ORIGINAL ARTICLE

lajss.org© Latin American Journal of Solids and Structures

www.lajss.org

The Subset Simulation method for structural reliability based on CNN-LSTM

Xinqi Zhang[a], Jun Hu[a]*

[a] School of Civil Engineering, Chongqing Jiaotong University, Chongqing 400074, China. Email: 971895608@qq.com, hujun@cqjtu.edu.cn

* Corresponding author

https://doi.org/10.1590/1679-7825/e9070

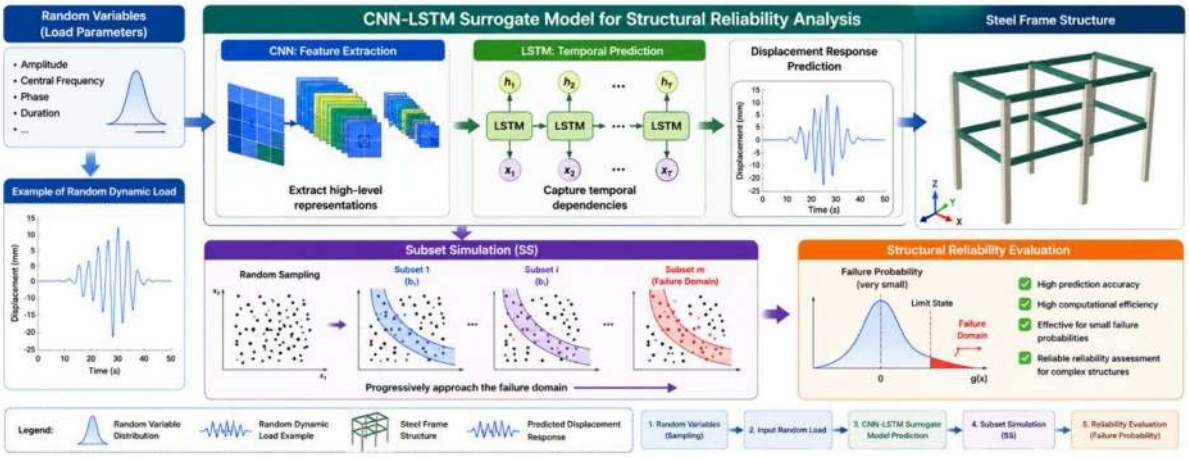
Abstract

Machine learning prediction of structural responses is highly efficient. The use of surrogate models to assess the reliability of small probability events is significant for engineering safety evaluations. This paper proposes a structural reliability analysis method that combines machine learning surrogate models with physical information and Subset Simulation. By using CNN to extract features from load and structural frequencies and then employing LSTM to predict responses based on the feature vector, the surrogate model is integrated with the SS method. This approach aims to address the issues of computational efficiency and accuracy in structural reliability analysis, especially for small failure probabilities. Case studies on planar truss systems and steel frame structures under random dynamic loads demonstrate the effectiveness of the method. The results show that the CNN-LSTM model achieves better prediction accuracy than LSTM. In terms of reliability results, the failure probability predicted by CNN-LSTM is closer to the finite element-Subset Simulation results, while the failure probability error of LSTM-Subset Simulation is 12.50 percentage points higher than that of CNN-LSTM-Subset Simulation. This method is significant for evaluating structural reliability.

Keywords

Structural reliability, Subset Simulation, CNN-LSTM, Surrogate model.

Graphical abstract



1 Introduction

In the reliability analysis of engineering structures, how to efficiently and accurately solve the reliability problems with small failure probabilities has always been a challenge. Especially in complex structures (He et al. 2022; Zhang et al. 2023), traditional probabilistic analysis methods, such as Monte Carlo (MC) simulation (Liu and Jun, 2001), often fail to effectively address this issue due to their enormous computational demands and repetitive calculations.

Received April 09, 2026. In revised form June 02, 2026. Accepted June 09, 2026. Available online June 11, 2026.
https://doi.org/10.1590/1679-7825/e9070

Latin American Journal of Solids and Structures. ISSN 1679-7825. Copyright © 2026. This is an Open Access article distributed under the terms of the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited

For complex structures, the high dimensionality and nonlinearity of the system make each computation time-consuming (Meng et al. 2022; Gao et al. 2024), leading to an almost unbearable consumption of computational resources during large-scale simulations. Both cross assumption (Siorikis et al. 2024) and moment methods (Wang et al. 2020; Guan, 2023) indeed rely on certain simplifying assumptions when solving reliability, and these assumptions are typically made to reduce computational complexity and improve computational efficiency. However, directly applying these assumptions in calculations will lead to inaccurate results, especially when dealing with structures that have high nonlinearity, strong coupling, or complex stochastic characteristics (Hu and Wang, 2023).Therefore, accurately and efficiently solving the low failure probabilities in complex structures has become one of the major challenges in the engineering field.

Subset Simulation (SS), as an effective reliability analysis method, has demonstrated significant advantages in the reliability analysis of complex structures in recent years (Chan et al. 2022; Yang et al. 2022). Unlike traditional MC methods, SS reduces the computational burden by constructing a series of subsets that progressively approach the failure domain, with importance sampling applied within each subset (Zhang and Ma, 2024). This method is capable of effectively estimating extremely low failure probabilities, making it particularly suitable for the reliability analysis of high-dimensional complex systems (Xu et al. 2024; Wang and Abdollah, 2021). By improving sampling efficiency, SS not only reduces computational costs when handling large-scale systems but also ensures computational accuracy (Zhang et al. 2023). Lee presented a SS-based variance-reduction method for evaluating network seismic fragility, addressing challenges like complex topologies and low failure probabilities, and efficiently generating fragility curves using intermediate failure events (Lee et al. 2025). Au developed an adaptive scheme for SS, optimizing the correlation parameter to reduce sample rejection and correlation. It improves efficiency in rare event simulations, especially for nonlinear, high-dimensional problems, and provides insights into scaling proposal PDFs (Elfverson et al. 2024). Li introduces ABCS-SS, combining Bayesian compressive sensing and SS, to efficiently estimate small failure probabilities in nonstationary geotechnical systems, significantly reducing computational time compared to ABCS-MCS (Li, 2025). To address the sample traversal issue in SS, Kinner (Kinner and DiazDelaO, 2025) combined ecological niches with SS, and the results showed that ecological niches can improve this problem. The reliability estimation results of generalized SS have poor stability. Xia (Xia and Liao, 2025) combined sampling techniques with generalized SS and found, through examples, that it is more efficient and ensures higher accuracy compared to traditional SS. As a result, it has been widely applied across various fields. However, the practical application of SS still faces some challenges, such as the need for significant computational resources when calculating the responses of complex structures.

With the development of machine learning, using machine learning to build surrogate models for predicting structural responses has become increasingly common (Sengodan, 2021; Shao and Bassem, 2022). Surrogate models, replacing finite element calculations, offer a new approach for small probability assessment. Zhang (Zhang et al. 2025) combined Transformer and LSTM to propose a hybrid model, which outperforms traditional learning models in predicting bridge seismic responses, offering higher efficiency and maintaining good prediction accuracy even in highly dynamic response segments. To better predict the seismic response of steel structures, Sukhnandan (Sukhnandan and Drosopoulos, 2025) established a large number of test sets and used ANN to accurately predict the seismic response of the steel structures. Tan (Tan et al. 2024) used LSTM as a predictive model to forecast the structural response for assessing the safety of train-bridge systems and verified the effectiveness of this method. In addition, in the intersection of SS and machine learning, some studies have used machine learning as a surrogate model to predict structural responses and perform reliability assessments. For example, Sabri (Sabri et al. 2024) compared NN, LSTM, and Bayesian models, with LSTM showing the best performance; Nguyen (Nguyen et al. 2022) also used DL surrogate models to replace finite element models for reliability evaluation. However, the predictions made by conventional learning models may be prone to overfitting and lack essential physical information, which means the reliability assessment results based on SS may not always be reasonable.

This paper combines CNN-LSTM with SS for structural reliability assessment. First, load and structural information such as mass, stiffness, and damping are input, and CNN is used to reduce the dimensionality and extract useful features. Then, LSTM is employed to train the surrogate model. Finally, the surrogate model replaces finite element analysis, and SS is used to complete the reliability evaluation.

2 Reliability Analysis Based on CNN-LSTM for SS

2.1 SS Theory

In structural reliability analysis, the MC simulation method is not limited by the complexity of the model and can provide a relatively accurate estimate of the structural failure probability when the sample size is sufficiently large. Therefore, it is the most commonly used numerical simulation method. However, when dealing with rare failure probability problems, such as the reliability of suspension bridges under wind field effects, traditional MC methods require significant computational effort and may even fail to converge. Even with improved sampling techniques to reduce sample variance and thus the sample size, the issue of large computational cost cannot be fundamentally solved. SS, based on the conditional probability multiplication theorem in probability theory, addresses this by introducing reasonable intermediate events. This approach transforms the solution of small probability problems into the product of a series of nested event probabilities, significantly reducing computational effort, as shown in Figure 1.

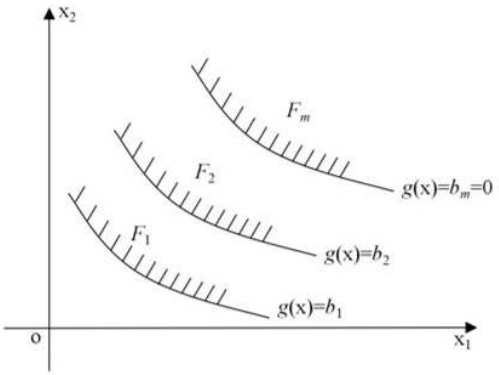


Figure 1. SS Method

As shown in Figure 1, F_i represents the failure domain of different subsets, and b_i represents different failure thresholds. SS introduces different failure thresholds $b_i(b_1 > b_2 > b_m)$ to the performance function $g(x)$. The structural failure probability is represented by P_f, and the structural failure probability in reliability analysis is expressed as:

$$P_{\mathrm{f}} = \iint \cdots \iint_{G(\mathbf{X})<0} f(\mathbf{X})\mathrm{d}\mathbf{X} \tag{1}$$

In the equation, $\mathbf{X}\left(x_1, x_2, \ldots, x_n\right)$ represents n random variables; $f(\mathbf{X})$ is the joint probability density function of $\mathbf{X}$; $G(\mathbf{X})$ is the limit-state function. $F_m \subset F_{m-1} \cdots \subset F_{m-i} \subset F_2 \subset F_1 = F$ represents the failure domain of the structure. By finding a series of subset failure domains $P_f(F)$ that satisfy the conditions, the structural failure probability H can be expressed as:

$$P_{\mathrm{f}}(F) = P\left(F_m\right) = \prod_{i=1}^{m} P\left(F_i \mid F_{i-1}\right) \tag{2}$$

A small failure probability event can be expressed as the product of the conditional probabilities of a series of larger failure events. The conditional failure probability of the i-th state, $P_i\left(F_i \mid F_{i-1}\right)$, is expressed as:

$$P_i\left(F_i \mid F_{i-1}\right) = \frac{1}{N_{i-1}} \sum I_{F_i}\left[g\left(x^{i-1}\right)\right] \tag{3}$$

In the equation, N_{i-1} represents the number of samples for the i-th state; $g\left(x^{i-1}\right)$ is the new sample value.

The indicator function $I_{F_i}[\cdot]$ for the i-th state is expressed as:

$$I_{F_i}[\cdot] = \begin{cases} 0 & g\left(x^{i-1}\right) \leqslant b_i \\ 1 & g\left(x^{i-1}\right) > b_i \end{cases} \tag{4}$$

The samples for the intermediate state F_i are generated using the Markov Chain MC (MCMC) method. By applying the MCMC method, new samples that conform to the target distribution can be generated from the existing samples.

2.2 Markov Chain MC

MCMC sampling can efficiently draw new samples from a subset that conform to the target distribution. For a target distribution with a joint probability density function $\pi(\cdot)$, the transition probability density function between adjacent states of the Markov chain is $p\left(x_{k+1} \mid x_k\right)$, x_{k} is the current state of the random variable, and $x_{\mathrm{k+1}}$ is the next state of the random variable. If the Markov chain satisfies the detailed balance condition:

$$\pi\left(x_k\right) p\left(x_{k+1} \mid x_k\right) = \pi\left(x_{k+1}\right) p\left(x_k \mid x_{k+1}\right) \tag{5}$$

If $p(\cdot)$ conforms to a symmetric proposal distribution, such as a Gaussian distribution, then both x_{k} and $x_{\mathrm{k+1}}$ follow the target distribution. The transition probabilities between adjacent states are the same, and the Markov chain can converge to the steady-state distribution of the probability density function $\pi(\cdot)$.

In SS, when sampling, it is necessary to choose an appropriate proposal distribution $p(\cdot)$ to ensure the stability of the Markov chain (Zhang and Ma, 2024). The Metropolis-Hastings (MH) algorithm is the most widely used MCMC algorithm, but because it samples from a vector perspective, its sampling efficiency is relatively low. Therefore, the improved MMH algorithm is adopted. This algorithm is based on the idea of random walk and samples each component of the random variable's dimensions. It addresses the issues of low sampling efficiency for high-dimensional random variables and the inability to traverse the entire sample space, thus improving the Markov chain generation efficiency and showing good adaptability in SS.

The MMH algorithm samples from the one-dimensional component $x_k^j (j = 1, 2, \ldots, n)$ of the current state sample, where the corresponding one-dimensional proposal distribution's probability density function is $q\left(\cdot \mid x_k^j\right)$. The generated candidate sample is ε_k^j, and the acceptance rate $\alpha\left(x_k^j, \varepsilon_k^j\right)$ of this candidate sample can be expressed as:

$$\alpha\left(x_k^j, \epsilon_k^j\right) = \min\left[1, \frac{q\left(x_k^j \mid \epsilon_k^j\right)\pi\left(\epsilon_k^j\right)}{q\left(\epsilon_k^j \mid x_k^j\right)\pi\left(x_k^j\right)}\right] \tag{6}$$

Random number $u \sim U(0,1)$. If $u < \alpha\left(x_k^j, \varepsilon_k^j\right)$, accept the candidate sample ε_k^j as the next state sample component x_{k+1}^j; otherwise, reject the sample and continue with x_k^j as the next state sample.

$$x_{k+1}^j = \begin{cases} \varepsilon_k^j, & \alpha\left(x_k^j, \varepsilon_k^j\right) > u \\ x_k^j, & \alpha\left(x_k^j, \varepsilon_k^j\right) \leqslant u \end{cases} \tag{7}$$

This paper adopts the improved MMH algorithm, which randomly samples each dimensional component of the random variable. If the candidate sample does not meet the acceptance rate condition, a new candidate sample is generated until the condition is satisfied. The next state sample of the Markov chain is repeatedly generated until the distribution of intermediate states in each subset is traversed.

2.3 CNN-LSTM-SubSim Reliability Analysis

SS is an efficient structural reliability analysis method that offers significant advantages in solving extremely small failure probability problems, particularly in the reliability assessment of large engineering structures such as bridges and high-rise buildings. However, traditional methods have obvious limitations in computational efficiency, especially when dealing with high-dimensional nonlinear problems. They often require extensive finite element calculations, leading to prohibitively high computational costs. Existing LSTM and its improved algorithms, along with other surrogate models, can effectively replace finite element calculations, significantly improving computational efficiency. However, fully data-driven machine learning models often lack physical constraints, and the predicted results may be overfitted. Therefore, it is necessary to incorporate physical features into the machine learning training process.

To address the above issues, this paper proposes an improved SS method based on CNN-LSTM. Starting from the basic principles of SS, the method constructs a surrogate model to enable fast prediction of responses and improve the reliability prediction of small probability events. The process includes the following steps: First, multiple structural responses are obtained based on finite element analysis. The load and structural frequency information are combined and input into the CNN for dimensionality reduction and feature extraction. Then, the LSTM is used to train and validate the CNN-LSTM surrogate model to predict the response. Subsequently, within the SS framework, this surrogate model replaces the original finite element model, enabling rapid evaluation of the responses of all subset samples and determining the performance function. This method significantly reduces the computational cost per simulation while retaining the advantage of SS in handling low failure probabilities, and the predicted responses are constrained by physical features. The specific process is as follows:

An engineering model is established using FEA software, and multiple dynamic response analyses are performed under different random conditions. The load and structural frequency information are then combined and input into the CNN for dimensionality reduction and feature extraction. Subsequently, the LSTM is used to train and validate the CNN-LSTM surrogate model to predict the response.

Use MC sampling to extract N sample points as the initial subset.

Input the initial subset into the response calculation program and predict the response of all samples using the surrogate model.

Use the MCMC method to continue generating N new samples that conform to the failure domain distribution, forming a new subset, and continue step 3 until $g(X) \leq 0$ is satisfied, obtaining the reliability. The specific process is shown in Figure 2.

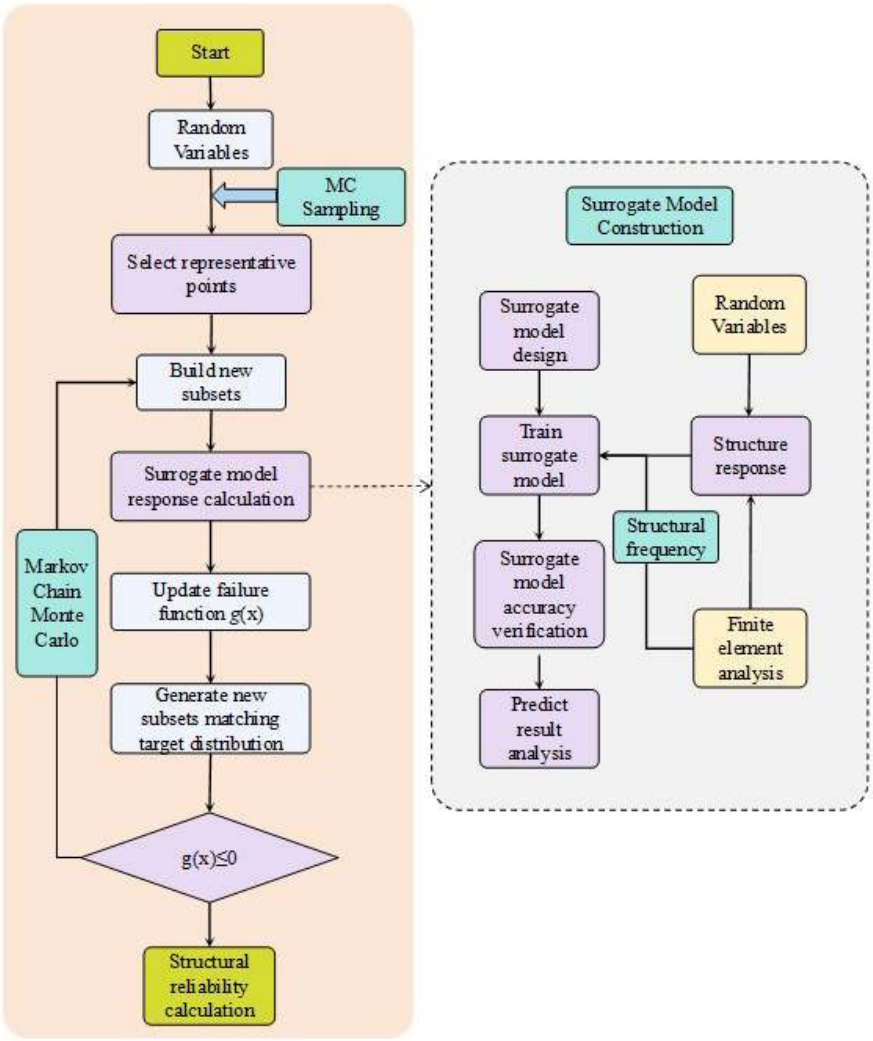


Figure 2. Flowchart of SS-CNN-LSTM

3 Response Prediction Based on CNN-LSTM

3.1 CNN

CNN are a class of deep learning models specifically designed for processing data with grid-like structures, such as images or time-series signals. The core idea of CNNs is to efficiently extract spatial features through local connections and weight sharing. A complete CNN architecture typically consists of several functional modules: the input layer, which receives the raw data (such as images or time-series signals); the convolutional layers, which perform feature mapping using a set of filters; the pooling layers, which reduce the dimensionality of the features; the fully connected layers, which integrate high-level features; and the output layer, which generates the final prediction results. This hierarchical structure enables CNNs to automatically learn abstract feature representations from low-level to high-level features. During feature extraction, the convolutional layers process the input data using learnable convolution kernels (also known as filters). The processing procedure is as follows:

$$y_j^k = f\left(k\sum_{i\in C_j} x_i^{k-1} * u_{ij}^k + b_j^k \right) \tag{8}$$

In the equation, x_i^{k-1} represents the activation value of the i-th feature map in the $k-1$-th layer, reflecting the feature response pattern of the previous layer; u_{ij}^k is the learnable convolution kernel connecting the feature maps i and j of adjacent layers; b_j^k is the bias term, used to adjust the activation threshold of the feature map; C_j specifies the set of input feature maps involved in the calculation of the current feature map; $*$ is the operator representing the convolution operation, which extracts local features through a sliding window approach; $f(\bullet)$ is the nonlinear activation function, typically chosen as the ReLU function to mitigate the vanishing gradient problem, and its expression is $f(X) = \max(0, X)$.

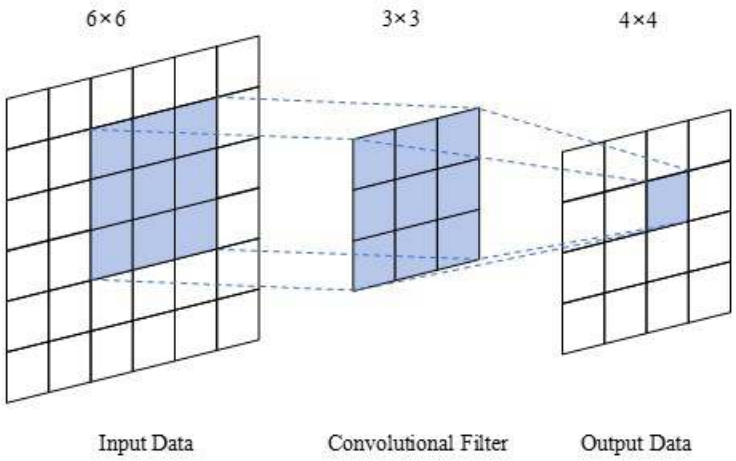


Figure 3. Convolution Operation Diagram

Figure 3 illustrates the convolution operation, with blue dashed arrows connecting the input data matrix to the output data matrix, showing the process of the convolution kernel sliding over the input data and performing convolution calculations. Specifically, a 3×3 convolution kernel slides over a 6×6 input data matrix with a certain stride, covering a 3×3 area each time. The convolution operation generates one element in the corresponding output data matrix, ultimately producing a 4×4 output matrix.

The pooling layer, as an essential component of a convolutional neural network, performs the core function of spatial dimension reduction on the feature maps output by the convolutional layer through downsampling. This layer uses a sliding window mechanism similar to the convolutional layer, stepping across the feature map with a fixed stride. In max pooling, the maximum value within each sliding window is retained, and this selective retention mechanism effectively highlights significant features while suppressing noise interference. Through pooling, the spatial resolution of the feature map is reduced, which not only decreases the computational load for subsequent layers of the network but also enhances the model's robustness to small shifts in the input data. Additionally, by reducing the dimensionality of intermediate features, the pooling layer helps control the size of the model's parameters, thereby lowering the risk of overfitting.

3.2 LSTM

LSTM networks were proposed to address the long-term dependency problem in traditional RNNs and have demonstrated excellent performance in time-series modeling. The core innovation of LSTM lies in its carefully designed memory cell structure and gating mechanism, which work together to selectively retain important historical information while filtering out irrelevant noise. Unlike the standard RNN unit, which only includes a simple tanh activation layer, the LSTM unit introduces three gating structures (input gate, forget gate, output gate) and a cell state, forming the internal structure shown in Figure 4.

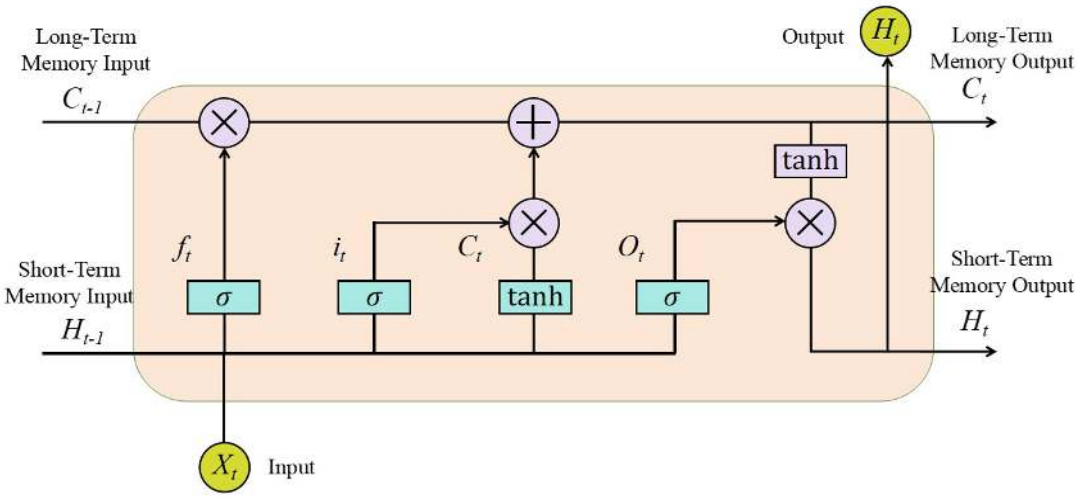


Figure 4. Internal Structure of LSTM Unit

The dynamic update process of the LSTM network follows the following gating computation rules[19]:

$$g_{\mathrm{i}} = \sigma\left(W_{\mathrm{i}}\left[H_{\mathrm{t-1}}x_t\right] + b_{\mathrm{i}}\right) \tag{9}$$

$$g_{\mathrm{f}} = \sigma\left(W_{\mathrm{f}}\left[H_{\mathrm{t-1}}x_t\right] + b_{\mathrm{f}}\right) \tag{10}$$

$$g_{\mathrm{o}} = \sigma\left(W_{\mathrm{o}}\left[H_{\mathrm{t-1}}X_t\right] + b_{\mathrm{o}}\right) \tag{11}$$

$$\tilde{C}_t = \tanh\left(W_{\mathrm{c}}\left[H_{\mathrm{t-1}}X_t\right] + b_{\mathrm{c}}\right) \tag{12}$$

$$C_t = g_{\mathrm{i}}^{*}\tilde{C}_t + g_{\mathrm{f}}^{*}C_{\mathrm{t-1}} \tag{13}$$

$$H_t = g_{\mathrm{o}}^{*}\tanh C_t \tag{14}$$

In the above equation, the variables are defined as follows: X_t represents the input feature vector at time step t; H_t denotes the hidden state output at time step t; C_t records the cell memory state at time step t. The network contains three key gating structures: input gate g_i, forget gate g_f, and output gate g_o, which are parameterized by the corresponding weight matrices W_i, W_f, W_o, and bias terms b_i, b_f, b_o. The memory unit update process is achieved through the weight matrices W_c and bias terms b_c, with all gating units employing the sigmoid activation function σ for nonlinear transformation. At each time step t, the system inputs the current input X_t, the previous time step's hidden state H_{t-1}, and memory state C_{t-1} into the gating system. Through the coordinated computation of gates g_i, g_f, and g_o, the network dynamically generates the hidden state H_t and the updated memory state C_t at the current time step. These state variables are passed as temporal memory carriers to the next time step, with all weight matrices and state vectors continually updated, allowing the network to adaptively learn the long-term dependencies in sequence data. This architecture ensures that the LSTM can retain important historical information while effectively filtering out irrelevant noise in time-series modeling tasks.

3.3 CNN-LSTM Prediction Response Model

This study uses a CNN-LSTM model with physical information constraints for response prediction. The overall architecture is shown in Figure 5.

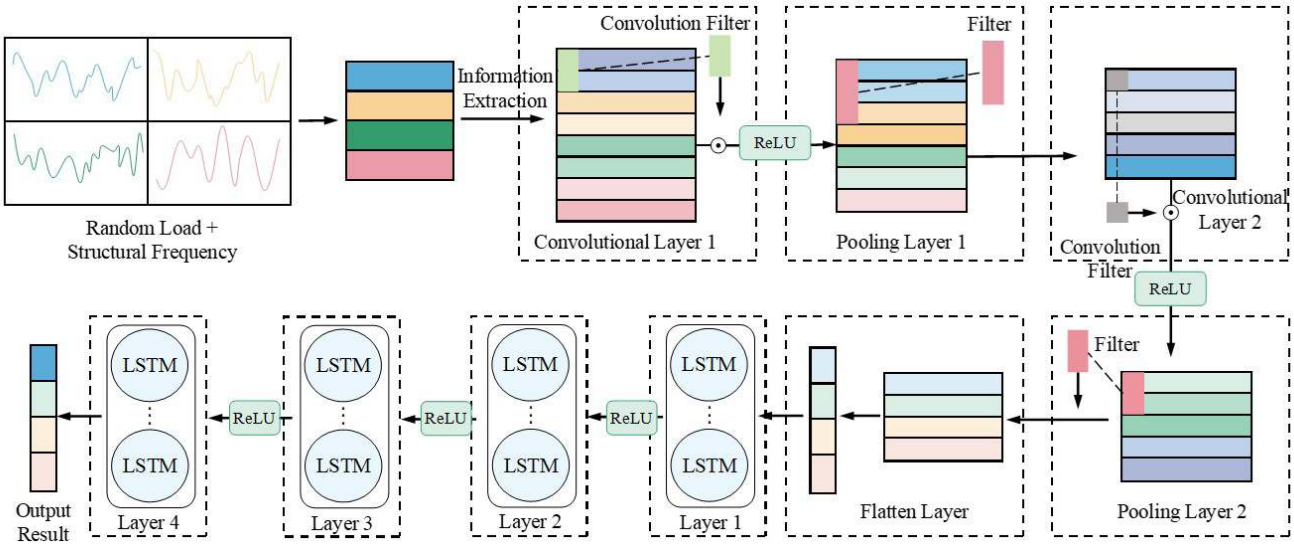


Figure 5. CNN-LSTM Response Prediction

The model first uses the CNN branch to extract local spatial features from the input sequence by stacking multiple convolutional-pooling layers to build hierarchical feature representations. Then, the extracted feature vectors are input into the LSTM branch for time-series modeling, where the gating mechanism learns the dynamic evolution patterns. A feature fusion layer is placed at the connection between the two branches, using channel concatenation to integrate both spatial and temporal features.

4 Case Study Analysis

This section presents two case studies: one for calculating the reliability of planar truss systems under stochastic harmonic excitation, and another for determining the dynamic reliability of two-story steel frame structures subjected to non-stationary loads. The truss example verifies the effectiveness of the Subset Simulation method for small failure probability estimation, while the steel frame example demonstrates the applicability and accuracy of the proposed physics-informed CNN-LSTM-SS framework for structural reliability analysis.

4.1 Reliability of Truss System Considering System Uncertainty and Random Load

This section uses the reliability analysis of a two-dimensional truss structure under random loading as an example to verify the computational performance and engineering applicability of SS. By establishing a probabilistic model that considers the randomness of material properties, geometric dimensions, and external loads, the SS method is applied to approximate the structure's failure domain layer by layer. The study investigates the method's computational accuracy and efficiency under small failure probabilities (on the order of 10^{-3}). Furthermore, a large-scale MC simulation with a sample size of 10^4 is used as the benchmark solution to compare the results of the two methods and validate the accuracy of the SS approach.

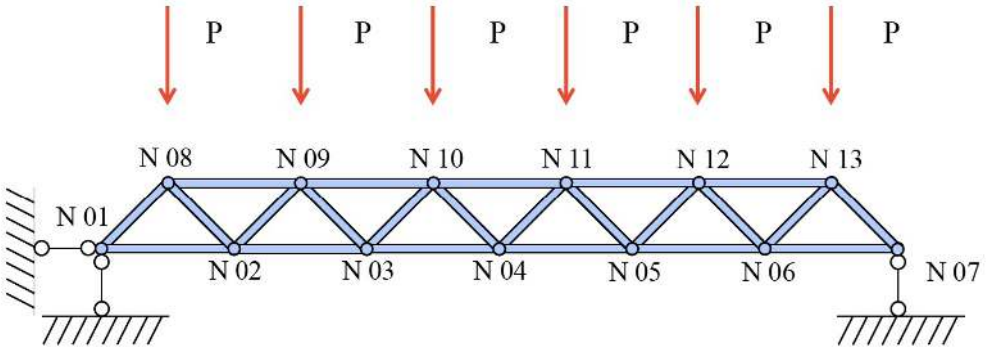


Figure 6. Truss Diagram

Figure 6 shows the schematic of a planar truss system under random loading. This two-dimensional truss structure consists of 13 nodes ($N01-N13$) and 23 members, arranged symmetrically. The bottom horizontal bars (4 meters) and inclined bars (2.828 meters) alternate to form stable triangular units. The top nodes are subjected to 6 vertical downward random loads (P). The performance function is expressed as follows:

$$g_1(x) = f_1(x_1, x_2, x_3) = R_1 - S_1 \tag{15}$$

In the above equation, $g_1(x)$ represents the performance function of the planar truss system under random loading; $f_1(x_1, x_2, x_3)$ refers to the random variables in the system; $R_1 = 0.12\,\mathrm{m}$ is the structural safety threshold; S_1 represent the displacements of the planar truss structure under random loading, considering the structural uncertainties.

In the reliability analysis of the truss structure, both material parameter uncertainties (such as member cross-sectional area and Young's modulus) and load randomness are comprehensively considered, defining a total of 10 random variables. These variables encompass two sources of uncertainty: structural resistance (geometric and material parameters) and external effects (node loads). The distribution types and parameters of these variables are shown in the table below:

As shown in Table 1, for the log-normal distribution variable 1, the mean and standard deviation are 2e-3 and 2e-4, respectively. For the Gumbel distribution, taking variables 5-10 as an example, the location parameter and scale parameter are 5e4 and 7.5e3, respectively.

Table 1 Probability Distribution of Random Variables

Random Variable	Physical Meaning	Probability Distribution	Characteristic Parameter 1	Characteristic Parameter 2
1	Horizontal bar cross-sectional area A1	Log-normal distribution	$\mu = 2e^{-3}$	$\delta = 2e^{-4}$
2	Diagonal bar cross-sectional area A2	Log-normal distribution	$\mu = e^{-3}$	$\delta = e^{-4}$
3	Horizontal bar Young's modulus E1	Log-normal distribution	$\mu = 2.1e^{11}$	$\delta = 2.1e^{10}$
4	Diagonal bar Young's modulus E2	Log-normal distribution	$\mu = 2.1e^{11}$	$\delta = 2.1e^{10}$
5-10	Vertical load at nodes F	Gumbel distribution	$\mu = 5e^{4}$	$\beta = 7.5e^{3}$

In the reliability analysis of the truss structure, the structural member system is clearly divided into two categories: horizontal members and inclined members. The mechanical properties of the horizontal members are determined by two random material parameters: cross-sectional area (A_1) and Young's modulus (E_1), while the inclined members are modeled using the same random parameters (i.e., cross-sectional area (A_2) and Young's modulus (E_2)).

MATLAB code is used to perform the finite element analysis of a two-dimensional planar truss structure, which is applied to calculate the maximum displacement of the structure under given loads. The finite element analysis of the truss structure is based on the linear static equilibrium equations, with the core being the solution of the displacement-force equilibrium equation:

$$\mathrm{Ku} = \mathrm{F} \tag{16}$$

Where: K is the global stiffness matrix, with dimensions $n_{\mathrm{dof}} \times n_{\mathrm{dof}}$ $\left(n_{\mathrm{dof}} = 2 \times n_{\mathrm{node}}\right)$. u is the displacement vector, which includes the displacements in the x and y directions for all nodes. F is the external force vector, which includes the external forces in the x and y directions for all nodes.

For each truss element eee, the local stiffness matrix in its local coordinate system is given by:

$$\mathrm{K}_e^{\mathrm{local}} = \frac{EA}{L}\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{17}$$

Where: E is the Young's modulus, A is the cross-sectional area, L is the length of the element.

Since the truss element may be inclined, the local stiffness matrix needs to be transformed into the global coordinate system. The coordinate transformation formula is as follows:

$$\mathrm{K}_e^{\mathrm{global}} = \mathrm{T}_e^T \, \mathrm{K}_e^{\mathrm{local}} \, \mathrm{T}_e \tag{18}$$

Where the coordinate transformation matrix T_e is expressed as:

$$\mathrm{T}_e = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & \cos\theta & \sin\theta \\ 0 & 0 & -\sin\theta & \cos\theta \end{bmatrix} \tag{19}$$

In the above equation, θ is the angle between the truss element and the xxx-axis.

The global stiffness matrix is assembled by combining the stiffness matrices of all the elements according to the following equation:

$$\mathrm{K} = \sum_{e=1}^{n_{\mathrm{elem}}} \mathrm{K}_e^{\mathrm{global}} \tag{20}$$

Where: A is the total number of truss elements (23). Each element's contribution is added to the global stiffness matrix K according to the degree of freedom numbering.

Since the truss structure has fixed hinge supports and roller supports, the displacements for certain degrees of freedom are restricted. The vertical and horizontal displacements of node 1 are constrained, and the vertical displacement of node 7 is restricted. The equilibrium equation is partitioned as follows:

$$\begin{bmatrix} \mathrm{K}_{cc} & \mathrm{K}_{cf} \\ \mathrm{K}_{fc} & \mathrm{K}_{ff} \end{bmatrix} \begin{bmatrix} \mathrm{u}_c \\ \mathrm{u}_f \end{bmatrix} = \begin{bmatrix} \mathrm{F}_c \\ \mathrm{F}_f \end{bmatrix} \tag{21}$$

Where: K_{cc} is the stiffness matrix between the fixed degrees of freedom, K_{ff} is the stiffness matrix between the free degrees of freedom, K_{cf} and K_{fc} are the coupling terms.

Since $\mathrm{u}_c = 0$ (support with no displacement), the equation simplifies to:

$$\mathrm{K}_{ff} \, \mathrm{u}_f = \mathrm{F}_f \tag{22}$$

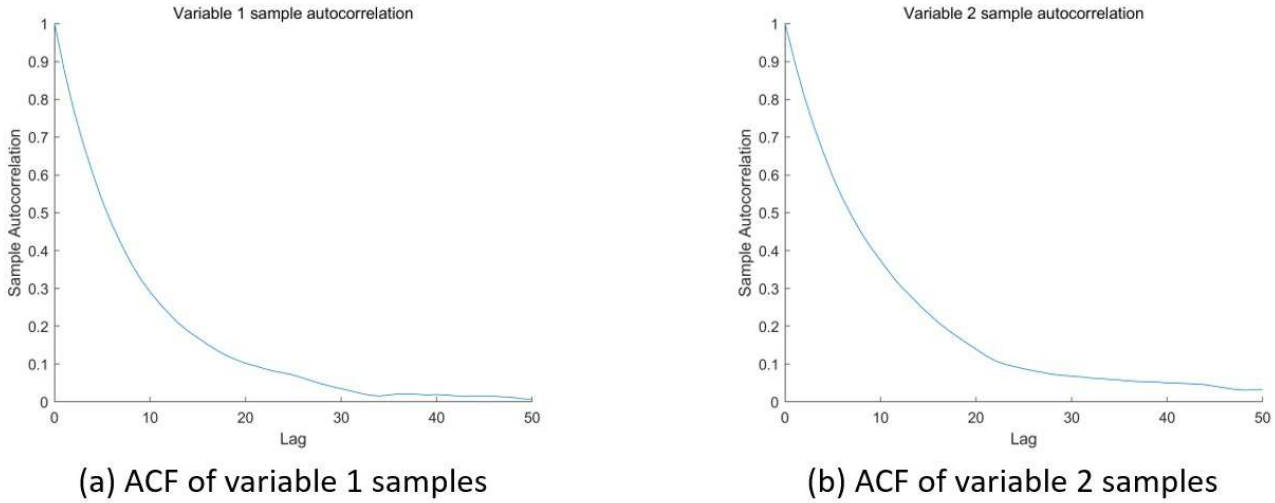
The displacements in different directions at the corresponding nodes, $\mathbf{u}_f$, are solved through the following system of linear equations:

$$\mathbf{u}_f = \mathbf{K}_{ff}^{-1}\mathbf{F}_f \tag{23}$$

Due to the symmetry of the structure and the influence of the load application positions, the vertical displacement of the middle node N04 is the largest, indicating that this node is the key control point for the overall stiffness of the structure. To quantify the structural reliability performance, the vertical displacement of the middle node N04 is selected as the evaluation index, reflecting the overall deformation behavior of the structure under the uncertainty of random material parameters (such as cross-sectional area and Young's modulus) and random loading. This displacement is then used for subsequent reliability assessment.

Before applying the SS method to solve the structural reliability, it is necessary to validate the accuracy of MCMC sampling. This is because the core of the SS method involves progressively approximating the true failure probability through a series of intermediate failure events, and MCMC sampling is the key tool for generating these intermediate failure samples. If there are biases or convergence issues in the MCMC sampling process, it will directly affect the final results of the SS, leading to inaccurate failure probability estimates. Therefore, before proceeding with the reliability analysis, it is essential to validate the implementation of the MCMC method to ensure that it can correctly generate statistically reasonable samples from the conditional probability distribution.

During the MCMC sampling process, neighboring samples often exhibit strong autocorrelation, which can affect the sampling efficiency and statistical independence. By calculating the autocorrelation function (ACF), the degree of correlation between samples can be evaluated. The autocorrelation of samples for variable 1 and variable 2 is shown in Figure 8.



(a) ACF of variable 1 samples (b) ACF of variable 2 samples

Figure 7. Autocorrelation of Random Variable Samples

From the above figure, it can be observed that the autocorrelation of samples for both variable 1 and variable 2 shows a rapid decay trend as the lag steps increase. When the lag step is 0, the sample autocorrelation coefficient is 1, which is the initial state of autocorrelation. As the lag step increases, the autocorrelation coefficient for both variables quickly decreases. Specifically, for variable 1, the autocorrelation coefficient approaches 0 when the lag step reaches about 30, while for variable 2, the autocorrelation coefficient decays to around 0.1 at a lag step of approximately 30, and then gradually approaches 0 as the lag steps continue to increase. This rapid decay characteristic indicates that the samples generated by MCMC sampling are essentially independent after a certain lag step, suggesting that the Markov chain has good mixing and ergodicity, effectively exploring the parameter space and generating representative samples. Therefore, from the perspective of sample autocorrelation, it can be concluded that the MCMC sampling results are accurate.

To quantitatively demonstrate the stability of the MCMC sampling used in Subset Simulation, Table 2 summarizes the key parameters and statistical indicators for both numerical examples, including the number of samples per subset, the conditional probability p_0, the burn-in setting, and the average acceptance rate. A short burn-in of 5 samples per chain is applied, while the initial seeds of the chains are drawn from the conditional distribution. The acceptance rates around 34% indicate satisfactory chain mixing. Together with the ACF plots and subset histograms, these quantitative and qualitative indicators confirm that the MCMC sampling is stable and suitable for the estimation of small failure probabilities in both examples.

Table 2 MCMC sampling parameters and statistical indicators for both numerical examples

Case	Samples per subset	Conditional probability p_0	Burn-in	Average acceptance rate
Truss	1000	0.1	5	33.8%
Steel frame	500	0.1	5	34.2%

A burn-in length of 5 samples was adopted for each Markov chain. This value was selected according to the recommendations of Proppe (Proppe,. 2017), who investigated the influence of different burn-in lengths on MCMC-based reliability algorithms and reported that a short burn-in period of approximately 5 samples provides a good compromise between computational efficiency and sampling accuracy, whereas larger burn-in values (e.g., 20 samples) mainly increase the computational cost. Therefore, a burn-in length of 5 samples was used in the present study to improve chain stability while maintaining computational efficiency. Larger burn-in values were also considered; however, their influence on the estimated failure probability was found to be limited, while the associated computational cost increased. Therefore, a burn-in length of 5 samples was selected as a balanced choice between efficiency and sampling stability.

Furthermore, a probability distribution test was conducted on the generated samples. This was done by comparing the sample probability density with the theoretical distribution values of the variables to verify their accuracy. The probability density histogram for any random variable sample in any subset generated by MCMC was calculated. Meanwhile, the corresponding probability density function curve (such as for Gumbel distribution, log-normal distribution, etc.) was plotted based on the theoretical distribution of the variable, and the two were compared, as shown in the following figure.

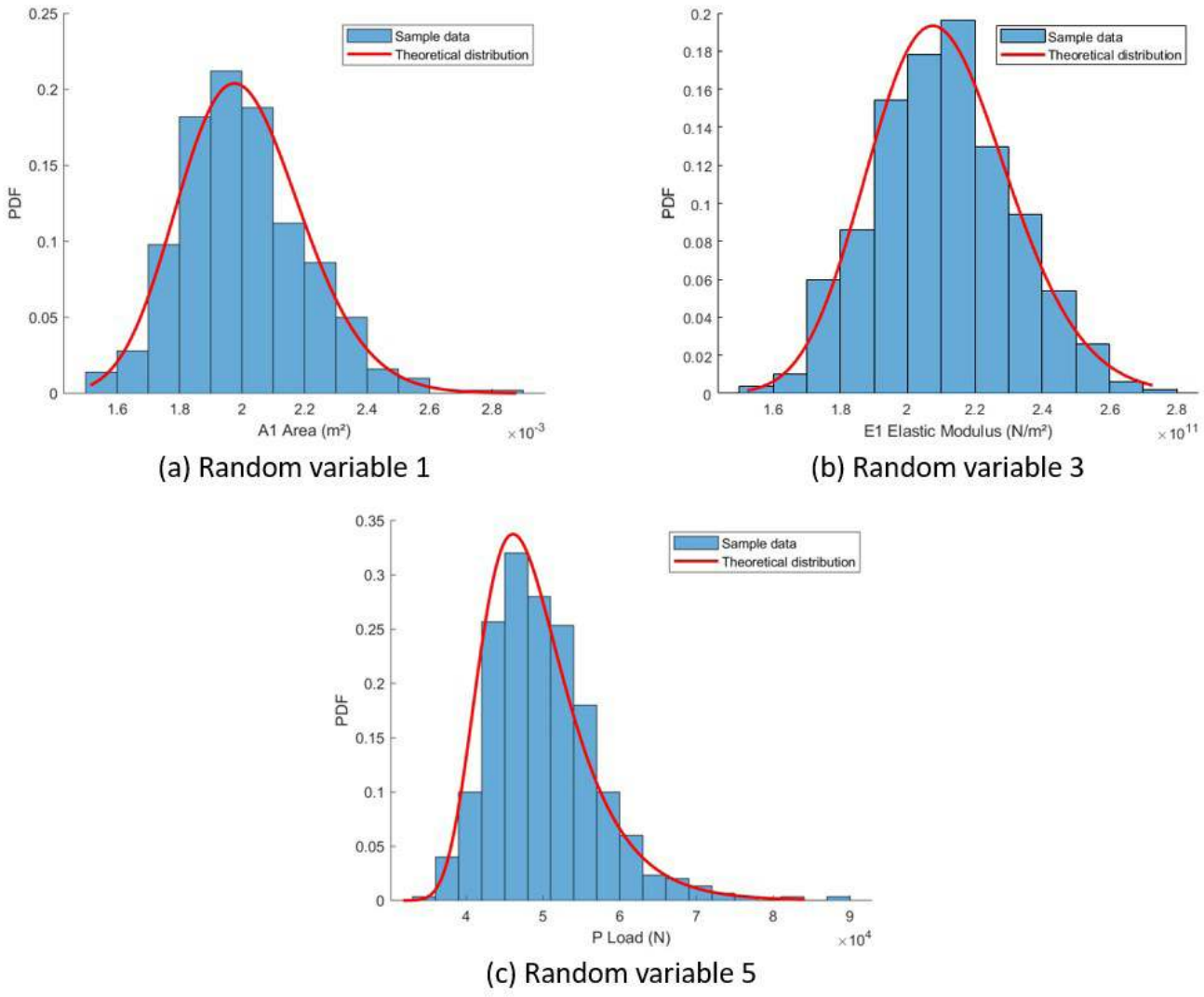


Figure 8. Comparison of Random Variable Probability Distribution with Theoretical Values

Figure 8 shows the comparison between the probability distributions of three random variables—cross-sectional

area A_1 of the horizontal bar, Young's modulus E_1, and load P—against their theoretical distribution curves for the first subset. By conducting a probability distribution test on the MCMC-generated samples, it was found that the sample probability distributions for variables 1, 3, and 5 match well with their theoretical distribution curves. This consistency fully validates that the MCMC sampling method can accurately capture the original statistical characteristics of each variable, providing a reliable data foundation for subsequent reliability calculations based on these samples.

After validating the accuracy of the MCMC samples, the SS method was employed for structural reliability calculation. To ensure the robustness of the results, three independent reliability analyses were performed on the same truss structure, with the final reliability indicator being the average of the three calculations. During the computation, the limit state function for structural failure was set to $g(x) = 0$, and the samples generated by MCMC were used to form subsets, progressively approximating the failure threshold. A total of three subsets were formed during the entire SS process, with the third subset precisely converging to the failure boundary of $g(x) = 0$. Each subset used the MCMC method for sample updates, with strict control over the acceptance probability to ensure the stability of the Markov chain. This hierarchical and progressive computational strategy, combined with repeated averaging, effectively reduced sampling errors, leading to a final reliability indicator with higher statistical significance. The displacement result distributions corresponding to different subsets are shown in the following figure:

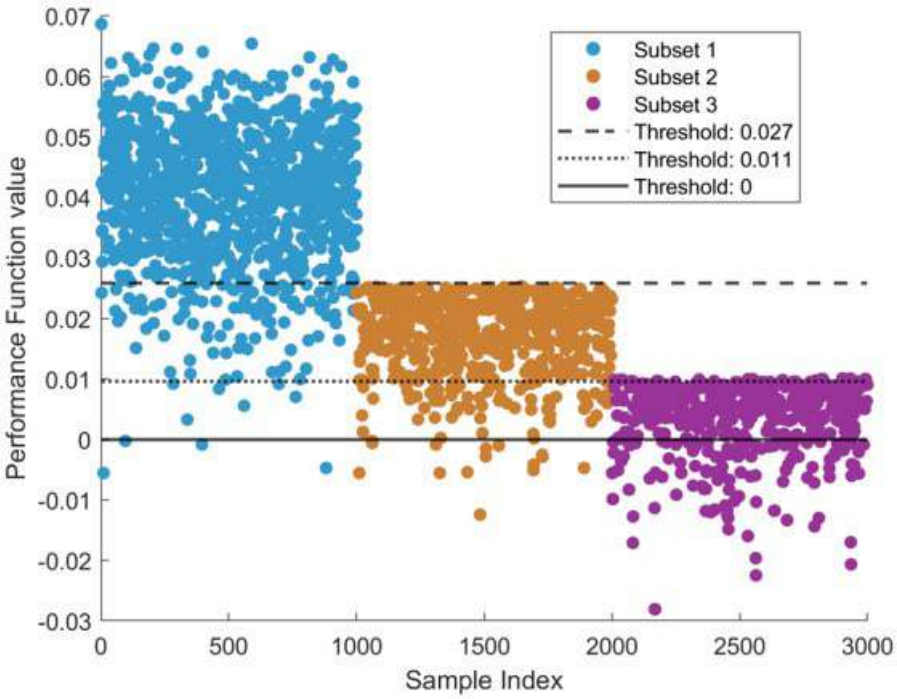


Figure 9. Distribution of the Function Values for Different Subsets

Figure 9 shows the three subsets (Subset 1-3) in the SS method for the truss structure reliability analysis, with each subset containing 1,000 samples. Subset 1 (0-1000) shows that the majority of the samples have a function value $g(x) > 0$, indicating that the structure still has a safe displacement allowance, representing the initial sampling phase. Subset 2 (1000-2000) presents a transitional state, with samples gradually approaching the failure boundary. Subset 3 (2000-3000) converges near the limit state, with some samples having a function value $g(x) \le 0$, indicating structural failure. This figure clearly reflects the process of SS, which searches the small probability failure domain through hierarchical sampling, effectively improving the accuracy and efficiency of structural reliability assessment.

Figure 10 shows the frequency distribution of function values for three subsets (Subset 1-3) in the SS method for the truss structure reliability analysis. Subset 1 (blue): Displays a smooth distribution, reflecting the initial sampling covering the entire safe domain, with a threshold value of 0.027. Subset 2 (orange): The samples shift significantly to the left, forming a peak at $g(x) \approx 0.02$, with a sharp drop in frequency afterwards (corresponding to a threshold of 0.011). Subset 3 (purple): The samples are further concentrated near the failure boundary, with almost no samples after $g(x) > 0.01$. The steep drop to the right of Subsets 2 and 3 is caused by the MCMC local sampling in the SS method. The algorithm gradually removes high-value samples and focuses computational resources on the failure boundary, forming a "narrowing" distribution.

This phenomenon demonstrates the algorithm's efficiency in capturing small probability failure events and further validates its effectiveness. The final failure probability is 0.00163, and the reliability result is 0.99837. The different subsets and failure samples are shown in the table 3 below:

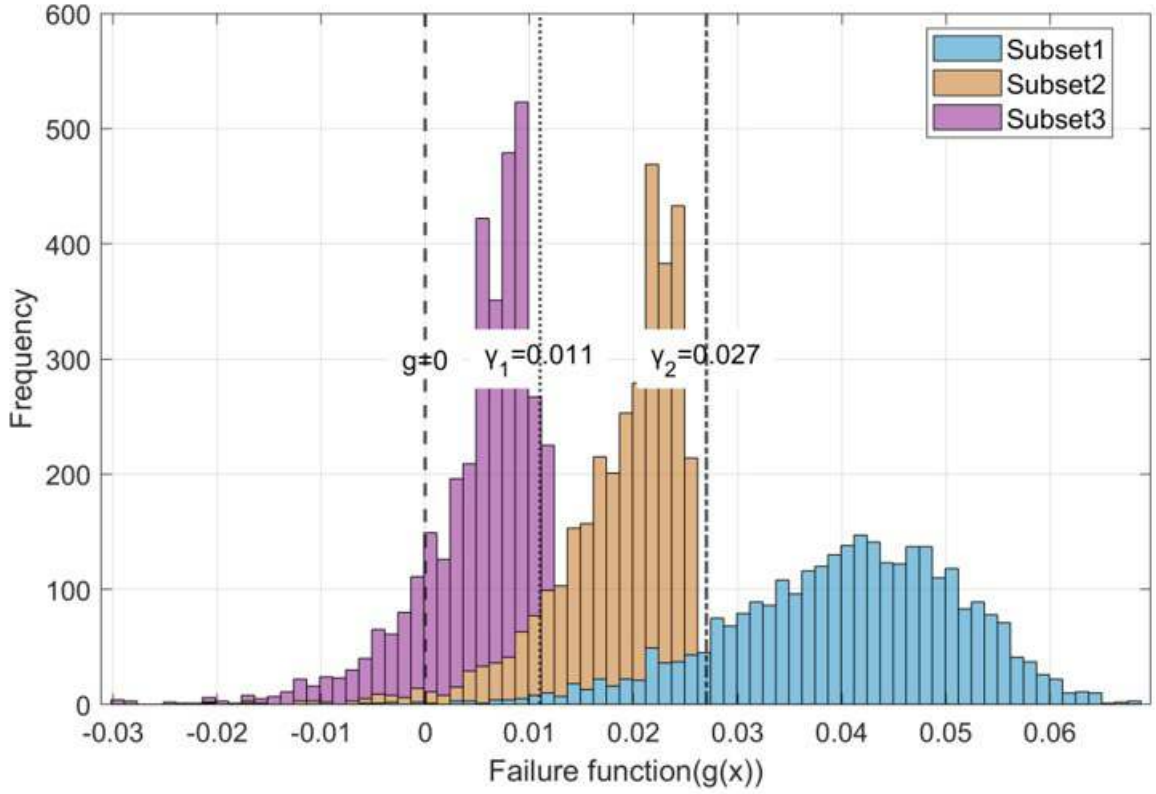


Figure 10. Frequency Distribution of Function Values for Different Subset

Table 3 Different Subsets and Failure Samples

Subset	Threshold	Number of Failure Samples
1	0.027	100
2	0.011	100
3	0	163

To validate the accuracy of the SS method, 50,000 random samples were generated using the MC method for benchmark comparison analysis. The MC method, as a classical reliability analysis method, provides statistically precise results when the sample size is sufficiently large, making it suitable as a benchmark solution for verifying the reliability of the SS method. The specific implementation process is as follows: First, 10 random variables are sampled with the same distribution, and 50,000 independent and identically distributed sample points are generated through direct MC sampling. Then, the displacement of the corresponding sample points is calculated, and the number of samples satisfying the failure condition ($g(x) \le 0$) is counted. The failure probability (Pf) and its coefficient of variation (COV) are calculated. The following figure shows the probability distribution of the reliability function calculated using the MC MC method.

The figure 11 shows the probability density distribution of the function values when calculating the structural reliability using the MC method. The distribution approximately follows a normal distribution, with the peak located at $g(x) \approx 0.04$, where the probability density is about 0.35, indicating that this region is the most likely range for the function value. The red dashed line represents $g(x) = 0$, which serves as the critical boundary between structural safety and failure. When the function value is below $g(x) > 0$, the structure is in a safe state, whereas when it reaches $g(x) \le 0$, the structure is in a failure state. By using MC sampling, the sample proportion within the failure domain $g(x) \le 0$ can be estimated, allowing for the calculation of failure probability and reliability.

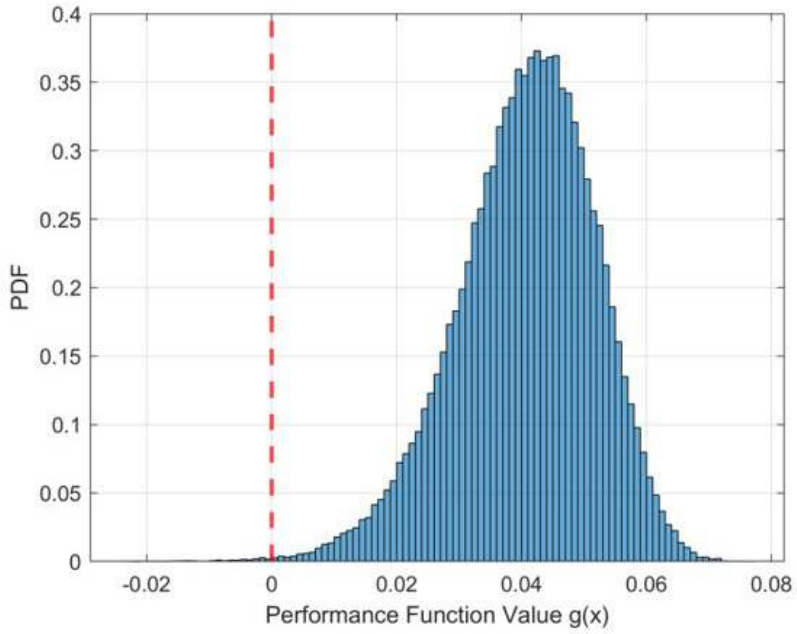


Figure 11. Probability Distribution of MC Function Values

From the figure 12 and table 4, it can be seen that as the sample size increases, the variance of the MCS method decreases, and the results approach the true value. The result curve for 50,000 samples stabilizes, and it closely aligns with the SS results (with a very small error), demonstrating that the SS method, while ensuring computational efficiency, can accurately capture small probability failure events. The results demonstrate that the SS method can accurately estimate small failure probabilities with significantly improved computational efficiency compared with direct Monte Carlo Simulation.

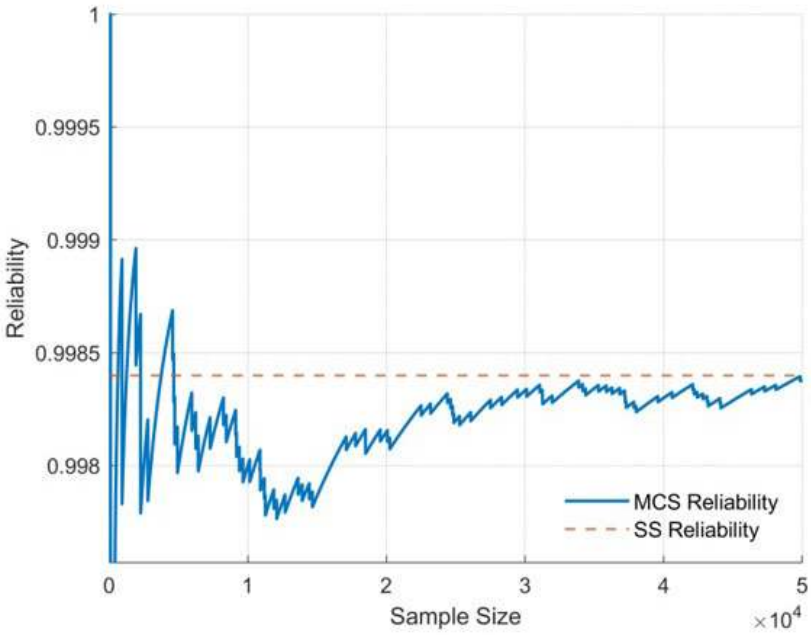


Figure 12. Comparison of MCS and SS Reliability Results

Table 4 Comparison of Reliability Results

Method	Reliability	Error
MCS	0.99846	0
SS	0.99837	9.01×10^{-5}

4.2 Dynamic Reliability of Steel Frame Under Random Loading

In the previous section, the advantages of the SS method in solving reliability were verified through a truss reliability case study. Unlike the first truss example, the present steel frame case explicitly incorporates the CNN-LSTM surrogate model into the Subset Simulation framework and is therefore used to validate the proposed physics-informed CNN-LSTM-SS method. In this section, the dynamic reliability of a two-story steel frame is solved to verify the accuracy and effectiveness of the surrogate model approach. Using finite element software, a two-story steel frame model is established, and the dynamic response is solved under multiple random loading conditions to obtain the response sample set of the system. The CNN-LSTM model is used to train the system's response sample set, establishing the surrogate model for predicting the system's response. The surrogate model then predicts the responses of all samples, enabling the resolution of time history analysis results for the samples, ensuring both accuracy and efficiency.



Figure 13. Steel Frame Model Diagram

The figure 13 illustrates a bidirectional rigidly connected portal frame, with two spans arranged continuously in the long direction, each spanning 6 meters, and evenly spaced columns. In the short direction, a single span is arranged with a span of 6 meters. The story height is 4 meters for the first floor and 3.3 meters for the second floor.

The model was created using ABAQUS finite element software, where the beams, columns, and vertical beams are modeled using beam elements. The cross-section of the beams (both horizontal and vertical) is I-shaped, while the column cross-section is rectangular. The specific dimensions of the cross-sections are shown in the figure 14.

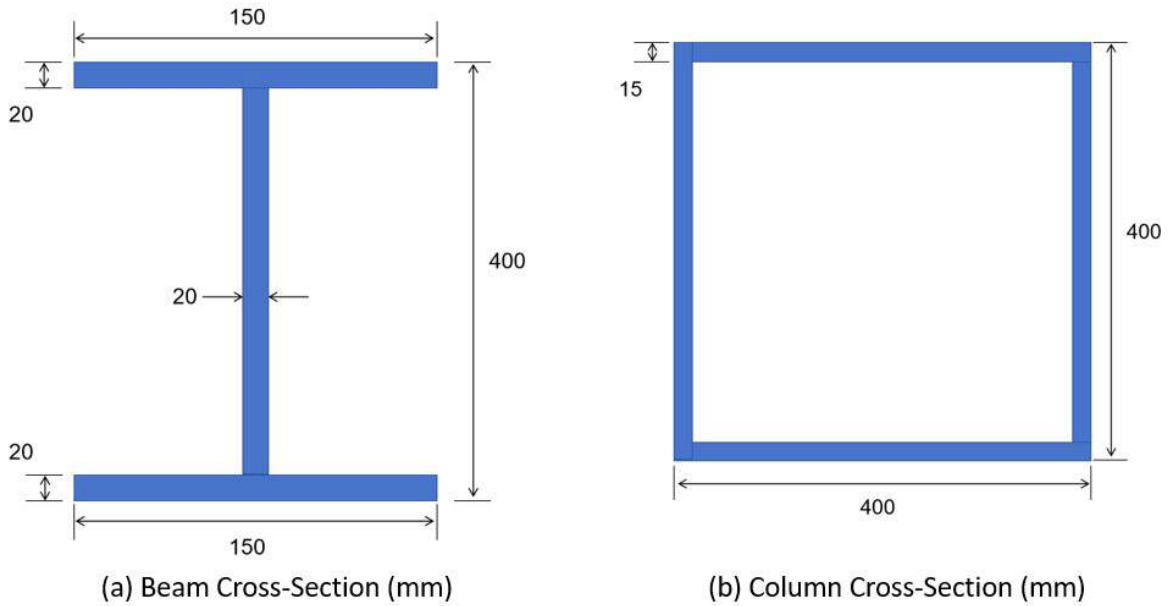


Figure 14. Steel Frame Model Diagram

The beams and columns are made of Q345 steel, with the following material parameters: Young's Modulus $E = 2.06 \times 10^5$ MPa, Poisson's Ratio $v = 0.3$, Yield Strength $f_y = 345\,\text{MPa}$. The column bases are fixed with constraints, fully restricting their translational and rotational degrees of freedom to simulate rigidly connected foundations in actual engineering. To account for geometric nonlinear effects, the large deformation option (NLGEOM) is enabled during the analysis. The meshing is controlled globally, with element sizes set to 600mm, and local mesh refinement is applied at the node connections to ensure accurate stress transfer.

In the ABAQUS finite element model, the dynamic analysis is divided into two steps. The first step is a static general analysis, simulating the initial equilibrium state of the structure under its own weight. At the same time, geometric nonlinearity (NLGEOM) is enabled to consider possible nonlinear effects, and damping parameters (Rayleigh damping coefficients $\alpha = 0.05$ and $\beta = 0.01$) are set to improve numerical convergence. The second step is an implicit dynamic analysis, simulating the dynamic response of the structure under lateral random dynamic loading. The lateral dynamic load is applied as a concentrated force at each beam-column node, with a duration of 50 seconds. The specific formula for the random dynamic load is as follows:

$$\begin{aligned} F(t) &= a_0 \cdot A(t) \cdot \sin(t) \\ A(t) &= \left(\sum_{k=1}^{6} \frac{e^{-\alpha t_o |t - t_k|}}{3} \right) \cdot \left(1 - \beta + 2\beta \sin\left(2\pi f_m t\right)\right) \end{aligned} \tag{24}$$

The above formula represents the lateral dynamic load, where $A(t)$ is the modulation function, f_m is the modulation frequency, 0.05 Hz. a_0, α, t_k, t_0, and β are random variables. Their distribution types and corresponding parameters are as follows:

As shown in Table 5, for the log-normal distribution variable a_0, the mean and standard deviation are 9e3 and 9e-3, respectively. For the Gamma distribution, taking variable t_0 as an example, its mean and variance are 1 and 0.5, respectively. The performance function expression of the steel frame under lateral random dynamic loading is as follows:

Table 5 Probability Distribution of Random Variables

Random Variables	Probability Distribution	Characteristic Parameter 1	Characteristic Parameter 2
a_0	Normal Distribution	$\mu = 8e^4$	$\delta = 10.5e^3$
α	Normal Distribution	$\mu = 0.8$	$\delta = 0.08$
t_k	Log-Normal Distribution	$\mu = 1.2$	$\delta = 0.5$
t_0	Gamma Distribution	$\mu = 1$	$\delta = 0.5$
β	Exponential Distribution	$\mu = 0.5$	$\lambda = 2$

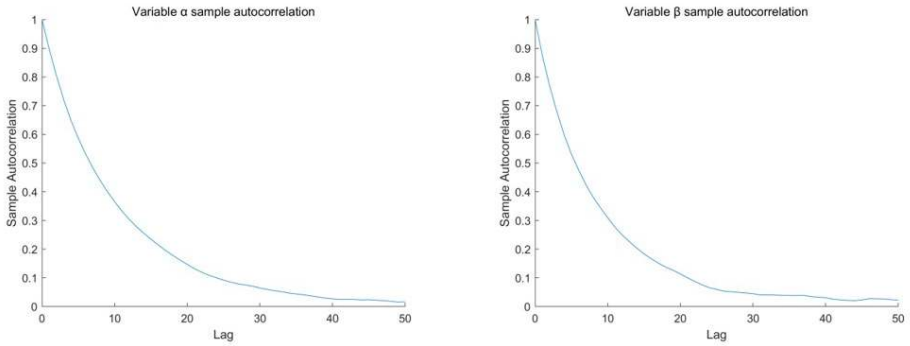
$$g_2(x) = f_2(x_1, x_2, x_3) = R_2 - S_2 \tag{25}$$

In the above equation, $g_2(x)$ is the performance function of the steel frame system under random loading, $f_2(x_1, x_2, x_3)$ represents the random variables in the system, $R_2 = 0.16\,\text{m}$ is the structural safety threshold, S_2 is the lateral displacement of the steel frame structure under random loading. Next, the proposed LSTM-SS method will be used to solve for the reliability, with the finite element—SS method serving as the benchmark solution. The results will also be compared with those obtained using the surrogate model—SS method to verify the efficiency and accuracy of the proposed method.

4.2.1 Finite Element-SS Method for Reliability Calculation

The high-precision benchmark solution is obtained based on the finite element-SS method, which directly couples the finite element model with the SS algorithm. Although the computational cost is high, the results can serve as the benchmark solution for reliability analysis. Each sample in the subsets is input into the finite element model to compute its response. A dynamic analysis steel frame model is established using ABAQUS, generating an inp format file. The system command is used to control ABAQUS and transfer loads. Results are read using the MATLAB function get_history_output.m and the Python script odbHistoryOutput.py.

Before performing reliability calculation using the finite element-SS method, the accuracy of MCMC sampling must first be validated. During the MCMC sampling process, neighboring samples often exhibit strong autocorrelation, which can affect sampling effectiveness and statistical independence. By calculating the autocorrelation function (ACF), the degree of correlation between the samples can be assessed. The autocorrelation of samples for variable A and variable B is shown in Figure 15.

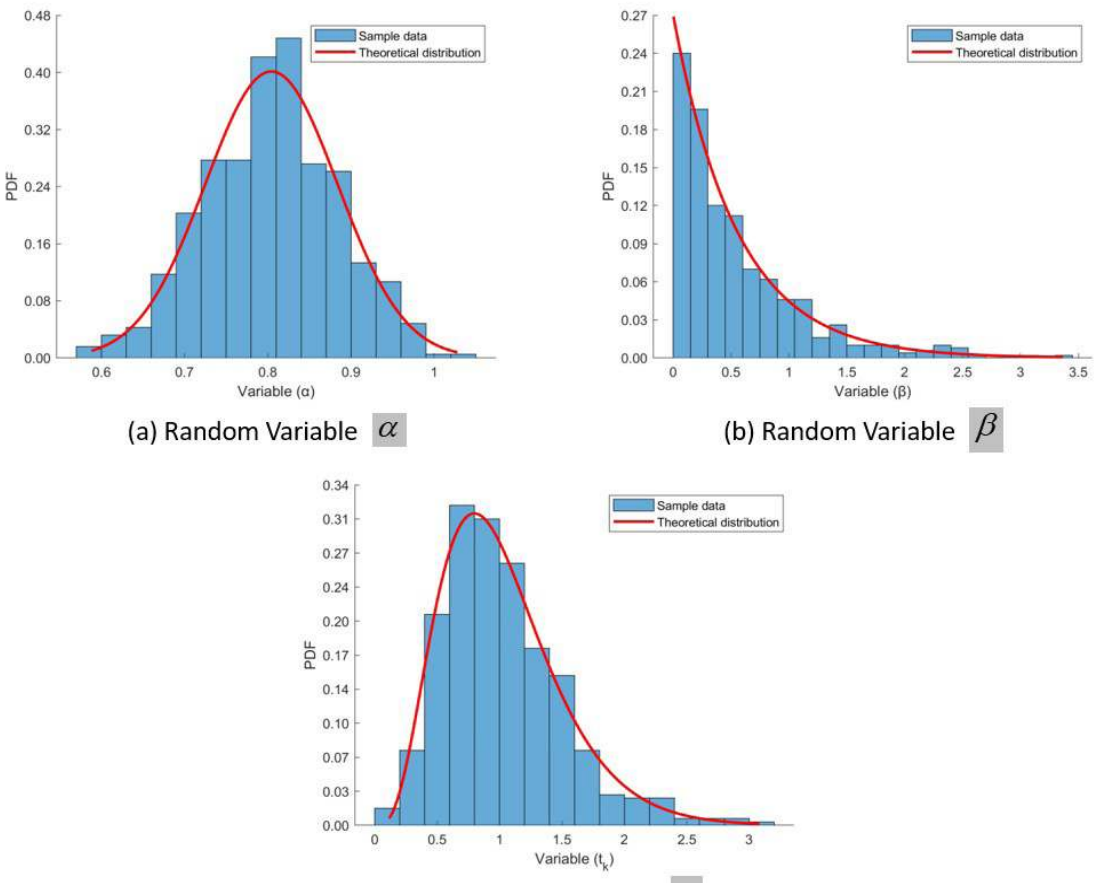


(a) The ACF of Variable Sample α (b)The ACF of Variable Sample β

Figure 15. Autocorrelation of Random Variable Samples

From the above figure, it can be seen that the autocorrelation of samples for both variable α and variable β rapidly decays as the lag step increases. When the lag step is 0, the sample autocorrelation coefficient is 1. As the lag step increases, the autocorrelation coefficients for both variables quickly decrease. Specifically, the autocorrelation coefficient for variable β approaches 0 when the lag step reaches approximately 30, while for variable α, the autocorrelation coefficient decays close to 0 when the lag step reaches around 35. Therefore, from the perspective of sample autocorrelation, it can be concluded that the MCMC sampling results are accurate.

After completing the autocorrelation test of the MCMC-generated samples, the independence of the Markov chain samples was verified. Additionally, a probability distribution test was further conducted on the generated samples. This involved comparing the sample probability density with the theoretical distribution values of the variables to verify their accuracy. The probability density histogram for any random variable sample in any subset generated by MCMC was calculated. Simultaneously, the corresponding probability density function curve was plotted based on the theoretical distribution of the variable, and the two were compared, as shown in the following figure:



(a) Random Variable α (b) Random Variable β

(c) Random Variable t_k

Figure 16. Comparison of Random Variable Probability Distribution with Theoretical Values

Figure 16 shows the comparison of the probability distributions of three random variables—load variables α, β, and t_k—with their theoretical distribution curves for the first subset. By performing a probability distribution test on the MCMC-generated samples, it was found that the sample probability distributions of variables α, β, and t_k align closely with their theoretical distribution curves. This consistency fully validates that the MCMC sampling method can accurately capture the original statistical characteristics of each variable, providing a reliable data foundation for subsequent reliability calculations based on these samples.

After validating the accuracy of the MCMC samples, the SS method was used to perform the structural reliability calculation. During the calculation, the limit state function for structural failure was set to $g(x)=0$, and subsets were generated from MCMC samples, progressively approximating the failure threshold. A total of two subsets were formed during the entire SS process, with the second subset precisely converging to the failure boundary. Each subset used the MCMC method for sample updates, and the acceptance probability was strictly controlled to ensure the stability of the Markov chain. The displacement result distributions corresponding to different subsets are shown in the following figure:

Figure 17 illustrates the progressive sampling process in the reliability analysis of the truss structure using the SS method, revealing the process of the structure transitioning from the safe state to the failure boundary through two consecutive subsets (Subset 1-2). Each subset contains 500 samples. Subset 1 (0-500) is generated using MC sampling, resulting in a more dispersed sample distribution, with some samples crossing the failure threshold. Subset 2 (500-1000) converges to the limit state, with more samples entering the failure domain, and the function values reaching $g(x)\approx -0.1$ within the failure domain. All sample responses in these subsets are obtained through ABAQUS model analysis, considering structural nonlinearity, which ensures accurate results. However, the computational cost is high, with a single analysis taking about 3 minutes.

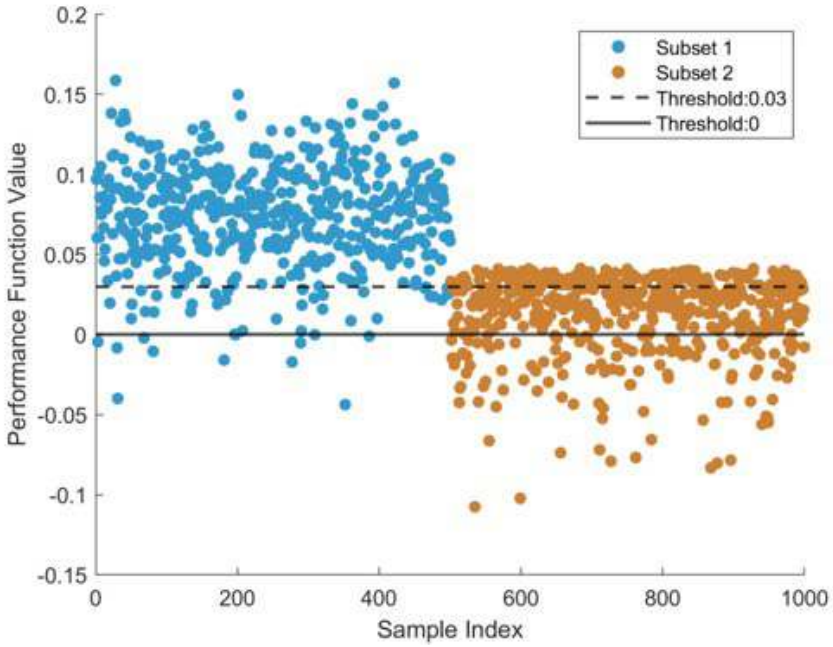


Figure 17. Distribution of Function Values for Different Subsets

Figure 18 shows the frequency distribution of the function values for two subsets (Subset 1-2) in the truss structure reliability analysis using the SS method. Subset 1 (blue): The frequency distribution is relatively flat, covering the function value range of -0.05 to 0.16, reflecting the initial sampling phase that broadly covers the safe domain, with the threshold set at 0.03. Subset 2 (orange): The sample distribution shifts to the left, with a peak forming at $g(x)\approx 0.03$, and a sharp drop in frequency afterward at $g(x)>0.03$ (corresponding to the threshold). This indicates that high-value samples have been removed. Some samples are concentrated around $g(x)=0$, and a few samples enter the negative value range ($g(x)<0$), representing the final convergence to the critical failure state.The steep drop on the right side of Subset 2 is the result of the SS method gradually removing high-value samples through MCMC and optimizing the allocation of computational resources, forming a “narrowing” distribution. This process efficiently captures rare failure events, with the final failure probability being 0.0224, validating the effectiveness of the algorithm. The reliability calculation result is 0.9776. The different subsets and the number of finite element model calls are shown in the table 6 below:

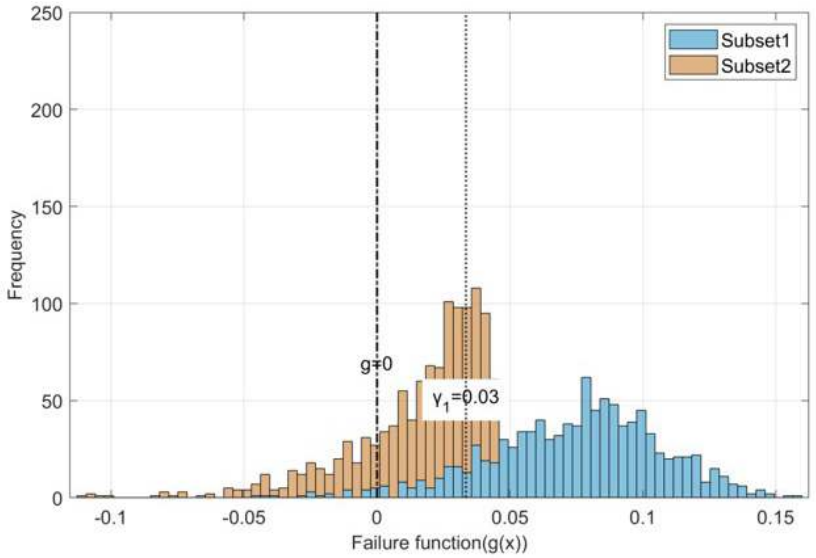


Figure 18. Distribution of Function Values for Different Subsets

Table 6 Failure Samples and FEM Calls for Different Subsets

Subset	Threshold	Number of Failure Samples	FEM Calls
1	0.03	50	500
2	0	112	500

4.2.2 Surrogate Model-SS Method for Reliability Calculation

In the surrogate model-based SS method, an efficient surrogate model is constructed to approximate the response, improving the efficiency of reliability analysis. Initially, the surrogate model is trained using the initial samples, and then it is applied to various levels of the SS to quickly predict the response values for a large number of samples, avoiding repeated calls to the finite element model.

Before performing the surrogate model-SS method for structural reliability calculation, the Latin Hypercube Sampling (LHS) method is used to generate 400 sample sets. This method ensures uniform coverage of each random variable within its probability distribution space through stratified sampling. After calculating the corresponding lateral loads for the 400 sample sets, the samples are input into finite element analysis software for batch computation, obtaining the lateral load-displacement response curve for each sample. The lateral displacement of the steel frame under lateral loading for any sample is shown in the figure 19:

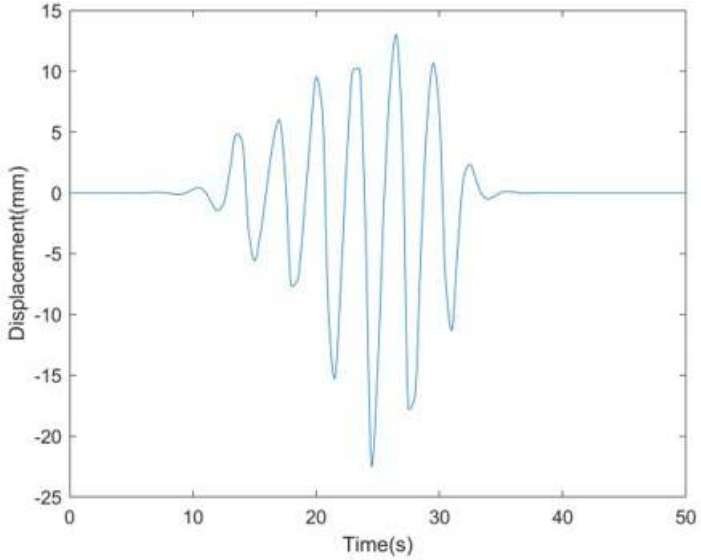


Figure 19. Steel Frame Displacement Response Diagram

After obtaining the displacement response data corresponding to different operating conditions and their parameters, a high-precision surrogate model is built using a CNN-LSTM hybrid neural network. The model's input layer is designed as a dual-branch structure: the CNN branch extracts the spatial features of the operating condition parameters (such as the correlation of load distribution), while the LSTM branch (including Gated Recurrent Units, GRU) handles the temporal dependencies of the displacement response. The first eight natural frequencies of the model are extracted and input into the CNN along with the load for feature extraction. The extracted features are then used for model training with LSTM. Table 7 shows the first eight natural frequencies of the model. In contrast, the traditional LSTM directly uses the load for model training.

Table 7 The first eight natural frequencies of the structure

Modal order	Frequency /Hz
1	1.52
2	4.51
3	5.07
4	8.96
5	14.48
6	22.59
7	27.58
8	42.27

The detailed architecture and training settings of the proposed CNN-LSTM surrogate model are summarized in Table 8 to improve reproducibility. The CNN module is mainly employed to extract and fuse the physical features contained in the random load parameters and structural natural frequencies, while the LSTM module is used to capture the temporal dependence of the structural dynamic response. Before model training, all input and output data are standardized using Z-score normalization. The model is trained using the Adam optimizer with the mean squared error loss function, and the validation dataset is used to monitor the convergence behavior during the training process. The detailed architecture and training settings of the proposed CNN-LSTM surrogate model are summarized in Table 8 to improve reproducibility.

Table 8 Configuration of the CNN-LSTM surrogate model

Item	Setting
Training samples	400 LHS samples
Dataset split	80% training / 20% validation
Input information	Random load parameters and first eight natural frequencies
Output	Lateral displacement time-history response
CNN layers	2 convolutional layers
Kernel size	3
Stride	1
Activation function	ReLU
Pooling	Max pooling, pool size=2
LSTM layer	1 LSTM layer
LSTM hidden units	96
Sequence length	50s response history
Fully connected layer	1 regression output layer
Loss function	Mean squared error
Optimizer	Adam
Initial learning rate	0.005
Batch size	16
Epochs	175 epochs
Random seed	Fixed for reproducibility
Software	MATLAB R2025a/ABAQUS

The final hyperparameters of the CNN-LSTM model were determined through preliminary numerical experiments. The network consists of 32 convolutional filters with a kernel size of 3 and 96 LSTM hidden units. The model was trained using the Adam optimizer with an initial learning rate of 0.005 and a batch size of 16 for 175 epochs. The 400 samples generated by the Latin Hypercube Sampling (LHS) method were randomly divided into a training dataset and a validation dataset with a ratio of 80:20. Consequently, 320 samples were used for model training, while the remaining 80 samples were reserved for validation. The standardized data is input into the neural network for training, with a separate test set used as validation data. This validation set is only used to assess the model's prediction performance and does not participate in the training process, ensuring the objectivity of the performance test. This surrogate model can be embedded into the SS method to replace traditional finite element calculations, enabling the prediction of structural responses under complex conditions. The following figure shows the training convergence graph of the steel frame surrogate model:

Figure 20 shows the loss function convergence curves for the LSTM and CNN-LSTM models. Both models' losses decrease gradually with the number of iterations and eventually stabilize. It can be observed that the CNN-LSTM model converges faster, with a lower loss value compared to the LSTM model, indicating better performance during training.

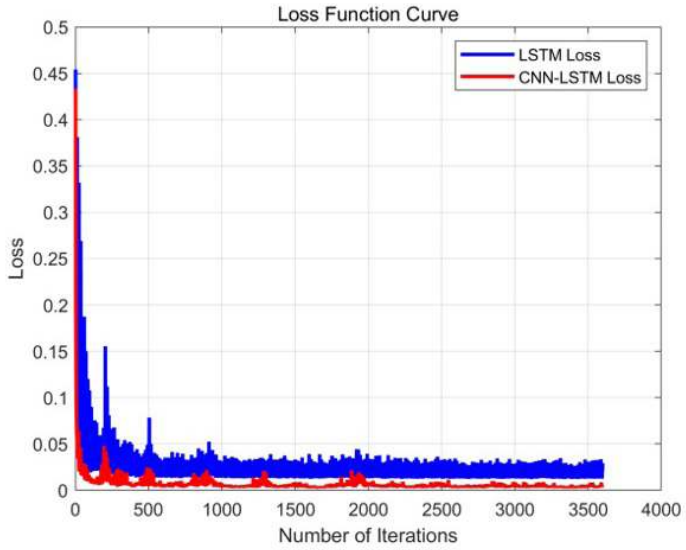


Figure 20. Training Convergence Diagram

Figure 21 compares the true displacement values (blue line) obtained from finite element analysis (FEA) with the predictions from the LSTM (magenta dashed line) and CNN-LSTM (red dashed line) models. The true displacement values serve as the reference, and both models track the general trend of the true data. However, the CNN-LSTM model, which incorporates additional physical information, aligns more closely with the true values, especially during the peak displacements.

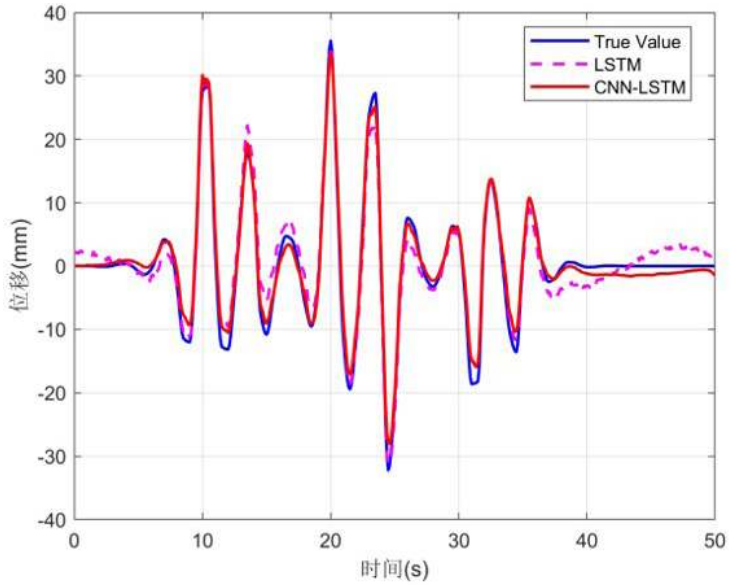


Figure 21. Comparison of Steel Frame Displacement Response

Figure 22 shows the prediction error analysis for the CNN-LSTM and LSTM models. In figure (a), the CNN-LSTM model (red) consistently has smaller prediction errors compared to the LSTM model (black), particularly during peak displacements. Figure (b) illustrates that the CNN-LSTM model has a more concentrated error distribution around zero, indicating higher accuracy and fewer large errors compared to the LSTM model. This demonstrates that the CNN-LSTM model, which incorporates physical information, is more accurate than the LSTM model.

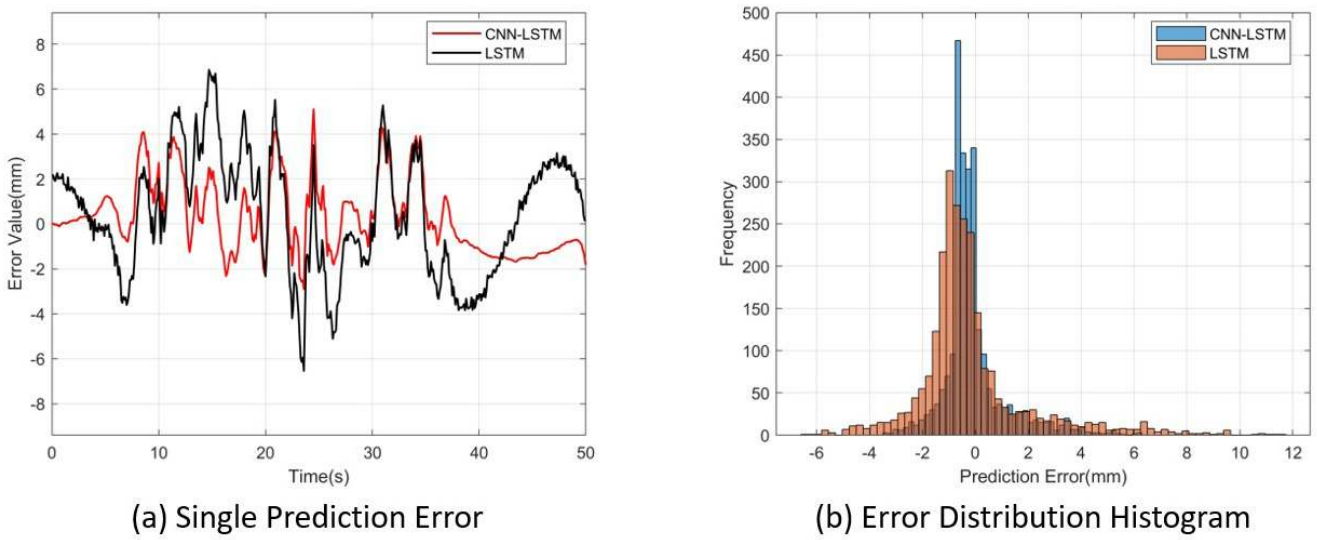


Figure 22. Prediction Error Diagram

Figure 23 compares the predicted displacements with the true displacements for both CNN-LSTM (blue) and LSTM (orange). The CNN-LSTM model shows a tighter clustering around the diagonal line, with an R^2 of 0.978, indicating better prediction accuracy than the LSTM model, which has an R^2 of 0.937.

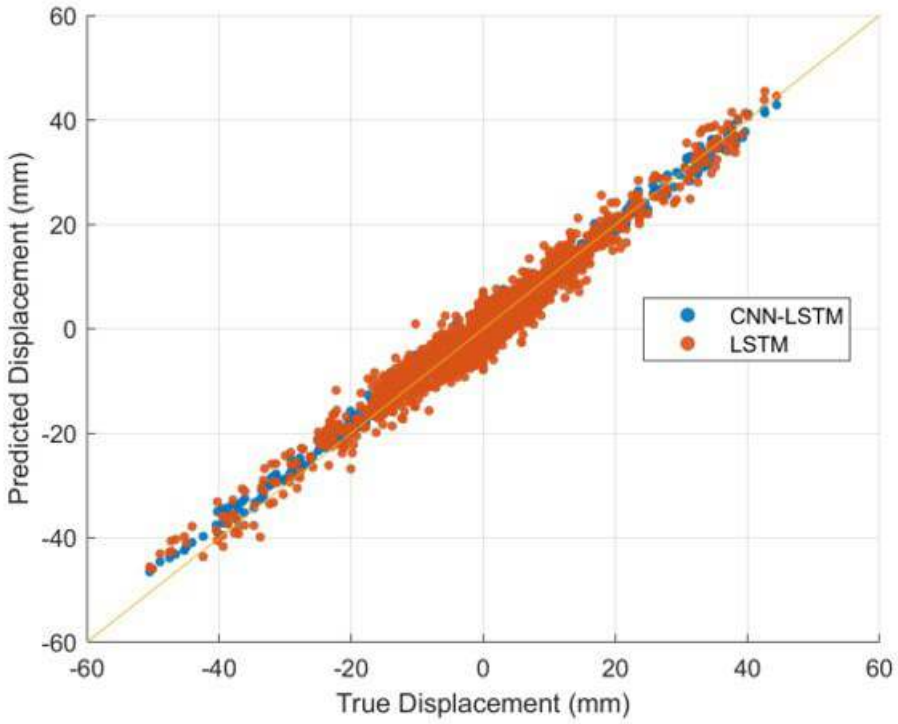


Figure 23. Model Prediction Accuracy Scatter Plot

After validating the accuracy of the surrogate model, SS is applied. In the finite element-SS method, the independence and accuracy of the MCMC-generated samples have already been verified. Take the SS method with CNN-LSTM as an example, the entire SS process forms 2 subsets, with the second subset precisely converging to the failure boundary. Each subset uses the MCMC method for sample updates, and the acceptance probability is strictly controlled to ensure the stability of the Markov chain. The displacement result distributions corresponding to different subsets are shown in the following figure:

Figure 24 shows the progressive sampling process in the reliability analysis of the steel frame structure using the SS method, revealing the process of the structure transitioning from the safe state to the failure boundary through two consecutive subsets (Subset 1-2). Each subset contains 500 samples, with Subset 1 (0-500) generated using MC sampling, resulting in a more dispersed sample distribution, with some samples crossing the failure threshold. Subset 2 (500-1000) converges to the limit state, with more samples entering the failure domain. All responses are predicted by the surrogate model, which reduces computational costs while maintaining the accuracy of the results.

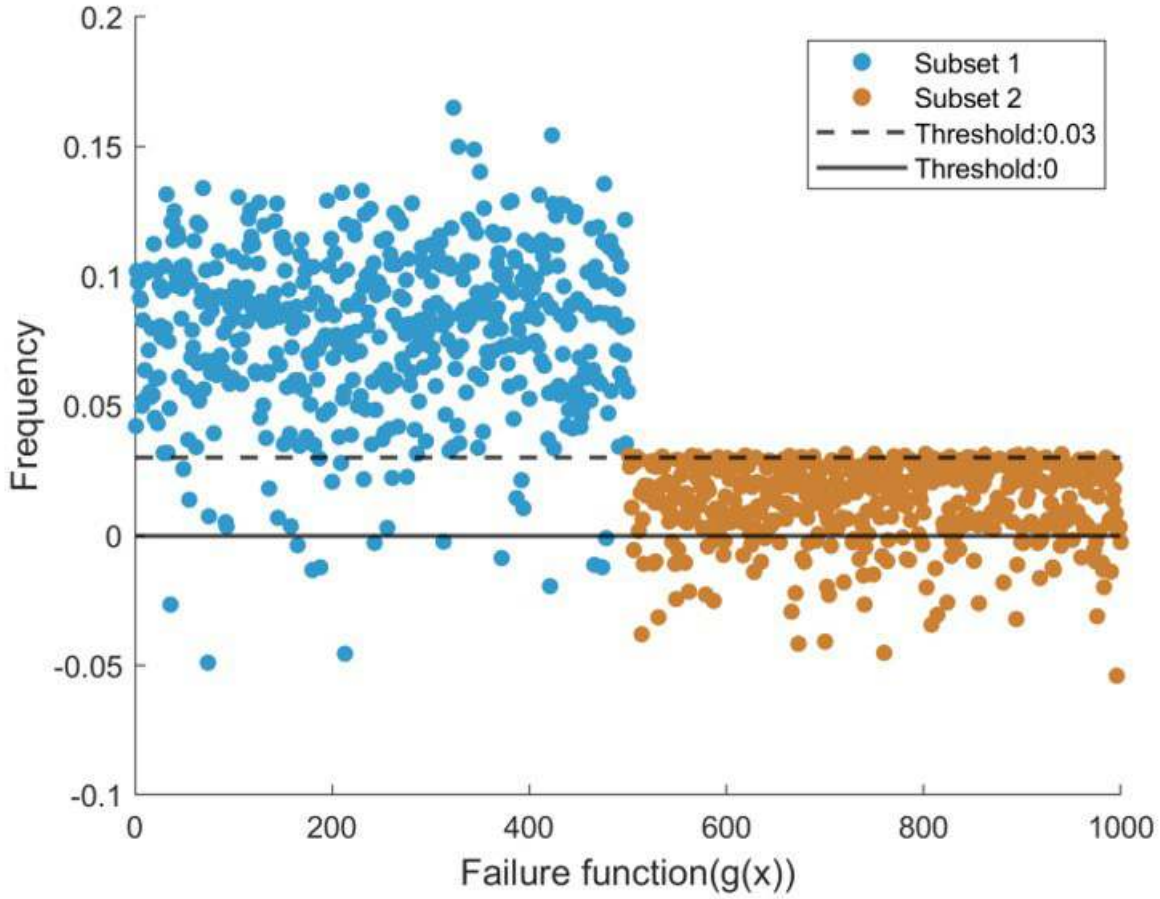


Figure 24. Distribution of Function Values for Different Subsets

Figure 25 shows the frequency distribution of the function values for two subsets (Subset 1-2) in the reliability analysis of the steel frame structure using the SS method. Subset 1 (blue): The frequency distribution is relatively flat, covering the function value range of-0.05 to 0.16, reflecting the initial sampling phase that broadly covers the safe domain, with the threshold set at 0.03. Subset 2 (orange): The sample distribution shifts to the left, forming a peak at $g(x) \approx 0.025$, and a sharp drop in frequency occurs after $g(x) > 0.025$, indicating that high-value samples have been removed. Some samples are concentrated near $g(x) = 0$, and a few samples enter the negative value range ($g(x) < 0$), representing the final convergence to the critical failure state. The steep drop on the right side of Subset 2 is the result of the SS method gradually removing high-value samples through MCMC and optimizing the allocation of computational resources, forming a “narrowing” distribution.

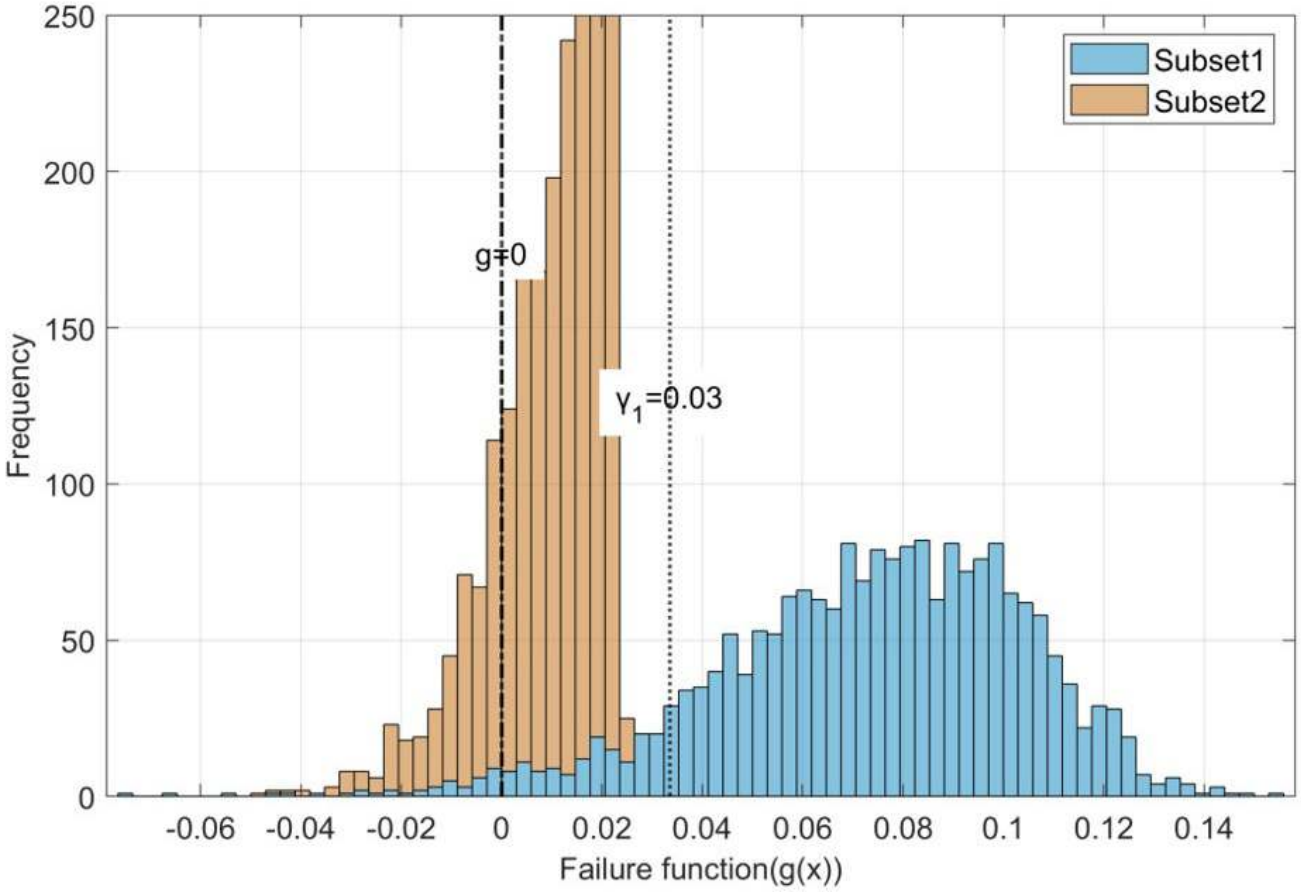


Figure 25. Distribution of Function Values for Different Subsets

Since structural reliability estimation is strongly governed by the prediction accuracy near the limit-state boundary, additional verification was conducted for representative samples around the failure region. Representative samples around the failure region were selected, and the predicted responses obtained by the LSTM and CNN-LSTM surrogate models were compared with the corresponding finite element solutions. The comparison results are summarized in Table 8 and Figures 26–27.

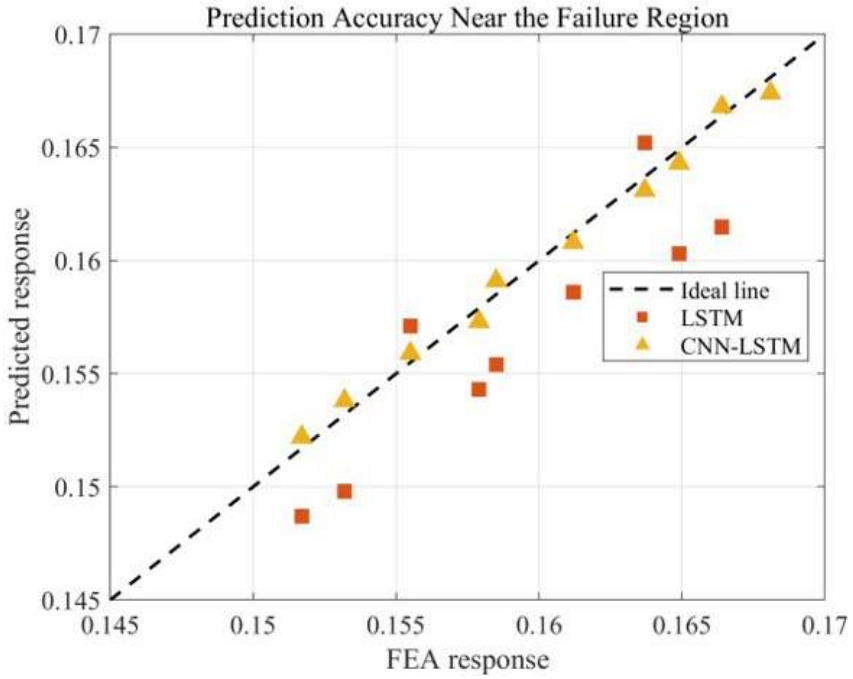


Figure 26. Prediction accuracy near the failure region

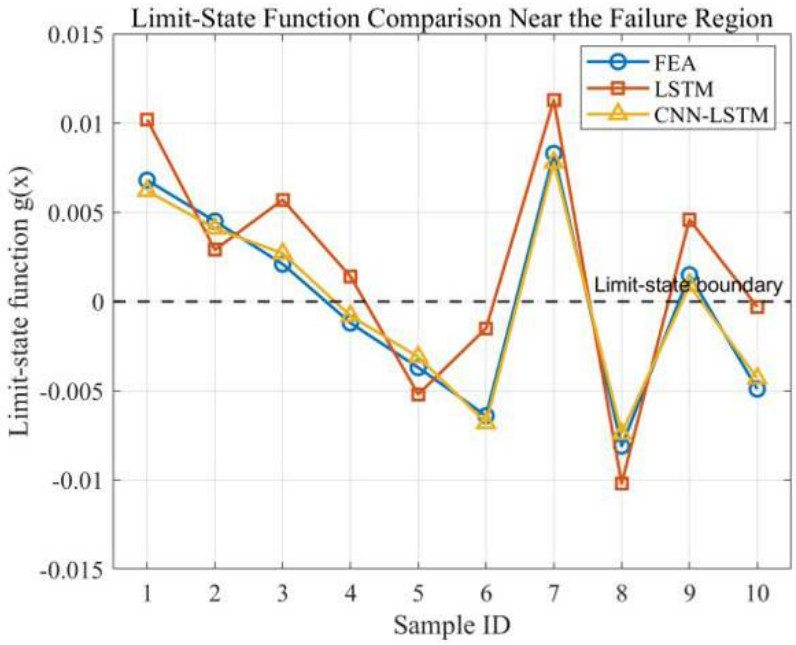


Figure 27. Limit-State function comparison near the failure region

As shown in Figure 26, the prediction results of the proposed CNN-LSTM surrogate remain significantly closer to the ideal prediction line than those of the conventional LSTM model near the failure region. Furthermore, Figure 27 demonstrates that the proposed CNN-LSTM model maintains better agreement with the FEA solutions in terms of the limit-state function evolution.

Table 9 further shows that the conventional LSTM model incorrectly classifies several samples around the failure region, while the proposed CNN-LSTM surrogate correctly identifies the safe/failure states for all representative samples. These results indicate that the proposed physics-informed CNN-LSTM framework provides improved reliability-oriented prediction capability near the limit-state boundary and is therefore more suitable for structural reliability estimation.

Table 9 Verification of surrogate prediction accuracy near the limit-state boundary

Sample	g_{FEA}	FEA response	LSTM response	CNN-LSTM response	FEA state	LSTM sate	CNN-LSTM state
1	0.0068	0.1532	0.1498	0.1538	Safe	Safe	Safe
2	0.0045	0.1555	0.1571	0.1559	Safe	Failure	Safe
3	0.0021	0.1579	0.1543	0.1573	Safe	Safe	Safe
4	-0.0012	0.1612	0.1586	0.1608	Failure	Safe	Failure
5	-0.0037	0.1637	0.1652	0.1631	Failure	Failure	Failure
6	-0.0064	0.1664	0.1615	0.1668	Failure	Failure	Failure
7	0.0083	0.1517	0.1487	0.1522	Safe	Safe	Safe
8	-0.0081	0.1681	0.1702	0.1674	Failure	Failure	Failure
9	0.0015	0.1585	0.1554	0.1591	Safe	Safe	Safe
10	-0.0049	0.1649	0.1603	0.1643	Failure	Failure	Failure

This process results in a final failure probability of 0.0174, with the reliability calculation result being 0.9826. The different subsets and the number of failure samples for both methods are shown in the table 9 below:

This table 10 compares the reliability indicators obtained using the SS-FEA, SS-LSTM, and SS-CNN-LSTM methods. The SS-FEA method serves as the benchmark solution, yielding a failure probability of 0.0224 and a reliability of 0.9776. After retraining with the expanded sample dataset, the SS-LSTM method produces a failure probability of 0.0189 with a failure probability error of 15.63%, corresponding to a reliability of 0.9811. In comparison, the proposed SS-CNN-LSTM method yields a failure probability of 0.0231 with a significantly smaller failure probability error of 3.13% and a reliability of 0.9769, which is much closer to the benchmark SS-FEA result. These results demonstrate that the introduction of the CNN component with embedded structural physical information effectively improves the surrogate prediction capability and enhances the accuracy of structural reliability estimation compared with the conventional LSTM model.

Table 10 Comparison of Reliability Indicators for Different Methods

Method	Failure Probability	Failure Probability Error	Reliability
SS-FEA	0.0224	0	0.9776
SS-LSTM	0.0189	15.63%	0.9811
SS-CNN-LSTM	0.0231	3.13%	0.9769

To further evaluate the statistical stability of the proposed method, five independent Subset Simulation (SS) runs with different random seeds were additionally conducted for the SS-FEA, SS-LSTM, and SS-CNN-LSTM methods. The corresponding statistical results are summarized in Table 11.

Table 11 Statistical results of repeated SS runs

Method	Mean Pf	Std	COV
SS-FEA	0.0225	0.0009	4.00%
SS-LSTM	0.0188	0.0012	6.38%
SS-CNN-LSTM	0.0230	0.0007	3.04%

And the repeated-run failure probability comparisons are illustrated in Figure 28.

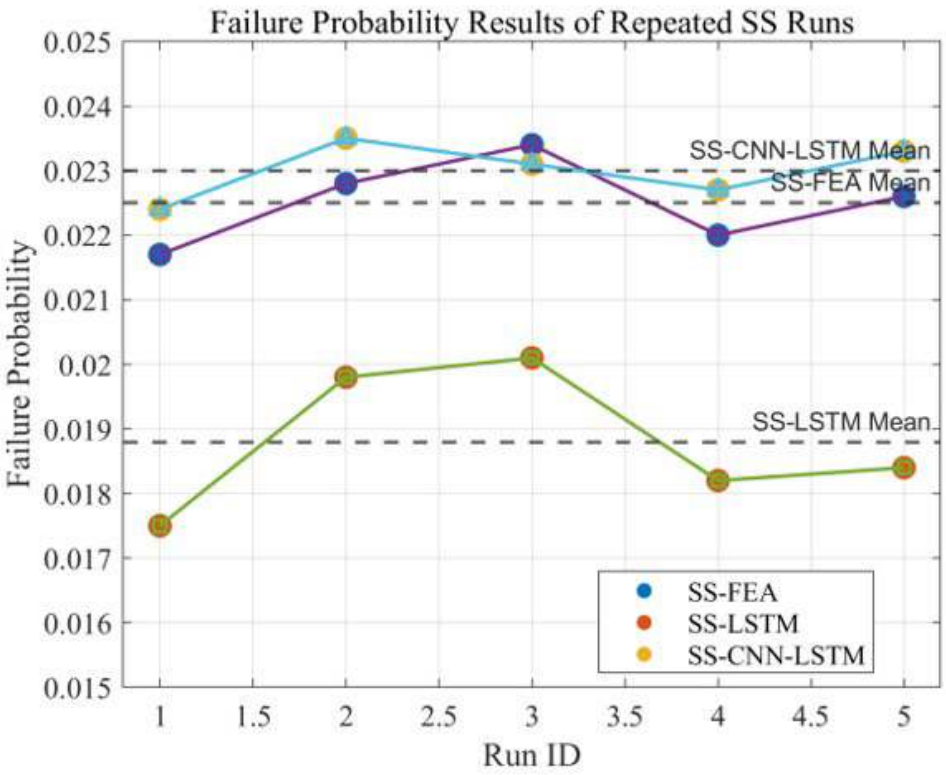


Figure 28. Failure probability results of repeated SS runs

It can be observed that the proposed SS-CNN-LSTM method consistently provides failure probability estimates close to the benchmark SS-FEA results under different random seeds. In contrast, the SS-LSTM method exhibits relatively larger deviations and statistical fluctuations. Furthermore, the SS-CNN-LSTM method achieves the smallest coefficient of variation (COV), indicating improved robustness and stability in the reliability estimation process.

These results demonstrate that the superior agreement between SS-CNN-LSTM and SS-FEA is not caused by random sampling variability, but rather results from the enhanced prediction capability of the proposed physics-informed CNN-LSTM surrogate framework.

5. Conclusions

In conclusion, this study introduces an innovative method for structural reliability analysis by combining the CNN-LSTM hybrid model with Subset Simulation (SS). The CNN-LSTM model leverages the strength of CNN to extract key features from the steel frame structural inputs, including the stochastic load parameters and the first eight natural frequencies of the frame. The CNN component efficiently reduces the dimensionality of these input data and captures important physical patterns of the steel frame that would otherwise be difficult to model directly. The following conclusions are drawn specifically from the analysis of the steel frame structure:

A CNN-LSTM-based surrogate model is proposed, which combines CNN for physical feature extraction and LSTM for temporal sequence learning. The CNN layer helps capture spatial dependencies and essential physical features, while the LSTM network captures the dynamic, time-dependent nature of the structural responses. The numerical results demonstrate that this hybrid model provides more accurate predictions for the considered framework structural systems, although further studies would be needed to generalize to other structural types.

By coupling the CNN-LSTM model with SS, this research significantly reduces computational costs while maintaining the precision of reliability predictions. The SS method, which progressively approaches the failure domain, benefits from the speed and accuracy of the CNN-LSTM surrogate model in predicting structural responses, thus facilitating efficient reliability analysis without the heavy computational burden of traditional finite element methods.

The CNN-LSTM model performs better than the traditional LSTM model. The key difference lies in the CNN component, which adds physical information to the model, improving its ability to predict complex structural responses. Specifically, the CNN-LSTM model exhibits a 3.13% smaller failure probability error compared to LSTM, where the error in LSTM is 15.63%. These results indicate that the CNN component enhances prediction accuracy within the tested framework structures, while further validation is required for other structural types.

Acknowledgement

This research is funded by the Chongqing Joint Training Base for Postgraduate (JDLHPYJD2020020).

Author's contributions: Conceptualization, J Hu; Methodology, XQ Zhang; Investigation, XQ Zhang; Writing-original draft, XQ Zhang; Writing-review & editing, XQ Zhang; Resources, XQ Zhang; Supervision, J Hu

Data Availability statement: Research data is only available upon request

Editor: Rogério José Marczak

Reference

Chan, Jianpeng, Iason Papaioannou, and Daniel Straub. "An adaptive subset simulation algorithm for system reliability analysis with discontinuous limit states." Reliability Engineering & System Safety 225 (2022): 108607.

Elfverson, Daniel, et al. "Adaptive multilevel subset simulation with selective refinement." SIAM/ASA Journal on Uncertainty Quantification 12.3 (2024): 932-963.

Gao, Jianxiong, et al. "A novel machine learning method for multiaxial fatigue life prediction: Improved adaptive neuro-fuzzy inference system." International Journal of Fatigue 178 (2024): 108007.

Guan, Xuefei. "Probabilistic modeling of threshold stress intensity factor for fatigue endurance reliability prediction." Probabilistic Engineering Mechanics 72 (2023): 103417.

He, Jin-Chao, et al. "Size effect in fatigue modelling of defective materials: Application of the calibrated weakest-link theory." International Journal of Fatigue 165 (2022): 107213.

Hu, Jun, and Junhan Wang. "First passage dynamic reliability analysis of non-stationary and non-gaussian buffeting random dynamic responses of long-span suspension bridge." International Journal of Structural Stability and Dynamics 23.20 (2023): 2350199.

Kinnear, Hugh J., and F. A. DiazDelaO. "Niching subset simulation." Probabilistic Engineering Mechanics 79 (2025): 103729.

Lee, Dongkyu, Ziqi Wang, and Junho Song. "Efficient seismic reliability and fragility analysis of lifeline networks using subset simulation." Reliability Engineering & System Safety 260 (2025): 110947.

Li, Peiping. "Active Learning of Small Failure Probabilities of Highly Nonstationary Geotechnical Systems by Adaptive Bayesian Compressive Sensing and Subset Simulation." ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering 11.1 (2025): 04024083.

Liu, Jun S., and Jun S. Liu. Monte Carlo strategies in scientific computing. Vol. 10. New York: springer, 2001.

Meng, Debiao, et al. “Multidisciplinary design optimization of engineering systems under uncertainty: a review.” International Journal of Structural Integrity 13.4 (2022): 565-593.

Nguyen, Truong-Thang, et al. “Seismic reliability analysis of building structures using subset simulation coupled with deep learning-based surrogate model.” Advances in Structural Engineering 25.11 (2022): 2301-2318.

Proppe, Carsten. “Markov chain Monte Carlo simulation methods for structural reliability analysis.” Procedia engineering 199 (2017): 1122-1127.

Sabri, Md Shayan, Furquan Ahmad, and Pijush Samui. “Machine learning-aided Monte Carlo simulation and subset simulation.” Transportation Research Record 2678.12 (2024): 864-886.

Sengodan, Ganapathi Ammasai. “Prediction of two-phase composite microstructure properties through deep learning of reduced dimensional structure-response data.” Composites Part B: Engineering 225 (2021): 109282.

Shao, Tianyu, and Bassem Andrawes. “Using machine learning to predict the seismic response of an SDOF RC structure with superelastic dampers.” International Journal of Civil Engineering 20.10 (2022): 1165-1180.

Siorikis, Vassilis G., et al. “Comparative study of UPV and IE results on concrete cores from existing structures.” Civil Engineering Journal September 10.9 (2024).

Sukhnandan, Jurad, and Georgios A. Drosopoulos. “A machine learning approach used to predict the peak displacement, base shear and fundamental frequency of multi-storey steel structures under seismic excitation.” Structures. Vol. 73. Elsevier, 2025.

Tan, Sui, et al. “Dynamic Response Prediction of Railway Bridges Considering Train Load Duration Using the Deep LSTM Network.” Applied Sciences 14.20 (2024): 9161.

Wang, Yutian, et al. “A confidence-based reliability optimization with single loop strategy and second-order reliability method.” Computer Methods in Applied Mechanics and Engineering 372 (2020): 113436.

Wang, Zeyu, and Abdollah Shafieezadeh. “Metamodel-based subset simulation adaptable to target computational capacities: the case for high-dimensional and rare event reliability analysis.” Structural and Multidisciplinary Optimization 64.2 (2021): 649-675.

Xia, Weili, and Zihan Liao. “Enhanced generalized subset simulation with multiple importance sampling for reliability estimation.” Computers & Structures 313 (2025): 107741.

Xu, Teng, et al. “Estimation of small failure probability in high-dimensional groundwater contaminant transport modeling using subset simulation coupled with preconditioned Crank-Nicolson MCMC.” Water Resources Research 60.12 (2024): e2024WR038260.

Yang, Zhiyong, et al. “Efficient estimation of cumulative distribution functions of multiple failure modes using advanced generalized subset simulation.” International journal for numerical and analytical methods in geomechanics 46.6 (2022): 1093-1108.

Zhang, Gao, et al. “Prediction of bridge structure response and resilience assessment under main-aftershock: LSTM-Transformer model based on adaptive learning rate framework.” Engineering Structures 345 (2025): 121449.

Zhang, Xin, et al. “Pressure and polymer selections for solid-state batteries investigated with high-throughput simulations.” Cell Reports Physical Science 4.3 (2023).

Zhang, Yuming, and Juan Ma. “A method of combined metamodel and subset simulation for reliability analysis of rare events.” Advances in Engineering Software 195 (2024): 103693.

Zhang, Zhenhao, et al. “Study on Static and Dynamic Reliability of Main Girder of Cable-stayed Bridge Based on Subset Simulation Method.” KSCE Journal of Civil Engineering 27.2 (2023): 657-669.