

# PINN-FORM: A new physics-informed neural network for reliability analysis with partial differential equation

Zeng Meng<sup>a</sup>, Qiaochu Qian<sup>a</sup>, Mengqiang Xu<sup>a</sup>, Bo Yu<sup>a,\*</sup>, Ali Rıza Yıldız<sup>b</sup>,  
Seyedali Mirjalili<sup>c</sup>

<sup>a</sup> Institute of Applied Mechanics, School of Civil Engineering, Hefei University of Technology, Hefei 230009, PR China

<sup>b</sup> Department of Automotive Engineering, Bursa Uludağ University, Görükle, Bursa, Turkey

<sup>c</sup> School of Information and Communication Technology, Griffith University, Nathan Campus, Brisbane QLD 4111, Australia

Received 16 April 2023; received in revised form 5 June 2023; accepted 5 June 2023

Available online 22 June 2023

## Abstract

The first-order reliability method (FORM) is commonly used in the field of structural reliability analysis, which transforms the reliability analysis problem into the solution of an optimization problem with equality constraint. However, when the limit state functions (LSFs) in mechanical and engineering problems are complex, particularly for implicit partial differential equations (PDEs), FORM encounters computation difficulty and incurs unbearable computational effort. In this study, the physics-informed neural network (PINN), which is a new branch of deep learning technology for addressing forward and inverse problems with PDEs, is applied as a black-box solution tool. For LSFs with implicit PDE expressions, PINN-FORM is constructed by combining PINN with FORM, which can avoid the calculation of the real structure response. Moreover, a loss function model with an optimization target item is established. Then, an adaptive weight strategy, which can balance the interplay between different parts of the loss function, is suggested to enhance the predictive accuracy. To demonstrate the effectiveness of PINN-FORM, five benchmark examples with LSFs expressed by implicit PDEs, including two-dimensional and three-dimensional problems, and steady state and transient state problems are tested. The results illustrate the proposed PINN-FORM not only is very accurate, but also can simultaneously predict the solutions of PDEs and reliability index within a single training process.

© 2023 Elsevier B.V. All rights reserved.

**Keywords:** Reliability; Partial differential equations; First-order reliability method; Physics-informed neural network

## 1. Introduction

For real-world mechanics and engineering systems, the physical and environmental factors, including material properties, structure size, and boundary conditions, are inherently stochastic [1–5]. Considering this feature, many reliability analysis methods have been suggested for structural design [6–10] and safety assessment. There are many popular reliability analysis methods, including sampling methods [11,12] (e.g. Monte Carlo simulation (MCS) [13–15]), most probable point (MPP) based methods (e.g. first-order reliability method (FORM) [16,17] and second-order reliability method (SORM) [18,19]), surrogate methods [20], expansion methods [21], and approximated

\* Corresponding author.

E-mail address: [yubochina@hfut.edu.cn](mailto:yubochina@hfut.edu.cn) (B. Yu).

integration methods [22,23]. Among them, FORM exhibits superiority in terms of its simplicity and efficiency, and thus it is widely utilized in various engineering problems [24,25].

FORM computes the failure probability by approximating the limit state function (LSF) using the first-order Taylor series expansion. This method maps the reliability analysis problem from the original design space to the standard normal space (U-space), and an optimization problem with an equality constraint is defined accordingly. Until now, a sequence of FORM algorithms has been constructed. One of the most widely utilized FORM algorithms is the Hasofer–Lind and Rackwitz–Fiessler (HL-RF) algorithm, which was reported that performs well for linear engineering problems [16]. However, this algorithm is not suitable for solving nonlinear problems, and thus it leads to the development of new algorithms. Yang et al. [26] introduced the stability transformation method to deal with the non-convergence issue of HL-RF. Then, Meng et al. [27] disclosed the different contributions of circumferential and tangential directions for iterative point, and a directional stability transformation method was further suggested. Yang et al. [28] modified the search direction of HL-RF, which reduces the risk of periodic oscillations. Zhu et al. [29] defined a new MPP optimization model using the cumulative density function, which could avoid the transformation for non-normal variables. Furthermore, many other methods were also developed, such as the hybrid self-adaptive conjugate algorithms (HSAC) [30] and the chaotic conjugate stability transformation method (CCSTM) [31]. Nevertheless, these methods still require repeated computing the real LSF and its sensitivities, which encounters an unbearable computational burden for engineering problems involving complex implicit LSFs. Moreover, when the LSF is expressed by implicit partial differential equations (PDEs), it leads to computational difficulties.

The surrogate method, i.e., metamodel, is capable of replacing complex physical models by simple mathematical models, which aims to solve complex problems with high efficiency. Since it is an easy-to-use method, it is often employed as a black-box prediction tool to tackle the low efficiency and computation barrier of implicit LSFs. The presentative surrogate methods include the response surface method [32,33], Kriging [34–37], support vector machine [38,39], and artificial neural networks (ANN). Among them, ANN [40,41] is deemed as an effective surrogate model that simulates the neural structure and mechanism of the human brain. Unlike other models, ANN is a pure data-driven model that can approximate LSFs with arbitrarily specified accuracy [42]. The key of ANN is the training process, which optimizes the loss function using the residual between the simulation values and the real sample values. A number of seminal works on the application of ANN for reliability analysis can be found in the works of Papadrakakis et al. [43,44], and the results disclosed that ANN is an effective tool. Deng et al. [45] formed different reliability methods based on ANN, including ANN-based FORM, ANN-based SORM, and ANN-based MCS. Gomes et al. [46] tested several reliability analysis methods and proved the high efficiency of ANN-based reliability methods. Xiang et al. [47] combined ANN and the weighted sampling method, which showed high performance for dealing with nonlinear and multi-variable engineering problems. Moreover, other improved ANN-based reliability methods were also developed, such as the local weighted average surrogate approach and the local best surrogate approach [48].

Despite the benefits of the neural networks, a huge amount of the real sample data is required to train them, while the collection of the training data is costly for real-world engineering and physics problems. Moreover, the sample data is often limited by some prior knowledge in engineering and physics problems, such as the conservation laws and constitutive equations. If the prior physics and engineering information are utilized in the neural networks, the amount of the real sample data required for training is greatly reduced. To achieve this, Raissi et al. [49] introduced the physics-informed neural network (PINN) recently, which successfully solved PDEs, including the Navier–Stokes and Allen–Cahn equations. For the PINN theory, the loss function is constructed using the residual of the PDE and the boundary conditions. Then, the PINN model is trained by optimizing the loss function, and the predictive results satisfy the physical laws. In general, the PINN model not only can approximate the known data, but also can automatically meet the prior information expressed by PDEs. Due to the addition of physics information in the training, PINN can obtain more accurate results with less real sample data than ANN. Recently, PINN has received growing attentions in various science and engineering fields, and many new PINN architectures, such as conservative PINNs [50] and nonlocal PINNs [51], were suggested to improve the robustness and efficiency of PINN. So far, PINN has been successfully applied to solve the issues of the fluid mechanics [52–54], solid mechanics [55–58], and material sciences [59,60]. When the LSFs are expressed by PDEs, it requires repeated solving PDEs for reliability assessment. This leads to the computational difficulty and inefficiency. As mentioned above, PINN is an effective black-box solution tool that can avoid the direct calculation for different types of

PDEs, thus the effectiveness and efficiency are guaranteed. Therefore, the application of PINN for solving reliability assessment problems has the potential to expand the range of applications and enhance performance of reliability computation method, particularly for implicit PDEs. If PINN is utilized, the reliability assessment problem and PDEs can be integrated into a single training process, thereby achieving high efficiency and accuracy. Despite this potential, the application of PINN in reliability assessment remains under-explored to date.

In this paper, PINN-FORM is constructed to address the reliability problems with implicit LSFs expressed by PDEs, in which PINN is utilized as a black-box solution tool for the optimization problem defined by FORM. Since it is unnecessary to calculate real LSFs, the computational difficulty and inefficiency are tackled. The rest of this paper is presented as follows. Section 2 presents a brief introduction to the background, including FORM, ANN, and PINN. Section 3 describes the details of PINN-FORM. Section 4 provides five illustrated examples. Some conclusions are summarized in Section 5.

## 2. Background

### 2.1. FORM

The structural reliability is an effective tool that measures the structural safety of engineering and scientific problems with uncertainties, and the reliability index and the failure probability are utilized [61]. The failure probability  $P_f$  is formulated by a multidimensional integral.

$$P_f = P[g(\mathbf{X}) < 0] = \int_{g(\mathbf{X}) < 0} f_{\mathbf{X}}(\mathbf{X}) d\mathbf{X} \quad (1)$$

where  $\mathbf{X} = (X_1, X_2, \dots, X_n)$  are random variables.  $g(\mathbf{X}) = 0$  denotes the LSF, and  $g(\mathbf{X}) < 0$  is failure state.  $f_{\mathbf{X}}$  is the joint probability density function of  $\mathbf{X}$ .

The reliability index  $\beta$  is computed by MPP  $\mathbf{U}^*$  by transforming  $\mathbf{X}$  to  $\mathbf{U}$  using Rosenblatt transformation, which has the minimum distance from the LSF to the origin in U-space. The relationship between the failure probability  $P_f$  and MPP  $\mathbf{U}^*$  is stated as follows:

$$P_f = \Phi(-\beta) = \Phi(-\|\mathbf{U}^*\|) \quad (2)$$

where  $\Phi(\cdot)$  denotes the cumulative distribution function. The reliability assessment problem is transformed into an optimization problem with an equality constraint.

$$\begin{aligned} \min_{\mathbf{U}} \beta &= \sqrt{\mathbf{U}^T \mathbf{U}}, \\ \text{s.t. } g(\mathbf{U}) &= 0 \end{aligned} \quad (3)$$

where  $\mathbf{U}$  represents the random variable vectors in U-space.

### 2.2. ANN

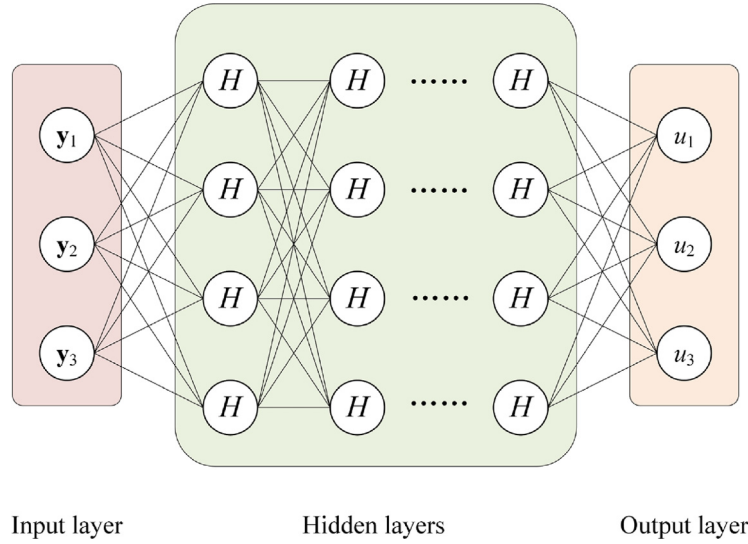
The structure and working principle of ANN are introduced in this subsection. ANN [62,63] consists of neurons situated on the input layer, the output layer, and several hidden layers. There exist functional relationships between the neurons in adjacent layers, while the output of one layer is served as the input of the next layer. The information flow is unidirectional, which propagates from the input layer to the output layer. A multi-layer feed-forward ANN architecture is shown in Fig. 1. The information transfer function is formulated by

$$\mathbf{z}^l = H(\mathbf{w}^l \cdot \mathbf{z}^{l-1} + \mathbf{b}^l), l = 1, 2, \dots, L \quad (4)$$

where  $\mathbf{z}^l$ ,  $\mathbf{w}^l$ , and  $\mathbf{b}^l$  are the output, the weight, and the bias of layer  $l$ , respectively.  $H(\cdot)$  denotes the activation function between neurons, which is utilized to introduce the nonlinearity for neural networks, and the hyperbolic tangent function is often applied. The loss function is formulated to train the ANN model, and it is written as

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N |u_{NN}^i - u^i|^2 \quad (5)$$

where  $N$  denotes the training sample number.  $u_{NN}^i$  denotes the output values of the neural network corresponding to sample  $i$ , and  $u^i$  denotes the real sample values. After the training process, the ANN model is established and can be used for prediction.



**Fig. 1.** Multi-layer feed-forward ANN architecture.

### 2.3. PINN

PINN was first proposed by Raissi et al. [49], which is a new numerical solver for different types of PDEs, and it is also an effective data-driven approach for solving inverse problem. A nonlinear PDE in general form is stated as follows:

$$u_t + \mathcal{N}[u; \lambda] = 0, \mathbf{y} \in \Omega, t \in [0, T] \quad (6)$$

where  $u = u(t, \mathbf{y})$  is the latent solution of the PDE, in which  $t$  denotes the temporal variable, and  $\mathbf{y}$  denotes the spatial variable.  $u_t$  is the partial derivative of  $u(t, \mathbf{y})$  with respect to  $t$ .  $\mathcal{N}[\cdot, \lambda]$  denotes a nonlinear operator that is parametrized by  $\lambda$ .  $\Omega$  denotes a subset of the Euclidean space  $\mathbb{R}^D$ . The left-hand-side of Eq. (6) is defined as  $f_{NN}(t, \mathbf{y})$ , which is computed by

$$f_{NN}(t, \mathbf{y}) = (u_{NN})_t + \mathcal{N}[u_{NN}; \lambda] \quad (7)$$

where  $u_{NN}(t, \mathbf{y})$  is the numerical solution of the PDE approximated by the neural network, in which the temporal variable  $t$  and the spatial variable  $\mathbf{y}$  are the inputs of PINN. The PINN model is learned by optimizing the loss function

$$\mathcal{L} = MSE_u + MSE_f \quad (8)$$

where

$$MSE_u = \frac{1}{N_u} \sum_{i=1}^{N_u} |u_{NN}(t_u^i, \mathbf{y}_u^i) - u^i|^2 \quad (9)$$

$$MSE_f = \frac{1}{N_f} \sum_{i=1}^{N_f} |f_{NN}(t_f^i, \mathbf{y}_f^i)|^2 \quad (10)$$

in which  $\{t_u^i, \mathbf{y}_u^i, u^i\}_{i=1}^{N_u}$  denotes the boundary and initial training data on  $u_{NN}(t, \mathbf{y})$  and  $\{t_f^i, \mathbf{y}_f^i\}_{i=1}^{N_f}$  denotes the training data for  $f_{NN}(t, \mathbf{y})$  at the collocation points.  $MSE_u$  enforces  $u_{NN}(t, \mathbf{y})$  to satisfy the boundary and initial conditions of  $u_{NN}(t, \mathbf{y})$ .  $MSE_f$  enforces  $f_{NN}(t, \mathbf{y}) = 0$ , which means  $u_{NN}(t, \mathbf{y})$  satisfies the relation of Eq. (6) at the collocation points. The PINN architecture is shown in Fig. 2.

For the forward problems, because the parameter  $\lambda$  is known, PINN is directly utilized to predict the solution of PDEs after training. However, when the inverse problems involve, the parameter  $\lambda$  remains unknown. Thus, some additional constraint data of  $u_{NN}(t, \mathbf{y})$  are also required for  $MSE_u$ . Then,  $\lambda$  can be determined after training.

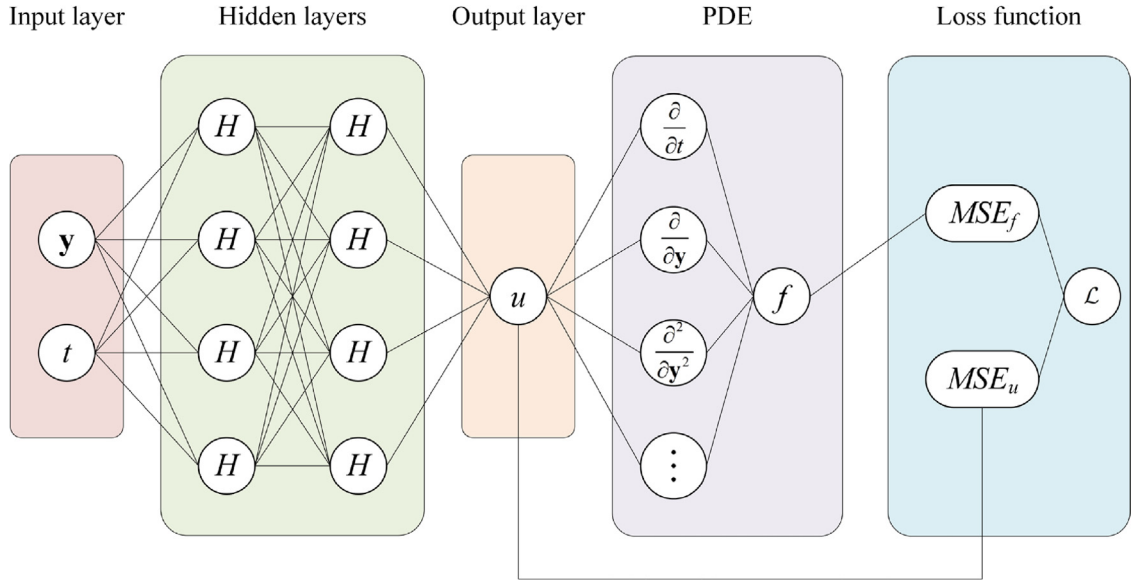


Fig. 2. PINN architecture.

PINN has the advantages of implementation simplicity and applicability to various PDEs. However, the training process usually requires much time. Thus, compared with other classical numerical techniques, e.g. the finite differential method and the finite element method (FEM), the efficiency of PINN is not high. Therefore, PINN should be deemed as an important supplement to traditional numerical methods for solving PDEs.

### 3. The proposed method

#### 3.1. PINN-form

Inspired by the process of solving inverse problems by PINN, the reliability problem is also solved by PINN in this study. The random variables are regarded as the parameters determined by the training process. The purpose is to determine the MPP by PINN, and then the reliability problem is tackled. An implicit LSF expressed by a nonlinear PDE is stated as below:

$$u_t + \mathcal{N}[u, \mathbf{X}] = 0, \mathbf{y} \in \Omega, t \in [0, T] \quad (11)$$

$$g(u) = 0 \quad (12)$$

where  $u = u(t, \mathbf{y})$  is the latent solution of the PDE, in which  $t$  denotes the temporal variable, and  $\mathbf{y}$  denotes the spatial variable vector.  $u_t$  is the partial derivative of  $u(t, \mathbf{y})$  with respect to  $t$ .  $\mathcal{N}[\cdot, \mathbf{X}]$  is a nonlinear operator parametrized by the random variables  $\mathbf{X}$ .  $\Omega$  is a subset of the Euclidean space  $\mathbb{R}^D$ . Eq. (12) represents the LSF. The left-hand-side of Eq. (11) is defined as  $f_{NN}(t, \mathbf{y})$ , which is computed by

$$f_{NN}(t, \mathbf{y}) = (u_{NN})_t + \mathcal{N}[u_{NN}, \mathbf{X}] \quad (13)$$

A new loss function is proposed by Eq. (3) and (8), which is described by

$$\mathcal{L} = MSE_u + MSE_f + MSE_g + \sqrt{\mathbf{U}^T \mathbf{U}} \quad (14)$$

where

$$MSE_u = \frac{1}{N_u} \sum_{i=1}^{N_u} |u_{NN}(t_u^i, \mathbf{y}_u^i) - u^i|^2 \quad (15)$$

$$MSE_f = \frac{1}{N_f} \sum_{i=1}^{N_f} |f_{NN}(t_f^i, \mathbf{y}_f^i)|^2 \quad (16)$$

$$MSE_g = \frac{1}{N_g} \sum_{i=1}^{N_g} |g_{NN}(t_g^i, \mathbf{y}_g^i)|^2 \quad (17)$$

where  $\{t_u^i, \mathbf{y}_u^i, u^i\}_{i=1}^{N_u}$  denotes the boundary and initial data on  $u_{NN}(t, \mathbf{y})$ ,  $\{t_f^i, \mathbf{y}_f^i\}_{i=1}^{N_f}$  denotes the data for  $f_{NN}(t, \mathbf{y})$  at the collocation points, and  $\{t_g^i, \mathbf{y}_g^i\}_{i=1}^{N_g}$  denotes the data at the points related to the limit state defined by the problem.  $MSE_u$  enforces  $u_{NN}(t, \mathbf{y})$  to satisfy the boundary and initial conditions of  $u(t, \mathbf{y})$ .  $MSE_f$  enforces  $f_{NN}(t, \mathbf{y}) = 0$ , which means  $u_{NN}(t, \mathbf{y})$  satisfies the relation of Eq. (11) at the collocation points. The loss function for traditional PINN only consists of two terms, but more information is required for introducing additional constraints on the neural network. Thus,  $MSE_g$  is added to the loss function to enforce  $g_{NN}(t, \mathbf{y}) = 0$ . Based on the above three terms, the parameters  $\mathbf{X}$  are enforced to satisfy Eqs. (11) and (12), which means they satisfy the implicit LSF for the reliability analysis. The purpose of the last term is to optimize  $\sqrt{\mathbf{U}^T \mathbf{U}}$  under the constraint of the LSF to determine the MPP. By optimizing the loss function, the random variables are determined, and then the reliability index  $\beta$  is evaluated. As a PDE solver, PINN requires more computational time than some classical numerical methods, such as FEM. However, traditional reliability methods repeatedly solve the PDEs, while PINN-FORM only needs to train the neural network once, thereby avoiding direct calculation of real LSFs. Moreover, the reliability problem and PDEs can be systematically solved within one training process.

### 3.2. The adaptive weight strategy

Wang et al. [64] found that assigning different weights for different terms of the loss function can balance the interplay between different parts, thereby improving the predictive accuracy of PINN. However, the values of  $MSE_u$ ,  $MSE_f$ ,  $MSE_g$ , and  $\sqrt{\mathbf{U}^T \mathbf{U}}$  have different orders of magnitude. Therefore, an adaptive weight parameter  $\alpha$  is applied to balance different parts of the loss function. The loss function using the adaptive weight strategy is formulated by

$$\mathcal{L} = \alpha MSE_{all} + \sqrt{\mathbf{U}^T \mathbf{U}} \quad (18)$$

where

$$MSE_{all} = MSE_u + MSE_f + MSE_g \quad (19)$$

$$\alpha = \frac{\ln(e + 5k \times 10^{-6})}{(MSE_{all})_1} \times 10^6 \quad (20)$$

where  $k$  denotes the iterative step of the training process,  $e$  is the natural constant, and  $(MSE_{all})_1$  is the value of  $MSE_{all}$  at the first iterative step in training. To balance the relationship between  $MSE_{all}$  and  $\sqrt{\mathbf{U}^T \mathbf{U}}$ , the weight parameter  $\alpha$  is constructed by using the values of  $(MSE_{all})_1$  and  $\ln(e + 5k \times 10^{-6})$ . When  $k = 0$ ,  $\ln(e + 5k \times 10^{-6}) = 0$  and  $\alpha = 1/(MSE_{all})_1 \times 10^6$ . When  $k$  increases,  $\ln(e + 5k \times 10^{-6})$  and  $\alpha$  gradually increase. In this way, the weight parameter  $\alpha$  simultaneously considers the orders of magnitude of  $MSE_{all}$  and  $\sqrt{\mathbf{U}^T \mathbf{U}}$ , which aims to enforce  $MSE_{all} = 0$  at the optimum, thereby improving the predictive accuracy of the PINN model.

### 3.3. Convergence criterion

The traditional iterative method only requires two successive steps to judge the convergence. However, during the training process of PINN, the magnitude of parameter variations between each iteration is typically very small, and the iterative history of PINN appears oscillation phenomenon, which can be seen in the related reference [58]. Thus, there is a challenge for selecting an appropriate convergence precision between two successive iteration steps. To address this issue, more iterative steps are used to ensure the convergence. According to our test, 100 iterative steps can provide enough accuracy. Thus, the stopping criterion is formulated as follows:

$$\|\mathbf{X}^k - \mathbf{X}^{k-100}\| / \|\mathbf{X}^k\| \leq \varepsilon, k = 100, 200, \dots \quad (21)$$



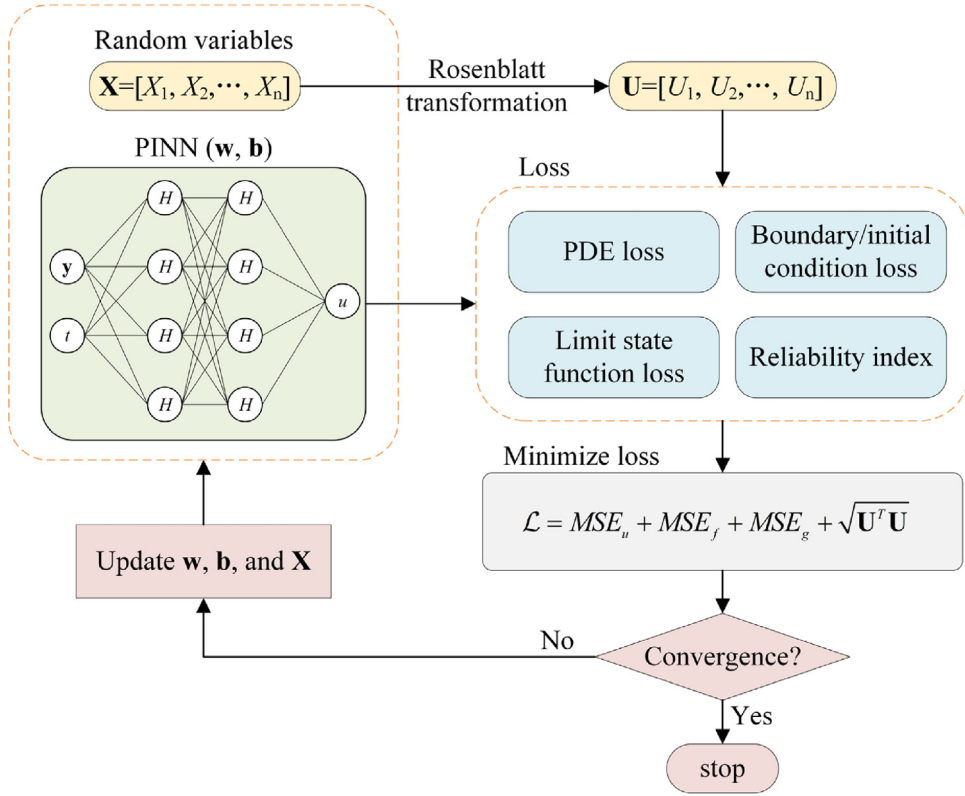


Fig. 3. Flowchart of PINN-FORM.

where  $\varepsilon$  is the allowable convergence precision. Considering the oscillation phenomenon, it is necessary to select an appropriate value of  $\varepsilon$ ; otherwise, it may lead to premature convergence or non-convergence. Therefore, the value of  $\varepsilon$  is set as  $10^{-6}$  in this paper.

### 3.4. Flowchart of PINN-FORM

Flowchart of the learning process for PINN-FORM is illustrated in Fig. 3, while the details of procedures are stated below:

- (1) Select the sample data  $\{t^i, \mathbf{y}^i\}_{i=1}^N$ , including  $\{t_u^i, \mathbf{y}_u^i\}_{i=1}^{N_u}$ ,  $\{t_f^i, \mathbf{y}_f^i\}_{i=1}^{N_f}$ , and  $\{t_g^i, \mathbf{y}_g^i\}_{i=1}^{N_g}$ , as the input of the PINN-FORM model.
- (2) Initialize the neural network parameters  $\mathbf{w}^0$ ,  $\mathbf{b}^0$ , and the random variables  $\mathbf{X}^0$ .
- (3) Transform the random variables  $\mathbf{X}^k$  to standard normal random variables  $\mathbf{U}^k$ .
- (4) Predict the neural network output  $\mathbf{u}$  with the sample information, and then the value of  $\mathbf{f}$  is calculated according to the information of the PDE.
- (5) Compute  $MSE_u$ ,  $MSE_g$ , and  $MSE_f$  to obtain the loss of the neural network.
- (6) Check the convergence by Eq. (21). If convergent, stop; else, update  $\mathbf{w}^k$ ,  $\mathbf{b}^k$ , and  $\mathbf{X}^k$  by optimizing the loss function, and go to step (3).

## 4. Illustrative examples

Five reliability analysis problems, different types of PDEs are solved by PINN-FORM, and the solutions are compared with those calculated by other reliability analysis methods (i.e. MCS, HL-RF, iHL-RF, CCSTM [31], and HSAC [30]). For Examples 2 and 4, the PDEs of MCS, HL-RF, iHL-RF, CCSTM, and HSAC are solved by FEM, and the analytical solutions of PDEs are used for the other three examples. The PINN architectures have 4 hidden layers, and each layer contains 20 neurons.

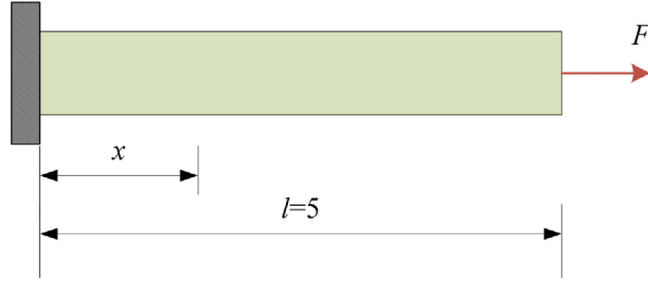


Fig. 4. A circular straight bar.

**Table 1**  
Results of Example 1.

| Methods                      | $N_{call}$      | Iterations      | $\beta$ | $\delta(\%)$ |
|------------------------------|-----------------|-----------------|---------|--------------|
| MCS                          | $1 \times 10^6$ | $1 \times 10^6$ | 2.9679  | –            |
| HL-RF                        | 28              | 7               | 2.9750  | 0.239%       |
| iHL-RF                       | 190             | 7               | 2.9750  | 0.239%       |
| CCSTM                        | 60              | 15              | 2.9750  | 0.239%       |
| HSAC                         | 116             | 29              | 2.9750  | 0.239%       |
| PINN-FORM                    | 0               | $5 \times 10^5$ | 2.9632  | 0.159%       |
| PINN-FORM( $\alpha = 10^4$ ) | 0               | $8 \times 10^4$ | 2.9435  | 0.822%       |
| PINN-FORM( $\alpha = 10^5$ ) | 0               | $2 \times 10^5$ | 2.9614  | 0.219%       |

#### Example 1. A circular straight bar

A circular straight bar is plotted in Fig. 4. The left side is fixed, and the right side is subjected to a load  $F$ . The Young's modulus  $E$ , the cross-sectional area  $A$ , and  $F$  are considered as normal random variables, where the values of means and standard deviations (SDs) are  $\mu_E = 206$ ,  $\sigma_E = 15$ ,  $\mu_A = 1$ ,  $\sigma_A = 0.1$ ,  $\mu_F = 100$ , and  $\sigma_F = 10$ . The maximum displacement allowable of the bar is  $u_{\max} = 1.5$ . For this problem, one boundary point and 500 collocation points on the bar are randomly generated, which are selected as training samples. For the circular straight bar, the governing equation is

$$\frac{du}{dx} = \frac{F}{AE} \quad (22)$$

The boundary condition is stated as follows:

$$(u)_{x=0} = 0 \quad (23)$$

The performance function is stated as follows:

$$g = u - 1.5 \quad (24)$$

The results of MCS, PINN-FORM, HL-RF, iHL-RF, CCSTM, and HSAC are listed in Table 1, in which “ $N_{call}$ ” represents the number of function calls.  $\delta$  is the relative error between MCS and other reliability methods. It can be seen that HL-RF, iHL-RF, CCSTM, and HSAC converge to the same reliability index and MPP. This is because these traditional reliability methods are performed based on the analytical solution. However, PINN-FORM is performed based on PINN, and thus the result is different from those obtained by traditional methods. Compared with iHL-RF, the relative error of PINN-FORM is only 0.397%. The number of function calls of HL-RF and iHL-RF algorithms are 28 and 412, respectively. However, the number of function calls of PINN-FORM is 0 by using PINN. In addition, the impact of adaptive weight strategy is also investigated. When the weight function  $\alpha$  is a constant value  $10^4$ , the relative error is 1.059%. When the value of is  $10^5$ , the relative error is 0.457%. This discloses that the adaptive weight strategy is helpful for improving the accuracy. The iterative curves of loss function and convergence criterion are shown in Fig. 5, it can be seen that the oscillation phenomenon appears during the training process, especially at the beginning phase of the iterative process. Thus, the proposed convergence criterion is necessary. In general, the results prove the effectiveness of PINN-FORM for computing the reliability index with an ordinary differential equation.



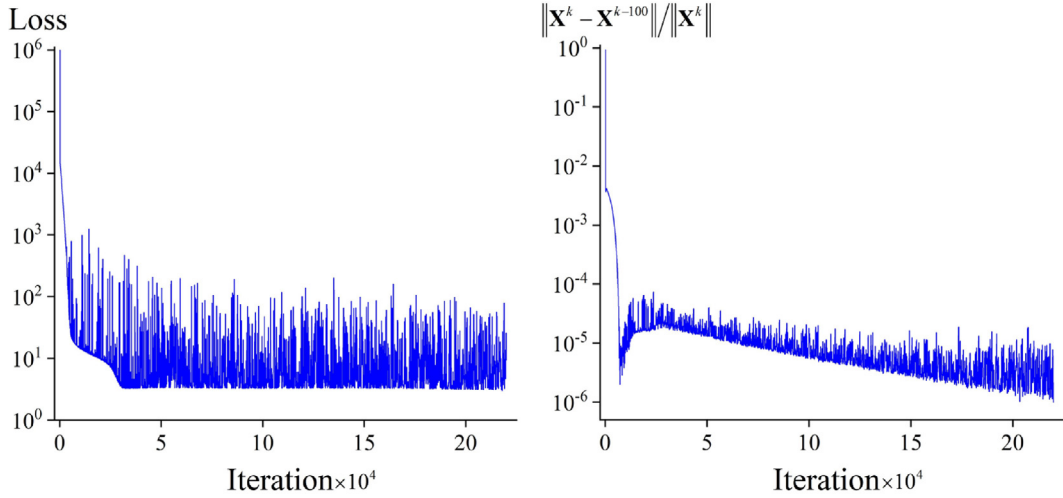


Fig. 5. The iterative curves of loss function and convergence criterion for Example 1.

**Example 2.** A thin square plate with uniform loads

A thin square plate with uniform loads is shown Fig. 6. The side length is 100, and the Poisson's ratio is  $\nu = 0.3$ . The pressures  $q$  and  $E$  are considered as the normal random variables, where the values of means and SDs are  $\mu_E = 200$ ,  $\sigma_E = 10$ ,  $\mu_q = 20$ , and  $\sigma_q = 1.2$ . The maximum vertical displacement allowable is  $|v|_{\max} = 8$ . 1000 boundary points on four sides and 5000 collocation points on the plate surface are randomly generated, which are selected as training samples. For the thin square plate, the governing equation is

$$\begin{cases} \frac{\partial \bar{\sigma}_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} = 0 \\ \frac{\partial \bar{\sigma}_y}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} = 0 \\ \frac{\partial u}{\partial x} = \frac{1}{E}(\bar{\sigma}_x - \nu \bar{\sigma}_y) \\ \frac{\partial v}{\partial y} = \frac{1}{E}(\bar{\sigma}_y - \nu \bar{\sigma}_x) \\ \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} = \frac{2(1+\nu)}{E} \tau_{xy} \end{cases} \quad (25)$$

The boundary conditions are stated as follows:

$$\begin{cases} (u)_{y=0} = 0, (v)_{y=0} = 0 \\ (\bar{\sigma}_x)_{x=0} = 0, (\tau_{xy})_{x=0} = 0 \\ (\bar{\sigma}_x)_{x=100} = 0, (\tau_{xy})_{x=100} = 0 \\ (\bar{\sigma}_y)_{y=100} = -q, (\tau_{xy})_{y=100} = 0 \end{cases} \quad (26)$$

Considering the symmetry, an additional boundary condition is introduced in Eq. (27).

$$(u)_{x=50} = 0 \quad (27)$$

The performance function is stated as follows:

$$g = |v| - 8 \quad (28)$$

The results of MCS, PINN-FORM, HL-RF, iHL-RF, CCSTM, and HSAC are listed in Table 2. Since HL-RF, iHL-RF, CCSTM, and HSAC are performed using the FEM, they converge to the same MPP and reliability index. However, PINN-FORM is performed using PINN, and thus the result is different from those obtained by traditional

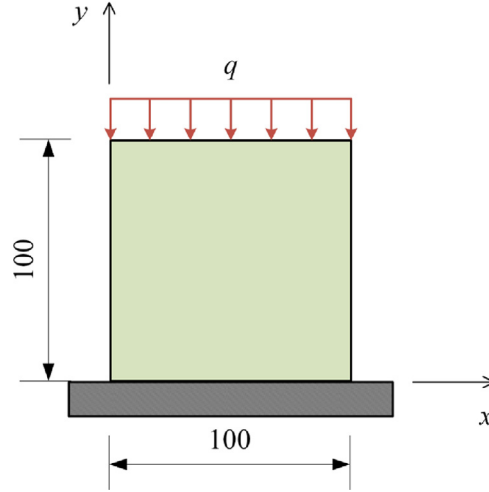


Fig. 6. A thin plate with the uniform loads.

**Table 2**  
Results of Example 2.

| Methods   | $N_{call}$      | Iterations      | $\beta$ | $\delta(\%)$ |
|-----------|-----------------|-----------------|---------|--------------|
| MCS       | $1 \times 10^5$ | $1 \times 10^5$ | 2.6589  | —            |
| HL-RF     | 15              | 5               | 2.6860  | 1.019%       |
| iHL-RF    | 133             | 5               | 2.6860  | 1.019%       |
| CCSTM     | 15              | 5               | 2.6860  | 1.019%       |
| HSAC      | 33              | 11              | 2.6860  | 1.019%       |
| PINN-FORM | 0               | $8 \times 10^5$ | 2.6629  | 0.150%       |

methods. Compared with iHL-RF, the relative error of PINN-FORM is only 0.860%. The number of function calls of HL-RF and iHL-RF algorithms are 15 and 222, respectively. However, the number of function calls of PINN-FORM is 0 by using PINN. The vertical displacement calculated by iHL-RF and PINN-FORM algorithms at the MPP is plotted in Fig. 7, and the relative error is plotted in Fig. 8. The relative error of vertical displacement sharply increases on the fixed supported side, and the maximum value appears at the corners. In general, the error is accepted for the reliability computation. The LSF is expressed by a complex system of PDEs, and there are five outputs of the PINN architecture: two displacement components ( $u$ , and  $v$ ) and three stress components ( $\bar{\sigma}_x$ ,  $\bar{\sigma}_y$ , and  $\tau_{xy}$ ). In this example, traditional reliability methods require repeated solving PDEs with FEM, but PINN is an efficient solving tool for PDEs. Thus, the computational burden can be reduced. Benefiting from PINN, PINN-FORM can avoid direct solving PDEs, and it is capable of simultaneous predicting the solution of the complex PDEs and solving the reliability problem with enough accuracy.

**Example 3.** A thin square plate with non-uniform loads

A thin square plate simply supported on four sides is shown in Fig. 9. The side length is  $\pi$ , and the plate thickness is  $t = 0.5$ . The load  $q = q_0 \sin x \sin y$  is applied on the plate surface.  $E$ ,  $q_0$  and  $\nu$  are considered as the normal random variables, where the values of means and SDs are  $\mu_E = 2 \times 10^8$ ,  $\sigma_E = 2 \times 10^7$ ,  $\mu_{q_0} = 5 \times 10^5$ ,  $\sigma_{q_0} = 2 \times 10^5$ ,  $\mu_\nu = 0.3$ , and  $\sigma_\nu = 0.1$ . The maximum displacement allowable is  $w_{\max} = 1.1$ . 1000 boundary points on four sides and 3000 collocation points on the plate surface are randomly generated, which are selected as training samples. For the thin square plate, the governing equation is

$$\nabla^4 w = \frac{q}{D} \quad (29)$$

where  $D$  is the bending stiffness of the plate that is expressed by

$$D = \frac{Et^3}{12(1 - \nu^2)} \quad (30)$$

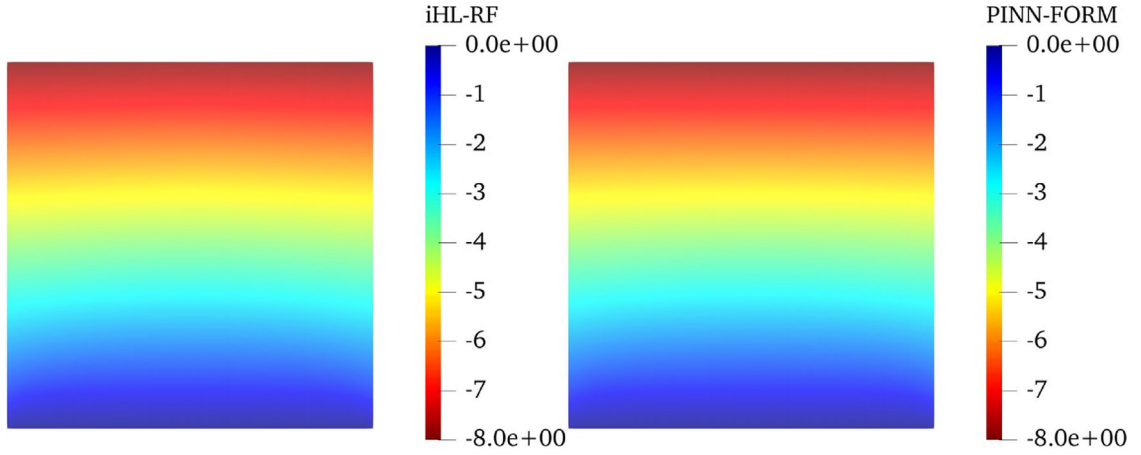


Fig. 7. Vertical displacement calculated by iHL-RF and PINN-FORM at the MPP for Example 2.

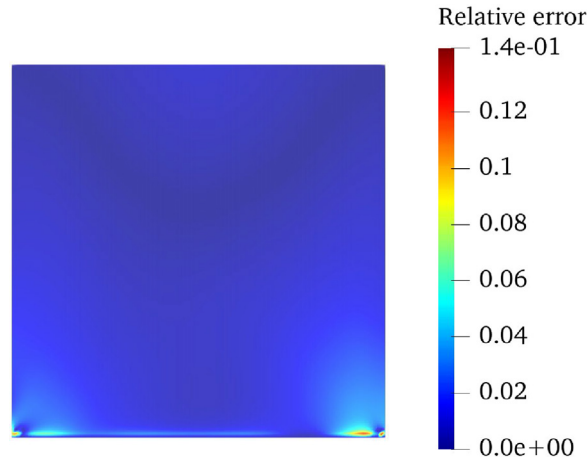


Fig. 8. The relative error of vertical displacement calculated by PINN-FORM at the MPP for Example 2.

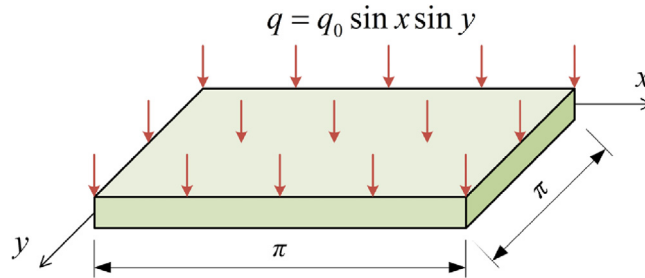


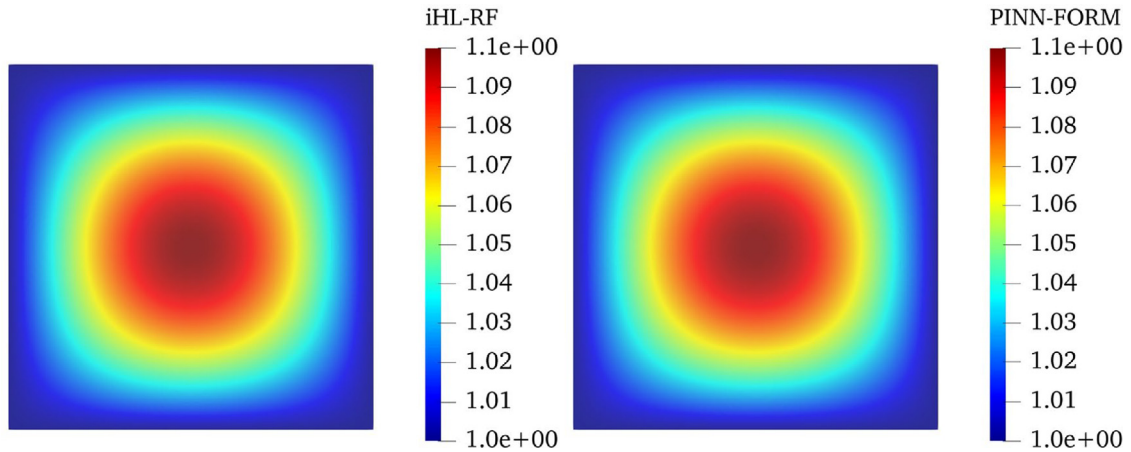
Fig. 9. A square thin plate with non-uniform load.

The boundary conditions and the performance function are stated as follows:

$$\begin{cases} (w)_{x=0,\pi} = 1, (w)_{y=0,\pi} = 1 \\ (\frac{\partial^2 w}{\partial x^2})_{x=0,\pi} = 0, (\frac{\partial^2 w}{\partial y^2})_{y=0,\pi} = 0 \end{cases} \quad (31)$$

**Table 3**  
Results of Example 3.

| Methods   | $N_{call}$      | Iterations      | $\beta$ | $\delta(\%)$ |
|-----------|-----------------|-----------------|---------|--------------|
| MCS       | $1 \times 10^6$ | $1 \times 10^6$ | 1.8851  | —            |
| HL-RF     | 32              | 8               | 1.8413  | 2.323%       |
| iHL-RF    | 123             | 5               | 1.8413  | 2.323%       |
| CCSTM     | 20              | 5               | 1.8413  | 2.323%       |
| HSAC      | 56              | 14              | 1.8413  | 2.323%       |
| PINN-FORM | 0               | $2 \times 10^5$ | 1.8346  | 2.679%       |

**Fig. 10.** Displacement calculated by iHL-RF and PINN-FORM at the MPP for Example 3.

$$g = w - 1.1 \quad (32)$$

The results of PINN-FORM, HL-RF, iHL-RF, CCSTM, and HSAC are listed in Table 3. Compared with iHL-RF algorithm, the relative error of PINN-FORM is only 0.364%. The number of function calls of HL-RF and iHL-RF algorithms are 32 and 364, respectively. However, the number of function calls of PINN-FORM is 0 by using PINN. The displacement calculated by iHL-RF and PINN-FORM algorithms at the MPP is plotted in Fig. 10, and the relative error is plotted in Fig. 11. It is obvious that the relative error is very small on the plate surface, and the values of relative error on four simply supported sides become larger. However, the maximum value of relative error is only  $1 \times 10^{-4}$ , which means the PINN-FORM offers enough accuracy. The LSF expressed by the PDE is up to the fourth order, which leads to high nonlinearity and incurs highly computational cost. However, PINN-FORM can avoid the calculation of the PDE. It can be concluded that PINN-FORM not only can simultaneously predict the high-order PDE and solve the reliability problem, but also exhibits validity and high accuracy.

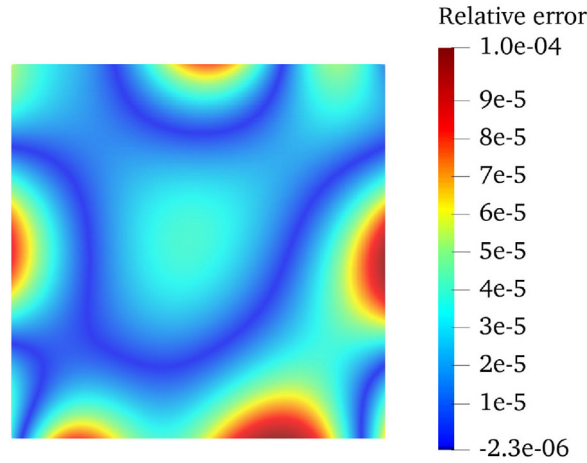
#### Example 4. A three-dimensional cube structure

In a cube structure, the length of each edge is 4, the heat source  $Q$  and the thermal conductivity  $k$  are considered as the normal random variables, where the values of means and SDs are  $\mu_Q = 1$ ,  $\sigma_Q = 0.05$ ,  $\mu_k = 1$ , and  $\sigma_k = 0.05$ . The temperature on each boundary surface is  $T_b = 1$ . The maximum temperature allowable is  $T_{\max} = 1.5$ . 2000 boundary points on each boundary surface and 5000 collocation points in the cube are randomly generated, which are selected as training samples. The model of the cube and the schematic diagram of collocation sample distribution is shown in Fig. 12. For the three-dimensional steady heat conduction problem, the governing equation is

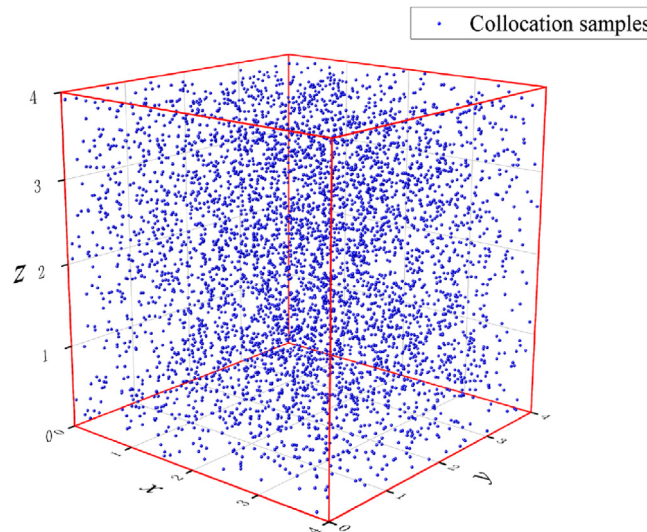
$$-k\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2}\right) = Q \quad (33)$$

The boundary conditions and the performance function are stated as follows:

$$(T)_{x=0,4} = 1, (T)_{y=0,4} = 1, (T)_{z=0,4} = 1 \quad (34)$$



**Fig. 11.** The relative error of displacement calculated by PINN-FORM at the MPP for [Example 3](#).



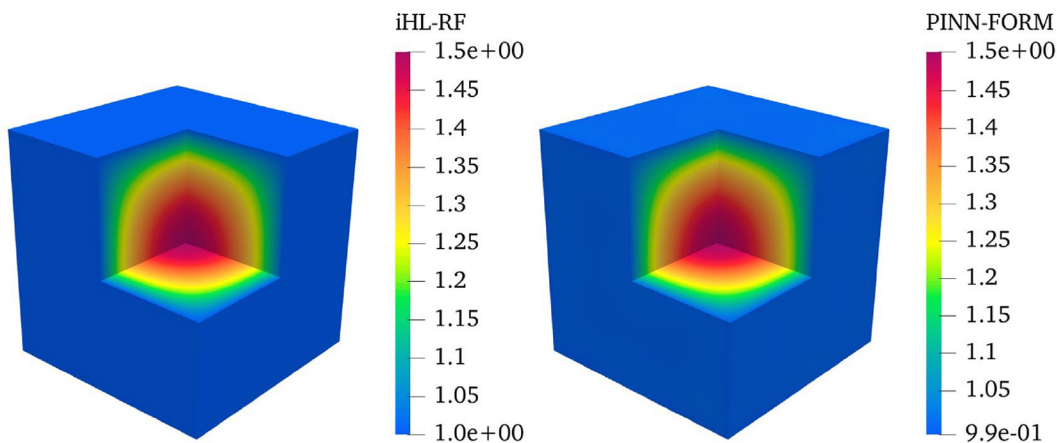
**Fig. 12.** The schematic diagram of sample distribution for the cub structure.

$$g = T - 1.5 \quad (35)$$

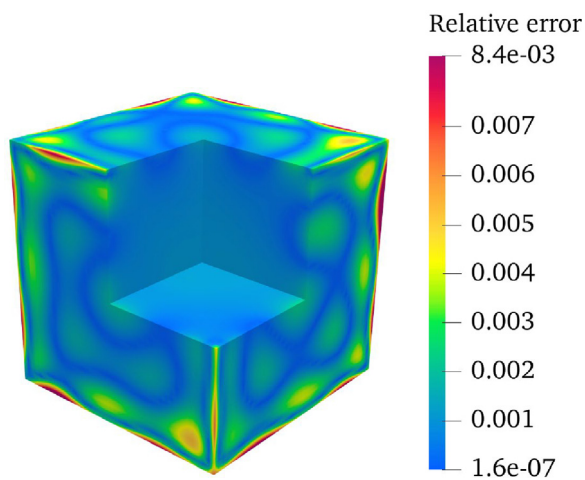
The results of PINN-FORM, HL-RF, iHL-RF, CCSTM, and HSAC are listed in [Table 4](#). The relative error of PINN-FORM is 0.816% compared with iHL-RF, so the accuracy is very high. The number of function calls of HL-RF and iHL-RF algorithms are respectively 18 and 267, while it is 0 for PINN-FORM. The temperature calculated by iHL-RF and PINN-FORM algorithms at the MPP is plotted in [Fig. 13](#), and the relative error is plotted in [Fig. 14](#). The relative error increases on the outer surface of the cube, and it reaches maximum value on the edges and corners. Nonetheless, the relative error inside the cube, especially at the center of the cube, is very small. Thus, the solution is very accurate for the reliability problem. The LSF is expressed by a three-dimensional PDE, which leads to highly computational cost. However, the calculation of PDEs can be avoided, so the practicability of PINN-FORM is proved. In general, PINN-FORM can simultaneously solve the three-dimensional PDE and the reliability problem with high accuracy.

**Example 5.** A wheel hub with functionally graded materials

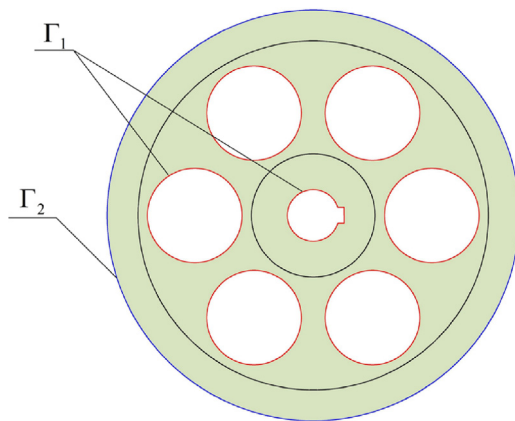
A wheel hub is shown in [Fig. 15](#), the density  $\rho$  and the specific heat  $c$  of the hub are considered as the normal random variables, where the values of means and SDs are  $\mu_\rho = 1$ ,  $\sigma_\rho = 0.2$ ,  $\mu_c = 2$ , and  $\sigma_c = 0.2$ . The maximum



**Fig. 13.** Temperature calculated by iHL-RF and PINN-FORM at the MPP for [Example 4](#).



**Fig. 14.** The relative error of temperature calculated by PINN-FORM at the MPP for [Example 4](#).

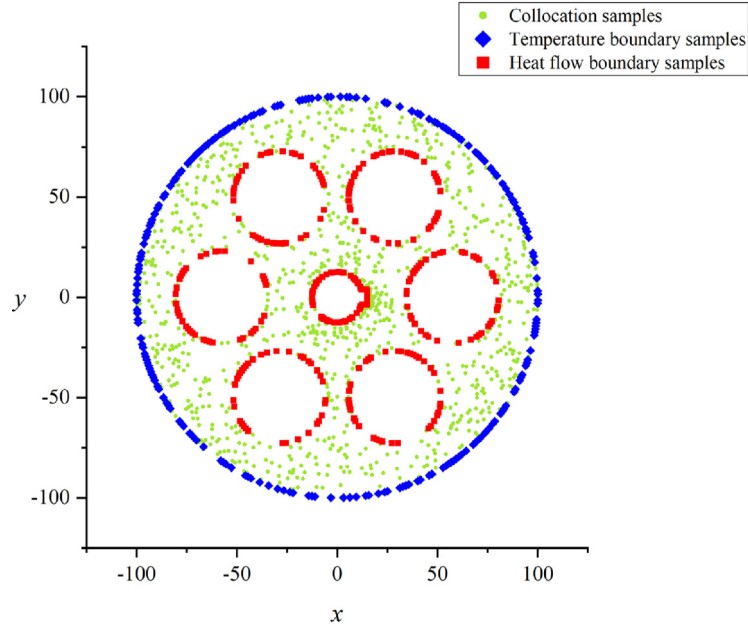


**Fig. 15.** A wheel hub.



**Table 4**  
Results of Example 4.

| Methods   | $N_{call}$      | Iterations      | $\beta$ | $\delta(\%)$ |
|-----------|-----------------|-----------------|---------|--------------|
| MCS       | $1 \times 10^5$ | $1 \times 10^5$ | 2.9440  | —            |
| HL-RF     | 18              | 6               | 2.9183  | 0.873%       |
| iHL-RF    | 597             | 11              | 2.9183  | 0.873%       |
| CCSTM     | 15              | 5               | 2.9183  | 0.873%       |
| HSAC      | 30              | 10              | 2.9183  | 0.873%       |
| PINN-FORM | 0               | $8 \times 10^5$ | 2.9421  | 0.065%       |



**Fig. 16.** The schematic diagram of sample distribution for the wheel hub.

temperature allowable is  $T_{\max} = 200$ . 3000 heat flow boundary points, 2000 temperature boundary points, and 10000 collocation points on the hub are randomly generated, which are selected as training samples. The schematic diagram of sample distribution of  $t = 1$  is shown in Fig. 16. For the transient heat conduction problem, the governing equation is stated as follows:

$$\frac{\partial}{\partial \mathbf{y}} \left( k(\mathbf{y}) \frac{T(\mathbf{y}, t)}{\partial \mathbf{y}} \right) + Q(\mathbf{y}, t) = \rho c \frac{\partial T}{\partial t}, t \in [0.1, 1] \quad (36)$$

where  $t$  denotes the temporal variable, and  $\mathbf{y}$  denotes the spatial variable. The heat source  $Q$  and the thermal conductivity  $k$  are respectively expressed as follows:

$$k = \frac{r^2}{100} + 10 \quad (37)$$

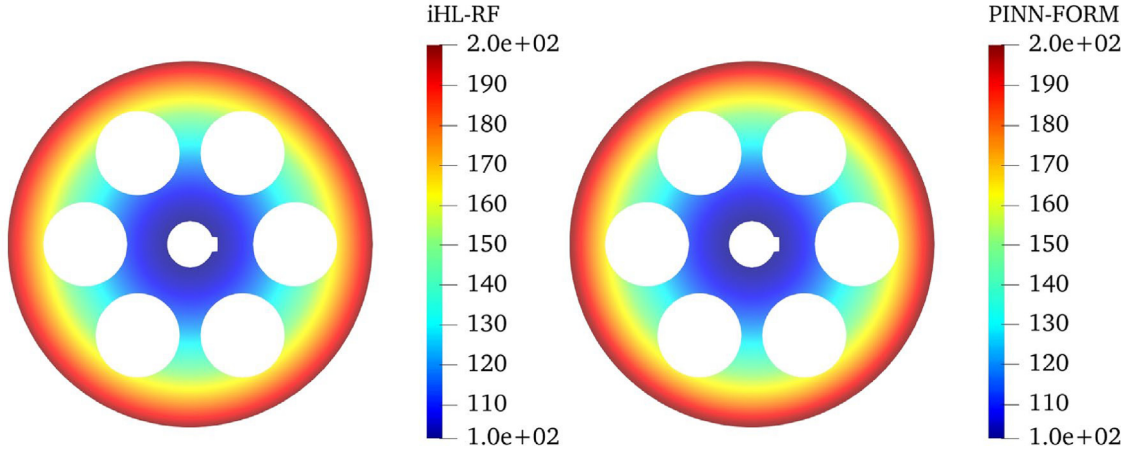
$$Q = \frac{r^2}{100} - \frac{r^2}{1250}t - 0.4t \quad (38)$$

where  $r = \sqrt{x^2 + y^2}$  is the distance from the collocation point to the center of the wheel hub. The heat flow boundary condition  $q(r, t)$  is applied on the spoke edge and the connection between the axle and the hub, and the temperature initial boundary condition  $T(r, 0.1)$  is applied on the rim edge, which are respectively expressed as follows:

$$T(r, 0.1) = \frac{r^2}{1000} + 100, r \in \Gamma_1 \quad (39)$$

**Table 5**  
Results of [Example 5](#).

| Methods   | $N_{call}$      | Iterations      | $\beta$ | $\delta(\%)$ |
|-----------|-----------------|-----------------|---------|--------------|
| MCS       | $1 \times 10^6$ | $1 \times 10^6$ | 2.3912  | —            |
| HL-RF     | 33              | 11              | 2.4153  | 1.008%       |
| iHL-RF    | 134             | 4               | 2.4153  | 1.008%       |
| CCSTM     | 51              | 17              | 2.4153  | 1.008%       |
| HSAC      | 30              | 10              | 2.4153  | 1.008%       |
| PINN-FORM | 0               | $2 \times 10^5$ | 2.4018  | 0.443%       |



**Fig. 17.** Temperature calculated by iHL-RF and PINN-FORM at the MPP for [Example 5](#).

$$q(r, t) = \frac{r^3}{5000}t + \frac{r}{5}t, \quad r \in \Gamma_2 \quad (40)$$

The performance function is stated as follows:

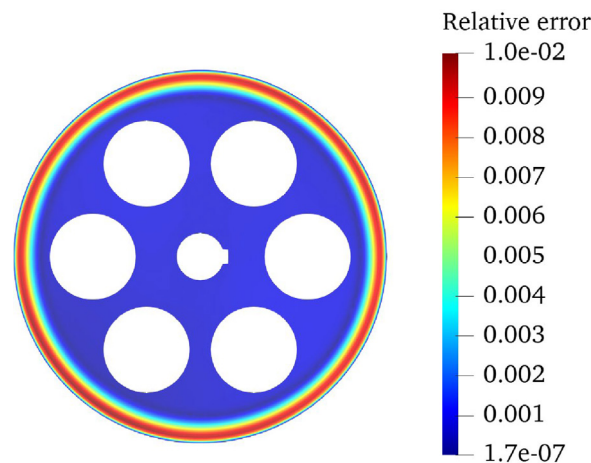
$$g = T(r, t) - 200 \quad (41)$$

The results of PINN-FORM, HL-RF, iHL-RF, CCSTM, and HSAC are listed in [Table 5](#). The relative error of PINN-FORM is 0.559%. The number of function calls of HL-RF and iHL-RF algorithms are 33 and 324, respectively. However, the number of function calls of PINN-FORM is 0 by using PINN. The temperature calculated by iHL-RF and PINN-FORM algorithms at the MPP is plotted in [Fig. 17](#), and the relative error is plotted in [Fig. 18](#). The relative error increases on the rim edge, but it is still less than  $1 \times 10^{-2}$ , which means the solution is accurate. The transient heat conduction problem involving functionally graded materials in a complex structure leads to highly computational cost. However, PINN-FORM can avoid the direct calculation of the PDE, and the results verify the effectiveness and practicability of PINN-FORM.

The above five examples involve different types of PDEs, including a system of PDEs, a high-order PDE, a three-dimensional cube structure, and a wheel hub, and the results show high accuracy and validity of PINN-FORM for solving various PDEs. The number of function calls of PINN-FORM is 0 owing to using PINN, which means PINN-FORM can avoid the computational difficulty. The computational burden incurred by repeated solving PDEs is also avoided. Moreover, the reliability analysis problem and the PDEs can be simultaneously solved after finishing the training process of PINN. Although the PDEs are complex and highly nonlinear, the error of PINN-FORM is small. In summary, the results show the effectiveness and practicability of PINN-FORM for solving reliability assessment problems involving implicit PDEs.

## 5. Conclusions

In this paper, PINN-FORM is proposed to address the reliability problems with complex implicit LSFs expressed by PDEs. PINN is utilized as a black-box solution tool for reliability problems defined by FORM, and a novel loss



**Fig. 18.** The relative error of temperature calculated by PINN-FORM at the MPP for [Example 5](#).

function is suggested to train the neural network. Additionally, an adaptive weight strategy is introduced to balance the interplay among different components of the loss function. Since the solving process of the reliability problem is integrated into the training process, it is unnecessary to compute the real LSFs directly, thereby circumventing the computational challenges posed by PDEs.

Five examples are tested to illustrate the practicability and effectiveness of PINN-FORM. The results reveal that the number of function calls of PINN-FORM is 0, which is significantly fewer than those for classical reliability methods. When the PDEs are complex and highly nonlinear, the superiority of PINN-FORM becomes more pronounced. Moreover, the adaptive weight strategy exhibits a positive impact on accuracy improvement. However, it should be noted that the learning process of PINNs often requires a huge number of iterations. Besides, there exists oscillation phenomenon in the training process, which potentially leads to convergence difficulties. Despite these limitations, PINN-FORM is a promising numerical technology with potential applications in reliability assessment, and further research and development of this technology will be explored in the future.

### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Data availability

Data will be made available on request.

### Acknowledgments

The support of the National Natural Science Foundation of China (Grant No. 11972143) is much appreciated.

### References

- [1] Z. Meng, Z. Zhang, H. Zhou, A novel experimental data-driven exponential convex model for reliability assessment with uncertain-but-bounded parameters, *Appl. Math. Model.* 77 (2020) 773–787.
- [2] I. Lee, U. Lee, P. Ramu, D. Yadav, G. Bayrak, E. Acar, Small failure probability: principles, progress and perspectives, *Struct. Multidiscip. Optim.* 65 (2022) 326.
- [3] D. Zhang, S. Shen, C. Jiang, X. Han, Q. Li, An advanced mixed-degree cubature formula for reliability analysis, *Comput. Methods Appl. Mech. Engrg.* 400 (2022) 115521.
- [4] D. Meng, Z. Lv, S. Yang, H. Wang, T. Xie, Z. Wang, A time-varying mechanical structure reliability analysis method based on performance degradation, *Structures* 34 (2021) 3247–3256.
- [5] B.Y. Ni, C. Jiang, P.G. Wu, Z.H. Wang, W.Y. Tian, A sequential simulation strategy for response bounds analysis of structures with interval uncertainties, *Comput. Struct.* 266 (2022) 106785.

- [6] Y. Li, Y. Wang, R. Ma, P. Hao, Improved reliability-based design optimization of non-uniformly stiffened spherical dome, *Struct. Multidiscip. Optim.* 60 (2019) 375–392.
- [7] Z. Wang, D. Zhao, Y. Guan, Flexible-constrained time-variant hybrid reliability-based design optimization, *Struct. Multidiscip. Optim.* 66 (2023) 89.
- [8] A. Kaveh, A. Zaerreza, Reliability-based design optimization of the frame structures using the force method and SORA-DM framework, *Structures* 45 (2022) 814–827.
- [9] J. Yang, H. Jensen, J. Chen, Structural optimization under dynamic reliability constraints utilizing probability density evolution method and metamodels in augmented input space, *Struct. Multidiscip. Optim.* 65 (2022) 107.
- [10] Z. Meng, Y. Pang, Y. Pu, X. Wang, New hybrid reliability-based topology optimization method combining fuzzy and probabilistic models for handling epistemic and aleatory uncertainties, *Comput. Methods Appl. Mech. Engrg.* 363 (2020) 112886.
- [11] A. Basudhar, S. Missoum, A sampling-based approach for probabilistic design with random fields, *Comput. Methods Appl. Mech. Engrg.* 198 (2009) 3647–3655.
- [12] K. Cheng, I. Papaioannou, Z. Lu, X. Zhang, Y. Wang, Rare event estimation with sequential directional importance sampling, *Struct. Saf.* 100 (2023) 102291.
- [13] A. Alban, H.A. Darji, A. Imamura, M.K. Nakayama, Efficient Monte Carlo methods for estimating failure probabilities, *Reliab. Eng. Syst. Saf.* 165 (2017) 376–394.
- [14] G. Stefanou, D. Savvas, M. Papadrakakis, Stochastic finite element analysis of composite structures based on mesoscale random fields of material properties, *Comput. Methods Appl. Mech. Engrg.* 326 (2017) 319–337.
- [15] Z. Zhu, X. Du, Reliability analysis with Monte Carlo simulation and dependent kriging predictions, *J. Mech. Des.* 138 (2016) 121403.
- [16] D. Yang, Chaos control for numerical instability of first order reliability method, *Commun. Nonlinear Sci.* 15 (2010) 3131–3141.
- [17] X. Du, W. Chen, Sequential optimization and reliability assessment method for efficient probabilistic design, *J. Mech. Des.* 126 (2004) 225–233.
- [18] J. Zhang, X. Du, A second-order reliability method with first-order efficiency, *J. Mech. Des.* 132 (2010) 101006.
- [19] I. Lee, Y. Noh, D. Yoo, A novel second-order reliability method (SORM) using noncentral or generalized Chi-Squared distributions, *J. Mech. Des.* 134 (2012) 100912.
- [20] D. Meng, S. Yang, Y. Zhang, S. Zhu, Structural reliability analysis and uncertainties-based collaborative design and optimization of turbine blades using surrogate model, *Fatigue Fract. Eng. Mater. Struct.* 42 (2019) 1219–1227.
- [21] S. Rahman, B.N. Rao, A perturbation method for stochastic meshless analysis in elastostatics, *Internat. J. Numer. Methods Engrg.* 50 (2001) 1969–1991.
- [22] S. Rahman, H. Xu, A univariate dimension-reduction method for multi-dimensional integration in stochastic mechanics, *Probabilist. Eng. Mech.* 19 (2004) 393–408.
- [23] B.D. Youn, K.K. Choi, K. Yi, Performance moment integration (PMI) method for quality assessment in reliability-based robust design optimization, *Mech. Based. Des. Struct.* 33 (2005) 185–213.
- [24] R.H. Lopez, A.J. Torii, L.F.F. Miguel, J.E. Souza Cursi, Overcoming the drawbacks of the FORM using a full characterization method, *Struct. Saf.* 54 (2015) 57–63.
- [25] S. ur Rehman, M. Langelaar, Efficient global robust optimization of unconstrained problems affected by parametric uncertainties, *Struct. Multidiscip. Optim.* 52 (2015) 319–336.
- [26] D. Yang, X. Li, G. Chen, Z. Meng, Accelerated stability transformation method for chaos control of discrete dynamical systems, *Nonlinear. Dynam.* 94 (2018) 1195–1213.
- [27] Z. Meng, G. Li, D. Yang, L. Zhan, A new directional stability transformation method of chaos control for first order reliability analysis, *Struct. Multidiscip. Optim.* 55 (2017) 601–612.
- [28] M. Yang, D. Zhang, X. Han, New efficient and robust method for structural reliability analysis and its application in reliability-based design optimization, *Comput. Methods Appl. Mech. Engrg.* 366 (2020) 113018.
- [29] S.-P. Zhu, B. Keshtegar, S. Chakraborty, N.-T. Trung, Novel probabilistic model for searching most probable point in structural reliability analysis, *Comput. Methods Appl. Mech. Engrg.* 366 (2020) 113027.
- [30] B. Keshtegar, S. Chakraborty, A hybrid self-adaptive conjugate first order reliability method for robust structural reliability analysis, *Appl. Math. Model.* 53 (2018) 319–332.
- [31] B. Keshtegar, Chaotic conjugate stability transformation method for structural reliability analysis, *Comput. Methods Appl. Mech. Engrg.* 310 (2016) 866–885.
- [32] N. Roussouly, F. Petitjean, M. Salaun, A new adaptive response surface method for reliability analysis, *Probabilist. Eng. Mech.* 32 (2013) 103–115.
- [33] Y. Jiang, L. Zhao, M. Beer, E. Patelli, M. Broggi, J. Luo, Y. He, J. Zhang, Multiple response surfaces method with advanced classification of samples for structural failure function fitting, *Struct. Saf.* 64 (2017) 87–97.
- [34] D. Wang, H. Qiu, L. Gao, C. Jiang, A single-loop Kriging coupled with subset simulation for time-dependent reliability analysis, *Reliab. Eng. Syst. Saf.* 216 (2021) 107931.
- [35] L. Chen, H. Qiu, L. Gao, Z. Yang, D. Xu, Exploiting active subspaces of hyperparameters for efficient high-dimensional Kriging modeling, *Mech. Syst. Signal. Process.* 169 (2022) 108643.
- [36] Z. Tayyab, Y. Zhang, Z. Wang, An efficient Kriging based method for time-dependent reliability based robust design optimization via evolutionary algorithm, *Comput. Methods Appl. Mech. Engrg.* 372 (2020) 113386.
- [37] C. Zhou, N.C. Xiao, M.J. Zuo, W. Gao, An improved Kriging-based approach for system reliability analysis with multiple failure modes, *Eng. Comput-Ger.* 38 (2022) 1813–1833.
- [38] C.M. Rocco, J.A. Moreno, Fast Monte Carlo reliability evaluation using support vector machine, *Reliab. Eng. Syst. Saf.* 76 (2002) 237–243.

- [39] H. Li, Z. Lü, Z. Yue, Support vector machine for structural reliability analysis, *Appl. Math. Mech.* 27 (2006) 1295–1303.
- [40] A.A. Chojaczyk, A.P. Teixeira, L.C. Neves, J.B. Cardoso, C. Guedes Soares, Review and application of artificial neural networks models in reliability analysis of steel structures, *Struct. Saf.* 52 (2015) 78–89.
- [41] T.I. Zohdi, A note on rapid genetic calibration of artificial neural networks, *Comput. Mech.* 70 (2022) 819–827.
- [42] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, *Neural Netw.* 2 (1989) 359–366.
- [43] M. Papadrakakis, N.D. Lagaros, Reliability-based structural optimization using neural networks and Monte Carlo simulation, *Comput. Methods Appl. Mech. Engrg.* 191 (2002) 3491–3507.
- [44] M. Papadrakakis, N.D. Lagaros, Reliability-based structural optimization using neural networks and Monte Carlo simulation, *Comput. Methods Appl. Mech. Engrg.* 191 (2002) 3491–3507.
- [45] J. Deng, D. Gu, X. Li, Z.Q. Yue, Structural reliability analysis for implicit performance functions using artificial neural network, *Struct. Saf.* 27 (2005) 25–48.
- [46] H.M. Gomes, A.M. Awruch, Comparison of response surface and neural network with other methods for structural reliability analysis, *Struct. Saf.* 26 (2004) 49–67.
- [47] Z. Xiang, J. Chen, Y. Bao, H. Li, An active learning method combining deep neural network and weighted sampling for structural reliability analysis, *Mech. Syst. Signal. Process.* 140 (2020) 106684.
- [48] C. Ren, Y. Aoues, D. Lemosse, E. Souza De Cursi, Ensemble of surrogates combining Kriging and artificial neural networks for reliability analysis with local goodness measurement, *Struct. Saf.* 96 (2022) 102186.
- [49] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *J. Comput. Phys.* 378 (2019) 686–707.
- [50] A.D. Jagtap, E. Kharazmi, G.E. Karniadakis, Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems, *Comput. Methods Appl. Mech. Engrg.* 365 (2020) 113028.
- [51] G. Pang, M. D’Elia, M. Parks, G.E. Karniadakis, nPINNs: Nonlocal physics-informed neural networks for a parametrized nonlocal universal Laplacian operator. Algorithms and applications, *J. Comput. Phys.* 422 (2020) 109760.
- [52] M. Raissi, A. Yazdani, G.E. Karniadakis, Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations, *Science* 367 (2020) 1026–1030.
- [53] Q. Zhu, Z. Zhao, J. Yan, Physics-informed machine learning for surrogate modeling of wind pressure and optimization of pressure sensor placement, *Comput. Mech.* (2022).
- [54] M. Raissi, Z. Wang, M.S. Triantafyllou, G.E. Karniadakis, Deep learning of vortex-induced vibrations, *J. Fluid Mech.* 861 (2019) 119–137.
- [55] J. Bai, T. Rabczuk, A. Gupta, L. Alzubaidi, Y. Gu, A physics-informed neural network technique based on a modified loss function for computational 2D and 3D solid mechanics, *Comput. Mech.* (2022).
- [56] W.Q. Hao, L. Tan, X.G. Yang, D.Q. Shi, M.L. Wang, G.L. Miao, Y.S. Fan, A physics-informed machine learning approach for notch fatigue evaluation of alloys used in aerospace, *Int. J. Fatigue* 170 (2023) 107536.
- [57] L. Ning, Z. Cai, H. Dong, Y. Liu, W. Wang, A peridynamic-informed neural network for continuum elastic displacement characterization, *Comput. Methods Appl. Mech. Engrg.* 407 (2023) 115909.
- [58] E. Haghighat, M. Raissi, A. Moure, H. Gomez, R. Juanes, A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics, *Comput. Methods Appl. Mech. Engrg.* 379 (2021) 113741.
- [59] Z. Fang, J. Zhan, Deep physical informed neural networks for metamaterial design, *IEEE Access* 8 (2020) 24506–24513.
- [60] Y. Chen, L. Lu, G.E. Karniadakis, L. Dal Negro, Physics-informed neural networks for inverse problems in nano-optics and metamaterials, *Opt. Express* 28 (2020) 11618.
- [61] X. Wang, W. Zhao, Y. Chen, X. Li, A first order reliability method based on hybrid conjugate approach with adaptive Barzilai–Borwein steps, *Comput. Methods Appl. Mech. Engrg.* 401 (2022) 115670.
- [62] Y. Guo, C. Wang, Z. Ma, X. Huang, K. Sun, R. Zhao, A new mesh smoothing method based on a neural network, *Comput. Mech.* 69 (2022) 425–438.
- [63] P. Thakolkaran, A. Joshi, Y. Zheng, M. Flaschel, L. De Lorenzis, S. Kumar, NN-EUCLID: Deep-learning hyperelasticity without stress data, *J. Mech. Phys. Solids* 169 (2022) 105076.
- [64] S. Wang, Y. Teng, P. Perdikaris, Understanding and mitigating gradient flow pathologies in physics-informed neural networks, *SIAM J. Sci. Comput.* 43 (2021) A3055–A3081.