variables $\mathbf{X} = (X_1, X_2, \dots, X_n)$ using the principle of equal probability transformation.

$$\mathbf{X} = F^{-1}(\phi(\mathbf{Z})). \quad (10)$$

(6) The original spatial variables $\mathbf{X} = (X_1, X_2, \dots, X_n)$ were substituted into the functional functions of the reliability analysis, and the structure response was calculated: $G_i = G(X_i)$. The number of times $G_i < 0$ was the number of failure sample points M . The unbiased estimate of the structural failure probability was [49] as follows:

$$P_f = \frac{M}{N}. \quad (11)$$

4. Structural Reliability Calculations Based on Dual Neural Network and Direct Integration for Correlated Variables

Structural reliability could be expressed as the probability of function $G > 0$. G could be expressed as a function of basic variables $\mathbf{X} = [x_1, x_2, \dots, x_n]$; that is, $G = g(x_1, x_2, \dots, x_n)$. The joint probability density function of the correlated variables was $f(x_1, x_2, \dots, x_n)$, introducing a weight function:

$$h(x) = \begin{cases} 1, & x > 0 \\ 0, & x \leq 0, \end{cases} \quad (12)$$

$$\begin{aligned} & y(x_1, x_2, \dots, x_n) \\ &= h(g(x_1, x_2, \dots, x_n)) f(x_1, x_2, \dots, x_n) \\ &= h(g(x_1, x_2, \dots, x_n)) f(x_1) f(x_2) \cdots f(x_n) \\ &\quad \cdot c(x_1, x_2, \dots, x_n; \theta). \end{aligned} \quad (13)$$

Structural reliability could be expressed as multiple integrals as follows:

$$\begin{aligned} P &= \int_{\mu_n - b_n \sigma_n}^{\mu_n + b_n \sigma_n} \cdots \int_{\mu_1 - a_1 \sigma_1}^{\mu_1 + a_1 \sigma_1} y(x_1, x_2, \dots, \\ & x_n) dx_1 dx_2 \cdots dx_n \\ &= \int_{\mu_n - b_n \sigma_n}^{\mu_n + b_n \sigma_n} \cdots \int_{\mu_1 - a_1 \sigma_1}^{\mu_1 + a_1 \sigma_1} h(g(x_1, x_2, \dots, x_n)) f(x_1) \\ &\quad \cdot f(x_2) \cdots f(x_n) \cdot c(x_1, x_2, \dots, x_n) dx_1 dx_2 \cdots dx_n. \end{aligned} \quad (14)$$

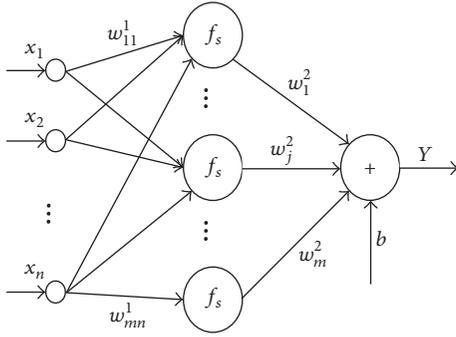


FIGURE 1: Original function network structure.

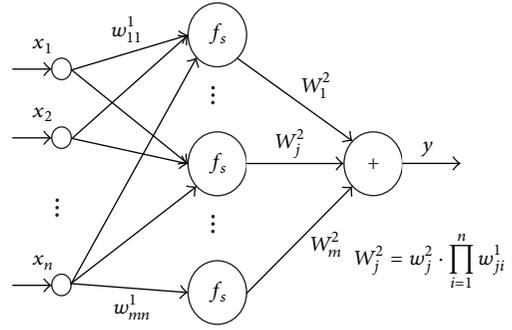


FIGURE 2: Integrand network structure.

In (16), μ_i and σ_i were the mean and standard deviation, respectively, of random variable i . a_i and b_i were positive integers, which were determined by accuracy requirements. In general, $a_i \geq 3$ and $b_i \geq 3$.

The construction of dual neural network for multiple integral calculation was discussed in the following section.

We constructed a single hidden layer BP neural network to establish the mapping relationship between the input variable X and the original function Y . The network structure is shown in Figure 1.

The relationships between the output variables and input variables could be expressed as

$$Y = \sum_{j=1}^m f_s \left(\sum_{i=1}^n w_{ji}^1 x_i + \vartheta_j \right) w_j^2 + b. \quad (15)$$

The derivation of (17) could be given by

$$\begin{aligned} y &= \frac{\partial^n Y}{\partial x_1 \partial x_2 \cdots \partial x_n} \\ &= \sum_{j=1}^m f_s^{(n)} \left(\sum_{i=1}^n w_{ji}^1 x_i + \vartheta_j \right) w_{j1}^1 w_{j2}^1 \cdots w_{jn}^1 w_j^2, \end{aligned} \quad (16)$$

where $W_j = w_{j1}^1 w_{j2}^1 \cdots w_{jn}^1 w_j^2$. Equation (16) could be rewritten as a function of the relationship between the output and input variables as follows:

$$y = \sum_{j=1}^m f_s^{(n)} \left(\sum_{i=1}^n w_{ji}^1 x_i + \vartheta_j \right) W_j. \quad (17)$$

The neural network completed the mapping relationship between the input variable X and the integrand y . The network structure is shown in Figure 2.

Let the original function of the integrand $y(x_1, x_2, \dots, x_n)$ be $Y(x_1, x_2, \dots, x_n)$. According to multiple integral theory, function Y is the weighted algebraic sum of each vertex in a hypercube $D = \{(x_1, x_2, \dots, x_n) \mid x_1^1 \leq x_1 \leq x_1^2, x_2^1 \leq x_2 \leq x_2^2, \dots, x_n^1 \leq x_n \leq x_n^2\}$ and could be expressed as

$$\begin{aligned} J &= \sum_{m_1=1}^2 \sum_{m_2=1}^2 \cdots \sum_{m_n=1}^2 (-1)^{\sum_{k=1}^n m_k} Y(x_1^{m_1}, x_2^{m_2}, \dots, x_n^{m_n}). \end{aligned} \quad (18)$$

Thus, the original function and integrand neural networks were combined as a dual neural network. The two networks, namely, integrand network and original function network, had three layers, n inputs, single output, and m unit in hidden layer. In the integrand network, the connection weights and threshold of input layer to hidden layer unit were w_{ji}^1 and ϑ_j , respectively; the connection weights and threshold of hidden layer to output layer were W_j and 0, respectively; and the activation function of the hidden layer units was $f_s^{(n)}$. In the original function network, the connection weights and threshold of input layer to hidden layer unit were w_{ji}^1 and ϑ_j , respectively; the connection weights and threshold of hidden layer to output layer were $W_j / \prod_{i=1}^n w_{ji}^1$ and b , respectively; and the activation function of the hidden layer units was f_s . When the integrand network approximated the integrand in the integral, the original function network simultaneously approximated the original function.

5. Numerical Examples

5.1. Example 1. The performance function of a system was assumed as $g(X) = x_1^2 - 3x_2$, where x_1 is the lognormal distribution random variables with mean value $\mu_1 = 12.42$, standard deviation $\sigma_1 = 2.484$, and coefficient of variation $\eta_1 = 0.20$. x_2 was an extreme value distribution of type I with mean value $\mu_2 = 17.75$, standard deviation $\sigma_2 = 5$, and coefficient of variation $\eta_2 = 0.2817$. The correlation coefficient of x_1 and x_2 was $\rho = -0.85$. The structural reliability was calculated.

Based on the Copula function, the 2D joint probability density function of the random variables x_1 and x_2 was established and could be written as

$$f(x_1, x_2) = f_1(x_1) f_2(x_2) c(F_1(x_1), F_2(x_2); \theta), \quad (19)$$

where

$$f_1(x_1) = \frac{1}{\sqrt{2\pi} x_1 \sigma_Y} \exp \left[-\frac{1}{2} \left(\frac{\ln x_1 - \mu_Y}{\sigma_Y} \right)^2 \right], \quad (20)$$

$$f_2(x_2) = \alpha \exp \{ -(x_2 - \delta) \alpha - \exp [-(x_2 - \delta) \alpha] \},$$

where $\mu_Y = 2.5$, $\sigma_Y = 0.198$, $\alpha = 0.246$, and $\delta = 15.4$.

Calculated by AIC and BIC criteria, in example 1, Plackett Copula, Frank Copula, number 16 Copula, and Gaussian

TABLE 2: Correlation parameter θ in example 1.

Copula function	Plackett Copula	Frank Copula	Number 16 Copula	Gaussian Copula
Correlation parameter θ	0.0229	-11.3478	0.0038	-0.89

TABLE 3: Training sample set of integrand network in example 1.

x_2	x_1	2.4920	2.9479	Y	3.4038	...	24.3741	24.8300
-32.2500		0	0	0	...	0	0	0
-30.6173		0	0	0	...	0	0	0
...	
42.8520	$1.1880e - 30$	$3.3068e - 19$	$6.7500e - 16$...	0	0	0	0
44.4847	$8.0728e - 31$	$2.7839e - 19$	$5.8612e - 16$...	0	0	0	0
46.1173	$5.4832e - 31$	$2.3675e - 19$	$5.0142e - 16$...	0	0	0	0
47.7500	$3.7267e - 31$	$2.0329e - 19$	$4.1622e - 16$...	0	0	0	0

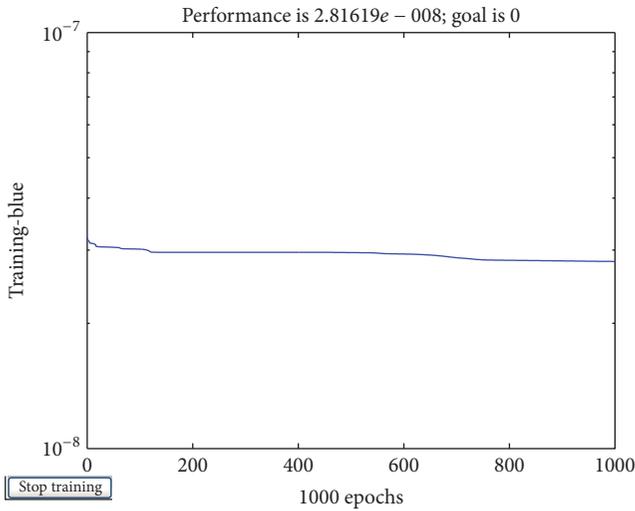


FIGURE 3: Training error curve of integrand network in example 1.

Copula were selected to construct the 2D joint probability density function of random variables x_1 and x_2 . The values of the correlation parameter θ of each Copula function were obtained from (3) and are shown in Table 2.

$f(x_1, x_2)$ and $g(X)$ were substituted for (13) to obtain the integrand.

The integrand neural network had 2 inputs, single output, and 20 units in hidden layer. The transfer functions of hidden layer and output layer were $f_s = e^x$ and Purelin function.

The ranges of x_1 and x_2 $[\mu_1 - 4\sigma_1, \mu_1 + 5\sigma_1]$ and $[\mu_2 - 10\sigma_2, \mu_2 + 6\sigma_2]$ were divided into 50 parts. Two variables crossed each other to form the network input samples. The network output value y of the corresponding sample points was calculated by the integrand. The training sample set is shown in Table 3.

This example used Levenberg-Marquardt training algorithm. Error convergence curve of training integrand network after 1000 steps is shown in Figure 3.

The relationship of the dual neural network was used to construct the original function network. The original

TABLE 4: Simulative sample set of original function network in example 1.

x_1	24.8300	24.8300	2.4920	2.4920
x_2	47.7500	-32.2500	47.7500	-32.2500

function network structure was the same as the integrand network. The integrand network training sample set of each vertex in a hypercube was calculated and simulated using the original function network. The sample set is shown in Table 4. The results were substituted into (18) to obtain the result of reliability.

For comparison, we provide the results from Nataf-Monte Carlo method and the proposed method. The results calculated using Nataf-Monte Carlo method with sampling 1,000,000 were viewed as accurate solutions. The relative errors are shown in Table 5.

The ratio of the calculation workloads of the Nataf-MCS and the proposed method can be obtained from the following equation:

$$\gamma = \frac{W_1 N_m}{W_1 N_{n1} + W_2 N_{n2}}, \quad (21)$$

where W_1 is a single workload for structural analysis. N_m is the calculation times of the Nataf-MCS. N_{n1} is the calculation times required to construct the network training sample. W_2 is a single workload for neural network training. N_{n2} is the neural network training times.

We can compare the computational efficiency of the two methods using the ratio γ of calculation workload. In this paper, $W_1 = 10W_2$. From the calculation in this example, $\gamma = 80$.

From Table 5, the accuracy of the proposed method was close to that of the Nataf-MCS. The maximum relative error was only 2.00%. Based on ratio $\gamma = 80$, the calculation workload of Nataf-MCS was 80 times higher than that of the proposed method. With regard to computational efficiency, the proposed method saves calculation time and greatly improves the computational efficiency of reliability.

TABLE 5: Example 1 reliability calculation results.

Method	Direct integral method based on dual neural network				
	Nataf-MCS	Plackett Copula	Frank Copula	Number 16 Copula	Gaussian Copula
Reliability	0.9609	0.9654	0.9528	0.9648	0.9799
Relative error (%)	0	0.47	0.84	0.41	2.00

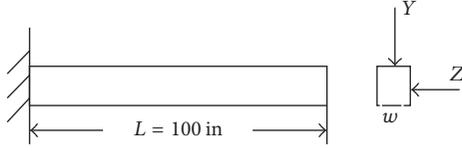


FIGURE 4: Cantilever structure.

TABLE 6: Statistical parameters and distribution types of variables in example 2.

Project	Y (lb)	Z (lb)	w (in)
Mean μ	988.72	506.28	1.98
Standard deviation σ	120.78	132.48	0.27
Edge distribution function	Normal	Normal	Normal

TABLE 7: Values of correlation parameter θ in example 2.

Copula function	Frank Copula	Gumbel Copula	Clayton Copula
Correlation parameter θ	32.47	7.66	9.05

5.2. Example 2. In the cantilever beam shown in Figure 4, Y is the horizontal load on the beam, Z is the lateral load on the beam, and w represents the beam width. The performance function of structure $g(Y, Z, w) = 45 - 43Y/w - 160Z/w^2$. Considering the correlation between variables, the reliability of the cantilever beam was analyzed [52].

Based on the experimental data in [52], the mean, standard deviation, and edge distribution functions of variables Y, Z, and w were obtained and are listed in Table 6.

Calculated by AIC and BIC criteria, in example 2, Frank Copula, Gumbel Copula, and Clayton Copula were selected to construct the 3D joint probability density function of random variables Y, Z, and w.

$$f(Y, Z, w) = f(Y) f(Z) f(w) \cdot c(F(Y), F(Z), F(w); \theta). \quad (22)$$

Based on the maximum likelihood estimation in (4), the values of each Copula function's correlation parameter θ were calculated and are listed in Table 7.

The 3D joint probability density function $f(Y, Z, w)$ and the performance function $g(Y, Z, w)$ in example 2 were substituted into (13) to calculate the integrand.

The structure of the integrand neural network could also be constructed. The integrand neural network had three inputs and single output. After adjustment, the number of

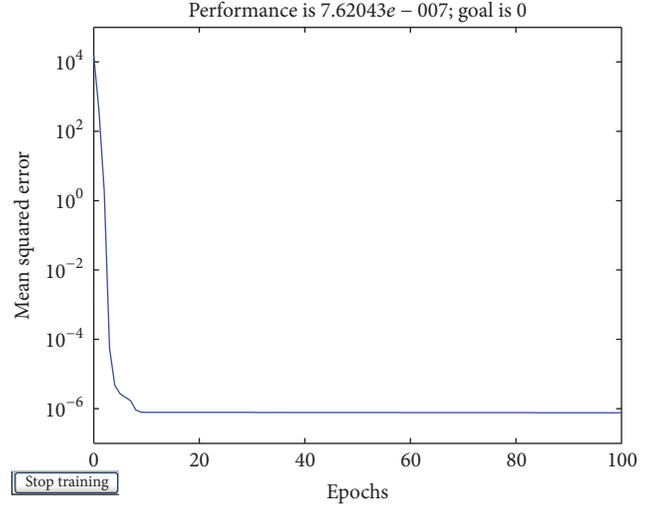


FIGURE 5: Training error curve of integrand network in example 2.

hidden layer units is $m = 10$. The transfer functions of hidden layer and output layer were $f_s = e^x$ and Purelin function, respectively.

Variables Y, Z, and w were separately divided into 30 parts in the range of $[\mu_i - 3\sigma_i, \mu_i + 3\sigma_i]$. The three variables intersect each other to form the integrand network input sample point. The network output value y of the corresponding sample point was calculated by the integrand function, and the training sample point set is shown in Table 8.

We used the Levenberg-Marquardt training algorithm. When the training reached 100 steps, the error of the integrand network reached the convergence state. The error convergence curve is shown in Figure 5.

The integrand network training sample set of each vertex in a hypercube was calculated and simulated with the original function network. The sample set is shown in Table 9. The results were substituted into (18) to obtain the result of reliability.

Nataf-MCS was introduced in Section 3. The sampling was 1,000,000 times, and the reliability of example 2 was 0.9998. In this example, $\gamma = 35.71$. Table 10 shows the reliability values calculated by the two methods.

In this example, three kinds of Archimedean Copula functions were selected to construct the joint probability density function of variables Y, Z, and w. The reliability of the proposed method was 0.85%, which was higher than that of Nataf-MCS. The ratio γ of calculation workload was 35.71, indicating that the proposed method has superior computational efficiency and can reduce the calculation time.

TABLE 8: Training sample set of integrand network in example 2.

Y	Z	w	1.17	1.23	...	2.74	2.80
626.40	108.84	3.7406e - 019		7.2479e - 019	...	2.0566e - 017	9.2777e - 019
626.40	136.25	6.1769e - 019		1.1926e - 018	...	3.3278e - 017	1.4721e - 018
...
626.40	903.70	1.1581e - 020		2.1592e - 020	...	4.9973e - 019	1.6704e - 020
...
1351.10	108.84	7.8195e - 021		1.4568e - 020	...	3.3560e - 019	1.1119e - 020
1351.10	136.25	1.2385e - 020		2.3063e - 020	...	5.2996e - 019	1.7472e - 020
...
1351.10	903.70	1.3654e - 022		2.5241e - 022	...	5.5421e - 021	1.6625e - 022

TABLE 9: Simulative sample set of original function network in example 2.

Y	626.40	1351.10	626.40	1351.10	626.40	1351.10	626.40	1351.10
Z	108.84	903.70	108.84	108.84	903.70	108.84	903.70	903.70
w	1.17	2.80	2.80	2.80	1.17	1.17	2.80	1.17

TABLE 10: Example 2 reliability calculation results.

Method	Nataf-MCS	Direct integral method based on dual neural network		
		Frank Copula	Gumbel Copula	Clayton Copula
Reliability	0.9998	0.9960	0.9913	0.9936
Relative error (%)	0	0.38	0.85	0.62

TABLE 11: Test data for soil parameters.

Number	LI	s_u (kPa)	s_u^{re} (kPa)	σ'_p (kPa)	σ'_v (kPa)
1	1.28	2.58	0.22	7.94	9.56
2	1.27	4.74	0.16	29.41	12.82
3	1.45	5.42	0.39	17.14	16.44
4	1.20	5.82	0.59	27.63	20.06
5	1.26	6.97	0.65	23.37	24.04
...
343	2.11	12.26	1.53	80.24	61.90
344	2.05	14.10	1.53	86.36	68.02
345	1.57	15.20	1.84	94.76	75.66

5.3. Example 3. The soil parameters of rock and soil include liquidity index (LI), undrained shear strength s_u , remolded undrained shear strength s_u^{re} , preconsolidation stress σ'_p , and vertical effective stress σ'_v . The experimental data in literature [53] were used as an example to illustrate the calculation steps of the proposed method. The performance function of the soil parameters $g(LI, s_u, s_u^{re}, \sigma'_p, \sigma'_v) = \sigma'_p(s_u^{re} + LI\sigma'_p) - (\sigma'_v - s_u)^2$ was considered, and the reliability of rock and soil was analyzed under normal use limit state. Test data for soil parameters are shown in Table 11.

The mean μ , standard deviation σ , and edge distribution functions type of each variable were obtained by analyzing the experimental data as shown in Table 12.

From the calculation, we obtained the correlation coefficient ρ_{ij} of any two variables to form the matrix ρ .

ρ

$$= \begin{matrix} & \text{LI} & s_u^{re} & s_u^{re} & \sigma'_p & \sigma'_v \\ \text{LI} & \begin{bmatrix} 1 & 0.05 & -0.50 & -0.06 & -0.21 \end{bmatrix} \\ s_u & \begin{bmatrix} 0.05 & 1 & 0.17 & 0.84 & 0.57 \end{bmatrix} \\ s_u^{re} & \begin{bmatrix} -0.50 & 0.17 & 1 & 0.30 & 0.42 \end{bmatrix} \\ \sigma'_p & \begin{bmatrix} -0.06 & 0.84 & 0.30 & 1 & 0.73 \end{bmatrix} \\ \sigma'_v & \begin{bmatrix} -0.21 & 0.57 & 0.42 & 0.73 & 1 \end{bmatrix} \end{matrix} \quad (23)$$

Given the positive and negative correlations among variables LI, s_u , s_u^{re} , σ'_p , and σ'_v , we need to select the appropriate Copula function from Elliptical Copula function to construct the joint probability density function $f(LI, s_u, s_u^{re}, \sigma'_p, \sigma'_v)$. Using AIC and BIC criteria, Gaussian Copula was selected for example 2. Afterwards, (3) was used to obtain any two variables related parameters θ_{ij} and to form matrix θ .

θ

$$= \begin{matrix} & \text{LI} & s_u^{re} & s_u^{re} & \sigma'_p & \sigma'_v \\ \text{LI} & \begin{bmatrix} 1 & 0.07 & -0.91 & -0.08 & -0.26 \end{bmatrix} \\ s_u & \begin{bmatrix} 0.07 & 1 & 0.25 & 0.88 & 0.67 \end{bmatrix} \\ s_u^{re} & \begin{bmatrix} -0.91 & 0.25 & 1 & 0.41 & 0.53 \end{bmatrix} \\ \sigma'_p & \begin{bmatrix} -0.08 & 0.88 & 0.41 & 1 & 0.78 \end{bmatrix} \\ \sigma'_v & \begin{bmatrix} -0.26 & 0.64 & 0.53 & 0.78 & 1 \end{bmatrix} \end{matrix} \quad (24)$$

TABLE 12: Statistical parameters and distribution types of variables in example 3.

Items	LI	s_u (kPa)	s_u^{re} (kPa)	σ'_p (kPa)	σ'_v (kPa)
Mean μ	1.25	31.01	2.51	105.82	66.63
Standard deviation σ	0.61	29.48	3.81	103.21	53.48
Edge distributionfunction	Lognormal	Lognormal	Lognormal	Lognormal	Lognormal

TABLE 13: Training sample set of integrand network in example 3.

LI	s_u	s_u^{re}	σ'_p	σ'_v	0.10	0.10	...	415.45	415.45
					0.10	25.32		201.85	227.07
0.10	0.10	0.10			$1.0728e - 006$	$5.3550e - 007$...	$4.1945e - 011$	$2.7672e - 011$
0.10	0.10	1.64			$3.3559e - 007$	$1.6819e - 007$...	$1.3897e - 011$	$9.2053e - 012$
...
0.10	0.10	13.94			$3.8546e - 009$	$1.9948e - 009$...	$2.5274e - 013$	$1.7285e - 013$
...
3.08	119.45	0.10			$1.3303e - 010$	$6.4296e - 011$...	$5.8267e - 014$	$3.7218e - 014$
3.08	119.45	1.64			$3.8597e - 011$	$1.8729e - 011$...	$1.7905e - 014$	$1.1483e - 014$
...
3.08	119.45	13.94			$2.4266e - 013$	$1.2159e - 013$...	$1.7823e - 016$	$1.1802e - 016$

The 5D joint probability density function of variables LI, s_u , s_u^{re} , σ'_p , and σ'_v can be expressed as the following equation:

$$\begin{aligned}
 & f(\text{LI}, s_u, s_u^{re}, \sigma'_p, \sigma'_v) \\
 &= f(\text{LI}) f(s_u) f(s_u^{re}) f(\sigma'_p) f(\sigma'_v) \\
 & \cdot c(F(\text{LI}), F(s_u), F(s_u^{re}), F(\sigma'_p), F(\sigma'_v); \theta).
 \end{aligned} \quad (25)$$

The 5D joint probability density function $f(\text{LI}, s_u, s_u^{re}, \sigma'_p, \sigma'_v)$ and the performance function $g(\text{LI}, s_u, s_u^{re}, \sigma'_p, \sigma'_v)$ were substituted into (13) to calculate the integrand.

The structure of the integrand neural network could also be constructed. The integrand neural network had five inputs and single output. After adjustment, the number of hidden layer units was $m = 5$. The transfer functions of hidden layer and output layer were $f_s = e^x$ and Purelin function, respectively.

Variables LI, s_u , s_u^{re} , σ'_p , and σ'_v are divided into 10 parts in the range of $[\mu_i - 4\sigma_i, \mu_i + 5\sigma_i]$. The five variables intersect each other to form the input sample points of the integrand network. The network output value γ of the corresponding sample points was calculated by the integrand function. Given that each variable follows the lognormal distribution, we get $\mu_i - 4\sigma_i = 0.1$ when the left interval of the variable $\mu_i - 4\sigma_i \leq 0$. The training sample set is shown in Table 13.

We used the Levenberg-Marquardt training algorithm. When the training reached 100 steps, the error of the integrand network reached the convergence state. The error convergence curve is shown in Figure 6.

We used the same method for the previous example. The sample points of the original function network were obtained and are shown in Table 14. The simulation values

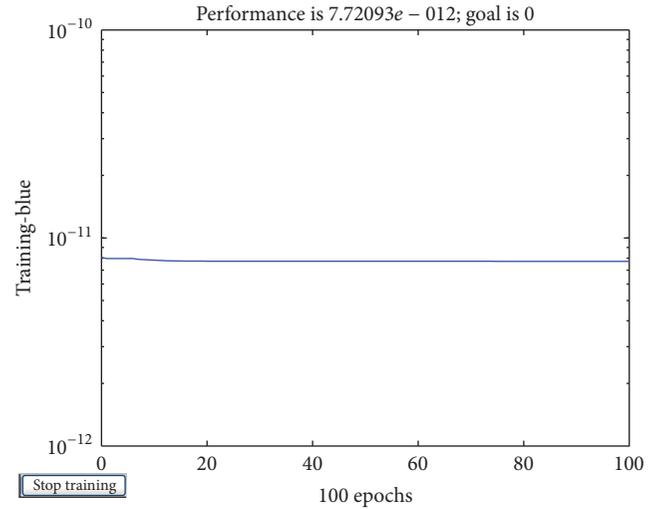


FIGURE 6: Training error curve of integrand network in example 3.

were substituted into (18) to obtain the reliability of the structure as shown in Table 15.

The calculation workload of the two methods was substituted into (21), thus obtaining $\gamma = 9.90$.

The reliability of 5D correlated variables is analyzed using example 3. The reliability of the proposed method was extremely close to that of Nataf-MCS, and the relative error was only 1.16%. From $\gamma = 9.90$, for the same problem, the workload of the two methods varied greatly. The neural network method has advantages in computational efficiency and reduces the calculation time for structural reliability analysis. This example shows that the proposed method can

TABLE 14: Simulative sample set of original function network in example 3.

LI	0.10	3.08	0.10	3.08	...	0.10	3.08
s_u	0.10	119.45	119.45	0.10	...	0.10	119.45
s_u^{re}	0.10	13.94	13.94	0.10	...	13.94	0.10
σ_p^f	0.10	415.45	415.45	0.10	...	0.10	415.45
σ_v^f	0.10	227.07	227.07	0.10	...	227.07	0.10

TABLE 15: Example 3 reliability calculation results.

Method	Nataf-MCS	Direct integral method based on dual neural network	
		Gaussian Copula	
Reliability	0.9594	0.9483	
Relative error (%)	0	1.16	

efficiently solve the reliability problem of multidimensional correlated variables.

6. Conclusion

Identifying the joint probability density function among variables is necessary to accurately calculate the structural reliability. This paper used the Copula function to construct the joint probability density function among correlated variables. According to Copula theory, the estimation of the edge probability density function among variables is independent of the selection of Copula function. The joint distribution function with arbitrary edge distribution and correlation structure could be constructed. This method was suitable for constructing joint probability density functions of 2D and multidimensional correlation problems. In addition, this paper proposed a Monte Carlo method based on Nataf inverse transformation. Nataf-MCS and the proposed method were used to solve the structural reliability among correlated variables. The proposed method could overcome the difficulty of solving multiple integral problems and achieve accurate calculation of reliability. However, we did not establish the relationship of the original function and every random variable because the training sample set originated from the integrand. We conveniently selected the sample set to improve the accuracy of calculating the reliability. Numerical examples showed that the proposed method is highly precise and efficient in calculating the reliability of multidimensional correlated variables.

Copula function is often used to construct the static variable correlation model. In many applications, the correlation coefficient among variables has the characteristics of being dynamic and time-varying. Thus, a Copula model must be constructed to describe the effect of these factors on structural reliability. This issue must be further explored.

In future work, the proposed method can also be extended for the reliability analysis of large and complex structural problems. In these cases, the workload of computing for dual neural network was negligible as compared with that of constructing network training samples. In addition, (21) can be simplified to $\gamma \approx N_m/N_n$. Hence, the proposed method shows efficiency and advantages in analyzing large and complex structural problems.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant no. 11262014 and Graduate Student Scientific Research Innovation Project of Autonomous Region under Grant no. B20171012805.

References

- [1] K. Goda, "Statistical modeling of joint probability distribution using copula: Application to peak and permanent displacement seismic demands," *Structural Safety*, vol. 32, no. 2, pp. 112–123, 2010.
- [2] B. J. Leira, "Probabilistic assessment of weld fatigue damage for a nonlinear combination of correlated stress components," *Probabilistic Engineering Mechanics*, vol. 26, no. 3, pp. 492–500, 2011.
- [3] D. Li, Y. Chen, W. Lu, and C. Zhou, "Stochastic response surface method for reliability analysis of rock slopes involving correlated non-normal variables," *Computers & Geosciences*, vol. 38, no. 1, pp. 58–68, 2011.
- [4] D.-Q. Li, X.-S. Tang, K.-K. Phoon, Y.-F. Chen, and C.-B. Zhou, "Bivariate simulation using copula and its application to probabilistic pile settlement analysis," *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 37, no. 6, pp. 597–617, 2013.
- [5] H. F. Burcharth, J. D. Sorensen, and E. Christian, "On the evaluation of failure probability of monolithic vertical wall breakwaters," *Proceedings of Wave Barriers in Deep Waters*, pp. 458–468, 1994.
- [6] A. Der Kiureghian, H.-Z. Lin, and S.-J. Hwang, "Second-order reliability approximations," *Journal of Engineering Mechanics*, vol. 113, no. 8, pp. 1208–1225, 1987.
- [7] S. Yue, "The bivariate lognormal distribution for describing joint statistical properties of a multivariate storm event," *Environmetrics*, vol. 13, no. 8, pp. 811–819, 2002.
- [8] A. C. Rencher, *Methods of Multivariate Analysis*, John Wiley & Sons, Inc., New York, NY, USA, 1995.
- [9] R. B. Nelsen, *An Introduction to Copulas*, Springer, New York, NY, USA, 2nd edition, 2006.

- [10] M. Sklar, "Fonctions de repartition an dimensions et leurs marges," *Publications de l'Institut de Statistique de l'Université de Paris*, vol. 8, pp. 229–231, 1959.
- [11] X. Zeng, J. Ren, Z. Wang, S. Marshall, and T. Durrani, "Copulas for statistical signal processing (Part I): Extensions and generalization," *Signal Processing*, vol. 94, no. 1, pp. 691–702, 2014.
- [12] Y. Li, B. Shi, and J. Zhang, "The copula joint function and its application in probabilistic seismic hazard analysis," *Earthquake Science*, vol. 30, no. 3, pp. 292–301, 2008.
- [13] X.-S. Tang, D.-Q. Li, C.-B. Zhou, and K.-K. Phoon, "Probabilistic analysis of load-displacement hyperbolic curves of single pile using Copula," *Rock and Soil Mechanics*, vol. 33, no. 1, pp. 171–178, 2012.
- [14] D. Huang, C. Yang, B. Zeng, and G. Fu, "A copula-based method for estimating shear strength parameters of rock mass," *Mathematical Problems in Engineering*, vol. 2014, Article ID 693062, 11 pages, 2014.
- [15] Z. Liu, X. Ma, J. Yang, and Y. Zhao, "Reliability modeling for systems with multiple degradation processes using inverse Gaussian process and copulas," *Mathematical Problems in Engineering*, vol. 2014, Article ID 829597, 2014.
- [16] Y. Liu and X. Fan, "Time-independent reliability analysis of bridge system based on mixed copula models," *Mathematical Problems in Engineering*, vol. 2016, Article ID 2720614, 13 pages, 2016.
- [17] X. Xu, J. Li, J. Gong, H. Deng, and L. Wan, "Copula-Based Slope Reliability Analysis Using the Failure Domain Defined by the g-Line," *Mathematical Problems in Engineering*, vol. 2016, Article ID 6141838, 11 pages, 2016.
- [18] I. Lee, K. K. Choi, and L. Zhao, "Sampling-based RBDO using the stochastic sensitivity analysis and dynamic kriging method," *Structural and Multidisciplinary Optimization*, vol. 44, no. 3, pp. 299–317, 2011.
- [19] I. Lee, K. K. Choi, Y. Noh, L. Zhao, and D. Gorsich, "Sampling-based stochastic sensitivity analysis using score functions for RBDO problems with correlated random variables," *Journal of Mechanical Design*, vol. 133, no. 2, Article ID 021003, 2011.
- [20] Y. Noh, K. K. Choi, and L. Du, "Reliability-based design optimization of problems with correlated input variables using a Gaussian Copula," *Structural and Multidisciplinary Optimization*, vol. 38, no. 1, pp. 1–16, 2009.
- [21] Y. Noh, K. K. Choi, and I. Lee, "Identification of marginal and joint CDFs using Bayesian method for RBDO," *Structural and Multidisciplinary Optimization*, vol. 40, no. 1-6, pp. 35–51, 2010.
- [22] Y. Noh, K. K. Choi, I. Lee, D. Gorsich, and D. Lamb, "Reliability-based design optimization with confidence level for non-Gaussian distributions using bootstrap method," *Journal of Mechanical Design*, vol. 133, no. 9, Article ID 091001, 2011.
- [23] Y. Noh, *Input Model Uncertainty and Reliability-Based Design Optimization with Associated Confidence Level*, The University of Iowa, Iowa, USA, 2009.
- [24] H. Cho, K. K. Choi, and D. Lamb, "Sensitivity developments for RBDO with dependent input variable and varying input standard deviation," *Journal of Mechanical Design*, vol. 139, no. 7, 9 pages, 2017.
- [25] Y. Noh, K. K. Choi, and I. Lee, "Reduction of ordering effect in reliability-based design optimization using dimension reduction method," *AIAA Journal*, vol. 47, no. 4, pp. 994–1004, 2009.
- [26] Y. Noh, K. K. Choi, I. Lee, D. Gorsich, and D. Lamb, "Reliability-based design optimization with confidence level under input model uncertainty due to limited test data," *Structural and Multidisciplinary Optimization*, vol. 43, no. 4, pp. 443–458, 2011.
- [27] Y. Noh, K. K. Choi, and I. Lee, "Comparison study between MCMC-based and weight-based Bayesian methods for identification of joint distribution," *Structural and Multidisciplinary Optimization*, vol. 42, no. 6, pp. 823–833, 2010.
- [28] H. Cho, K. K. Choi, I. Lee, and D. Lamb, "Design Sensitivity Method for Sampling-Based RBDO with Varying Standard Deviation," *Journal of Mechanical Design*, vol. 138, no. 1, Article ID 011405, 2016.
- [29] Y.-G. Zhao and T. Ono, "A general procedure for first/second-order reliability method (FORM/SORM)," *Structural Safety*, vol. 21, no. 2, pp. 95–112, 1999.
- [30] Y.-G. Zhao and T. Ono, "Third-moment standardization for structural reliability analysis," *Journal of Structural Engineering*, vol. 126, no. 6, pp. 724–731, 2000.
- [31] Y.-G. Zhao and T. Ono, "Moment methods for structural reliability," *Structural Safety*, vol. 23, no. 1, pp. 47–75, 2001.
- [32] Q. Gong, J. Zhang, C. Tan, and C. Wang, "Neural networks combined with importance sampling techniques for reliability evaluation of explosive initiating device," *Chinese Journal of Aeronautics*, vol. 25, no. 2, pp. 208–215, 2012.
- [33] V. Papadopoulos, D. G. Giovanis, N. D. Lagaros, and M. Papadrakakis, "Accelerated subset simulation with neural networks for reliability analysis," *Computer Methods Applied Mechanics and Engineering*, vol. 223/224, pp. 70–80, 2012.
- [34] R. Douc, F. Maire, and J. Olsson, "On the use of Markov chain MONte Carlo methods for the sampling of mixture models: a statistical perspective," *Statistics and Computing*, vol. 25, no. 1, pp. 95–110, 2015.
- [35] A. Genz and A. Malik, "Remarks on algorithm 006: An adaptive algorithm for numerical integration over an N-dimensional rectangular region," *Journal of Computational and Applied Mathematics*, vol. 6, no. 4, pp. 295–299, 1980.
- [36] M. Papadrakakis, V. Papadopoulos, and N. D. Lagaros, "Structural reliability analysis of elastic-plastic structures using neural networks and Monte Carlo simulation," *Computer Methods Applied Mechanics and Engineering*, vol. 136, no. 1-2, pp. 145–163, 1996.
- [37] P. A. M. Lopes, H. M. Gomes, and A. M. Awruch, "Reliability analysis of laminated composite structures using finite elements and neural networks," *Composite Structures*, vol. 92, no. 7, pp. 1603–1613, 2010.
- [38] Y.-L. Zuo, H.-H. Zhu, and X.-J. Li, "An ANN-based four order moments method for geotechnical engineering reliability analysis," *Rock and Soil Mechanics*, vol. 34, no. 2, pp. 513–518, 2013.
- [39] J. Cheng and Q. S. Li, "Reliability analysis of structures using artificial neural network based genetic algorithms," *Computer Methods Applied Mechanics and Engineering*, vol. 197, no. 45-48, pp. 3742–3750, 2008.
- [40] W. Meng, Z. Yang, X. Qi, and J. Cai, "Reliability analysis-based numerical calculation of metal structure of bridge crane," *Mathematical Problems in Engineering*, vol. 2013, Article ID 260976, 5 pages, 2013.
- [41] A. H. Elhewy, E. Mesbahi, and Y. Pu, "Reliability analysis of structures using neural network method," *Probabilistic Engineering Mechanics*, vol. 21, no. 1, pp. 44–53, 2006.
- [42] H. B. Li, Y. He, and X. B. Nie, "Structural reliability calculation method based on the dual neural network and direct integration method," *Neural Computing and Applications*, pp. 1–9, 2016.

- [43] H. Dai and Z. Cao, "A Wavelet Support Vector Machine-Based Neural Network Metamodel for Structural Reliability Assessment," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 4, pp. 344–357, 2017.
- [44] U. Cherubini, E. Luciano, and W. Vecchiato, "Copula Methods in Finance," *Copula Methods in Finance*, pp. 1–293, 2013.
- [45] H. Akaike, "A New Look at the Statistical Model Identification," *IEEE Transactions on Automatic Control*, vol. 19, no. 6, pp. 716–723, 1974.
- [46] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.
- [47] Y. Hao, X. Rong, L. Ma, P. Fan, and H. Lu, "Uncertainty analysis on risk assessment of water inrush in karst tunnels," *Mathematical Problems in Engineering*, vol. 2016, Article ID 2947628, 2016.
- [48] S. Mishra, O. A. Vanli, B. P. Alduse, and S. Jung, "Hurricane loss estimation in wood-frame buildings using Bayesian model updating: Assessing uncertainty in fragility and reliability analyses," *Engineering Structures*, vol. 135, pp. 81–94, 2017.
- [49] A. Ang, -S. Tang W, and -S. Tang, *Probability Concepts in Engineering Planning and Design*, John Wiley and Sons, New York, NY, USA, 1984.
- [50] A. Nataf, "Determination des distributions de probabilités dont les marges sont données," *Comptes Rendus De L'Académie Des Sciences*, vol. 255, pp. 42–43, 1962.
- [51] A. Der Kiureghian and P.-L. Liu, "Structural reliability under incomplete probability information," *Journal of Engineering Mechanics*, vol. 112, no. 1, pp. 85–104, 1986.
- [52] C. Jiang, W. Zhang, B. Wang, and X. Han, "Structural reliability analysis using a copula-function-based evidence theory model," *Computers & Structures*, vol. 143, pp. 19–31, 2014.
- [53] J. Ching and K. Phoon, "Corrigendum: Modeling parameters of structured clays as a multivariate normal distribution," *Canadian Geotechnical Journal*, vol. 49, no. 12, pp. 1447–1450, 2012.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

