



ARTICLE

SL-COA: Hybrid Efficient and Enhanced Coati Optimization Algorithm for Structural Reliability Analysis

Yunhan Ling¹, Huajun Peng², Yiqing Shi^{1,*}, Chao Xu¹, Jingzhen Yan¹, Jingjing Wang¹ and Hui Ma^{3,*}

¹China Academy of Machinery, Beijing Research Institute of Mechanical & Electrical Technology Co., Ltd., Beijing, 100083, China

²AVIC Guizhou Anda Aviation Forging Co., Ltd., Anshun, 561000, China

³School of Mechanical and Electrical Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China

*Corresponding Authors: Yiqing Shi. Email: shiyiqing1005@163.com; Hui Ma. Email: mahui@uestc.edu.cn

Received: 02 December 2024; Accepted: 07 February 2025; Published: 11 April 2025

ABSTRACT: The traditional first-order reliability method (FORM) often encounters challenges with non-convergence of results or excessive calculation when analyzing complex engineering problems. To improve the global convergence speed of structural reliability analysis, an improved coati optimization algorithm (COA) is proposed in this paper. In this study, the social learning strategy is used to improve the coati optimization algorithm (SL-COA), which improves the convergence speed and robustness of the new heuristic optimization algorithm. Then, the SL-COA is compared with the latest heuristic optimization algorithms such as the original COA, whale optimization algorithm (WOA), and osprey optimization algorithm (OOA) in the CEC2005 and CEC2017 test function sets and two engineering optimization design examples. The optimization results show that the proposed SL-COA algorithm has a high competitiveness. Secondly, this study introduces the SL-COA algorithm into the MPP (Most Probable Point) search process based on FORM and constructs a new reliability analysis method. Finally, the proposed reliability analysis method is verified by four mathematical examples and two engineering examples. The results show that the proposed SL-COA-assisted FORM exhibits fast convergence and avoids premature convergence to local optima as demonstrated by its successful application to problems such as composite cylinder design and support bracket analysis.

KEYWORDS: Hybrid reliability analysis; single-loop interactive hybrid analysis; most probability point; metaheuristic algorithms; coati optimization algorithm

1 Introduction

With the increasing complexity of engineering structures, structural reliability analysis (SRA) plays an increasingly important role in engineering design, operation, and maintenance. Traditional structural reliability analysis methods, while capable of obtaining relatively accurate results, suffer from extremely low computational efficiency when faced with small failure probabilities or large-scale finite element analyses. Therefore, the development of efficient and accurate structural reliability analysis methods has become a research hotspot. This study aims to address this challenge by improving existing optimization algorithms and combining them with the first-order reliability method (FORM) to propose a new structural reliability analysis framework for solving reliability analysis problems in complex engineering structures. This research not only has significant theoretical value but also boasts broad application prospects, providing engineers with more reliable and efficient analysis tools for engineering design.



SRA is not only beneficial to the systematic adjustment of structural safety factors but also a key factor in structural design and operation maintenance [1–3]. SRA plays a very important role in preventing higher-level risks in the life expectancy of engineering structures [4,5]. Higher-level risks in engineering structures can include catastrophic failures, such as bridge collapses, building collapses, and aircraft crashes. For example, the collapse of the I-35W Mississippi River bridge in Minneapolis in 2007 resulted in significant loss of life and property damage [6]. Reliability analysis can help prevent such failures by ensuring that engineering structures are designed to withstand expected loads and environmental conditions with high reliability. At the same time, it can also help engineers design more safe and reliable mechanical systems [7–10].

SRA is usually performed by calculating or estimating the probability that the research object violates the limit state function (LSF) during its life cycle [11,12]. Monte Carlo simulation (MCS) is a very important tool in the field of SRA, which has been widely studied based on the original MCS [13]. In addition, the use of analytical methods to complete SRA has also been widely studied. Common analytical methods include FORM [14], second-order reliability method (SORM) [15], first-order second-moment (FOSM) [16] and so on. Compared with using MCS to complete SRA, the analytical method requires lower computational cost. While analytical methods such as FORM generally require lower computational costs compared to MCS for low-dimensional problems, they face significant limitations in high-dimensional problems. This is because the complexity of the reliability analysis increases exponentially with the number of dimensions. In high-dimensional spaces, analytical methods may struggle to find the Most Probable Point (MPP) efficiently, leading to increased computational costs and potential inaccuracies. Furthermore, the assumptions made in analytical methods, such as local linear approximations of the LSF, may become less valid in high-dimensional spaces, further compromising the accuracy of the results. Although the use of MCS to complete structural reliability analysis can obtain a more accurate failure probability or reliability index. However, with the increasing complexity of engineering structures, the computational efficiency of MCS is very low in the face of engineering examples with small failure probability or large-scale finite element analysis [17,18]. Therefore, the subset simulation (SS) method [19], importance sampling (IS) method [20], and line sampling (LS) [21] method have been developed. In addition, the use of surrogate models instead of simulation or complex constraint functions has a very effective effect on reducing the number of assessments of limit states. Common surrogate models include the Kriging model [22–24], polynomial chaos expansion (PCE) [25], response surface method (RSM) [26], artificial neural network (ANN) [27], or support vector regression (SVR) method [28,29]. However, in the face of high-dimensional problems, it may be impossible to establish a reliable alternative model for SRA [30,31] due to overfitting or underfitting. Relatively speaking, the accuracy of the surrogate model-assisted hybrid reliability analysis method depends largely on the reliability of the surrogate model and the selection of training sample points [32,33]. However, the accuracy of the analytical reliability method depends more on searching the MPP, that is, the point [34,35] with the smallest distance from the origin in the standard normal space. This makes the calculation cost required in the process of reliability analysis by analytical method relatively lower. Therefore, this study mainly considers the FORM.

In recent years, researchers have continued to refine and expand the capabilities of FORM. Zhang et al. [36] proposed an enhanced finite step length (EFSL) method to improve the efficiency and robustness of solving complex nonlinear reliability-based design optimization (RBDO) problems. Yang et al. [37] proposed an efficient local adaptive Kriging approximation method with a single-loop strategy (LAKAM-SLS) to enhance the computational efficiency of Kriging-based RBDO. These developments aim to enhance the accuracy, efficiency, and robustness of FORM, particularly in addressing the challenges posed by complex and high-dimensional engineering structures. By incorporating these recent advancements, the field of SRA is continuously evolving, striving to provide more reliable and efficient methods for assessing structural safety and preventing catastrophic failures.

In the past few years, researchers have done a lot of research on SRA based on analytical methods. Among them, the traditional MPP search methods include the gradient method or conjugate gradient method. Although for low-dimensional reliability analysis problems, traditional FORM often obtains stable and accurate analysis results. Studies have shown that the relaxed Hasofer–Lind and Rackwitz–Fiessler (RHL-RF) [38], and non-negative constraint method (NNCM) [39] are inefficient in solving nonlinear convex LSFs [40]. These methods often face the problem of over-reliance on parameter settings or non-convergence of results. In addition, the convex problem may also be a big challenge for gradient FORM, and its computational cost may increase significantly in some solving processes. The meta-heuristic algorithm can obtain accurate results or close to the exact solution, which avoids the problem that the gradient method cannot obtain the reliability analysis solution due to the inability to converge [41,42].

As a non-gradient optimization algorithm, the meta-heuristic algorithm solves the optimization problem by simulating biological behavior or physical processes [43]. At present, there have been many studies on heuristic algorithm-assisted FORM. Zhu et al. [40] proposed an improved particle swarm optimization (PSO) algorithm-assisted FORM. The results show that the new FORM has higher stability and computational efficiency in the solution of MPP for complex problems. Pedroso [44] combines a typical differential evolution algorithm with FORM. From the calculation results, it can be seen that it has a very good performance in finite element examples that require large-scale simulation. Zhong et al. [45] proposed a new Harris Hawk optimization algorithm (HHO). The combination of the HHO algorithm and FORM improves the efficiency and stability of solving MPP points in high-dimensional examples. In FORM, the most critical issues of non-gradient search include. 1) The accuracy and robustness of MPP search; 2) Improve the reliability analysis ability of complex engineering structure models. If the simulation strategy is difficult or even impossible to obtain effective reliability analysis results, FORM can provide acceptable analysis results with low computational difficulty; 3) Improve the computational efficiency of FORM for complex reliability analysis problems. It is still important to develop a meta-heuristic algorithm with higher global convergence performance and faster convergence efficiency, which plays an important role in improving the efficiency, accuracy, and stability of searching MPP points.

Coati optimization algorithm (COA) [46] is a heuristic optimization algorithm, which has been proved to have good global development and local search capabilities. At the same time, it is also an optimization algorithm that does not need to apply any parameters to control. However, in the process of solving MPP, problems such as inaccurate calculation results or low computational efficiency may still be encountered. In this study, the social learning strategy is used to improve the COA (SL-COA), aiming to improve computational efficiency, global convergence ability, and robustness of the algorithm. By testing the benchmark test sets CEC2005 and CEC2017 and four optimization design examples, the SL-COA algorithm is demonstrated to improve the convergence speed and robustness. Secondly, this study combines the SL-COA algorithm with FORM to construct a reliability analysis method. Finally, the applicability and effectiveness of the SL-COA algorithm in the field of reliability analysis are verified by four mathematical examples and two engineering examples.

The rest of the paper is structured as follows: In [Section 2](#), the knowledge of FORM and COA is briefly introduced. Then, in [Section 3](#), the proposed SL-COA and the novel framework are explained in detail. In [Section 4](#), both mathematical and engineering examples are conducted to test the effectiveness of the enhanced COA and proposed hybrid reliability analysis framework, and then their results are fully discussed. Ultimately, [Section 5](#) provides a conclusion and prospects for this study.

2 Review of Related Knowledge

2.1 First-Order Reliability Method

SRA plays an important role in evaluating the safety and reliability of engineering structures under load and degradation. Estimating the failure probability based on a specific LSF is the most important goal of SRA [40]:

$$P_f = \int_{g(\mathbf{x}) \leq 0} \cdots \int f_x(x_1, \dots, x_n) d_{x_1} \cdots d_{x_n} \quad (1)$$

where $g(\mathbf{x})$ denotes the LSF, $f_x(\mathbf{x})$ denotes the joint probability density function (PDF) of n -dimensional random variable set \mathbf{x} . Because of the difficulty of solving the integral, researchers have proposed a method to solve the approximate solution close to the accurate reliability index by searching MPP. As shown in Fig. 1, a schematic diagram for searching MPP in a certain probability space is given.

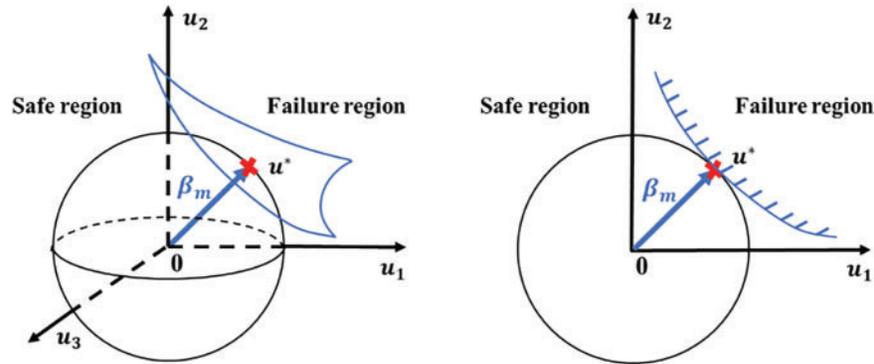


Figure 1: Schematic representation mixed model in the U -space

2.1.1 First-Order Reliability Method Based on Gradient

Eqs. (2) and (3) give the calculation formula of MPP point and reliability index by gradient method. This method is a reliability analysis method that combines the design point search based on an iterative gradient with the local linear approximation of the LSF in the standard normal probability space.

$$\mathbf{U}_{k+1} = \frac{\nabla^T g(\mathbf{U}_k) \mathbf{U}_k - g(\mathbf{U}_k)}{\nabla^T g(\mathbf{U}_k) \boldsymbol{\alpha}_k} \boldsymbol{\alpha}_{k+1} \quad (2)$$

$$\beta_{k+1} = \frac{\nabla^T g(\mathbf{U}_k) \mathbf{U}_k - g(\mathbf{U}_k)}{\nabla^T g(\mathbf{U}_k) \boldsymbol{\alpha}_k} \quad (3)$$

where $\boldsymbol{\alpha}$ denotes the normalized sensitivity vector. Furthermore, $\mathbf{U}_{k+1} = \beta_{k+1} \boldsymbol{\alpha}_{k+1}$. The iterative formulas of different gradient reliability analysis methods are different. The element in \mathbf{U}_k is $u_i = \Phi^{-1}[F_x(x_i)]$, which denotes the normal standard random variable. Eq. (4) gives the calculation formula of \mathbf{U}_k .

$$\mathbf{X}_k = \sigma^e \mathbf{U}_k + \boldsymbol{\mu}^e \quad (4)$$

where

$$\sigma^e = \frac{\varphi\{\Phi^{-1}[F_x(x_i)]\}}{f_x(x_i)} \quad (5)$$

$$\mu^e = x_i - \sigma^e \Phi^{-1} [F_x(x_i)] \tag{6}$$

where, $F_x(x_i)$ and $f_x(x_i)$ denotes the cumulative distribution function (CDF) and the PDF of original variable x_i . The calculation of α_{k+1} is the most important difference between different gradient reliability analysis methods, for example:

- 1). Hasofer–Lind and Rackwitz–Fiessler method (HL-RF) [38].

HL-RF is one of the most classic forms. The iterative formula of this method is given by Eq. (7).

$$\alpha_k = -\frac{\nabla g(\mathbf{U}_k)}{\|\nabla g(\mathbf{U}_k)\|} \tag{7}$$

- 2). Conjugate HL-RF algorithm [2]

To improve the performance of HL-RF, researchers proposed a conjugate HL-RF algorithm. α_k can be computed as follows:

$$\alpha_k = -\frac{\mathbf{U}_k + \lambda_k \mathbf{d}_k}{\|\mathbf{U}_k + \lambda_k \mathbf{d}_k\|} \tag{8}$$

where λ can be computed by Eq. (9).

$$\lambda_{k+1} = \begin{cases} \lambda_k/c \|\mathbf{U}_{k+1} - \mathbf{U}_k\| \geq \|\mathbf{U}_k - \mathbf{U}_{k-1}\| \\ \lambda_k \|\mathbf{U}_{k+1} - \mathbf{U}_k\| < \|\mathbf{U}_k - \mathbf{U}_{k-1}\| \end{cases} \tag{9}$$

where $1.2 < c < 1.5$. The calculation formula of conjugate gradient vector \mathbf{d}_k is shown as Eq. (10).

$$\mathbf{d}_{k+1} = -\nabla g(\mathbf{U}_k) - \frac{\|\nabla g(\mathbf{U}_k)\|^2}{\|\nabla g(\mathbf{U}_{k-1})\|^2} \mathbf{d}_{k-1} \tag{10}$$

where $\mathbf{d}_0 = \mathbf{0}$. With the help of scalar factor $\frac{\|\nabla g(\mathbf{U}_k)\|^2}{\|\nabla g(\mathbf{U}_{k-1})\|^2}$, α_k no longer coincides with the sensitivity vector of the previous iteration, reducing the periodicity risk.

- 3). Finite step length method (FSL) [36]

The radial sensitivity vector in FSL is calculated by Eq. (11).

$$\alpha_{k+1} = \frac{\mathbf{U}_k - \lambda \nabla g(\mathbf{U}_k)}{\|\mathbf{U}_k - \lambda \nabla g(\mathbf{U}_k)\|} \tag{11}$$

where λ is bigger than 0. In the iterative process, the calculation formula is the same as that shown in Eq. (9). The iterative instability of highly nonlinear LSF is controlled by changing factor c .

2.1.2 First-Order Reliability Method Based on Heuristic Algorithm

The computational efficiency and stability of the gradient-based FORM are good for low and nonlinear reliability analysis problems. However, in the face of moderate nonlinear problems and moderate dimensional reliability problems, acceptable reliability analysis results may not be obtained. The heuristic algorithm is very effective in obtaining acceptable reliability analysis results for complex problems. By transforming the reliability analysis problem into the following Eq. (12):

$$\min_x f(\mathbf{x}) = \|\mathbf{U}_k\| + \rho \max\{0, g(\mathbf{x})\} \tag{12}$$

where $f(\mathbf{x})$ denotes the fitness function. $g(\mathbf{x})$ denotes the LSF. ρ is the penalty factor [5]. It can be seen from Eq. (12) that the method improves the efficiency of solving the reliability index by solving the MPP in the failure domain.

Fig. 2 shows the calculation flow of the first-order reliability method using the heuristic algorithm.

Step 1: Initialize the heuristic algorithm and initialize the relevant parameters.

Step 2: Build the fitness function based on the reliability analysis problem by Eq. (12).

Step 3: A heuristic optimization algorithm is used to optimize the fitness function.

Step 4: Use the Eq. (13) to determine whether the current optimization results are accurate. $\varepsilon = 1 \times 10^{-6}$ is used in the current study. If the current optimization result does not satisfy Eq. (13), implement Step 5; Otherwise, Step 6 is implemented.

$$(\|\beta_k - \beta_{k-1}\|)/\beta_k < \varepsilon \quad (13)$$

Step 5: Update related parameters such as the penalty factor and return to Step 2.

Step 6: Output the current reliability analysis result.

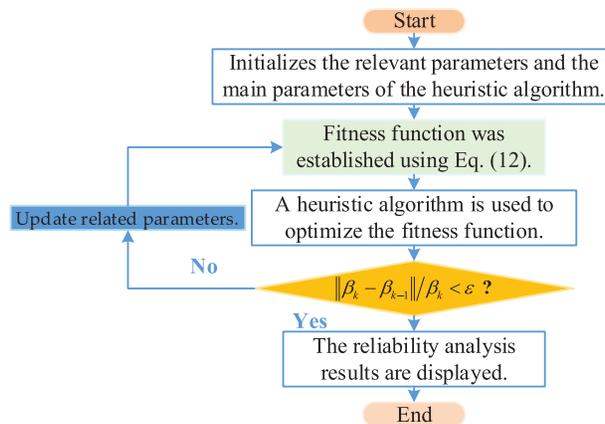


Figure 2: First-order reliability method flow based on the heuristic algorithm

2.2 Coati Optimization Algorithm

The batch normalization (BN) layer is an optimization method widely used in deep neural networks proposed by Ioffe [47] aiming to improve the training speed of models and reduce the impact of gradient vanishing and exploding problems. The BN layer normalizes the inputs between network layers, giving them zero mean and unit variance, thereby enhancing training stability. In CNNs, BN layers are typically inserted between convolutional layers and activation function layers. The COA is a novel metaheuristic algorithm introduced by Dehghani in 2022 [46]. It sets up a mathematical model to mimic the behavior of coati in nature. The basic idea of COA is the simulation of the two natural behaviors of coatis:

- i. Capturing and pursuing iguanas,
- ii. Evasion of potential threats.

Correspondingly, the execution process of COA is elucidated and mathematically represented in two distinct phases: exploration and exploitation.

Initialization: The initial positions of the coatis in the search space are randomly set using:

$$X_i: x_{i,j} = lb_j + r \cdot (ub_j - lb_j), i = 1, 2, \dots, N, j = 1, 2, \dots, m \tag{14}$$

where X_i is the position of the i th coati in search space, $x_{i,j}$ is the value of the j th decision variable, N is the number of coatis, m is the number of decision variables, r is a random real number between 1 and 2, and lb_j and ub_j are the lower bound and upper bound of the j th decision variable, respectively. After initialization, the individual coati starts to perform the two phases.

Phase 1: Hunting strategy (exploration phase)

The initial stage of population adjustment among the coatis within the search space involves simulating their behaviors when facing iguana attacks. It is assumed that half of the coatis choose to climb the tree, while the other half opt to stay on the ground, anticipating the iguana's descent. This methodology encourages coatis to migrate to different positions within the search space, highlighting the ability of COA to explore the entire problem-solving domain globally.

Climbing Coatis: It is postulated that the best member of the population occupies the position of the iguana. Therefore, the position of the coatis rising from the tree is mathematically simulated using Eq. (15).

$$X_i^{P1}: x_{i,j}^{P1} = x_{i,j} + r \cdot (Iguana_j - I \cdot x_{i,j}),$$

For

$$i = 1, 2, \dots, \left\lfloor \frac{N}{2} \right\rfloor \text{ and } j = 1, 2, \dots, m. \tag{15}$$

where X_i^{P1} is the new position calculated for the i th coati, $x_{i,j}^{P1}$ is the value of its j th dimension, r is random variable in the interval $[0, 1]$, $Iguana_j$ represents the iguana's position in the j th dimension, I is an integer, which is randomly selected from the set $\{1, 2\}$, and $\lfloor \cdot \rfloor$ is the floor function.

Ground Coatis: Once the iguana descends to the ground, it is positioned at a random location within the search space. Subsequently, the coatis on the ground navigates the search space, a process replicated using Eqs. (16) and (17).

$$Iguana^G: Iguana_j^G = lb_j + r \cdot (ub_j - lb_j), j = 1, 2, \dots, m \tag{16}$$

$$X_i^{P1}: x_{i,j}^{P1} = \begin{cases} x_{i,j} + r \cdot (Iguana_j^G - I \cdot x_{i,j}), & F_{Iguana}^G < F_i \\ x_{i,j} + r \cdot (x_{i,j} - Iguana_j^G), & \text{else} \end{cases}$$

For

$$i = \left\lfloor \frac{N}{2} \right\rfloor + 1, \left\lfloor \frac{N}{2} \right\rfloor + 2, \dots, N \text{ and } j = 1, 2, \dots, m \tag{17}$$

where $Iguana^G$ represents the position of the iguana on the ground, F is the value of the objective function.

If a newly calculated position for each coati improves the value of the objective function, it is considered suitable for the update process; otherwise, the coati maintains its current position. This update criterion applies to all coatis with indices $i = 1, 2, \dots, N$, as defined by Eq. (18).

$$X_i = \begin{cases} X_i^{P1}, & F_i^{P1} < F_i \\ X_i, & \text{else} \end{cases} \quad (18)$$

Phase 2: Escaping strategy (exploitation phase)

The second stage of updating the coati positions is mathematically structured to mirror the natural response of coatis when they encounter and evade predators. When faced with a predator's attack, coatis adopt a strategy that leads them to a safer spot near their current position. This behavior underscores the coatis' capacity for local search and exploiting their surroundings.

Local Position Update: To emulate this behavior, a random position is generated in close proximity to the current location of each coati, as determined by Eq. (19).

$$lb_j^{local} = \frac{lb_j}{t}, \quad ub_j^{local} = \frac{ub_j}{t}, \quad \text{where } t = 1, 2, \dots, T$$

$$X_i^{P2}: x_{i,j}^{P2} = x_{i,j} + (1 - 2r) \cdot (lb_j^{local} + r \cdot (ub_j^{local} - lb_j^{local})),$$

$$i = 1, 2, \dots, N, \quad j = 1, 2, \dots, m \quad (19)$$

where lb_j^{local} and ub_j^{local} represent the local lower and upper bounds for the j th decision variable, lb_j and ub_j denote the lower bound and upper bound of the j th decision variable, t is the iteration counter, X_i^{P2} is the new position calculated for the i th coati on the second phase of COA and $x_{i,j}^{P2}$ is the value of its j th dimension.

Position Acceptance: The freshly computed position is considered acceptable if it enhances the value of the objective function, which is assessed using the condition expressed in Eq. (20).

$$X_i = \begin{cases} X_i^{P2}, & F_i^{P2} < F_i \\ X_i, & \text{else} \end{cases} \quad (20)$$

where F is the objective function.

One iteration of the COA is completed after updating the positions of all coatis in the search space during the first and second phases. This cycle continues until the final iteration of the algorithm is reached. At the end of the COA run, the best solution obtained throughout all iterations of the algorithm is presented as the output. The various steps involved in the COA implementation are illustrated in Fig. 3 through a flowchart.

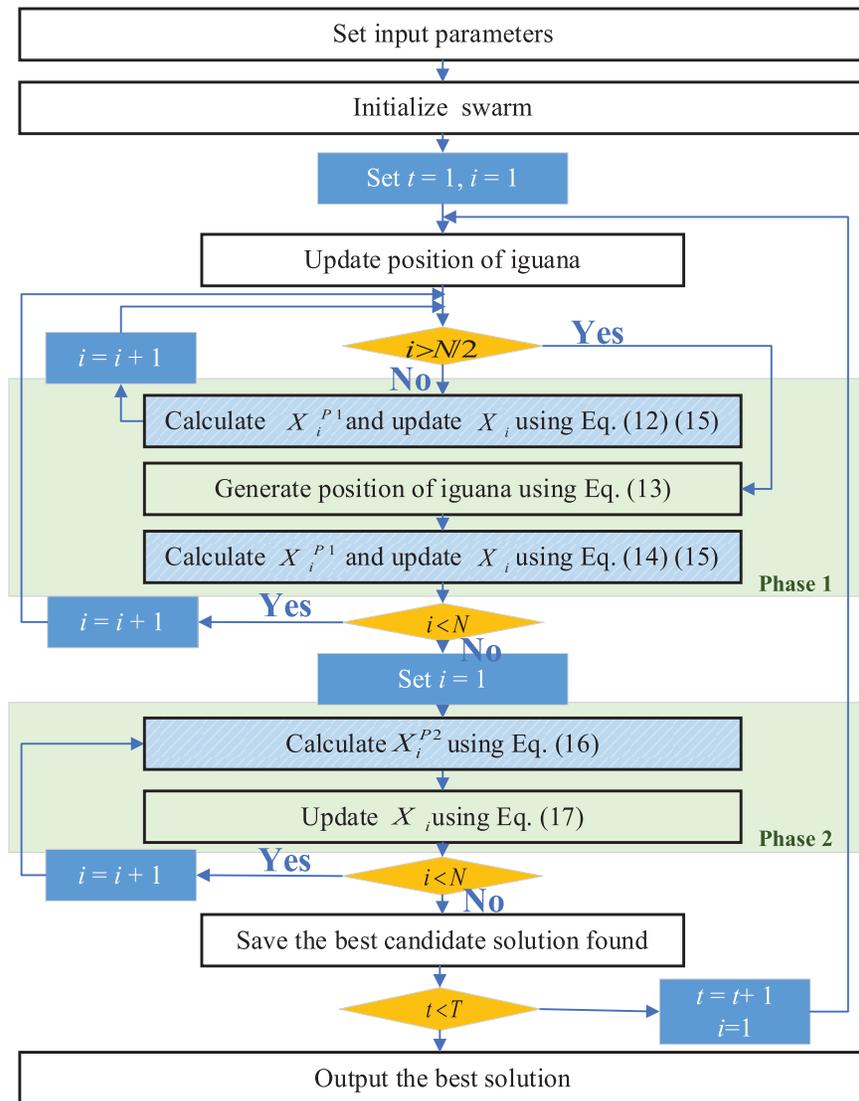


Figure 3: Block diagram of COA

3 Improved COA-Based FORM

In this section, the proposed hybrid reliability analysis model is discussed. Firstly, the improved COA is explained, and after that, the framework of this analysis model is defined and illustrated.

3.1 Social Learning-Adapted COA

The status monitoring of the forging press machine uses 4 types of sensors, including 4 channels of displacement signals, 4 channels of impact force signals, 2 channels of hydraulic signals, and 2 channels of air pressure signals. Each set of data collected consists of one sample with 12 channels. According to the duration of approximately 8 s for each forging operation on the forging press machine, the length for each signal is set at 8000 data points with a sampling frequency of 1000 Hz.

This paper introduces a novel method for COA enhancement called SL-COA. Social learning is a basic behavior of the biotic population. In the field of swarm intelligence algorithms, this concept is initially

explained in another bioinspired algorithm called PSO [48]. It is a swarm intelligence algorithm proposed by Kennedy and Eberhart in 1995, inspired by the social behavior of birds and fish. The core concept of the algorithm is as follows.

Considering a D -dimensional space consists of n particles, the velocity and position of each particle are defined as

$$V_i = (v_{i1}, v_{i2}, \dots, v_{iD}), i = 1, 2, \dots, N \quad (21)$$

$$X_i = (x_{i1}, x_{i2}, \dots, x_{iD}), i = 1, 2, \dots, N \quad (22)$$

In each iteration, the above expression is updated as

$$V_{id} = wV_{id-1} + c_1r_1(P_{id} - X_{id}) + c_2r_2(P_{gd} - X_{id}) \quad (23)$$

$$X_{id+1} = X_{id} + V_{id}t \quad (24)$$

where V_{id} is the velocity in d th iteration, P_{id} is the best position of an individual particle, and P_{gd} is the best position of the whole swarm, w is inertia weight and c is learning factor.

This expression elucidates the concept that an individual's velocity within a group can be characterized as a blend of influences from their prior velocity, self-awareness, and the collective influence of the group. Actually, this idea is also suitable for most of the swarm intelligence algorithms [48–51]. In this paper, this idea is used to improve the learning process of the behavior of coatis. That is the Eqs. (15), (17) and (19).

There are two types of indexes related to the swarm learning process in PSO, that is inertia weight w and learning factor c . The inertia weight is a parameter measured by the influence of the prior velocity of the swape. A higher value of w results in less frequent velocity changes, as it diminishes the impact of velocity adjustments. The learning factor is employed to fine-tune both self-awareness and the collective impact of the group. Conversely, a greater value of c leads to a more pronounced velocity change [49,50]. As for COA, the learning factor c is adopted in both phases 1 and 2. It improves the convergence speed in *Phase 1* and regulates the swarm diversity in *Phase 2*, preventing premature convergence to suboptimal solutions.

In PSO, the learning index is treated as a constant. However, it should be noticed that as the iterations progress, the disparity between each individual varies. In response to this variation, the learning index should be mathematically linked to it [49]. Consequently, a variable learning index is introduced as $c = c_{\max} - (c_{\max} - c_{\min}) \frac{t}{T}$, where t is the current iteration, T is total iteration, c_{\max} and c_{\min} are the maximum and minimum learning index respectively. This expression represents that with iterations progress, the learning index gradually decreases from c_{\max} to c_{\min} .

Therefore, the updated process (Eqs. (15), (17), and (19)) of COA is changed to Eqs. (25)–(27), respectively.

$$X_i^{P1}: x_{i,j}^{P1} = x_{i,j} + cr \cdot (Iguana_j - I \cdot x_{i,j}) \quad (25)$$

$$X_i^{P1}: x_{i,j}^{P1} = \begin{cases} x_{i,j} + cr \cdot (Iguana_j^G - I \cdot x_{i,j}), & F_{Iguana}^G < F_i \\ x_{i,j} + cr \cdot (x_{i,j} - Iguana_j^G), & else \end{cases} \quad (26)$$

$$X_i^{P2}: x_{i,j}^{P2} = x_{i,j} + c(1 - 2r) \cdot (lb_j^{local} + r \cdot (ub_j^{local} - lb_j^{local})) \quad (27)$$

The block diagram of the algorithm is shown in Fig. 4.

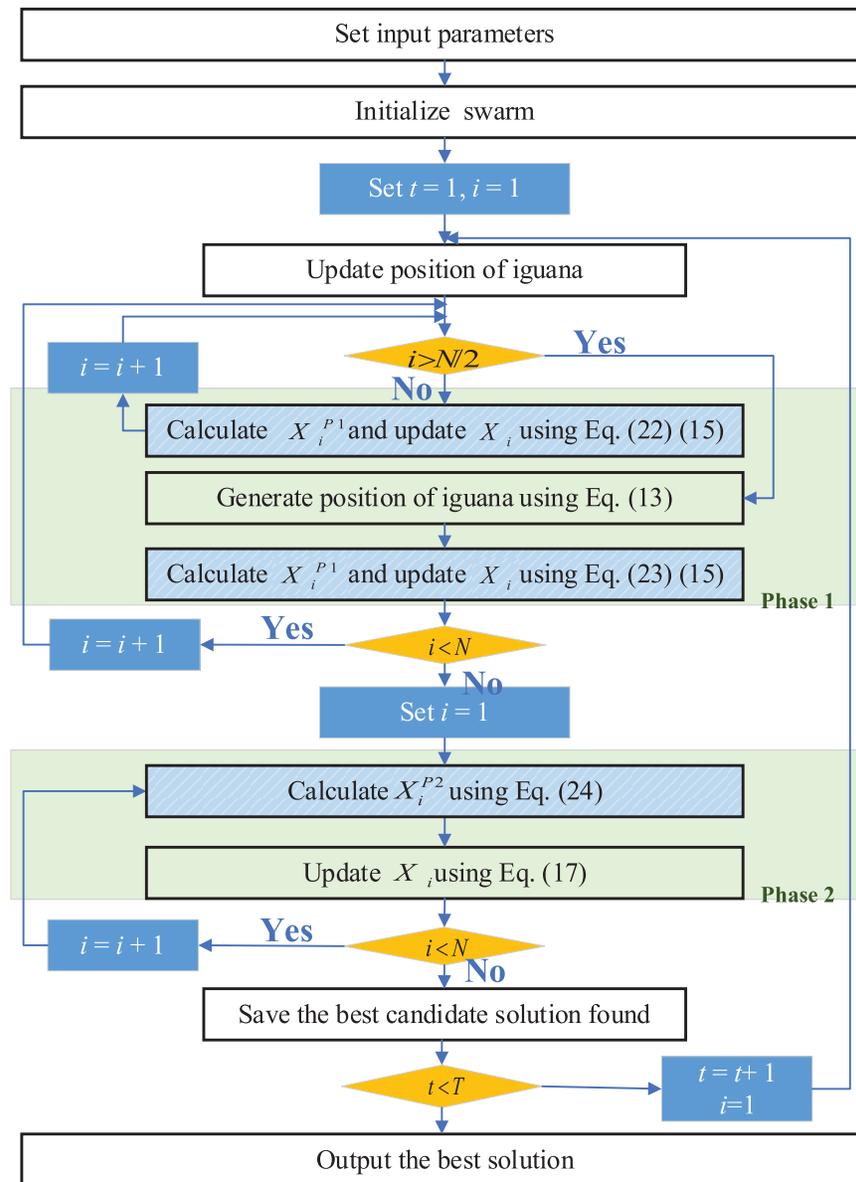


Figure 4: Block diagram of SL-COA

3.2 The Proposed Framework for FORM

By combining the improved algorithm with FORM, a new reliability analysis framework is constructed. The framework applies the SL-COA algorithm to the MPP search process and then converts the output results into reliability index β and failure probabilities P_f . The proposed reliability analysis framework is shown in Fig. 5.

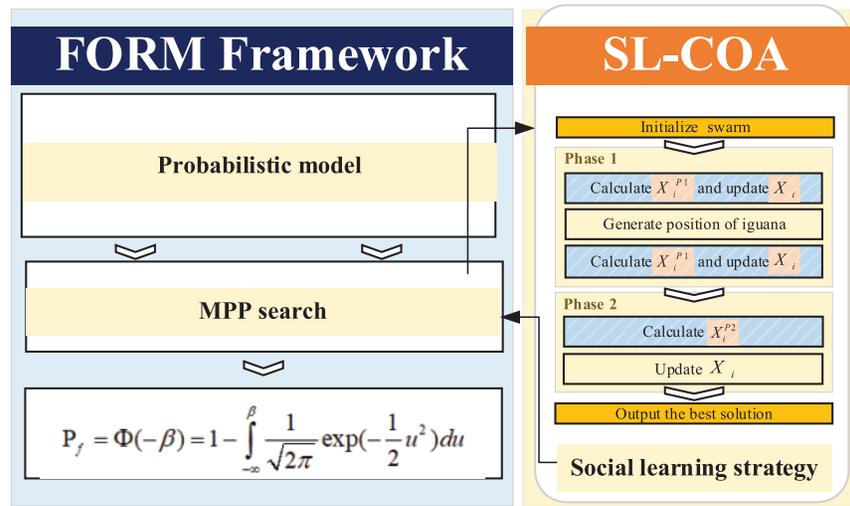


Figure 5: Block diagram of SL-COA-based FORM framework

4 Study Case and Discussion

In the following section, the performance of the improved optimization algorithm and the proposed reliability analysis framework are tested in sequence.

4.1 Examples for Illustration

In this example, the objective function shown in Eq. (28) is optimized to compare the effects of different learning factors on the performance of the optimization algorithm SL-COA. Among them, the number of populations is set to 20, and the maximum number of iterations is set to 50.

$$F(\mathbf{x}) = \sum_{i=1}^{100} -x_i \sin\left(\sqrt{-x_i \sin(\sqrt{|x_i|})}\right) \quad (28)$$

where variable x is the normal distribution variable. The lower bound of the variable is 500. The upper bound of the variable is 500. Table 1 shows the optimization results of different learning factors c and adaptive learning factors after 50 times of optimization.

Table 1: Optimization results in different learning factors

Methods	Min	Average	Max	Cov
COA	-41,898.28	-41,569.50	-16,037.74	0.1300
$c = 2.5$	-41,898.29	-41,415.68	-32,337.47	0.0378
$c = 2$	-41,898.29	-41,515.93	-30,054.15	0.0410
$c = 1.5$	-41,898.29	-41,749.64	-38,592.40	0.0123
SL-COA $c = 1$	-41,898.11	-41,513.82	-30,049.21	0.0402
$c = 0.5$	-41,898.23	-38,039.66	-18,341.69	0.1561
$c = 0.01$	-15,465.81	-12,070.21	-9259.90	0.0969
$c = c_{\max} - (c_{\max} - c_{\min}) \frac{t}{T}$	-41,898.29	-41,751.04	-40,917.10	0.0050

where $c_{\max} = 2.1$, $c_{\min} = 0.8$. It can be seen from the results in the table that the stability of the optimal solution obtained by using a fixed learning factor is relatively low. Comparing the average value of the optimal solution obtained in the repeated experiments, it can be found that the searchability of the SL-COA algorithm decreases as the learning factor c decreases. It can be seen that the optimal solution obtained by the proposed adaptive learning factor change strategy has higher accuracy and higher stability.

4.2 Examples of Optimization Procedure

The optimization performance of SL-COA is firstly evaluated under some mathematical and engineering cases. This discussion is conducted by comparing with another metaheuristic algorithm. That includes WOA (2016) [52], OOA (2023) [53], Rime Optimization Algorithm (RIME) algorithm (2023) [54], the multi-verse optimizer (MVO) (2015) [55], the tree-seed algorithm (TSA) (2015) [56] as well as its original version COA (2023) [46].

4.2.1 Mathematical Examples

Firstly, CEC2005 is used to justify the performance of the SL-COA. There are totally 25 functions in this setting: Unimodal function ($f_1 \sim f_5$), Basic multimodal function ($f_6 \sim f_{12}$), Extended multimodal function ($f_{13} \sim f_{14}$) and Mixed composite function ($f_{15} \sim f_{25}$). Among these problems, $f_{15} \sim f_{25}$ are believed to have most difficulties due to its complex structure. In this section, $f_1 \sim f_{14}$ are used to evaluate the basic optimization abilities of the proposed algorithm. The search history, trajectory curve, average curve as well as convergence curve of SL-COA is illustrated in Fig. 6. These problems vary in complexity and dimensionality, making them suitable for evaluating the performance of optimization algorithms across a wide range of scenarios.

The results demonstrate that SL-COA effectively identifies optimal solutions for each function. The search history graph, which displays position swaps after each iteration, exhibits a high concentration near the optimization point in the search space. The trajectory curve, representing the fitness of the first dimension, steadily decreases as the iterations progress. It initially exhibits steep changes but tends to stabilize towards the end of the optimization process. The average curve, depicting the mean fitness of all search candidates, and the convergence curve, showcasing the best individual's performance, collectively provides an overview of the optimization's overall performance. It's important to observe that when testing the proposed SL-COA under function F7, there is a fluctuation throughout the entire iteration process, as evidenced by the trajectory curve. In this case, the vertical coordinates of the trajectory curves represent the function value, and the horizontal coordinates represent the iteration number. However, this fluctuation remains within the range of $[-1, 0]$, and considering the good performance of the entire swarm, it can be deemed acceptable on the whole.

After that, CEC2017 is used to future evaluate the performance of the optimization. The function details of this testing set is listed in Table 2. CEC2017 contains 30 optimization problems, categorized into four groups: unimodal functions, multimodal functions, hybrid functions, and composition functions. Similar to the CEC2005 test set, these problems are designed to be challenging and representative of real-world optimization problems. The CEC2017 test set includes more complex and high-dimensional problems compared to the CEC2005 test set, making it a more rigorous benchmark for evaluating the performance of optimization algorithms.

The key parameters of each algorithm are listed in Table 3. Additionally, the size of the population is set to 30, the maximum iteration number is 1000. Each function is tested 50 times, and the average, standard deviation and errors of the results are calculated.

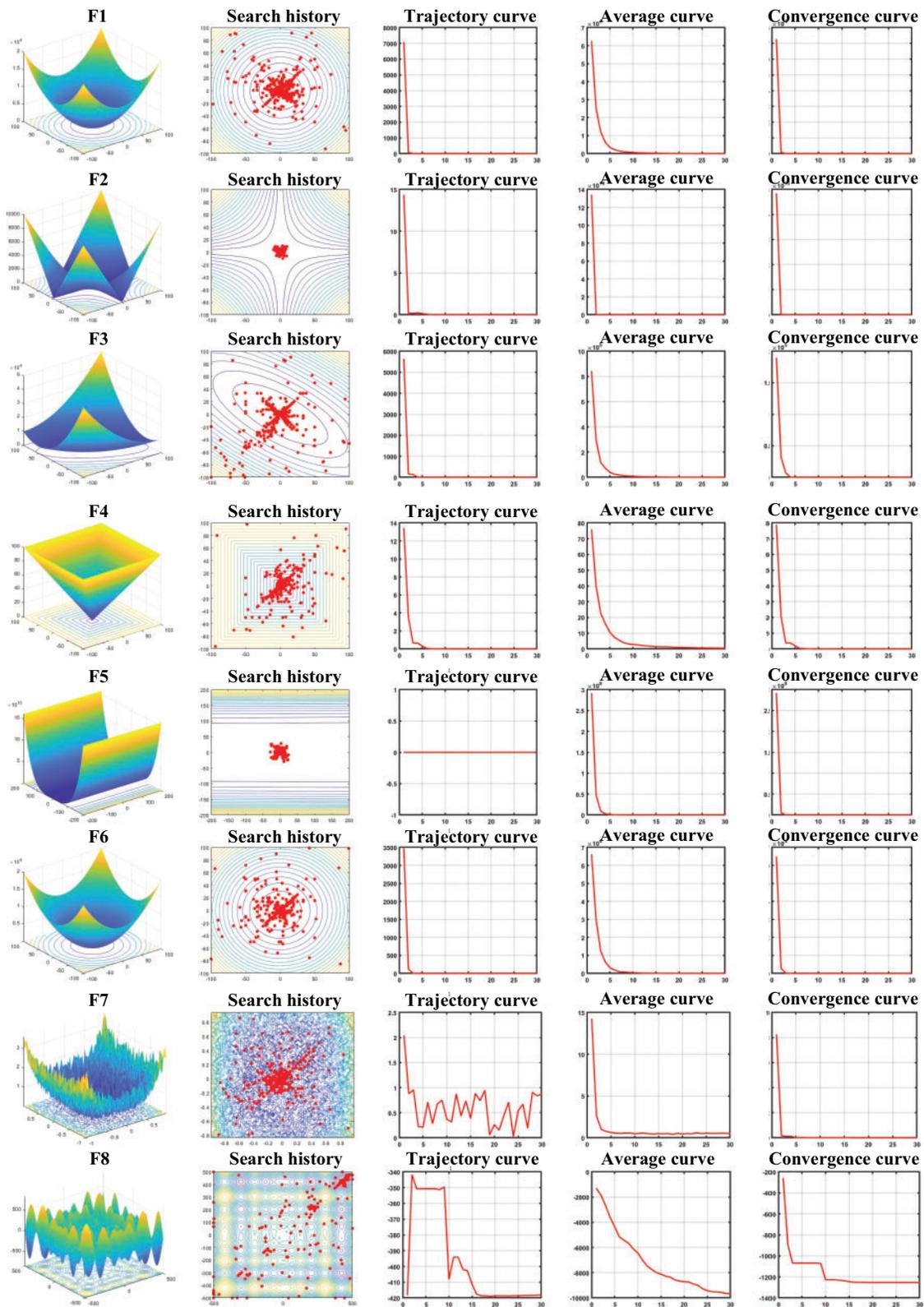


Figure 6: (Continued)

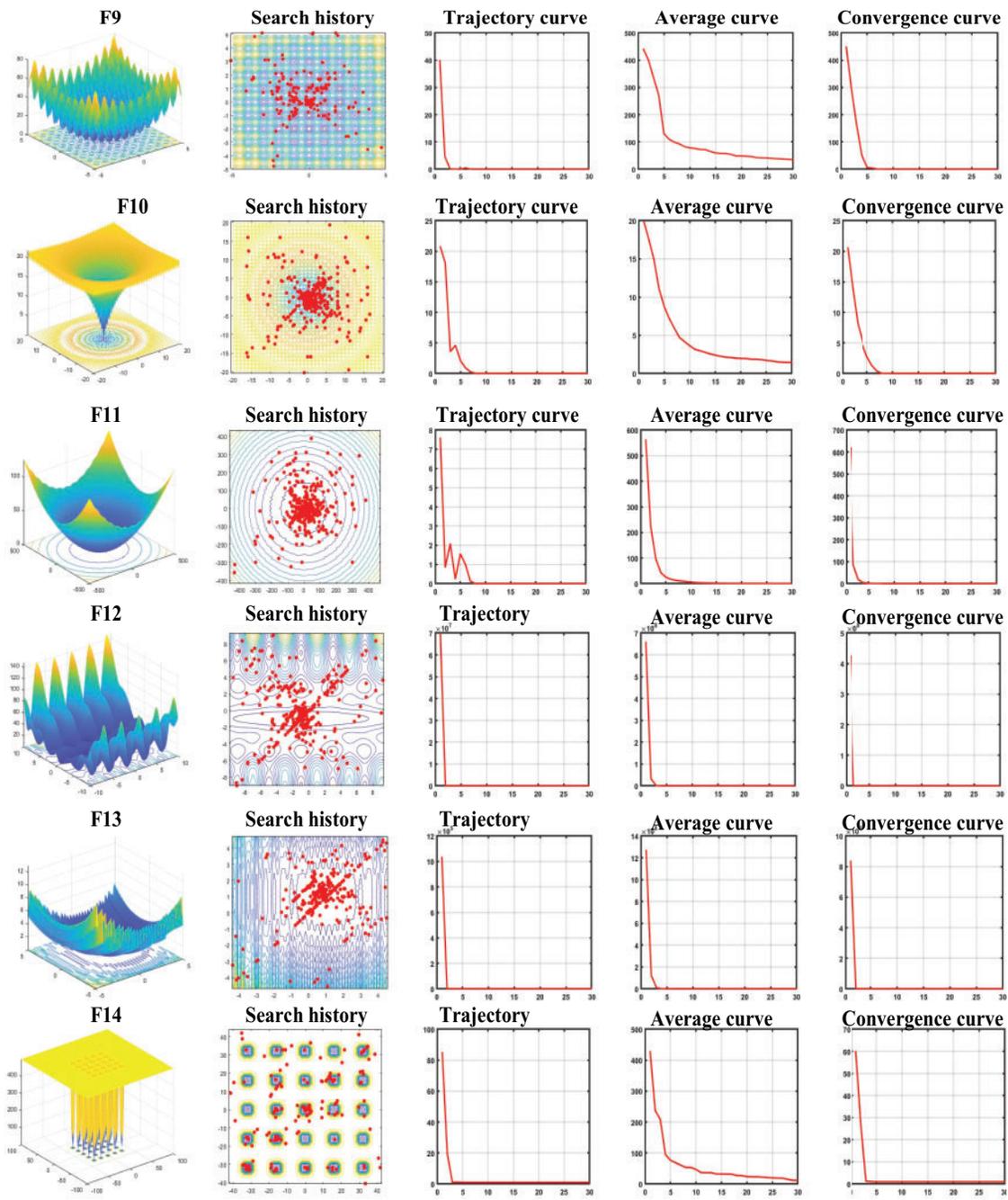


Figure 6: Test results for CEC2005

Table 2: Function details of CEC2017

Types	No.	Description	f
Unimodal functions	F3	Shifted and Rotated Zakharov function	300
	F4	Shifted and Rotated Rosenbrocks function	400
	F5	Shifted and Rotated Rastrigins function	500
	F6	Shifted and Rotated Expanded Scaffers F6 function	600
Multimodal functions	F7	Shifted and Rotated Lunacek Bi_Rastrigin function	700
	F8	Shifted and Rotated Non-Continuous Rastrigins function	800
	F9	Shifted and Rotated levy function	900
	F10	Shifted and Rotated Schwefels function	1000
Hybrid functions	F11	Hybrid function 1 ($N = 3$)	1100
	F12	Hybrid function 2 ($N = 3$)	1200
	F13	Hybrid function 3 ($N = 3$)	1300
	F14	Hybrid function 4 ($N = 4$)	1400
	F15	Hybrid function 5 ($N = 4$)	1500
	F16	Hybrid function 6 ($N = 4$)	1600
	F17	Hybrid function 6 ($N = 5$)	1700
	F18	Hybrid function 6 ($N = 5$)	1800
	F19	Hybrid function 6 ($N = 5$)	1900
	F20	Hybrid function 6 ($N = 6$)	2000
Composition functions	F21	Composition function 1 ($N = 3$)	2100
	F22	Composition function 2 ($N = 3$)	2200
	F23	Composition function 3 ($N = 4$)	2300
	F24	Composition function 4 ($N = 4$)	2400
	F25	Composition function 5 ($N = 5$)	2500
	F26	Composition function 6 ($N = 5$)	2600
	F27	Composition function 7 ($N = 6$)	2700
	F28	Composition function 8 ($N = 6$)	2800
	F29	Composition function 9 ($N = 3$)	2900
	F30	Composition function 10 ($N = 3$)	3000

Table 3: Parameter values for heuristic algorithms

Algorithms	Parameters
SL-COA	$c_{max} = 2.1; c_{min} = 0.8$
COA	/
WOA	$a_1 = [2, 0]; a_2 = [-2, -1]; b = 1$
OOA	/
RIME	$W = 5;$
MVO	$MEP_{Max} = 1; WEP_{Min} = 0.2$
TSA	$c_1 = c_2 = c_3 = [0, 1]; P_{min} = 1; P_{max} = 4$

The output results are analyzed to compute their mean value and standard deviation. Subsequently, each algorithm is assigned a ranking based on the magnitude of their errors. A comprehensive breakdown of the results can be found in the “CEC2017 Results” appendix. The sum of rankings across all functions and the mean value for each algorithm are presented in Table 4. Additionally, the convergence curve of each algorithm is shown in Fig. 7 and the box diagram illustrates the distribution of optimization results shown in Fig. 8.

Table 4: Rank results based on the errors of the heuristic algorithm under CEC2017

	SL-COA	COA	WOA	OOA	RIME	MVO	TSA	FORM
Sum	46	81	140	171	75	86	179	156
Mean	1.7037	3.0000	5.1852	6.3333	2.7778	3.1852	6.6296	5.7939
Total rank	1	3	5	7	2	4	8	6

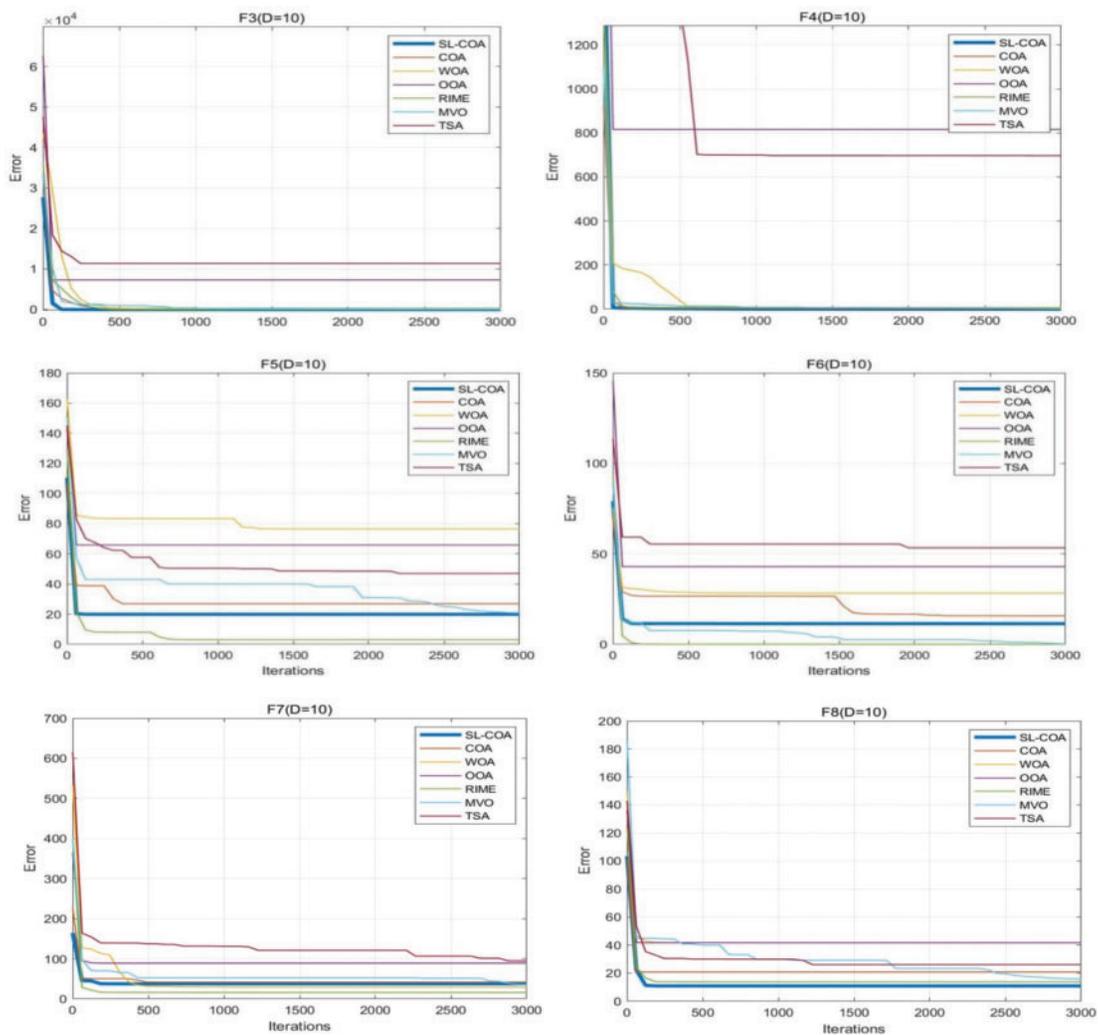


Figure 7: (Continued)

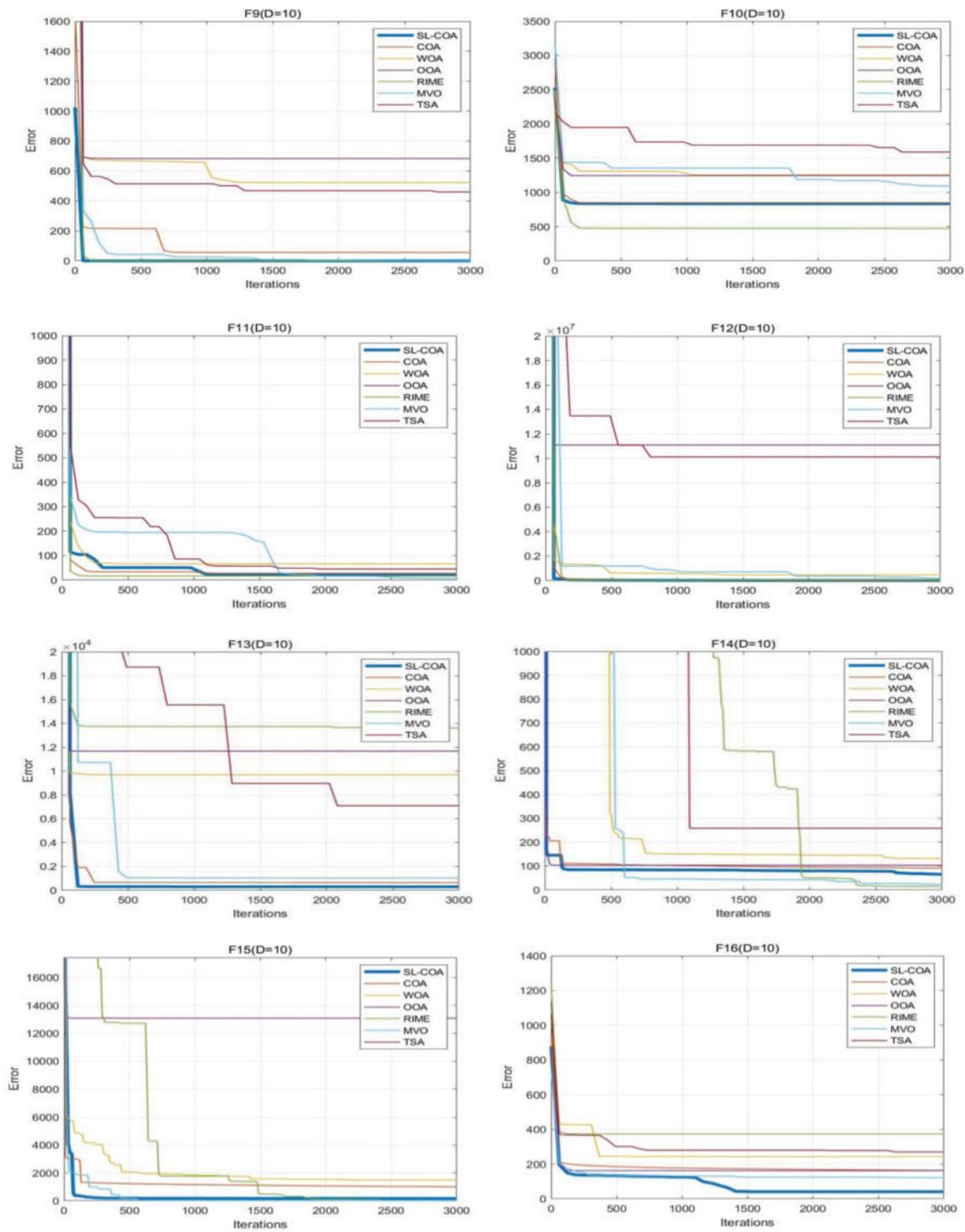


Figure 7: (Continued)

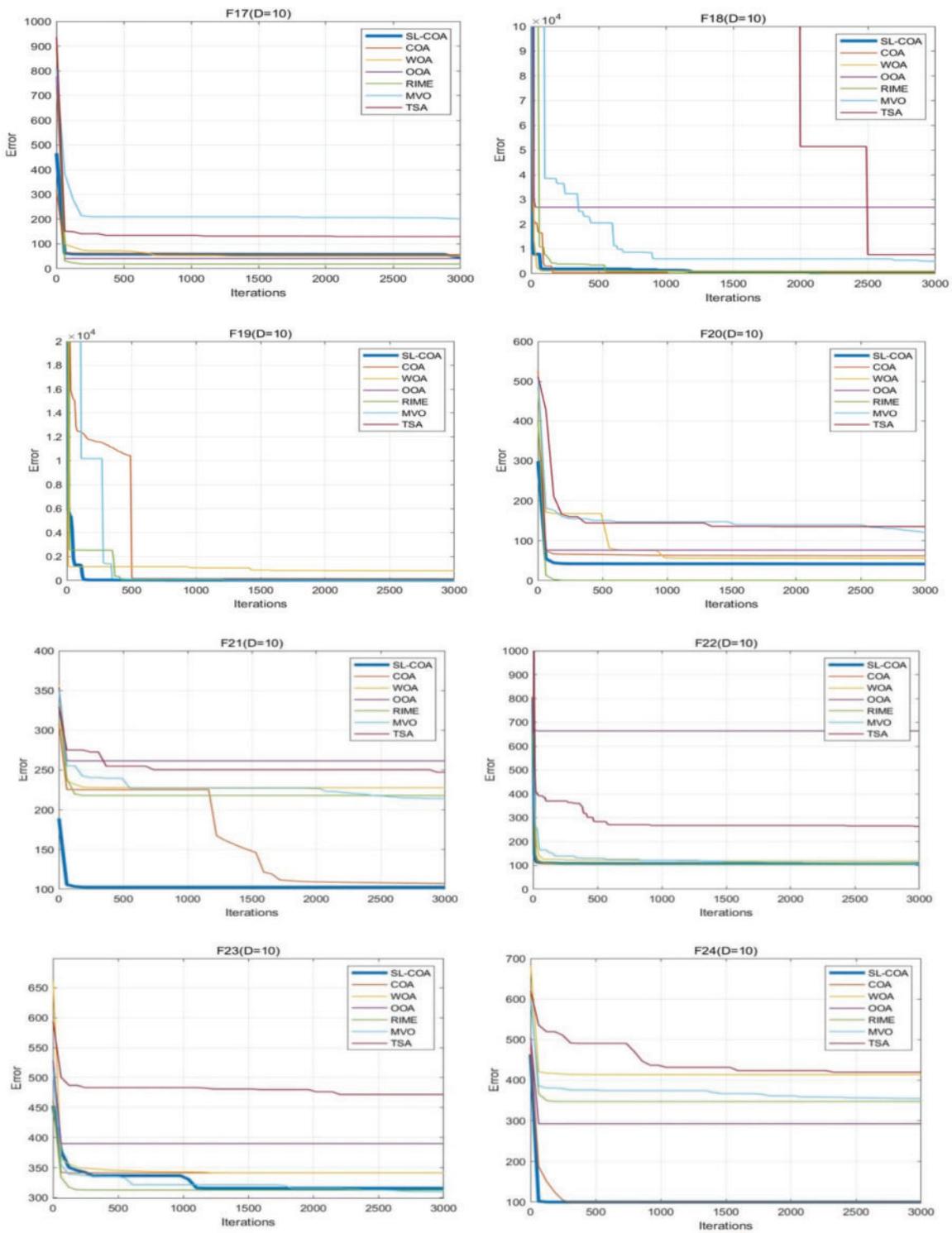


Figure 7: (Continued)

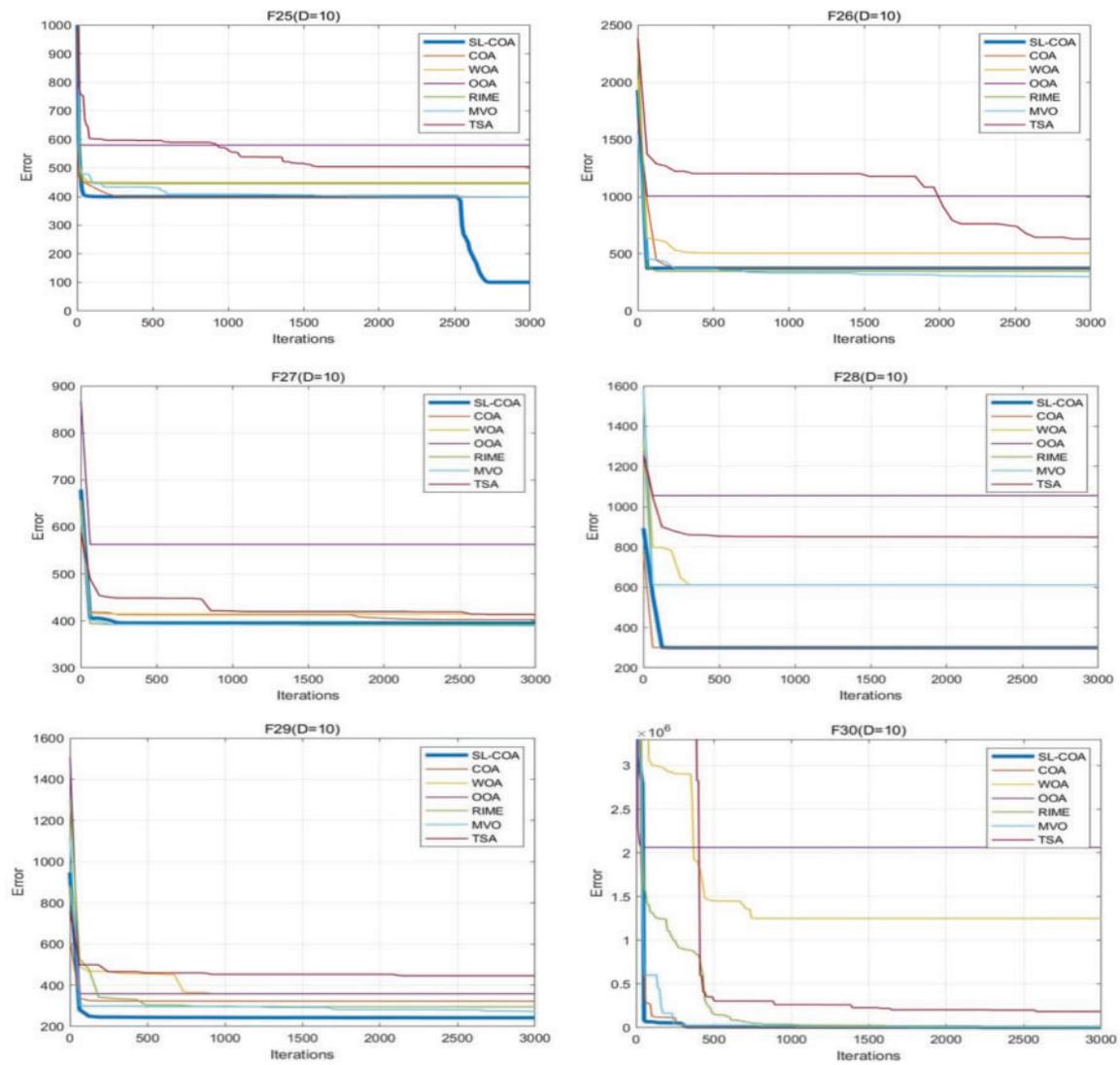


Figure 7: Convergence curve of CEC2017

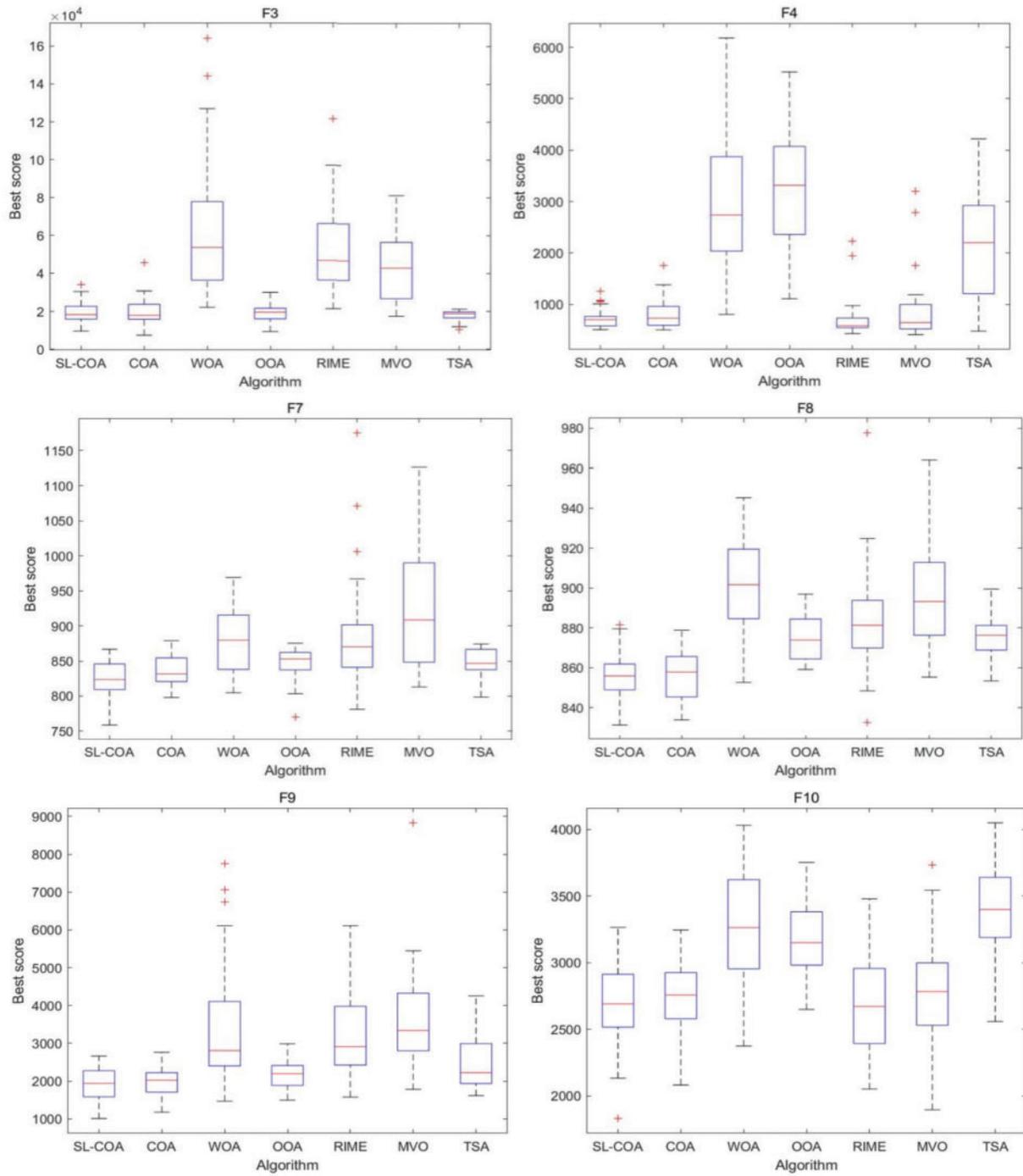


Figure 8: (Continued)

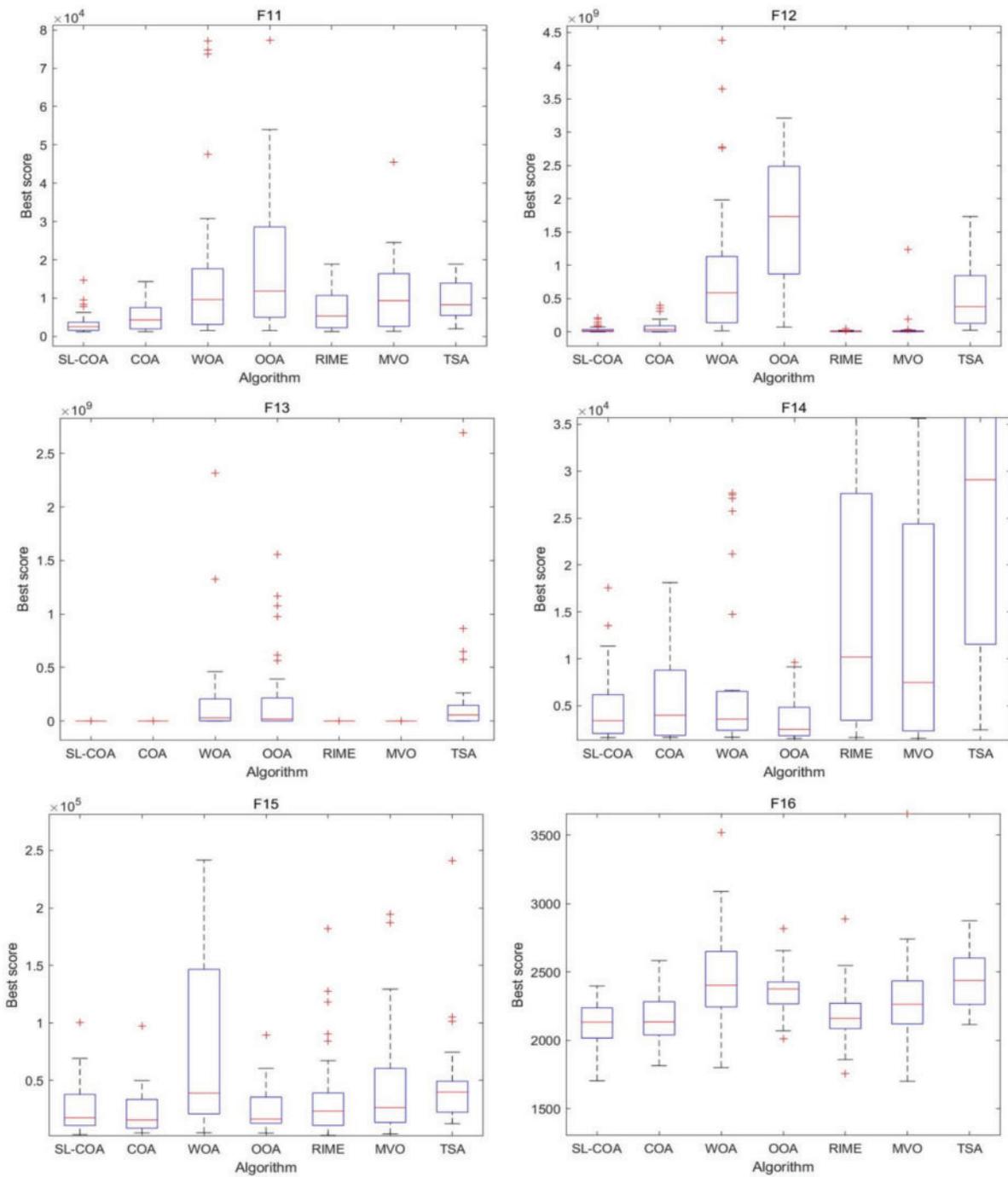


Figure 8: (Continued)

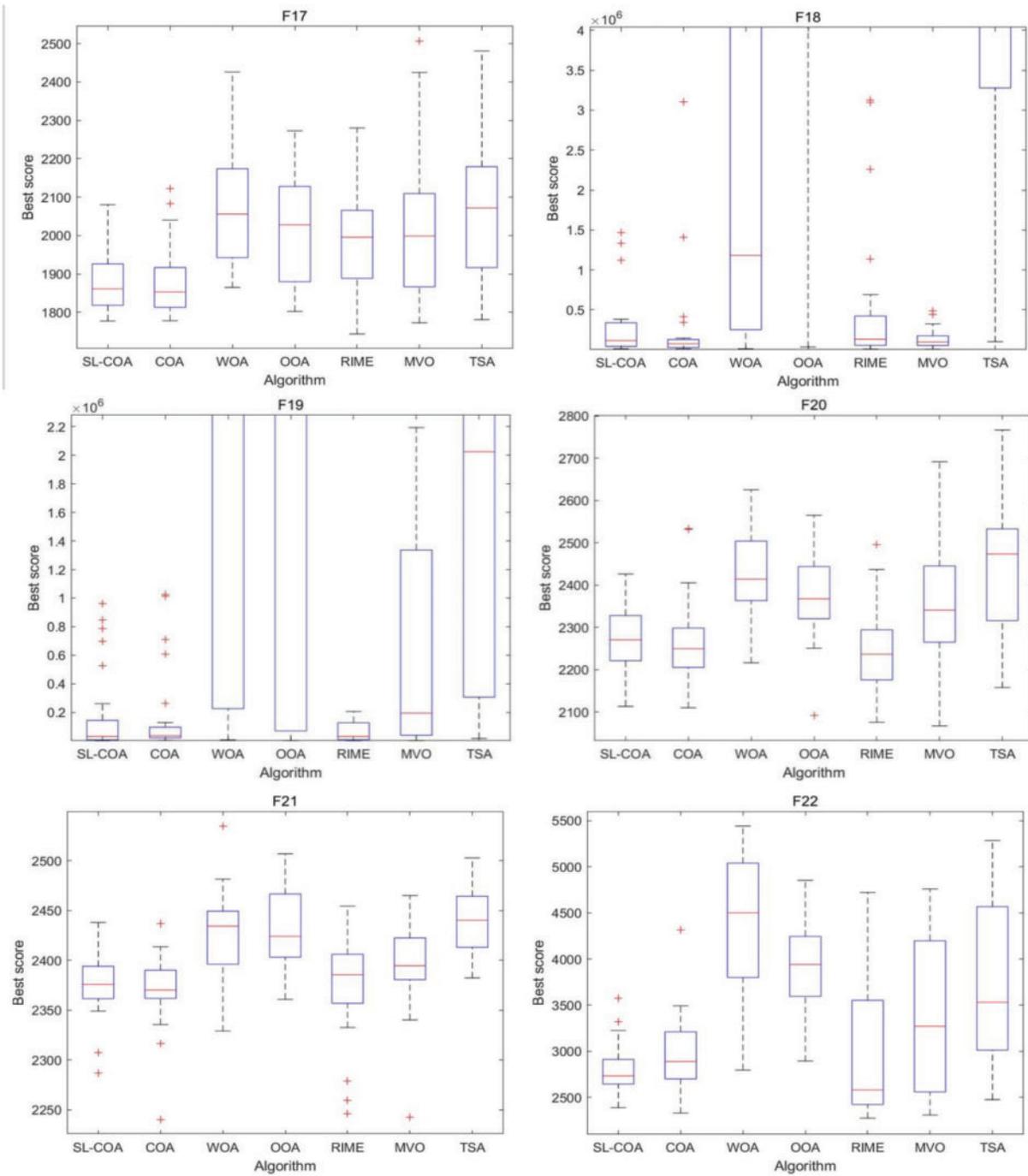


Figure 8: (Continued)

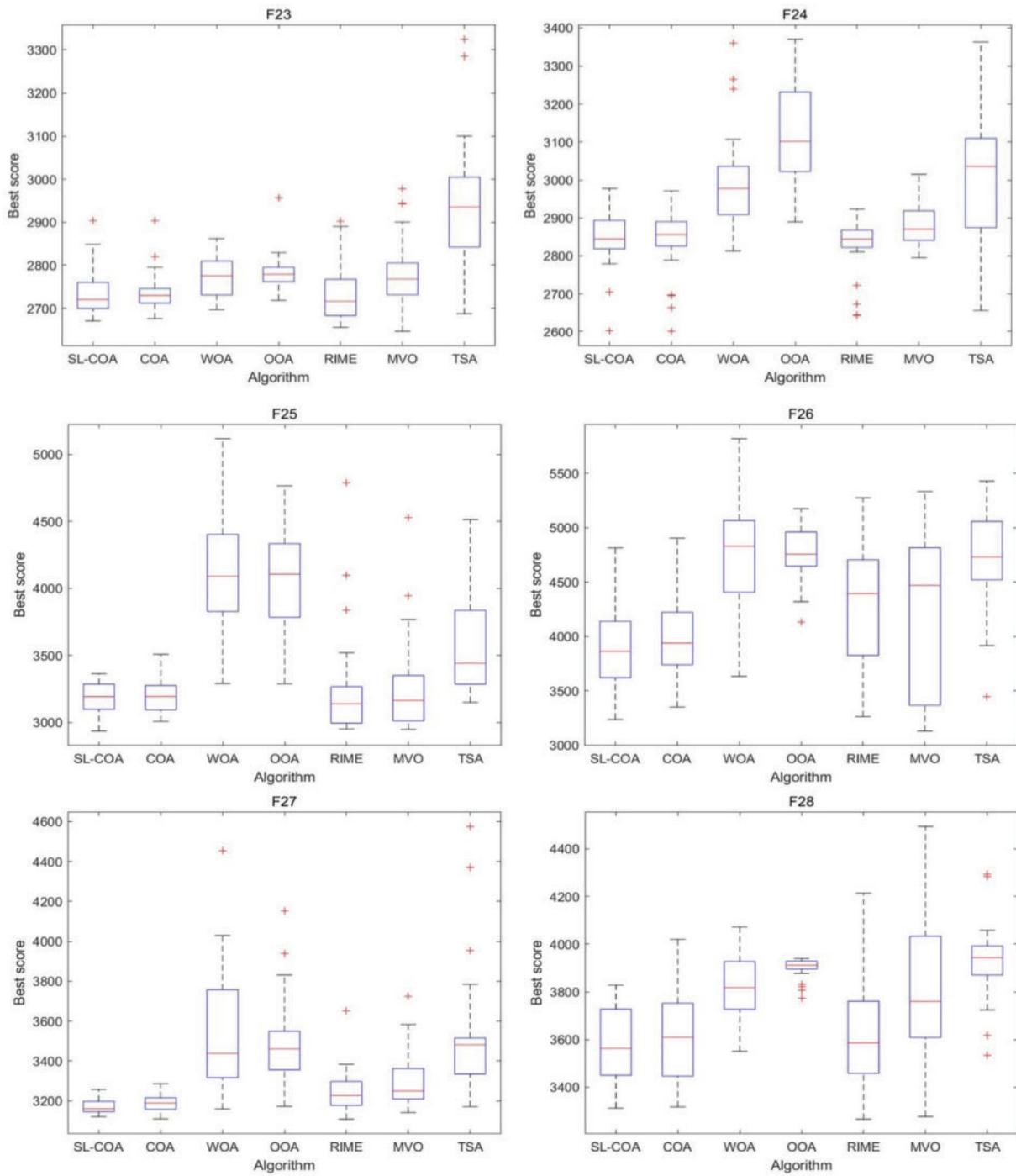


Figure 8: (Continued)

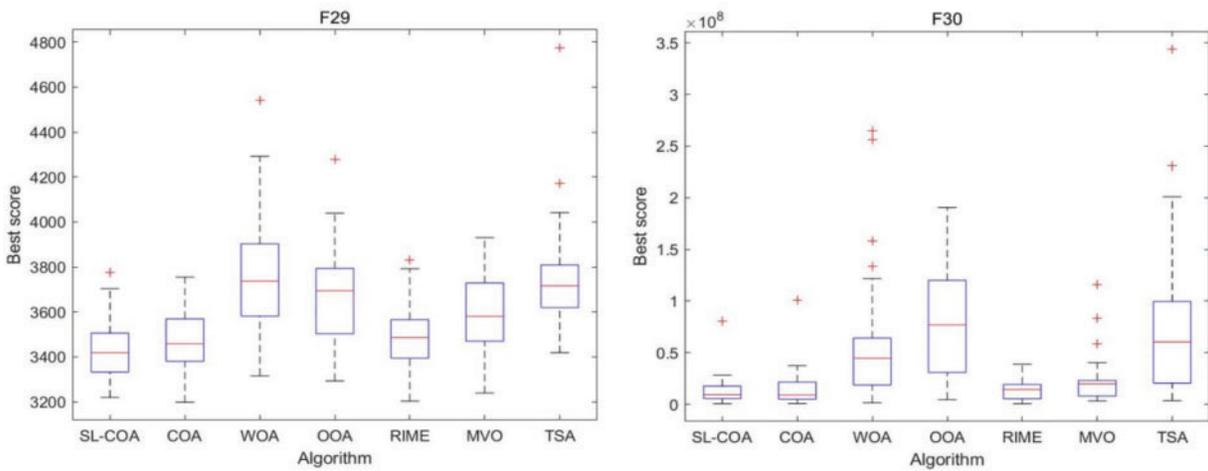


Figure 8: Box diagram of CEC2017

It could be found that SL-COA performs better on most of the functions. When comparing it to the original version (COA), it becomes evident that SL-COA offers distinct advantages for most functions, except for F10, F11, and F17. When compared to other heuristic algorithms, this algorithm demonstrates a notable superiority in terms of both average values and standard deviation. The convergence curve graphs illustrate the convergence performance of different algorithms. It can be observed that SL-COA exhibits faster convergence and effectively avoids local optima during the search process. The box graph visually displays the output distribution of the proposed algorithm, effectively illustrating both the average and standard deviation of the results.

Upon closer examination, while it's true that RIME occasionally exhibits better performance with specific test functions, the cumulative rank sum of RIME remains significantly higher than that of SL-COA. This may be attributed to its frequent failure to address certain hybrid and composite functions. RIME is believed to have a better optimization performance than COA, however, it cannot match the proposed SL-COA overall. Additionally, it is worth noting that when tackling composition functions, SL-COA demonstrates a clear advantage. While the error in certain problems (e.g., F30) is still large, it consistently delivers superior results when compared to other algorithms for the majority of cases.

4.2.2 Engineering Examples

In this section, four examples with engineering backgrounds are tested to evaluate the performance of the SL-COA. These comparisons are also conducted between SL-COA and the six heuristic algorithms mentioned above.

Example 1. Welded beam design problem

The objective of this problem is to find the optimal dimensions of a welded beam that minimizes the cost while satisfying certain constraints on its performance and safety (shown in Fig. 9) [57]. The design of the welded beam involves four variables: height (h), length (l), thickness (t), and width (b). The objective function to minimize is the cost of the beam, which is given by:

$$\text{Consider: } \vec{x} = [x_1 x_2 x_3 x_4] = [h l t b]$$

$$\text{Minimize: } f(\vec{x}) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2)$$

$$\text{Subject to: } g_1(\vec{x}) = \tau(\vec{x}) - \tau_{max} \leq 0,$$

$$g_2(\vec{x}) = \sigma(\vec{x}) - \sigma_{max} \leq 0,$$

$$g_3(\vec{x}) = \delta(\vec{x}) - \delta_{max} \leq 0,$$

$$g_4(\vec{x}) = x_1 - x_4 \leq 0,$$

(29)

$$g_5(\vec{x}) = P - P_c(\vec{x}) \leq 0,$$

$$g_6(\vec{x}) = 0.125 - x_1 \leq 0,$$

$$g_7(\vec{x}) = 1.10471x_1^2 + 0.04811x_3x_4(14.0 + x_2) - 5.0 \leq 0$$

Parameter range: $0.1 \leq x_1, x_4 \leq 2, 0.1 \leq x_2, x_3 \leq 10$

$$\text{where } \tau(\vec{x}) = \sqrt{(\tau')^2 + \frac{2\tau'\tau''x_2}{2R} + (\tau'')^2},$$

$$\tau' = \frac{P}{\sqrt{2x_1x_2}}, \tau'' = \frac{MR}{J},$$

$$M = P \left(L + \frac{x_2}{2} \right),$$

$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2} \right)^2},$$

$$J = 2 \left(\sqrt{2x_1x_2} \left(\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2} \right)^2 \right) \right)$$

$$\sigma(\vec{x}) = \frac{6PL}{x_4x_3^2}, \delta(\vec{x}) = \frac{4PL^3}{Ex_3^3x_4}$$

$$P_c(x) = 4.013E \sqrt{\frac{x_3^2x_4^2}{6}}$$

$$P = 6000lb, L = 14in, \delta_{max} = 0.25in,$$

$$E = 30 \times 10^6 psi, G = 12 \times 10^6 psi,$$

$$\tau_{max} = 13600 psi, \sigma_{max} = 30000 psi.$$

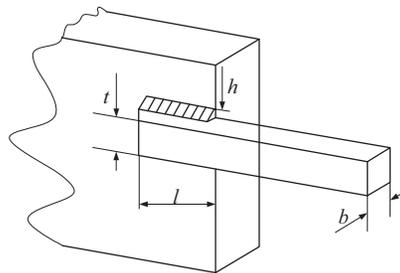


Figure 9: Welded beam design

The comparison results are listed in [Table 5](#).

Table 5: Results of welded beam design problem

	SL-COA	COA	WOA	OOA	RIME	MVO	TSA
h	0.1843	0.1950	0.1367	0.5444	0.3050	0.2043	0.2021
l	3.7468	3.4130	5.8852	2.4602	2.4823	3.2614	3.4554
t	9.0428	9.1607	8.9525	5.5247	7.3807	9.0443	8.9331
b	0.2057	0.2051	0.2096	0.5504	0.3084	0.2057	0.2109
f	1.7088	1.7176	1.9167	3.2136	2.0601	1.7954	1.7379

Example 2. Tension/compression spring design problem

The tension/compression spring design problem is another classical optimization problem in engineering (shown in Fig. 10) [58]. The objective of this problem is to determine the optimal dimensions and parameters of a tension or compression spring to meet certain performance requirements while minimizing manufacturing or material costs. The comparison of different algorithms is shown in Table 6.

Consider: $\vec{x} = [x_1 x_2 x_3] = [dDN]$

Minimize: $f(\vec{x}) = (x_3 + 2)x_2x_1^2$

Subject to: $g_1(\vec{x}) = 1 - \frac{x_2^3x_3}{71785x_1^4} \leq 0,$

$$g_2(\vec{x}) = \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_4)} + \frac{1}{5108x_1^2} \leq 0, \tag{30}$$

$$g_3(\vec{x}) = 1 - \frac{140.45x_1}{x_2^2x_3} \leq 0$$

$$g_4(\vec{x}) = \frac{x_1 + x_2}{1.5} - 1 \leq 0$$

Parameter range: $0.05 \leq x_1 \leq 2, 0.25 \leq x_2 \leq 1.3, 2 \leq x_3 \leq 15.$

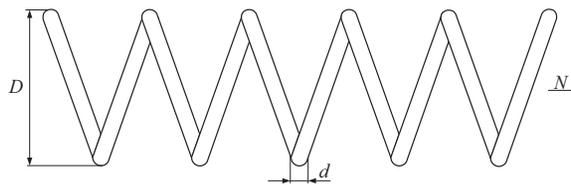


Figure 10: Tension/compression spring design

Table 6: Results of tension/compression spring design problem

	SL-COA	COA	WOA	OOA	RIME	MVO	TSA
d	0.0500	0.0500	0.0567	0.0613	0.0690	0.0500	0.0542
D	0.3163	0.3158	0.4900	0.6364	0.9336	0.3159	0.4186
N	14.1837	14.2419	6.3108	3.9444	2.0000	14.2345	8.4397
f	0.0127	0.0127	0.0131	0.0142	0.0178	0.0128	0.0128

Example 3. Three-bar truss design problem

This problem involves the optimization of the dimensions and geometry of a truss structure composed of three bars (or members) and typically used to support loads or distribute forces within a structure (shown in Fig. 11) [59]. The objective is to find the optimal design that minimizes certain criteria, such as weight or cost while ensuring that the truss can support the applied loads and maintain structural stability. The results are shown in Table 7.

Consider: $\vec{x} = [x_1 x_2] = [S1 S2]$

Minimize: $f(\vec{x}) = (2\sqrt{2}x_1 + x_2)l$

Subject to: $g_1(\vec{x}) = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2x_1^2 + 2x_1x_2}}P - \sigma \leq 0,$

$$g_2(\vec{x}) = \frac{x_2}{\sqrt{2x_1^2 + 2x_1x_2}}P - \sigma \leq 0, \quad (31)$$

$$g_3(\vec{x}) = \frac{1}{\sqrt{2x_2 + x_1}}P - \sigma \leq 0,$$

Parameter range: $0 \leq x_i \leq 1, i = 1, 2$

where $l = 100 \text{ cm}; P = 2 \text{ kN}/(\text{cm}^2); \sigma = 2 \text{ kN}/(\text{cm}^2)$

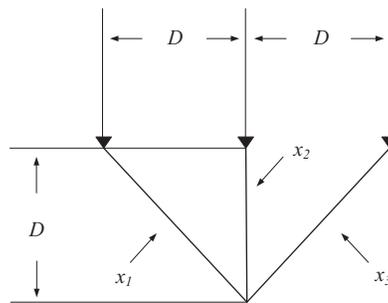


Figure 11: Three-bar truss design

Table 7: Results of three-bar truss design problem

	SL-COA	COA	WOA	OOA	RIME	MVO	TSA
S1	0.7876	0.7880	0.7978	0.7299	0.7377	0.7894	0.7881
S2	0.4114	0.4102	0.3830	0.6064	0.5757	0.4063	0.4099
f	263.8768	263.8962	263.9554	267.0859	266.2181	263.8963	263.9069

Example 4. Pressure vessel design problem

The pressure vessel design problem (show in Fig. 12) is an important engineering optimization problem related to the design and sizing of pressure vessels [60]. Pressure vessels are containers designed to hold and store fluids or gases at high pressure. They are used in various industries, including petrochemical, aerospace, and manufacturing, and must be designed to safely contain the pressurized contents while minimizing material and manufacturing costs. The results are shown in Table 8.

Consider: $\vec{x} = [x_1 x_2 x_3 x_4] = [T_s T_h RL]$

Minimize: $f(\vec{x}) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$

Subject to: $g_1(\vec{x}) = -x_1 + 0.0193x_3 \leq 0,$

$g_2(\vec{x}) = -x_3 + 0.00954x_3 \leq 0,$ (32)

$g_3(\vec{x}) = -\pi x_3^2x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0,$

$g_4(\vec{x}) = x_4 - 240 \leq 0,$

Parameter range: $0 \leq x_1, x_2 \leq 99, 10 \leq x_3, x_4 \leq 200$

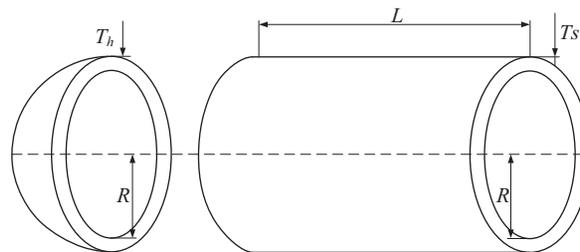


Figure 12: Pressure vessel design problem

Table 8: Results of pressure vessel design problem

	SL-COA	COA	WOA	OOA	RIME	MVO	TSA
T_s	1.2402	1.2655	1.3753	4.7514	1.2593	1.2411	1.3132
T_h	0.6130	0.6223	0.6365	7.4678	0.6225	0.6135	0.6626
R	64.2557	65.2254	65.2252	54.6361	65.2262	64.2681	64.9965
L	14.2442	10.0000	10.0000	65.3512	10.0000	14.7786	11.4516
f	7236.8021	7344.0831	7880.9304	79339.1339	7322.6112	7275.4761	7871.8263

It is shown that the utilization of SL-COA produces the best design for all four problems. The convergence curve is shown in Fig. 13. It is evident that SL-COA exhibits swift convergence and possesses the capability to thoroughly explore the solution space, thereby avoiding premature convergence to local optima. The number of iterations needed may vary depending on the complexity of the problems, but SL-COA consistently maintains a leading position among other algorithms. It's worth mentioning that RIME, known for its strong performance in mathematical scenarios, no longer exhibits its advantages in this engineering context. This could be attributed to its suboptimal optimization for solving certain polynomial functions, which mirrors its performance in mathematical problems (F7, F25, F27). These findings underscore the competitive edge held by the proposed SL-COA in addressing real world engineering challenges.

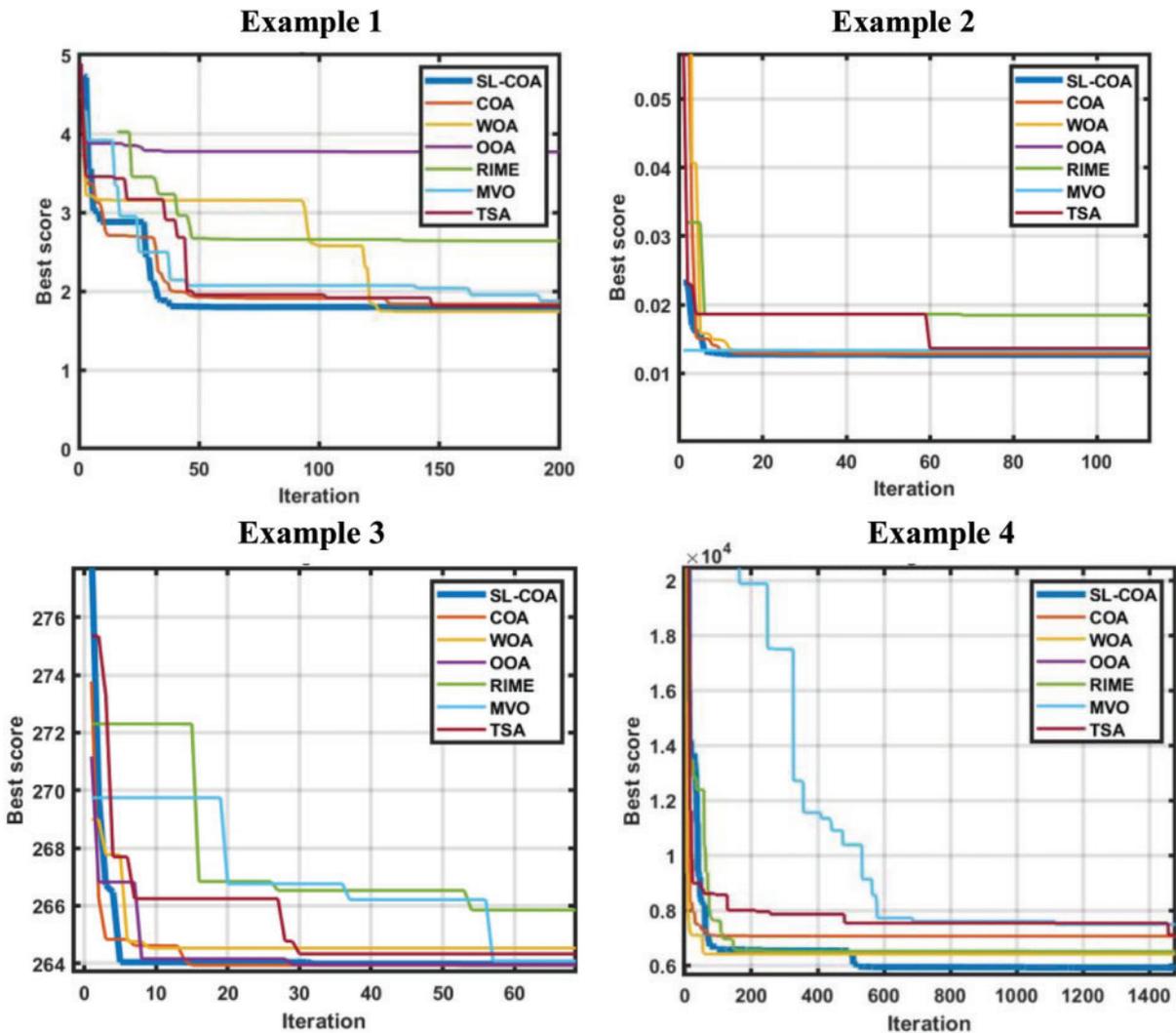


Figure 13: Convergence curve of 4 engineering problems

4.3 Examples of Reliability Analysis

In this section, the proposed SL-COA-based FORM framework is evaluated by mathematical examples and engineering examples. The proposed SL-COA FORM framework is compared with the FORM based on Sequential Quadratic Programming (SQP) or other optimization algorithms. In addition, the reliability analysis results of MCS are considered to be accurate reliability analysis results, which can be used to refer to other reliability analysis results.

4.3.1 Mathematical Examples

Example 1. Explicit performance function

Example 1 showcases reliability analysis problems with different explicit performance functions. Simulating the failure domain of the structural performance functions demonstrates how to effectively utilize the SL-COA algorithm to find the MPP, thereby calculating the reliability index. A mathematically straightforward problem involving three uncertain variables is tested in this section. We examine two distinct

performance functions, denoted as g_1 and g_2 , and their corresponding formulations.

$$g_1 = y - x_1x_2 \tag{33}$$

$$g_2 = 3y - x_1^2x_2 \tag{34}$$

where x_1 and x_2 are normally distributed variables with mean values of $\mu_1 = 3.0$ and $\mu_2 = 2.0$. The standard deviations $\sigma_1 = 0.3$ and $\sigma_2 = 0.2$, respectively. The interval variable $y \in [8, 12]$. It is obvious that $y^* = 8$. The reliability index β and the number of function calls are tested. The results are shown in [Table 9](#).

Table 9: Results of the explicit performance function

Example 1.1 $g_1 = y - x_1x_2$				Example 1.2 $g_2 = 3y - x_1^2x_2$			
	Call	β	Error		call	β	Error
MCS	10^7	2.2197	-	MCS	10^7	1.3902	-
FORM-SQP	14	0.0000	1.0000	FORM-SQP	17	0.0000	1.0000
FORM-SL-COA	50	2.1782	0.0187	FORM-SL-COA	30	1.3800	0.0074
FORM-PSO	170	2.2476	0.0126	FORM-PSO	190	1.3583	0.0229
FORM-COA	60	2.3840	0.0740	FORM-COA	40	1.5148	0.0896
FORM-WOA	60	1.6327	0.2644	FORM-WOA	100	0.8188	0.4110
FORM-OOA	110	2.3860	0.0749	FORM-OOA	140	0.4056	0.7082
FORM-RIMA	120	2.2434	0.0107	FORM-RIMA	80	1.6172	0.1633
FORM-MVO	150	0.0055	0.9975	FORM-MVO	40	1.5905	0.1441
FORM-TSA	200	0.7350	0.6689	FORM-TSA	20	1.0114	0.2725

The convergence curve is shown in [Fig. 14](#).

This example illustrates the effective resolution of the problem using the proposed method, which delivers high accuracy and efficient function calls. In the heuristic algorithm-based FORM, the determination of function calls occurs at the conclusion of their interactions.

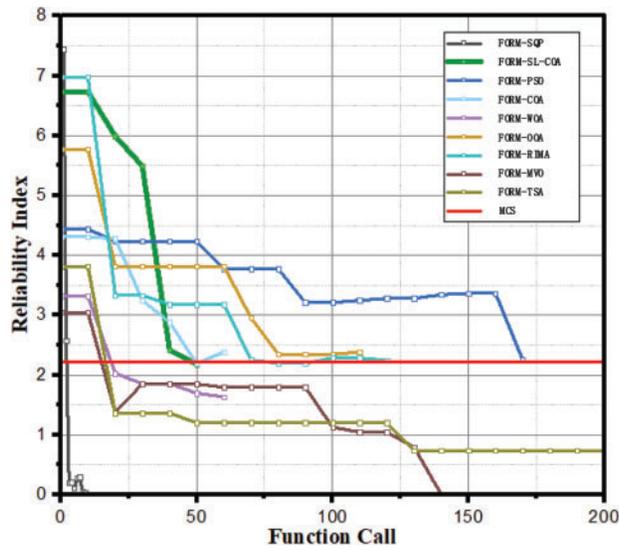
In Example 1.1, the output shows the third fewest errors, yet it delivers exceptional performance in terms of the number of function calls. In Example 1.2, it yields the most precise results while utilizing the second lowest number of function calls.

Example 2. Nonlinear performance function

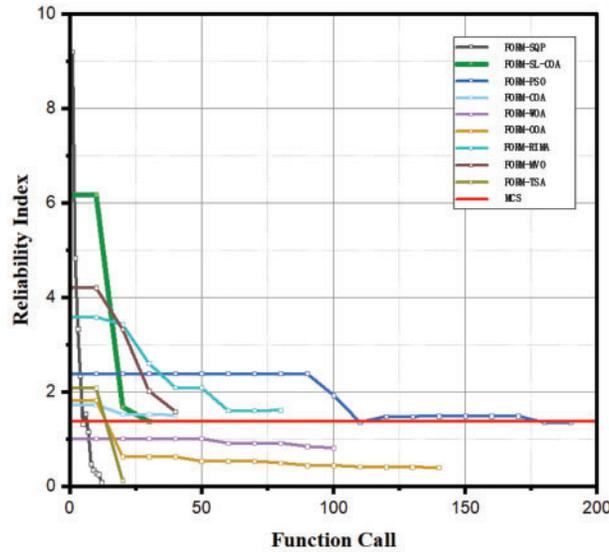
Example 2 presents reliability analysis problems with nonlinear performance functions. Nonlinear performance functions are common in practical engineering structures, such as those resulting from material nonlinearity or geometric nonlinearity in structural responses. Example 2 proves the effectiveness and accuracy of the SL-COA algorithm in handling nonlinear reliability analysis problems, which is crucial for evaluating the safety of complex engineering structures. A nonlinear problem is solved in this example. The LSF is

$$g = \sin\left(\frac{5}{2}x_1\right) - \frac{(x_1 + 4)(x_2 - 1)}{20} + y \tag{35}$$

where x_1 and x_2 are normally distributed variables with $\mu_1 = 1.5$, $\sigma_1 = 1$ and $\mu_2 = 2.5$, $\sigma_2 = 1$. The interval variable $y \in [2, 2.5]$. The results are shown in [Table 10](#).



(a) Example 1.1



(b) Example 1.2

Figure 14: Convergence curve of Example 1

Table 10: Results of the nonlinear performance function

	Call	β	Error
MCS	10^7	1.8748	–
FORM-SQP	14	0.0787	0.9580
FORM-SL-COA	20	1.8925	0.0148
FORM-PSO	120	1.8498	0.0133
FORM-COA	40	1.6074	0.1426
FORM-WOA	30	1.7821	0.0494
FORM-OOA	80	1.9486	0.0394

(Continued)

Table 10 (continued)

	Call	β	Error
FORM-RIMA	150	2.0602	0.0989
FORM-MVO	50	1.4629	0.2197
FORM-TSA	60	1.9470	0.0385

Example 3. Performance function with multiple independent interval variable

Example 3 showcases performance functions with multiple independent interval variables. In practical engineering, certain parameters (such as geometric dimensions, material properties, etc.) may have uncertainties, which are often represented in the form of interval variables. For the third example, an LSF with two interval variables is used to test the proposed framework. The performance function is

$$g = -10y_1 + y_2x_2 - \ln(x_1) \quad (36)$$

where x_1 and x_2 are normally distributed variables with $\mu_1 = 5, \sigma_1 = 3$ and $\mu_2 = 1.5, \sigma_2 = 0.9, y_1 \in [0, 1.5], y_2 \in [1, 1.5]$. The results are shown in [Table 11](#).

Table 11: Results of function with multiple independent interval variables

	Call	β	Error
MCS	10^7	1.4889	–
FORM-SQP	9	0.5340	0.6414
FORM-SL-COA	40	1.4775	0.0077
FORM-PSO	110	1.1352	0.2375
FORM-COA	80	1.2984	0.1279
FORM-WOA	50	1.6700	0.1216
FORM-OOA	140	1.2549	0.1572
FORM-RIMA	160	1.1281	0.2423
FORM-MVO	20	1.1555	0.2239
FORM-TSA	200	0.6594	0.5571

Example 4. Performance function with various probability distribution

Example 4 presents performance functions with different probability distributions (such as Weibull, Gumbel, Exponential, and Normal distributions). In practical engineering, the probability of distributions of different parameters may vary, adding complexity to the reliability analysis. A performance function with both normal and non-normal random variables is considered as below.

$$g = x_1x_2y - x_3x_4^2/8 \quad (37)$$

The means, standard derivations, and non-normal random variables are considered as shown in [Table 12](#).

The results are shown in [Table 13](#).

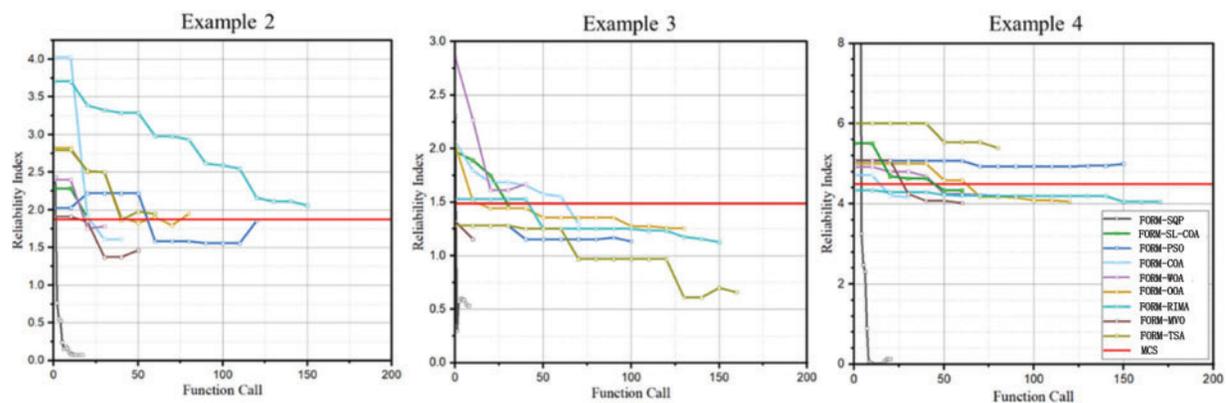
The convergence curve of Example 2, Example 3, and Example 4 is shown in [Fig. 15](#).

Table 12: Random variable for Example 4

Variable	Mean	Standard deviation	Distribution
x_1	4	0.1	Weibull
x_2	0.875	0.1	Gumbel
x_3	100	100	Exponential
x_4	150	10	Normal

Table 13: Results of function with various probability distributions

	Call	β	Error
MCS	10^7	4.4895	–
FORM-SQP	33	0.0000	1.0000
FORM-SL-COA	40	4.3342	0.0346
FORM-PSO	150	4.9943	0.1124
FORM-COA	30	4.1644	0.0724
FORM-WOA	80	4.2017	0.0641
FORM-OOA	120	4.0550	0.0968
FORM-RIMA	170	4.0494	0.0980
FORM-MVO	60	4.0226	0.1040
FORM-TSA	80	5.3949	0.2017

**Figure 15:** Convergence curve of Examples 2, 3, and 4

From the calculation results of the above mathematical examples, it can be seen that SL-COA performs well. Firstly, when the gradient-based FORM cannot obtain acceptable solution results, the FORM based on a heuristic algorithm can obtain effective analysis results. Secondly, compared with other optimization algorithms, the reliability index obtained by SL-COA is closer to the reference result calculated by MCS. This reflects the computational performance of SL-COA, that is, the algorithm can find more acceptable MPP points.

Since the SL-COA-based FORM framework meets the end iteration condition defined in this paper with very few function calls. The reliability index value with more function calls is worth exploring. Therefore, extra testing is conducted by setting a fixed function call number. The results are shown in Table 14.

Table 14: Results with different function calls

Example 1.1: explicit performance function									
	MCS				FORM-SL-COA				
Call	10^7	40	80	100	10^3	10^4	10^5	10^6	10^7
Beta	2.2197	2.1682	2.1782	2.2046	2.2046	2.2046	2.2046	2.2046	2.2046
Error	–	0.0232	0.0187	0.0068	0.0068	0.0068	0.0068	0.0068	0.0068

Example 1.2: explicit performance function									
	MCS				FORM-SL-COA				
Call	10^7	40	80	100	10^3	10^4	10^5	10^6	10^7
Beta	1.3902	1.3800	1.3800	1.4001	1.4001	1.3991	1.3991	1.3991	1.3991
Error	–	0.0074	0.0074	0.0071	0.0071	0.0064	0.0064	0.0064	0.0064

Example 2: nonlinear performance function									
	MCS				FORM-SL-COA				
Call	10^7	40	80	100	10^3	10^4	10^5	10^6	10^7
Beta	1.8748	1.8925	1.8926	1.9001	1.8988	1.8889	1.8889	1.8889	1.8862
Error	–	0.0095	0.0095	0.0135	0.0128	0.0075	0.0075	0.0075	0.0061

Example 3: performance function with multiple independent interval variable									
	MCS				FORM-SL-COA				
Call	10^7	40	80	100	10^3	10^4	10^5	10^6	10^7
Beta	1.4889	1.4775	1.4822	1.4842	1.4842	1.4844	1.4844	1.4844	1.4844
Error	–	0.0077	0.0045	0.0032	0.0032	0.0030	0.0030	0.0030	0.0030

Example 4: performance function with various probability distribution									
	MCS				FORM-SL-COA				
Call	10^7	40	80	100	10^3	10^4	10^5	10^6	10^7
Beta	4.4895	4.3342	4.3342	4.3979	4.4279	4.4289	4.4299	4.4299	4.4299
Error	–	0.0346	0.0346	0.0204	0.0137	0.0135	0.0133	0.0133	0.0133

The range of function call values employed in this testing spans from 40 to 10^7 . This range encompasses the minimum requirement necessary to meet the convergence conditions and extends to the maximum value required by the MCS comparison algorithm. The results demonstrate that the suggested framework can attain greater accuracy with an increased number of function calls, thereby suggesting that the specific interaction values can be adjusted to meet the accuracy requirements in practical engineering applications.

4.3.2 Engineering Examples

In this section, two structural reliability analysis cases are conducted to evaluate the performance of the proposed framework.

Example 1: Composite cylinder analysis problem

In this example, a composite cylinder case [61] is analyzed. The composite cylinder consists of an inner cylinder and an outer cylinder (shown in Fig. 16). The inner cylinder has an inner diameter of a , and an outer diameter of c , while the outer cylinder has an inner diameter of c and an outer diameter of b . The variable c is an interval variable. The cylinder is subjected to various loading conditions, and the reliability analysis aims to assess the cylinder's ability to withstand these loads without failing. The results of composite cylinder design problem are shown in Table 15.

Consider: $\vec{x} = [x_1 x_2 x_3] = [abc]$

Subject to: $g_1(\vec{x}) = S - S_1 \leq 0$,

Parameter: $x_1 \sim N(36.07, 0.3607^2)$,

$x_2 \sim N(44.45, 0.4445^2)$,

$30.48 \leq x_3 \leq 40.96$,

(38)

where $S = a_0 + \sum_{i=1}^3 b_i x_i + \sum_{i=1}^3 c_i x_i^2 + \sum_{j=1}^2 \sum_{j=i+1}^3 d_{ij} x_i x_j + e_{123} x_1 x_2 x_3$

$a_0 = 1306319, b_1 = -34404, b_2 = -28761.7, b_3 = -24292.4$,

$c_1 = -5, c_2 = -10.9, c_3 = -9.4$,

$d_{12} = 761.1, d_{13} = 645.8, d_{23} = 550.9$,

$e_{123} = -14.1$

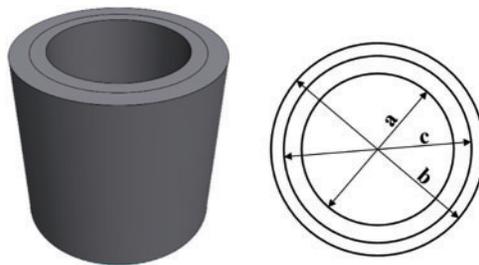


Figure 16: Composite cylinder analysis

Table 15: Results of composite cylinder design problem

	Call	β	Error
MCS	10^7	2.2272	-
FORM-SQP	22	1.8637	0.1632
FORM-SL-COA	60	2.2409	0.0062
FORM-PSO	90	3.0996	0.3917
FORM-COA	120	3.7800	0.6972
FORM-WOA	110	1.9855	0.1085
FORM-OOA	90	3.0165	0.3544
FORM-RIMA	150	3.1092	0.3960

(Continued)

Table 15 (continued)

	Call	β	Error
FORM-MVO	80	3.2920	0.4781
FORM-TSA	70	5.0900	1.2854

Example 2: Support bracket analysis problem

In this case, a support bracket within an aircraft braking structure is analyzed (the green triangular prism shown in Fig. 17). The material used for this structure is structural steel, and the presence of the triangular ribs in the middle enhances the structural strength. The bracket is subjected to forces generated during braking, and the reliability analysis aims to assess its ability to withstand these forces without failing. Using finite element simulation analysis, the results are obtained and fitted in MATLAB to obtain the LSF. The analysis based on one of the functions is performed below. The results of support bracket are shown in Table 16.

Consider: $\vec{x} = [x_1, x_2, x_3] = [LHD]$

Subject to: $g_1(\vec{x}) = 3.4675 - 0.04308x_1 - 0.04707x_2 - 0.02351x_3 \leq 0$,

Parameter:

$$x_1 \sim N(30, 1^2), \tag{39}$$

$$x_2 \sim N(30, 1^2),$$

$$x_3 \sim N(2.5, 0.1^2)$$

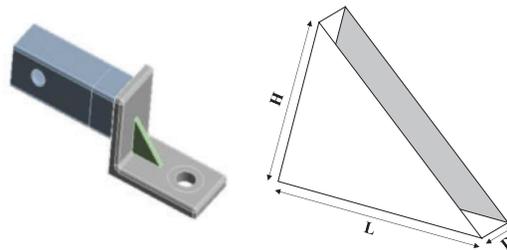


Figure 17: Support bracket analysis

Table 16: Results of support bracket design problem

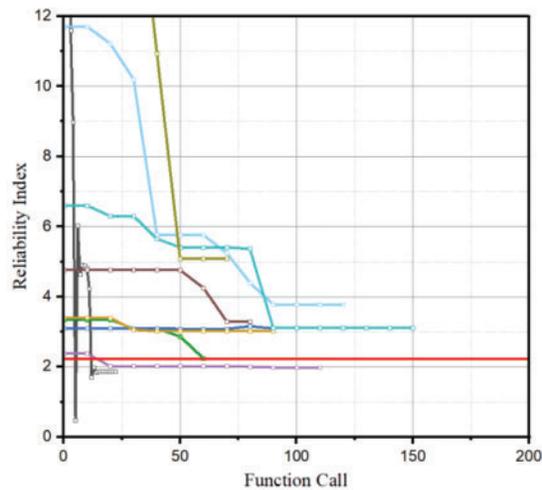
	Call	β	Error
MCS	10^7	0.9152	–
FORM-SQP	15	0.9043	0.0119
FORM-SL-COA	70	0.9035	0.0128
FORM-PSO	70	2.8210	2.0824
FORM-COA	120	2.0524	1.2425
FORM-WOA	80	1.6221	0.7723
FORM-OOA	120	1.1727	0.2813

(Continued)

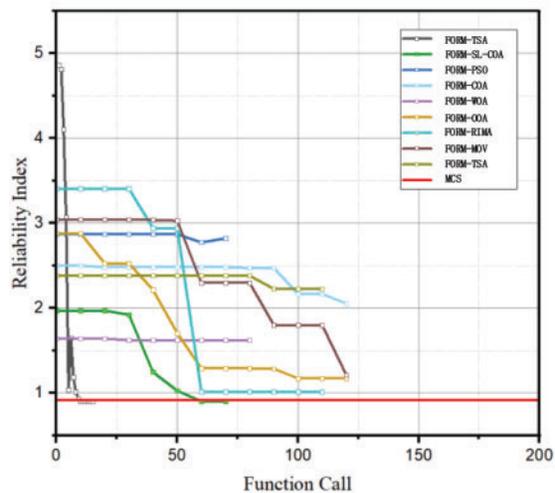
Table 16 (continued)

	Call	β	Error
FORM-RIMA	110	1.0158	0.1099
FORM-MVO	120	1.2116	0.3239
FORM-TSA	120	2.2269	1.4333

The proposed method successfully solves the engineering problems by delivering acceptable results with minimal function calls in each example. The error in β determined by the proposed methods, when compared to the results obtained through MCS, is approximately 1%. The convergence curve is shown in Fig. 18. The SQP-based FORM exhibits superior performance in these two engineering examples. Although it fails to solve the problems in Section 4.2.1, the results in this section might serve as a reminder that the SQP-based analysis framework is not entirely impractical for addressing hybrid reliability analysis problems. Overall, this result proves the capability of the SL-COA base FORM framework to tackle real engineering problems successfully.



(a) Example 1



(a) Example 2

Figure 18: Convergence curve of two engineering problems

5 Conclusions

As a typical analytical strategy for reliability analysis, FORM faces many challenges. In the face of solving complex engineering problems with MCS or improved MCS strategies, FORM may be able to use its low computational difficulty to provide an acceptable analysis result. The main purpose of FORM is to find the exact location of the MPP point. However, there are the following problems: 1) In the past, FORM based on gradients may not be able to obtain convergent results for some reliability analysis problems; 2) Although the FORM using the heuristic algorithm can often obtain convergent reliability analysis results, it may not be able to obtain the acceptable MPP position because the heuristic algorithm is trapped in local optimization. To this end, this study proposes an improved COA optimization algorithm (SL-COA), which is committed to improving the accuracy and robustness of the optimization results. By introducing the SL-COA algorithm into FORM, the effectiveness and practicability of FORM in complex structural reliability analysis are improved.

The main work of this study includes: A new optimization algorithm SL-COA is obtained by improving the COA algorithm through a social learning strategy, which improves its efficiency and robustness in solving medium-dimensional complex problems. The improved effect of the SL-COA algorithm is tested by CEC2005 and CEC2017 test function sets and four engineering examples, which reflect the effectiveness and competitiveness of the algorithm. The performance of FORM based on SL-COA is tested in four mathematical examples and two engineering examples, which shows the effectiveness and practicability of SL-COA in the field of reliability analysis.

New analytical strategies that require lower computational costs and are suitable for multiple reliability problems are also a research direction in future research. In the development of automotive components and systems, reliability and safety are key concerns. SL-COA can be applied to optimize the design of automotive parts, such as engines, transmissions, and body structures, to improve their reliability and durability.

Acknowledgement: We would like to express our sincere gratitude to Beijing Research Institute of Mechanical & Electrical Technology for providing us with the essential venue support for conducting our experiments. The generosity and assistance have been instrumental in the successful completion of this research.

Funding Statement: This research was funded by the National Key Research and Development Program (Grant No. 2022YFB3706904).

Author Contributions: Study conception and design: Yunhan Ling, Huajun Peng, Yiqing Shi, Hui Ma; data collection: Yunhan Ling, Huajun Peng, Yiqing Shi, Chao Xu, Jingzhen Yan; analysis and interpretation of results: Yunhan Ling, Huajun Peng, Hui Ma; draft manuscript preparation: Yunhan Ling, Huajun Peng, Jingjing Wang. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data will be made available by the corresponding author upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Liu X, Li T, Zhou Z, Hu L. An efficient multi-objective reliability-based design optimization method for structure based on probability and interval hybrid model. *Comput Methods Appl Mech Eng.* 2022;392:114682. doi:10.1016/j.cma.2022.114682.

2. Wang X, Zhao W, Chen Y, Li X. A first order reliability method based on hybrid conjugate approach with adaptive Barzilai-Borwein steps. *Comput Methods Appl Mech Eng.* 2022;401:115670. doi:10.1016/j.cma.2022.115670.
3. Wang Z, Xiao F, Cao Z. Uncertainty measurements for Pythagorean fuzzy set and their applications in multiple-criteria decision making. *Soft Comput.* 2022;26:9937–52. doi:10.1007/s00500-022-07361-9.
4. Li XQ, Song LK, Bai GC. Recent advances in reliability analysis of aeroengine rotor system: a review. *Int J Struct Integr.* 2022;13(1):1–29. doi:10.1108/IJSI-10-2021-0111.
5. Ou Y, Cai Y, Zeng L, Zhao W, Chen Y. A teaching-learning-based optimization algorithm for reliability analysis with an adaptive penalty coefficient. *Structures.* 2024;65:106695. doi:10.1016/j.istruc.2024.106695.
6. Hao S. I-35W bridge collapse. *J Bridge Eng.* 2010;15(5):608–14. doi:10.1061/(ASCE)BE.1943-5592.0000090.
7. Li W, Li C, Gao L, Xiao M. Risk-based design optimization under hybrid uncertainties. *Eng Comput.* 2022;38:1–13. doi:10.1007/s00366-020-01196-4.
8. Li W, Xiao M, Garg A, Gao L. A new approach to solve uncertain multidisciplinary design optimization based on conditional value at risk. *IEEE Trans Autom Sci Eng.* 2020;18:356–68. doi:10.1109/TASE.2020.2999380.
9. Yang M, Zhang D, Han X. New efficient and robust method for structural reliability analysis and its application in reliability-based design optimization. *Comput Methods Appl Mech Eng.* 2020;366:113018. doi:10.1016/j.cma.2020.113018.
10. Meng Z, Pang Y, Pu Y, Wang X. New hybrid reliability-based topology optimization method combining fuzzy and probabilistic models for handling epistemic and aleatory uncertainties. *Comput Methods Appl Mech Eng.* 2020;363:112886. doi:10.1016/j.cma.2020.112886.
11. Meng D, Yang S, Yang H, De JAM, Correia J, Zhu SP. Intelligent-inspired framework for fatigue reliability evaluation of offshore wind turbine support structures under hybrid uncertainty. *Ocean Eng.* 2024;307:118213. doi:10.1016/j.oceaneng.2024.118213.
12. Zhu SP, Keshtegar B, Bagheri M, Hao P, Trung NT. Novel hybrid robust method for uncertain reliability analysis using finite conjugate map. *Comput Methods Appl Mech Eng.* 2020;371:113309. doi:10.1016/j.cma.2020.113309.
13. Yang S, Meng D, Yang H, Luo C, Su X. Enhanced soft Monte Carlo simulation coupled with SVR for structural reliability analysis. In: *Proceedings of the Institution of Civil Engineers-Transport*; Emerald Publishing Limited; 2024. p. 1–41.
14. Liu X, Gong M, Zhou Z, Xie J, Wu W. An improved first order approximate reliability analysis method for uncertain structures based on evidence theory. *Mech Based Des Struct Mach.* 2023;51:4137–54. doi:10.1080/15397734.2021.1956324.
15. Hu Z, Mansour R, Olsson M, Du X. Second-order reliability methods: a review and comparative study. *Struct Multidiscpl Optim.* 2021;64:1–31. doi:10.1007/s00158-021-03013-y.
16. Yang Z, Ching J. A novel reliability-based design method based on quantile-based first-order second-moment. *Appl Math Model.* 2020;88:461–73. doi:10.1016/j.apm.2020.06.038.
17. Meng Z, Pang Y, Wu Z, Ren S, Yildiz AR. A novel maximum volume sampling model for reliability analysis. *Appl Math Model.* 2022;102:797–810. doi:10.1016/j.apm.2021.10.025.
18. Li XQ, Song LK, Bai GC, Li DG. Physics-informed distributed modeling for CCF reliability evaluation of aeroengine rotor systems. *Int J Fatigue.* 2023;167:107342. doi:10.1016/j.ijfatigue.2022.107342.
19. Abdollahi A, Moghaddam MA, Monfared SAH, Rashki M, Li Y. A refined subset simulation for the reliability analysis using the subset control variate. *Struct Saf.* 2020;87:102002. doi:10.1016/j.strusafe.2020.102002.
20. Tabandeh A, Jia G, Gardoni P. A review and assessment of importance sampling methods for reliability analysis. *Struct Saf.* 2022;97:102216. doi:10.1016/j.strusafe.2022.102216.
21. Papaioannou I, Straub D. Combination line sampling for structural reliability analysis. *Struct Saf.* 2021;88:102025. doi:10.1016/j.strusafe.2020.102025.
22. Meng D, Yang S, de Jesus AMP, Zhu SP. A novel Kriging-model-assisted reliability-based multidisciplinary design optimization strategy and its application in the offshore wind turbine tower. *Renew Energy.* 2023;203:407–20. doi:10.1016/j.renene.2022.12.062.

23. Meng D, Yang S, De JAM, Fazeres-Ferradosa T, Zhu SP. A novel hybrid adaptive Kriging and water cycle algorithm for reliability-based design and optimization strategy: application in offshore wind turbine monopile. *Comput Methods Appl Mech Eng.* 2023;412:116083. doi:10.1016/j.cma.2023.116083.
24. Ling Y, Shi Y, Hou H, Pan L, Chen H, Liang P, et al. Enhanced dung beetle optimizer for Kriging-assisted time-varying reliability analysis. *AIMS Math.* 2024;9(10):29296–332. doi:10.3934/math.20241420.
25. Zhou Y, Lu Z, Yun W. Active sparse polynomial chaos expansion for system reliability analysis. *Reliability Eng Syst Saf.* 2020;202:107025. doi:10.1016/j.res.2020.107025.
26. Meng D, Li Y, He C, Guo J, Lv Z, Wu P. Multidisciplinary design for structural integrity using a collaborative optimization method based on adaptive surrogate modelling. *Mater Design.* 2021;206:109789. doi:10.1016/j.matdes.2021.109789.
27. Afshari SS, Enayatollahi F, Xu X, Liang X. Machine learning-based methods in structural reliability analysis: a review. *Reliability Eng Syst Saf.* 2022;219:108223. doi:10.1016/j.res.2021.108223.
28. Luo C, Keshtegar B, Zhu SP, Niu X. EMCS-SVR: hybrid efficient and accurate enhanced simulation approach coupled with adaptive SVR for structural reliability analysis. *Comput Methods Appl Mech Eng.* 2022;400:115499. doi:10.1016/j.cma.2022.115499.
29. Keshtegar B, Seghier MEAB, Zio E, Correia JA, Zhu SP, Trung N-T. Novel efficient method for structural reliability analysis using hybrid nonlinear conjugate map-based support vector regression. *Comput Methods Appl Mech Eng.* 2021;381:113818. doi:10.1016/j.cma.2021.113818.
30. Meng D, Yang H, Yang S, Zhang Y, De Jesus AM, Correia J, et al. Kriging-assisted hybrid reliability design and optimization of offshore wind turbine support structure based on a portfolio allocation strategy. *Ocean Eng.* 2024;295:116842. doi:10.1016/j.oceaneng.2024.116842.
31. Xue Y, Deng Y. Extending set measures to orthopair fuzzy sets. *Fuzziness Knowl-Based Syst.* 2022;30:63–91. doi:10.1142/S0218488522500040.
32. Yang S, Meng D, Wang H, Chen Z, Xu B. A comparative study for adaptive surrogate-model-based reliability evaluation method of automobile components. *Int J Struct Integr.* 2023;14:498–519. doi:10.1108/IJSI-03-2023-0020.
33. Yang S, Meng D, Wang H, Yang C. A novel learning function for adaptive surrogate-model-based reliability evaluation. *Philos Trans R Soc A.* 2024;382(2264):20220395. doi:10.1098/rsta.2022.0395.
34. Yang S, Meng D, Guo Y, Nie P, Jesus AMDE. A reliability-based design and optimization strategy using a novel MPP searching method for maritime engineering structures. *Int J Struct Integr.* 2023;14:809–26. doi:10.1108/IJSI-06-2023-0049.
35. Zhu SP, Keshtegar B, Chakraborty S, Trung NT. Novel probabilistic model for searching most probable point in structural reliability analysis. *Comput Methods Appl Mech Eng.* 2020;366:113027. doi:10.1016/j.cma.2020.113027.
36. Zhang D, Zhang J, Yang M, Wang R, Wu Z. An enhanced finite step length method for structural reliability analysis and reliability-based design optimization. *Struct Multidiscipl Optim.* 2022;65:231. doi:10.1007/s00158-022-03294-x.
37. Yang M, Zhang D, Wang F, Han X. Efficient local adaptive Kriging approximation method with single-loop strategy for reliability-based design optimization. *Comput Methods Appl Mech Eng.* 2022;390:114462. doi:10.1016/j.cma.2021.114462.
38. Keshtegar B, Zhu SP. Three-term conjugate approach for structural reliability analysis. *Appl Math Model.* 2019;76:428–42. doi:10.1016/j.apm.2019.06.022.
39. Roudak MA, Karamloo M. Establishment of non-negative constraint method as a robust and efficient first-order reliability method. *Appl Math Model.* 2019;68:281–305. doi:10.1016/j.apm.2018.11.021.
40. Zhu SP, Keshtegar B, Seghier MEAB, Zio E, Taylan O. Hybrid and enhanced PSO: novel first order reliability method-based hybrid intelligent approaches. *Comput Methods Appl Mech Eng.* 2022;393:114730. doi:10.1016/j.cma.2022.114730.
41. Yang S, Guo C, Meng D, Guo Y, Guo Y, Pan L, et al. MECSBO: multi-strategy enhanced circulatory system based optimisation algorithm for global optimisation and reliability-based design optimisation problems. *Collab Intell Manuf.* 2024;6(2):e12097. doi:10.1049/cim2.12097.
42. Kokash N. An introduction to heuristic algorithms. *Dep Inform Telecommun.* 2005:1–8.

43. Vinod Chandra SS, Harshil Anand A. Nature inspired meta heuristic algorithms for optimization problems. *Computing*. 2022;104(2):251–69. doi:10.1007/s00607-021-00955-5.
44. Pedroso DM. FORM reliability analysis using a parallel evolutionary algorithm. *Struct Saf*. 2017;65:84–99. doi:10.1016/j.strusafe.2017.01.001.
45. Zhong C, Wang M, Dang C, Ke W, Guo S. First-order reliability method based on Harris Hawks Optimization for high-dimensional reliability analysis. *Struct Multidiscipl Optim*. 2020;62(4):1951–68. doi:10.1007/s00158-020-02587-3.
46. Dehghani M, Montazeri Z, Trojovská E, Trojovský P. Coati optimization algorithm: a new bio-inspired meta-heuristic algorithm for solving optimization problems. *Knowl Based Syst*. 2023;259:110011. doi:10.1016/j.knosys.2022.110011.
47. Ioffe S. Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv:1502.03167. 2015.
48. Kennedy J, Eberhart R. Particle swarm optimization. In: *Proceedings of ICNN'95—International Conference on Neural Networks; 1995; Perth, WA, Australia: IEEE*. Vol. 4, p. 1942–8. doi:10.1109/ICNN.1995.488968.
49. De OMAM, Stutzle T, Van den EK, Dorigo M. Incremental social learning in particle swarms. *IEEE Trans Syst Man Cybern Part B (Cybern)*. 2010;41:368–84.
50. Lu P, Wen F, Li Y, Chen D. Individual behaviors, social learning, and swarm intelligence: real case and counterfactuals. *Expert Syst Appl*. 2022;207:117878. doi:10.1016/j.eswa.2022.117878.
51. Shafiq M, Ali ZA, Israr A, Alkhamash EH, Hadjouni M. A multi-colony social learning approach for the self-organization of a swarm of UAVs. *Drones*. 2022;6:104. doi:10.3390/drones6050104.
52. Mirjalili S, Lewis A. The whale optimization algorithm. *Adv Eng Softw*. 2016;95:51–67. doi:10.1016/j.advengsoft.2016.01.008.
53. Dehghani M, Trojovský P. Osprey optimization algorithm: a new bio-inspired metaheuristic algorithm for solving engineering optimization problems. *Front Mech Eng*. 2023;8:1126450. doi:10.3389/fmech.2022.1126450.
54. Su H, Zhao D, Heidari AA, Liu L, Zhang X, Mafarja M. RIME: a physics-based optimization. *Neurocomputing*. 2023;532:183–214. doi:10.1016/j.neucom.2023.02.010.
55. Mirjalili S, Mirjalili SM, Hatamlou A. Multi-verse optimizer: a nature-inspired algorithm for global optimization. *Neural Comput Appl*. 2016;27:495–513. doi:10.1007/s00521-015-1870-7.
56. Kiran MS. TSA: tree-seed algorithm for continuous optimization. *Expert Syst Appl*. 2015;42:6686–98. doi:10.1016/j.eswa.2015.04.055.
57. Kamil AT, Saleh HM, Abd-Alla IH. A multi-swarm structure for particle swarm optimization: solving the welded beam design problem. *J Phys: Conf Series*. 2021;1804(1):012012. doi:10.1088/1742-6596/1804/1/012012.
58. Tzanetos A, Blondin M. A qualitative systematic review of metaheuristics applied to tension/compression spring design problem: current situation, recommendations, and research direction. *Eng Appl Artif Intell*. 2023;118:105521. doi:10.1016/j.engappai.2022.105521.
59. Min F, Huang H. An improved dragonfly optimization algorithm for solving numerical and three-bar truss optimization problems. In: *2021 5th Asian Conference on Artificial Intelligence Technology (ACAIT); Haikou, China; 2021*. p. 204–18.
60. Mohammed H, Rashid T. A novel hybrid GWO with WOA for global numerical optimization and solving pressure vessel design. *Neural Comput Appl*. 2020;32(18):14701–18. doi:10.1007/s00521-020-04823-9.
61. Gunawan S, Azarm S, Wu J, Boyars A. Quality-assisted multi-objective multidisciplinary genetic algorithms. *AIAA J*. 2003;41:1752–62. doi:10.2514/2.7293.