Cauchy Mutation Chaotic Coati Optimization Algorithm

Yu-Wei Song, Wei-Zhong Sun*, Jie-Sheng Wang, Yu-Liang Qi, Xun Liu, Yuan-Zheng Gao

Abstract—**Coati Optimization Algorithm (COA) is a novel heuristic algorithm that simulates the intricate behavioral repertoire of coatis manifestations during their pursuit of iguanas as well as their evasive tactics against predators. Aiming to elevate the convergence characteristics regarding speed and precision of the original algorithmic process, and addressing the issue of susceptibility to local optima, an enhanced Coati Optimization Algorithm is suggested based on Cauchy mutation and chaotic maps. Firstly, the Cauchy mutation is embedded in the process of coatis hunting for iguanas. Subsequently, each of ten chaotic maps was incorporated into the probing phase of the COA, generating ten unique enhanced versions. This augments the algorithm's precision in optimization, bolsters the equilibrium between exploration and exploitation and enhances its itinerancy and non-redundancy. The enhanced optimization algorithm exhibiting the aggregated optimal performance is selected from ten types of different enhanced COA and used for subsequent comparisons with other algorithms for function optimization and engineering optimization. Thirty benchmark functions from the CEC-BC-2022 datasets were incorporated for evaluation on performance metrics of the improved COA with Cauchy mutation and ten types of chaotic maps, and then the performance of the Chebyshev map enhanced COA and other six swarm intelligent algorithms are compared for optimization purposes. Finally, optimization was performed on four distinct engineering design problems. The outcomes of the simulation experiment evidence that the advanced Cauchy mutation chaotic COA yields satisfactory outcomes in addressing both function optimization and engineering optimization. The algorithm exhibits superiority in balancing exploration and exploitation during the iterative procedure of optimization, thereby enhancing convergence precision.**

Manuscript received December 17, 2023; revised April 5, 2024. This work was supported by the Basic Scientific Research Project of Institution of Higher Learning of Liaoning Province (Grant No. LJKZ0293), and Postgraduate Education Reform Project of Liaoning Province (Grant No. LNYJG2022137).

Yu-Wei Song is a postgraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: 2296087986@qq.com).

Wei-Zhong Sun is a lecturer of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (Corresponding author, phone: 86-0412-2538355; fax: 86-0412-2538244; e-mail: weizhongsun@126.com).

Jie-Sheng Wang is a professor of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: wang_jiesheng@126.com).

Yu-Liang Qi is a postgraduate student at School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan 114051, China (E-mail: 2281424682@qq.com).

Xun Liu is a postgraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: heidengxiaguo@163.com).

Yuan-Zheng Gao is a postgraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: 2452837718@qq.com).

Index Terms—**Coati optimization algorithm, Cauchy mutation, Chaotic maps, Function optimization, Engineering optimization**

I. Introduction

Optimization problems encompass a category of mathematical challenges whose underlying goal entails is to identify the supremely satisfactory solution within a given problem context [1]. Generally, optimization problems require to solve a target function while satisfying a set of constraints [2]. This involves many different algorithms and techniques, including heuristic search, stochastic search, gradient descent, etc. Optimization problems are extensively employed across a myriad of domains, including machine learning, artificial intelligence and engineering design. To cope with the escalating intricacy of optimization problems, a proliferating set of algorithms for optimization purposes has been introduced to address those challenging problems. The optimization algorithm has many advantages, which are described as follows. (1) Locate the supremely satisfactory solution within a reasonable time-frame. (2) Address intricate and multi- dimensional issues. (3) Flexibility and universality. (4) Provide global optimal solutions and avoid local optimal solutions [3].

As algorithms for optimization purposes continue to evolve and develop, heuristic algorithms have gradually emerged [4]. Heuristic algorithms rely on experience and intuition, striving to provide a practical resolution for every occurrence of a combination optimization issue while ensuring the cost remains reasonable and to discover an approximate optimal solution within a limited time-frame. Due to the stochastic of the algorithm, the feasible solutions are often different each time. Heuristic algorithms are ordinarily employed in scenarios involving the resolution of the problems associated with the optimization of combination structures, such as the traveling salesman problem and graph coloring problem, etc [5]. The solution space of such problems is usually very large, and may even be exponential, so it is often not practical to use traditional exhaustive methods to solve such problems. The deployment of a heuristic approach enables the provision of an approximate solution to the problem at an acceptable cost, so it has wide application value in practice. Meta-heuristic algorithms for optimization purposes represent an enhancement over heuristic algorithms for optimization purposes, arising from the concurrent utilization of stochastic methods and partial searching strategies. These algorithms are typically constructed with reliance on intuition or experience, and they are capable of generating a feasible solution to a problem within a tolerable amount of expenditure, while the degree of deviation from the supremely satisfactory solution is notwithstanding its lack of predictability beforehand [6-7]. In solving the complex optimization problems, these algorithms usually find a relatively optimal solution within a reasonable time-frame. Nonetheless, the utilization of heuristic algorithms does not ensure the attainment of the global optimum solution, as they might merely provide an approximation towards the ultimate optimal resolution for the aforementioned problem. Within the domain of meta-heuristic algorithms specifically designed for optimization objectives, two crucial exploration strategies are employed: exploration for enhancing diversity and exploitation for intensification [8]. Probing refers to the capacity to traverse the space exploration domain effectively and globally. The probing phase continues in when the algorithm aims to uncover as many promising regions in the space exploration domain as possible [9], which is associated with avoiding local optima and addressing local optima. In contrast, exploitation is in the vicinity of probing to enhance local optimization [10]. The meta-heuristic optimization algorithm has many advantages when confronted with intricate and grand-scale optimization dilemmas. It can find the approximate optimal solution with less computational cost and deal with non-differentiable, nonlinear, multi-peak, high-dimensional and complex global optimization problems.

Across scientific probing and practical implementation, meta-heuristic algorithms for optimization purposes have gained prominence as an indispensable tool. Owing to these distinctive features, many researchers have engineered a comprehensive collection of meta-heuristic algorithms for optimization purposes for deployment in the realm of optimization, such as Tabu Search (TS) [11], Simulated Annealing (SA), Hill Climbing Algorithm (HCA) [12], Genetic Algorithm (GA) [13], and so on. Nowadays, the majority of meta-heuristic algorithms designed for optimization purposes have their roots in bionics, and these algorithms have gained widespread application in the field of optimization, such as Mayfly Optimization Algorithm (MOA) [14], Grasshopper Optimization Algorithm (GOA) [15], Dwarf Mongoose Optimizer (DMO) [16], Firefly Algorithm (FA) [17], White Shark Optimizer (WSO) [18], Whale Optimization Algorithm (WOA) [19], Marine Predators Algorithm (MPA) [20], Slime Mould Algorithm (SMA) [21], Reptile Search Algorithm (RSA) [22], Osprey Optimization Algorithm (OOA) [23], Gravitational Search Algorithm (GSA) [24], and so on.

Cauchy mutation represents a stochastic perturbation method derived from the Cauchy distribution, which is utilized in algorithms for optimization purposes to boost variety within the population and uphold the algorithm's comprehensive exploitation capacity. In traditional genetic algorithms, mutation operations are usually implemented by stochastic changing the binary state of some bits. On the other hand, the Cauchy mutation based genetic algorithm can achieve variation by adding a variable of an individual within a Cauchy distribution stochastic number. The Cauchy distribution is a density function of probability akin to the normal distribution, albeit with broader tails and slower convergence. Introducing Cauchy mutation can boost population diversity, upgrade the algorithm's global exploitation skills and assist in preventing convergence to the local optimal solutions. These properties of the Cauchy mutation can be applied to optimization problems in a clever way. Chen et al. explored a hybrid artificial bumblebee algorithm by incorporating elite opposite learning and Cauchy mutation for the optimization of spherical curve shapes [25]. Bao et al. enhanced the teaching-learning-based optimization algorithm through the implementation of a chaotic operator and Cauchy distribution, thereby refining its performance and applicability [26]. Zhang et al. innovated an adaptive enhancement of the sand cat swarm algorithm by using an optimal neighborhood disruption strategy and Cauchy distribution, thereby fostering its computational efficiency and problem-solving capabilities [27].

A chaotic map generates a stochastic sequence originating from a straightforward deterministic system. Typical chaotic sequences are characterized by several key features, such as nonlinearity, dependence on initial conditions, ergodicity, stochasticness [28], strange attractors (chaotic attractors), fractional persistence, global stability, local instability, long-term unpredictability, orbital instability, bifurcation, as well as universality. The chaotic map is utilized for generating chaotic sequences, where a stochastic output is derived from simple deterministic systems [29].

In the realm of optimization, chaotic maps can serve as a substitute for pseudo-stochastic number generators, creating chaotic numbers within the range of 0 to 1. Experimental outcomes verify that the fitness function values obtained by using chaotic maps for generating stochastic numbers. Employing chaotic maps instead of conventional uniform distribution stochastic number generators can lead to superior outcomes, especially when numerous local optima exist in the space exploration domain, facilitating global optimal solution discovery. Indeed, chaos mapping has been utilized to enhance a multitude of algorithms. Chaotic Particle Swarm Optimization (CPSO) is improve to enhance the search process by incorporating chaotic mapping, thereby improving stochastic and global exploration capabilities. This algorithm exhibits widespread applications in many fields, such as function optimization, constraint optimization and dynamic programming [30]. Chaotic Genetic Algorithm (CGA) combines GA operations, such as crossover and mutation, with chaotic mapping to enhance global search capability and convergence speed. In complex engineering optimization and combination optimization problems, CGA demonstrates superior performance [31]. The chaotic Simulated Annealing Algorithm (CSA) incorporates chaotic mapping into the SA algorithm to accelerate convergence and enhance global search capability. It exhibits high-quality solutions for complex optimization problems [32]. Chaotic Artificial Neural Network (CANN) is a neural network based on chaotic dynamics that enhances training speed and performance by incorporating chaotic mapping. CANN has achieved significant accomplishments in pattern recognition, image processing and speech recognition domains [33].

Furthermore, chaos-oriented theories have found applications in areas like cryptography, DNA computation, image manipulation, and nonlinear circuits. These collective efforts manifest that across diverse engineering domains, the newly developed algorithm exhibits enhanced performance and superior beneficial precision in comparison to its fundamental counterpart.

The Coati Optimization Algorithm (COA), a novel heuristic algorithm, effectively emulates the inherent behavior of coatis while they hunt for iguanas and evade potential threats from predators. This article proposes an enhanced Coati Optimization Algorithm based on Cauchy mutation and chaotic maps. First, the Cauchy mutation was embedded in the process of coatis hunting for iguanas. Subsequently, each of ten chaotic maps was incorporated into the probing phase of the COA, generating ten unique enhanced versions. This augments the algorithm's precision in optimization, bolsters the equilibrium between exploration and exploitation, amplifies its itinerant capacity and minimizes redundancy. In addition, the enhanced COA with the premier integrated outcome is selected from ten types of different algorithms for optimization purposes formed after the enhancement, and the subsequent comparison and engineering optimization with other algorithms for optimization purposes is carried out. Throughout the process of enhancing the algorithm, thirty benchmark functions from the CEC-BC-2022 datasets were incorporated for evaluation. First, the performance of the enhanced COA with Cauchy mutation and ten types of chaotic maps is tested. After that, the performance of the Chebyshev map enhanced Coati Optimization Algorithm, and other six types of algorithms for optimization purposes, which have the best comprehensive effect, are tested. Finally, optimization was performed on four distinct engineering design challenges. The outcomes of the simulation experiments proved that the advanced Cauchy mutation chaotic Coati Optimization Algorithm yields satisfactory outcomes in addressing both function optimization and engineering optimization problems. The algorithm exhibits superiority in balancing exploration and exploitation during the iterative procedure of optimization, thereby enhancing convergence precision.

II. Coati Optimization Algorithm

In the realm of computer science and optimization, single-objective optimization concerns to select the most favorable solution from all potential alternatives relying on a particular metric, generally aimed at minimizing or maximizing the objective function. Conversely, artificial intelligence algorithms for optimization purposes emulate the evolutionary processes or individual learning experiences found in nature. These algorithms quest for optimal solutions by progressively exploring the solution space through iterative searches.

The coatis, an avid diurnal mammal, primarily inhabits the southwestern region of North America, Mexico, Central America and South America [34]. Being an omnivore, its diet is exceedingly diverse, encompassing invertebrates, small vertebrates and the green iguana. Given the frequent occurrence of this large iguana in trees, coatis collaborate in hunting them. Some coatis ascend trees to scare the green iguanas into jumping to the ground, while others promptly pounce on them during the ensuing chaos. However, coatis are still at risk from predators. Jaguars, tigers and foxes are predators of coatis. Thus, the Coati Optimization Algorithm (COA) is characterized as a meta-heuristic optimization algorithm, derived from the mimicry of coatis' methodologies for pursuing iguanas and their adaptive reactions to avoid predators. In the following sections, the algorithm initialization process of the COA is discussed and two stages of probing and development and mathematical modeling of the COA are described.

A. Algorithm Initialization Process

In the Coati Optimization Algorithm (COA), the coatis act as population constituents. Each coati's location within the space exploration domain is deemed as the significance of the determining decision variable (spatial location). Therefore, every coati's spatial location symbolizes a potential solution regarding the optimization issue under consideration. Due to the stochastic of coati locations, it is imperative to initialize the population based on the problem's upper and lower constraints, as shown in Eq. (1).

$$s_i(j) = lb_j + r\cdot(ub_j - lb_j), i = 1,2,\ldots,N, j = 1,2,\ldots,m. \quad (1)$$

In the COA, the coati's population is mathematically represented by the following matrix S, which are called as the population matrix. In this matrix, each row signifies a potential solution, while the columns thereof depict suggested parameters for the problem's variables, which is shown in Eq. (2).

$$S = \begin{bmatrix} S_1 \\ \vdots \\ S_i \\ \vdots \\ S_N \end{bmatrix}_{N\times m} = \begin{bmatrix} s_1(1) & \cdots & s_1(j) & \cdots & s_1(m) \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ s_i(1) & \cdots & s_i(j) & \cdots & s_i(m) \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ s_N(1) & \cdots & s_N(j) & \cdots & s_N(m) \end{bmatrix}_{N\times m} \quad (2)$$

where, the j-dimensional spatial location of the i-th coatis individual is denoted with N representing the population magnitude of the coatis. The variable m refers to the decision variables' dimension, r denotes a stochastic real number within the range [0, 1], and represents the upper and lower bounds of the j-dimensional decision variable.

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(x_1) \\ \vdots \\ F(x_i) \\ \vdots \\ F(x_N) \end{bmatrix}_{N\times 1} \quad (3)$$

where, F represents the fitness vector of the population of coatis, which is based on the optimal fitness values obtained for each coati and is used to calculate the suitability.

In the COA, the quality metric for the candidate solution is equivalent to the importance $Iguana^t_{ground}$ of the optimal fitness value of $s^t_i(j)$. In effect, in a population, the individual that causes the objective function to evaluate the optimal worth is called the best population individual. Because candidate solutions are updated during algorithmic iterations, the optimal population of individuals is updated with each iteration.

B. Mathematical Model of COA

The COA is an iterative procedure for refining the spatial location. These two inherent behaviors are exhibited by the coatis. These behaviors include:

(1) Strategies for coatis to attack iguanas.

(2) Strategies used by coatis to escape predators.

Thus, the population of COA is updated in two distinct stages.

(1) Probing phase (Phase 1) : strategies for hunting and attacking iguanas

In Phase 1, the mathematical model for updating the coatis population is derived from the simulation of their behavioral patterns when attacking iguanas. In this behavioral pattern, a collective of coatis ascend trees near the iguana and drive it away, while other coatis wait below the tree until the iguana descends to the ground. Upon landing, the coatis attack and kill it. This strategy enables the COA to transition to various locations within the space exploration domain, showcasing its comprehensive probing capability in the problem domain.

Upon designing the COA, assume the spatial location of the optimal individual s_{best}^t within the current population to represent the location of the iguana. Suppose half the coatis climb trees to hunt for iguana, and the other half await the iguana's descent to the ground. Hence, Eq. (4) is employed to numerically represent the coatis' location within the tree.

$$s_i^{t+1}(j) = s_i^t(j) + r \cdot (s_{best}^t(j) - I \cdot s_i^t(j)),\ i = 1,2,\ldots,\left[\frac{N}{2}\right] \quad (4)$$

Upon the iguana's descent to a stochastic location on the ground within the search space, the coatis on the ground initiate movement within the search space with the stochastic spatial location serving as the central point of their trajectory, and update the location by using Eq. (5)-(6).

$$Iguana_{ground}^t(j) = lb_j + r \cdot (ub_j - lb_j),\ j = 1,2,\ldots,m \quad (5)$$

$$s_i^{t+1}(j) = \begin{cases} s_i^t(j) + r \cdot (Iguana_{ground}^t(j) - I \cdot s_i^t(j)), \\ \qquad\qquad if \quad F(Iguana_{ground}^t) < F(s_i^t) \\ s_i^t(j) + r \cdot (s_i^t(j) - Iguana_{ground}^t(j)), \quad else \end{cases}$$
$$i = \frac{N}{2}+1, \frac{N}{2}+2, \ldots, N \quad (6)$$

where, t denotes the current iteration count, r represents a stochastic number within the range [0, 1], and T signifies a stochastic integer chosen between 1 and 2. So the $Iguana_{ground}^t$ stands for the updated spatial location of the iguana following its descent to the ground. $s_i^t(j)$ signifies the j-dimensional attribute value of the i-th entity in the present iteration.

If the computed updated location of the present coatis enhances the performance of the target function, it is considered as satisfactory. Otherwise, the coatis remain in their previous spatial location. That is to perform a greedy choice, as shown in Eq. (7).

$$s_i^{t+1} = \begin{cases} s_i^{t+1}, & if \quad F(s_i^{t+1}) < F(s_i^t) \\ s_i^t, & else \end{cases} \quad (7)$$

(2) Development stage (Phase 2) : strategies for coatis to attack iguanas

In Phase 2, The mathematical model underlying the procedure of refining the coatis' spatial locations within the search domain, represents a simulation of the natural behavior of these creatures as they evade predators. Upon a predator's assault on a coatis' entity, the coati flees from its current location. The coatis' maneuvers are spatial locations them in a secure proximity to their present site, which is the development capability of COA in partial searching. In order to replicate these behavioral patterns, a stochastic spatial location is computed near each coati's site, as illustrated by Eq. (8)-(9).

$$lb_j^{local} = \frac{lb_j}{t}, ub_j^{local} = \frac{ub_j}{t}, t = 1,2,\ldots,T \quad (8)$$

$$s_i^{t+1}(j) = s_i^t(j) - (1-2r) \cdot (lb_j^{local} + r \cdot (ub_j^{local} - lb_j^{local})),$$
$$i = 1,2,\ldots,N \quad (9)$$

where, r signifies a stochastic digit ranging from [0, 1], while T represents the uppermost limit of iterations. It can be observed that ub_j^{local} and lb_j^{local} serve as the upper and lower bounds for the updating of the j-th dimension variable with respect to the iteration count. If the recently computed spatial location bolster the prominence of the objective function, it is deemed appropriate, and simulation proceeds with the execution of greedy selection once more, as shown in Eq. (7). Otherwise, the coati remains in its previous spatial location.

III. Cauchy Mutation and Chaotic Coatis Optimization Algorithm

A. Cauchy Mutation

In response to the tendency of the COA to converge to the local optima, a novel approach relying on Cauchy mutation is further proposed. The Cauchy mutation draws from the continuous probability distribution known as the Cauchy distribution, characterized by a re

latively small peak at the origin and a gradual decline from the peak to zero, resulting in a more uniform mutation range. Introducing cauchy mutation during the update of the optimal individual location in the population enhances the heterogeneity of the population, improve the universal search capacity and expand the space-searchable domain of the algorithm. The mutation formula is defined as:

$$Cauchy\_s = b \times \tan(\pi \cdot (randn[1,d] - 0.5)), b = 0.01 \quad (10)$$

where, stochastic number generator randn(1,d) is used to generate stochastic vectors with Normal distribution, which enhances the stochastic of the generated cauchy distribution. By adding tan function to map the stochastic variables of Normal distribution to the interval (-π,π), the resulting cauchy distribution has a wider range of angles. By adjusting the parameter b in the tan function (here is 0.01), it can control the scale of the generated Cauchy distribution.

During the probing phase, the spatial location of the optimal individual s_{best}^t in the population is hypothetically assigned as the spatial location of the iguana, as the iguana's location where it falls from the tree due to being frightened by the coatis is prone to reaching local optima. To enable individuals to find optimal solutions more effectively, Cauchy mutation is incorporated into the probing phase algorithm. Similarly, in the exploitation phase, the spatial location of the optimal individual s_{best}^t in the population is assumed to be the location of the coati pursued by the predator. During the procedure of evading the predator, the coati tends to choose safe locations that lead to local optima. Therefore, cauchy mutation is also integrated into the exploitation phase algorithm. The method of incorporating cauchy mutation is identical in both phases, and its mathematical representation is provided below.

$$S_{new\_best} = S_{best} + S_{best} \times Cauchy(d) \tag{11}$$

Upon testing, the convergence curve of the COA with the incorporation of Cauchy mutation demonstrates a significantly superior performance compared to the original COA, so COA with Cauchy mutation is embedded in the following algorithm enhancement.

B. *Chaotic Maps Expressions and Visualized Graphics*

Ten most prevalent chaotic maps are employed to enhance the COA based on Cauchy mutation. The map expressions of these ten most prevalent chaotic maps can be seen in Eq. (12) to Eq. (21), and their visualized graphics are shown in Fig. 1.

(1) Chebyshev map (CH)

$$s_{k+1} = \cos(k \cos^{-1}(s_k)) \tag{12}$$

(2) Circle map (CI)

$$s_{k+1} = (s_k + b - (\frac{a}{2\pi})\sin(2\pi s_k)) \bmod(1) \tag{13}$$

where, $a = 0.5$, $b = 0.2$.

(3) Gauss map (GA)

$$s_{k+1} = \begin{cases} 0, s_{k=0} \\ \dfrac{1}{s_k \bmod(1)}, otherwise \end{cases}$$
$$\frac{1}{s_k \bmod(1)} = \frac{1}{s_k} - \left[\frac{1}{s_k}\right] \tag{14}$$

(4) Iterative map (IT)

$$s_{k+1} = \sin(\frac{a\pi}{s_k}) \tag{15}$$

where, $a \in (0,1)$.

(5) Logistic map (LO)

$$s_{k+1} = a s_k (1 - s_k) \tag{16}$$

where, x_k is the k -th chaotic number, k is the number of iterations, $x \in (0,1)$, $x_0 \notin (0, 0.25, 0.5, 0.75, 1)$, $a = 4$.

(6) Piecewise map (PI)

$$s_{k+1} = \begin{cases} \dfrac{s_k}{p}, 0 \le s_k < p \\ \dfrac{s_k - p}{0.5 - p}, p \le s_k < \dfrac{1}{2} \\ \dfrac{1 - p - s_k}{0.5 - p}, \dfrac{1}{2} \le s_k < 1 - p \\ \dfrac{1 - s_k}{p}, 1 - p \le s_k < 1 \end{cases} \tag{17}$$

where, $p \in (0,1)$, $x \in (0,1)$.

(7) Sine map (SI)

$$s_{k+1} = \frac{a}{4}\sin(\pi s_k) \tag{18}$$

where, $a \in (0, 4]$.

(8) Singer map (SG)

$$s_{k+1} = \mu(7.86 s_k - 23.31 s_k^2 + 28.75 s_k^3 - 13.302875 s_k^4) \tag{19}$$

where, $\mu \in (0.9, 1.08)$.

(9) Sinusoidal map (SO)

$$s_{k+1} = a s_k^2 \sin(\pi s_k) \tag{20}$$

where, a =2.3

(10) Tent map (TE)

$$s_{k+1} = \begin{cases} \dfrac{s_k}{\alpha}, s_k < \alpha \\ \dfrac{(1 - s_k)}{1 - \alpha}, s_k \ge \alpha \end{cases} \tag{21}$$

where, $\alpha \in (0,1)$.

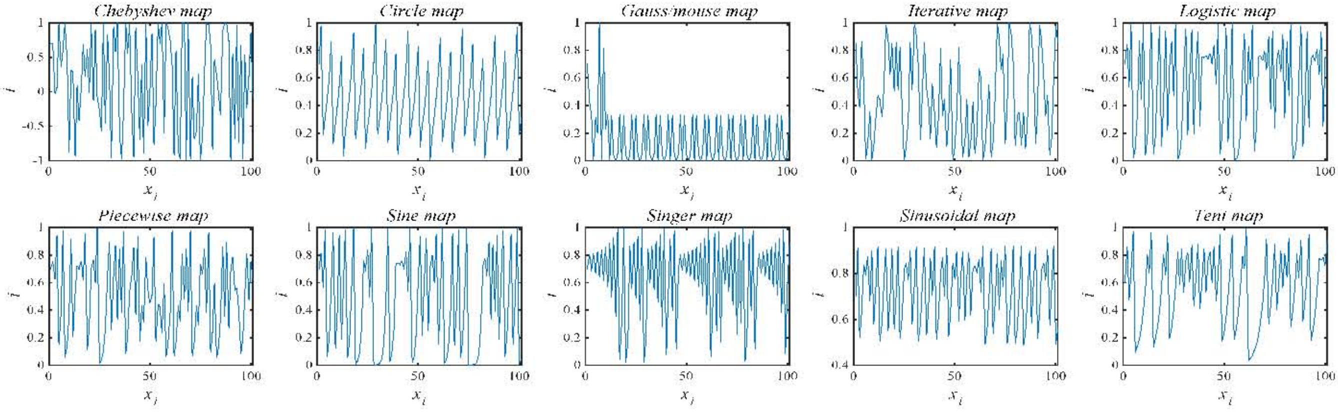


Fig. 1 Ten chaos visualizations.

C. Flowchart of Cauchy Mutation Chaotic COA

To enhance the searching precision of the algorithm, chaotic maps are introduced in the probing phase of COA. Ten distinct chaotic mappings are incorporated into Eq. (6) to replace the original stochastic numbers within the range [0, 1]. Whilst preserving the unchanged value range, these ten chaotic mappings possess unique variation curves, which alters the respective step sizes. This serves to enhance the traversal capability and non-repeatability of COA, thereby its convergence velocity and precision are amplified. The Chebyshev map serves as a prototype to update the spatial location of coatis. Eq. (6) after adding chaos map becomes:

$$s_i^{t+1}(j)=\begin{cases} s_i^t(j)+Cs\cdot(Iguana_{ground}^{t}(j)-I\cdot s_i^t(j)), \\ \qquad\qquad if \quad F(Iguana_{ground}^{t})<F(s_i^t) \\ s_i^t(j)+Cs\cdot(s_i^t(j)-Iguana_{ground}^{t}(j)), \ else \end{cases} \quad (22)$$

$$Cs=((s_{t+1}+1)\times Value)/2 \quad (23)$$

$$s_{k+1}=\cos(k\cos^{-1}(s_k)) \quad (24)$$

$$Value=\left(\frac{C\_Iter}{M\_Iter}\right)^{\frac{1}{6}} \quad (25)$$

where, t is the present iterative cycle, $s\in(0,1), s(1)=0.7$.

The algorithm flow of the enhanced COA based on Cauchy mutation and chaotic maps is shown in Fig. 2.

IV. Emulation Experiments and Result Interpretations

A. CEC2022 Function Optimization

Twelve test functions in the CEC-BC-2022 are adopted. To ensure the fairness of the experimental tests, the maximum iteration numbers for both the original COA and the enhanced Cauchy mutation chaotic COA for optimization purposes are set to 1000, with a population magnitude of 30. The optimization outcomes of these test functions are utilized to comprehensively demonstrate the superiority of the enhanced COA.

(1) CEC2022 test functions

These twelve functions chosen from the CEC-BC-2022 test suite in Ref. [1] utilized in this article all possess a 10-dimensional configuration. The test functions span a same range, which is uniformly [-100, 100]. The selected functions encompass four categories: Unimodal function f_1 ; Basic functions f_2-f_5 ; Hybrid functions f_6-f_8 ; Composition functions f_9-f_{12}.

(2) Cauchy mutation chaotic COA to solve CEC-BC-2022 test functions

This article chooses twelve test functions in the CEC-BC-2022 to compare the output of the enhanced COA with the original algorithm. Here, Cauchy mutation based employ and ten types of chaotic maps to enhance the COA, with each function having a dimension of 10 and the maximum iteration number 1000 for each algorithm. The experiments run for 30 iterations, recording the optimal solutions obtained in each of the 30 runs. Subsequently, mathematical statistics are conducted on the experimental outcomes to demonstrate the superiority of the enhanced COA over the original algorithm in subsequent proof. The mathematical statistical tests primarily utilize average values and variances for statistical analysis, facilitating a comprehensive evaluation of the experimental outcomes.

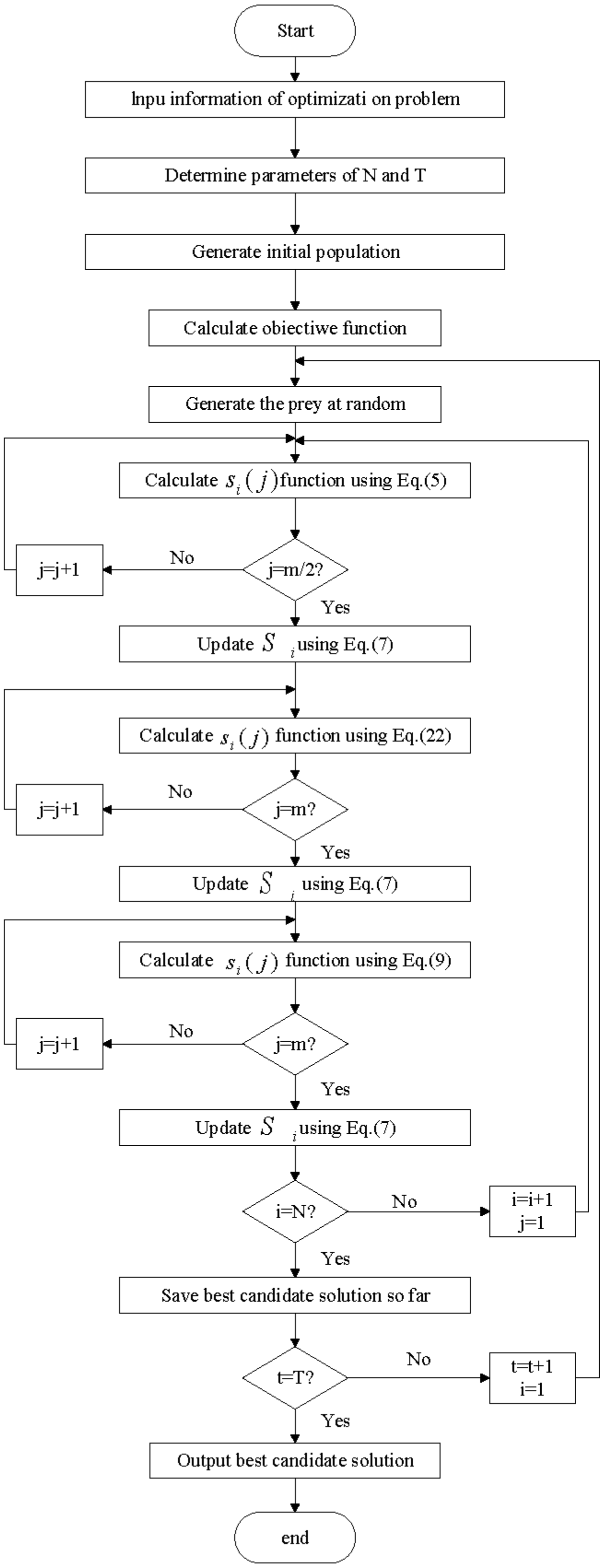
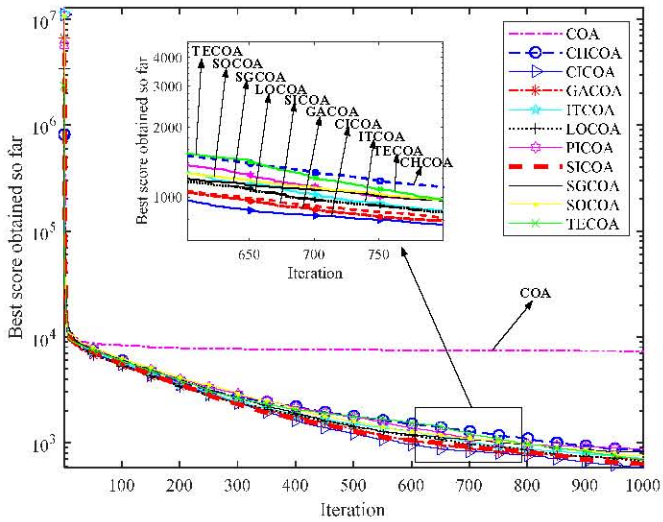


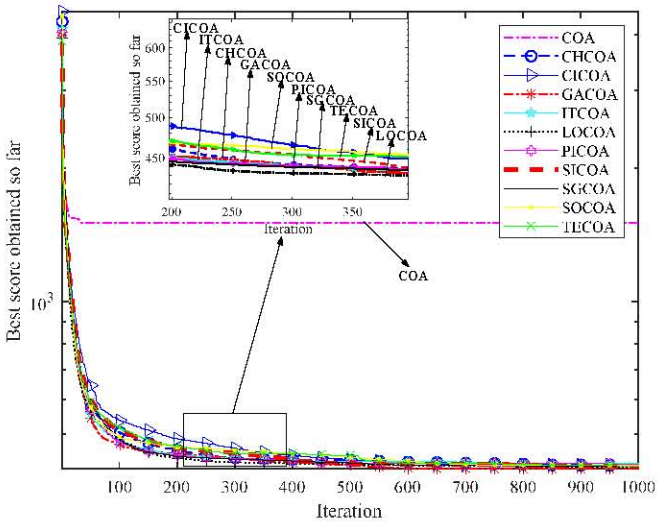
Fig. 2 Flow chart of enhanced Cauchy mutation chaotic COA.

The experimental outcomes have been organized in Table I, which displays the outcomes of the best solutions, average values and variances. To visually understand these outcomes, convergence curve plots were also created, as shown in Fig. 3. The graph observes that the convergence output of COA after improvement is superior to that of the original algorithm for the majority of functions. Overall, these experimental outcomes demonstrate the potential of the enhanced COA in practical applications and provide evidence for its suitability for a more extensive array of optimization issues.

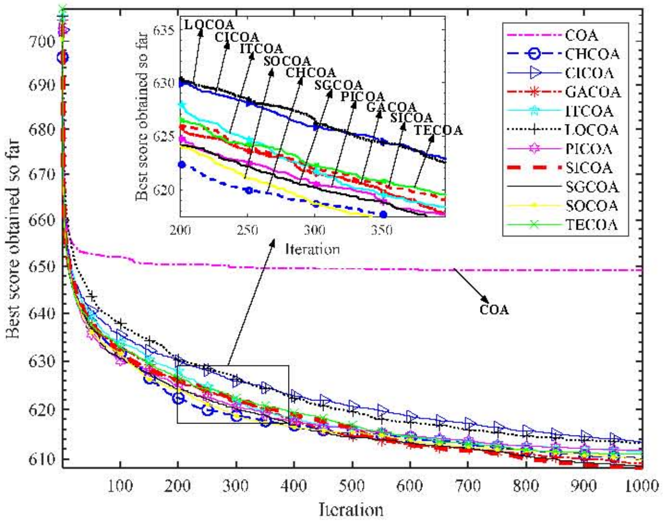
By examining the graphs in Fig. 3 and the statistics in Table I, the following conclusions can be summarized. The enhanced CHCOA achieves the smallest optimal values for functions $f_2, f_6, f_8 - f_{10}$, the smallest average values for functions $f_3 - f_5, f_7 - f_{10}, f_{12}$, and the smallest variances for functions f_{12}. The enhanced CICOA achieves the smallest optimal value for function f_5, the smallest average value for function f_1, and the smallest variance for function f_{10}. The enhanced GACOA achieves the smallest optimal value for function f_1, the smallest average value for function f_2. The enhanced ITCOA achieves the smallest optimal value for functions f_7, and the smallest variance for functions f_7. The enhanced LOCOA achieves the smallest average value for function f_{11}, and the smallest variances for function f_5, f_8, f_{11}. The enhanced SICOA achieves the smallest optimal value for function f_{12}, and the smallest variances for function f_2, f_3. The enhanced SGCOA achieves the smallest optimal values for function f_3, f_{11}, the smallest variance for functions f_4. The enhanced SOCOA achieves the smallest average value for function f_6, the smallest variances for functions f_6, f_9. The enhanced TECOA achieves the smallest optimal value for functions f_4, and the smallest variance for functions f_1. Reliance on the outcomes obtained in Table I, it can be ascertained that the COA enhanced by introducing the Chebyshev map has the best effect. Therefore, in the following comparison with other algorithms for optimization purposes, the COA enhanced by introducing the Chebyshev map is selected for further algorithm performance verification.



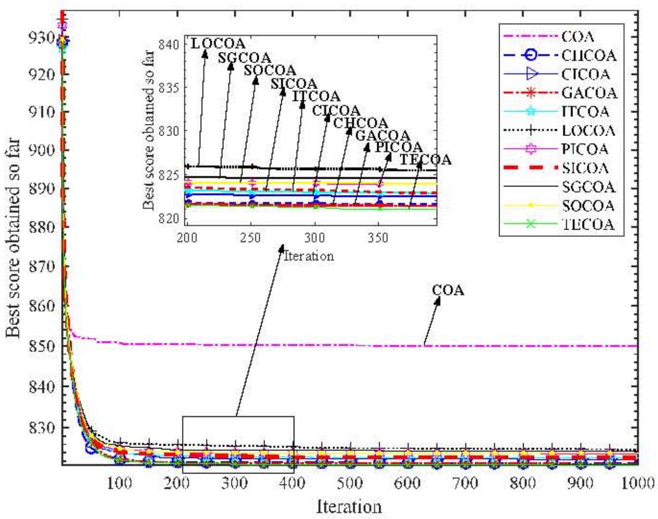
(1) f_1



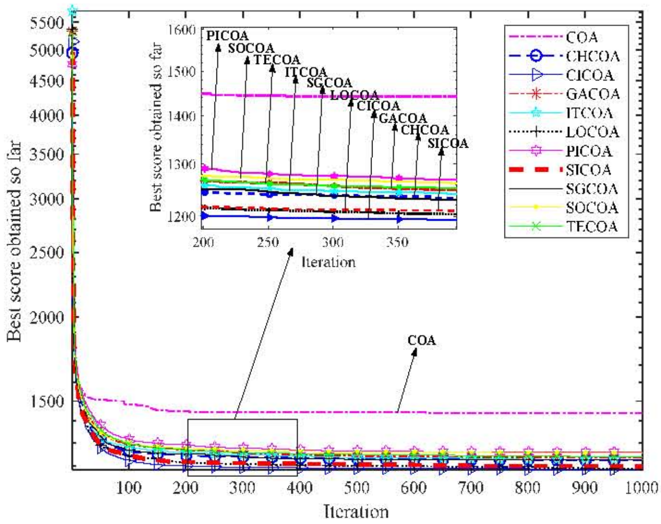
(2) f_2



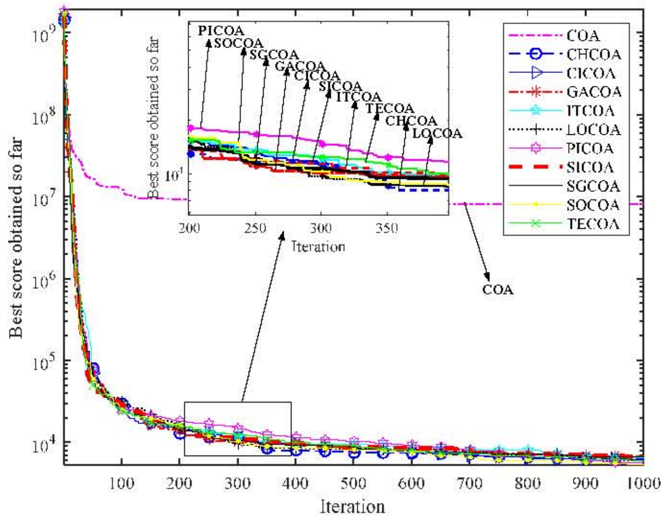
(3) f_3



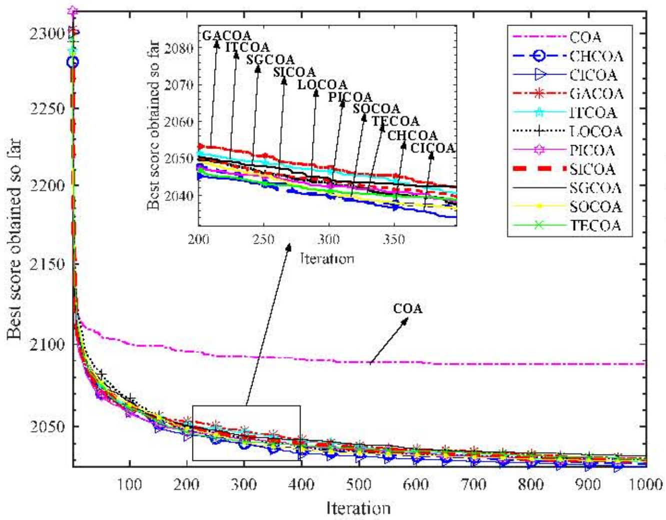
(4) f_4



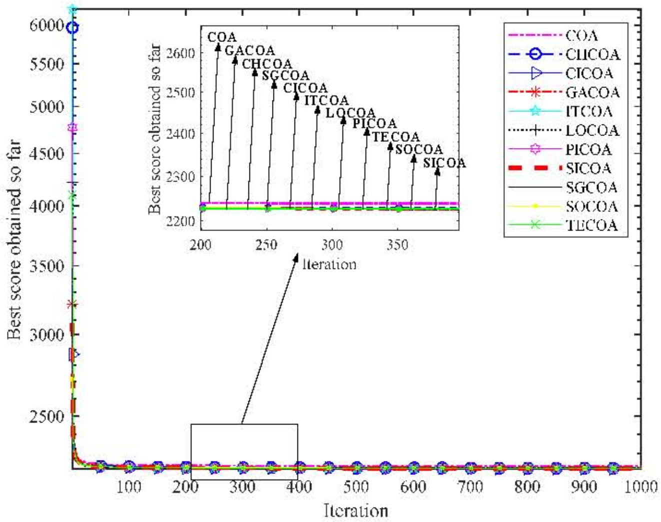
(5) f_5



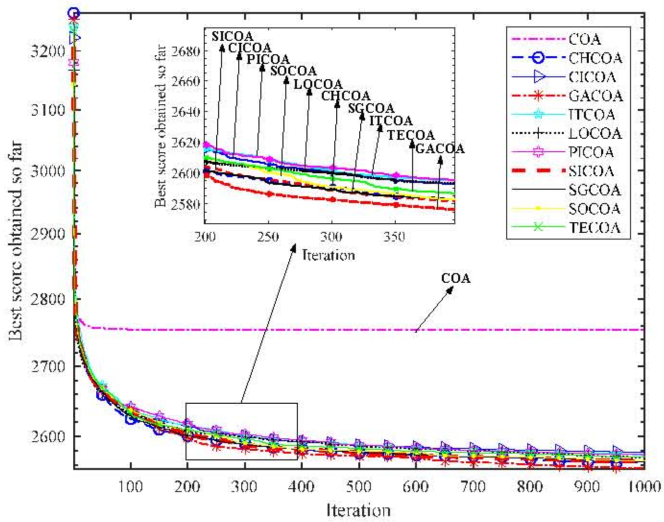
(6) f_6



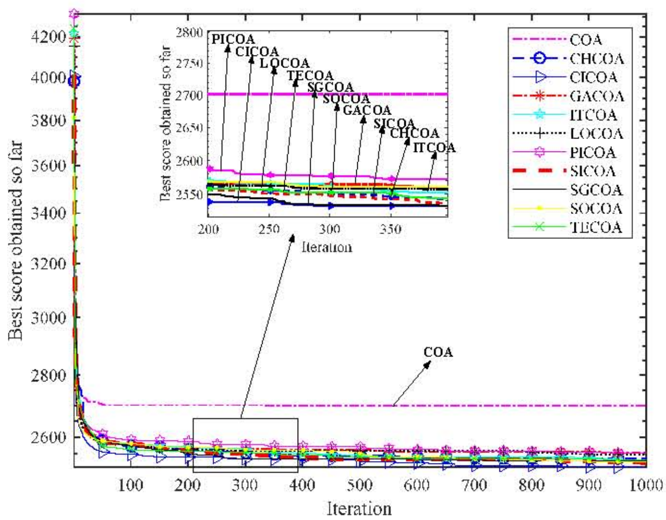
(7) f_7



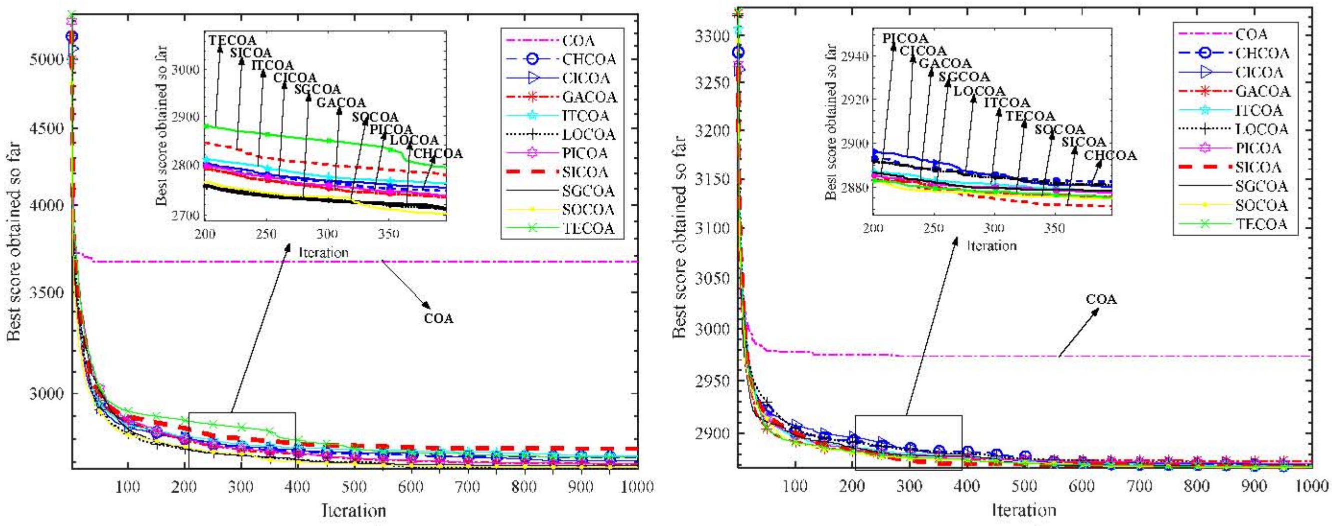
(8) f_8



(9) f_9



(10) f_{10}



(11) f_{11} (12) f_{12}

Fig. 3 The convergence graph of CEC-BC-2022 test functions solved by enhanced ten chaotic maps COA.

TABLE I. PERFORMANCE COMPARISON OUTCOMES OF CEC-2022 FUNCTION OPTIMIZATION

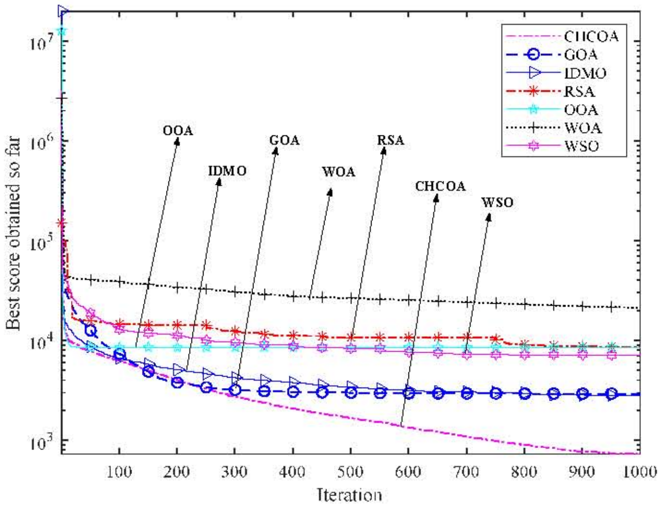
Function		COA	CHCOA	CICOA	GACOA	ITCOA	LOCOA	PICOA	SICOA	SGCOA	SOCOA	TECOA
f_1	Best	8.69E+03	1.96E+03	4.15E+02	**3.76E+02**	4.43E+02	3.97E+02	4.89E+02	4.18E+02	5.22E+02	4.65E+02	1.22E+03
	Ave	7.34E+03	8.36E+02	**5.85E+02**	6.92E+02	7.13E+02	6.72E+02	8.73E+02	6.32E+02	8.18E+02	7.49E+02	6.83E+02
	Std	1.86E+03	9.54E+02	7.25E+02	6.98E+02	7.68E+02	7.71E+02	8.02E+02	7.20E+02	1.02E+03	1.00E+03	**4.82E+02**
f_2	Best	1.77E+03	**4.00E+02**	4.11E+02	4.04E+02	4.00E+02	4.09E+02	4.74E+02	4.01E+02	4.09E+02	4.51E+02	4.71E+02
	Ave	1.51E+03	4.25E+02	4.26E+02	**4.18E+02**	4.29E+02	4.21E+02	4.29E+02	4.23E+02	4.23E+02	4.25E+02	4.24E+02
	Std	6.72E+02	2.97E+01	3.58E+01	2.76E+01	3.18E+01	2.89E+01	3.82E+01	**2.75E+01**	3.04E+01	2.93E+01	2.80E+01
f_3	Best	6.47E+02	6.13E+02	6.07E+02	6.13E+02	6.19E+02	6.03E+02	6.05E+02	6.03E+02	**6.02E+02**	6.03E+02	6.28E+02
	Ave	6.49E+02	**6.08E+02**	6.13E+02	6.09E+02	6.11E+02	6.13E+02	6.12E+02	6.08E+02	6.08E+02	6.10E+02	6.11E+02
	Std	8.51E+00	7.66E+00	9.99E+00	**6.55E+00**	7.52E+00	8.48E+00	7.59E+00	6.63E+00	7.01E+00	8.38E+00	8.46E+00
f_4	Best	8.48E+02	8.13E+02	8.18E+02	8.14E+02	8.26E+02	8.26E+02	8.23E+02	8.23E+02	8.18E+02	8.29E+02	**8.12E+02**
	Ave	8.50E+02	**8.21E+02**	8.22E+02	8.21E+02	8.23E+02	8.25E+02	8.24E+02	8.23E+02	8.24E+02	8.24E+02	8.21E+02
	Std	7.48E+00	6.49E+00	8.38E+00	7.48E+00	5.75E+00	6.73E+00	8.05E+00	5.61E+00	**5.49E+00**	6.00E+00	7.23E+00
f_5	Best	1.59E+03	1.17E+03	**1.10E+03**	1.22E+03	1.15E+03	1.19E+03	1.30E+03	1.10E+03	1.32E+03	1.22E+03	1.16E+03
	Ave	1.44E+03	**1.18E+03**	1.19E+03	1.24E+03	1.23E+03	1.19E+03	1.26E+03	1.20E+03	1.22E+03	1.26E+03	1.24E+03
	Std	1.55E+02	9.96E+01	1.14E+02	1.18E+02	1.20E+02	**9.43E+01**	1.43E+02	1.41E+02	1.31E+02	1.22E+02	1.28E+02
f_6	Best	5.52E+05	**2.53E+03**	6.74E+03	5.22E+03	4.65E+03	2.90E+03	4.43E+03	5.25E+03	9.56E+03	7.39E+03	3.60E+03
	Ave	8.14E+06	6.20E+03	6.22E+03	6.79E+03	6.41E+03	6.07E+03	5.72E+03	6.69E+03	6.63E+03	**5.30E+03**	6.01E+03
	Std	1.11E+07	2.41E+03	4.38E+03	3.14E+03	2.81E+03	3.27E+03	3.45E+03	3.71E+03	3.49E+03	**2.17E+03**	2.85E+03
f_7	Best	2.07E+03	2.03E+03	2.03E+03	2.03E+03	**2.02E+03**	2.03E+03	2.03E+03	2.03E+03	2.04E+03	2.03E+03	2.02E+03
	Ave	2.09E+03	**2.03E+03**	2.03E+03	2.03E+03	2.03E+03	2.03E+03	2.03E+03	2.03E+03	2.03E+03	2.03E+03	2.03E+03
	Std	1.84E+01	8.96E+00	1.03E+01	1.23E+01	**8.03E+00**	1.30E+01	1.07E+01	1.07E+01	1.04E+01	1.16E+01	1.22E+01
f_8	Best	2.24E+03	**2.22E+03**	2.23E+03	2.22E+03	2.23E+03	2.22E+03	2.22E+03	2.22E+03	2.22E+03	2.23E+03	2.22E+03
	Ave	2.24E+03	**2.22E+03**	2.22E+03	2.22E+03	2.22E+03	2.23E+03	2.23E+03	2.22E+03	2.22E+03	2.23E+03	2.22E+03
	Std	8.31E+00	3.92E+00	5.73E+00	3.48E+00	3.11E+00	**1.81E+00**	1.82E+00	5.16E+00	5.54E+00	2.02E+00	4.08E+00
f_9	Best	2.78E+03	**2.53E+03**	2.57E+03	2.54E+03	2.53E+03	2.57E+03	2.60E+03	2.60E+03	2.56E+03	2.63E+03	2.57E+03
	Ave	2.75E+03	**2.56E+03**	2.58E+03	2.56E+03	2.57E+03	2.57E+03	2.57E+03	2.57E+03	2.57E+03	2.57E+03	2.57E+03
	Std	3.59E+01	2.90E+01	3.64E+01	2.28E+01	4.05E+01	4.29E+01	3.14E+01	2.26E+01	3.74E+01	**2.24E+01**	3.95E+01
f_{10}	Best	2.55E+03	**2.50E+03**	2.50E+03	2.65E+03	2.50E+03	2.50E+03	2.63E+03	2.51E+03	2.50E+03	2.61E+03	2.63E+03
	Ave	2.70E+03	**2.50E+03**	2.51E+03	2.55E+03	2.53E+03	2.55E+03	2.55E+03	2.52E+03	2.53E+03	2.53E+03	2.53E+03
	Std	1.25E+02	5.69E+01	**2.86E+01**	6.32E+01	5.38E+01	6.17E+01	6.09E+01	4.98E+01	5.05E+01	4.97E+01	5.35E+01

	Best	3.41E+03	2.61E+03	2.61E+03	2.61E+03	2.61E+03	2.61E+03	2.61E+03	2.61E+03	**2.60E+03**	2.61E+03	2.92E+03
f_{11}	Ave	3.67E+03	2.72E+03	2.71E+03	2.69E+03	2.73E+03	**2.67E+03**	2.69E+03	2.76E+03	2.69E+03	2.68E+03	2.72E+03
	Std	4.11E+02	1.41E+02	1.22E+02	1.20E+02	1.43E+02	**8.45E+01**	1.39E+02	1.70E+02	1.26E+02	1.05E+02	1.65E+02
	Best	2.97E+03	2.87E+03	2.87E+03	2.87E+03	2.87E+03	2.87E+03	2.88E+03	**2.86E+03**	2.87E+03	2.87E+03	2.87E+03
f_{12}	Ave	2.97E+03	**2.87E+03**	2.87E+03	2.87E+03	2.87E+03	2.87E+03	2.87E+03	2.87E+03	2.87E+03	2.87E+03	2.87E+03
	Std	5.99E+01	**4.27E+00**	7.73E+00	1.51E+01	5.23E+00	7.15E+00	7.52E+00	5.30E+00	4.96E+00	5.07E+00	7.13E+00

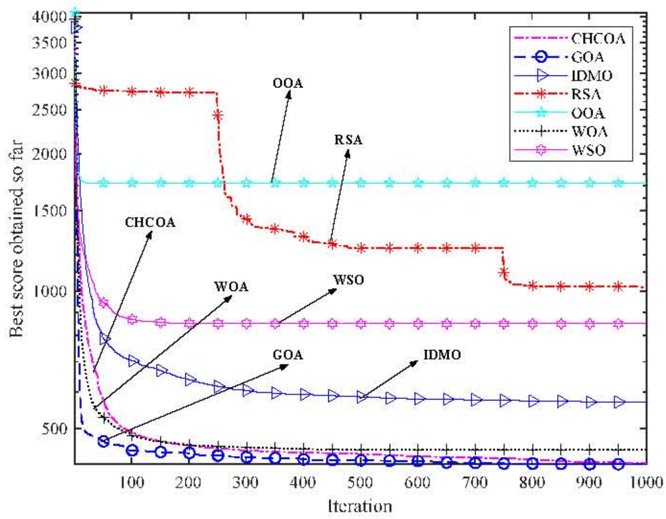
B. Comparison with Other Intelligent Algorithms for Optimization Purposes

By utilizing the chosen twelve test functions in the CEC-BC-2022, the output of the COA enhanced by Cauchy mutation and the chaotic map is compared with other intelligent optimization algorithms. The present study employs the COA with Chebyshev mapping improvement, alongside six other intelligent optimization algorithms, namely GOA, IDMO, RSA, OOA, WOA and WSO. Each of the chosen functions possesses a dimension of 10, with a maximum iteration number of 1000 generations for each algorithm. Each algorithm is executed in a loop for 30 iterations, and the optimal solution obtained in each loop is recorded. Experimental outcomes undergo mathematical statistical analyses to demonstrate the dominance of the improved COA compared to other intelligent optimization algorithms in terms of efficacy. The use of average and variance in statistical testing is beneficial for analyzing experimental results. The optimal values, average values and variances obtained from the experiments are shown in Table II. The convergence curves obtained from the experiments have been organized in Fig. 4, from which can be illustrated that the improved CHCOA outperforms the other six intelligent optimization algorithms for most of the functions.

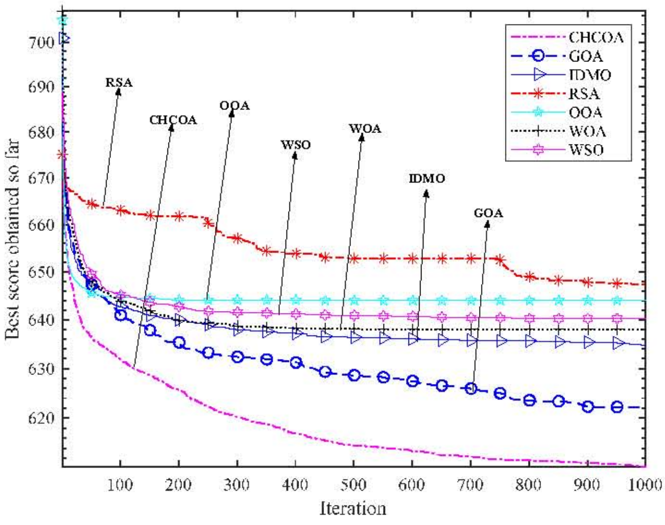
By examining the graphs in Fig. 4 and the statistics in Table II, the following conclusions can be summarized. Through the experimental outcomes of CEC-2022 function optimization, it can be found that the CHCOA exhibits superior output in terms of the optimal and average values, compared to the other six intelligent optimization algorithms. Furthermore, its relatively small variance demonstrates better stability. By comparing the optimization outcomes of the six algorithms in Table II, the following conclusions can be achieved. The enhanced CHCOA achieves the smallest optimal values for functions $f_2-f_4, f_7, f_8, f_{12}$, the smallest average values for functions f_1, f_3-f_5, f_7-f_{12}, and the smallest variances for functions $f_1, f_4, f_7-f_8, f_{11}-f_{12}$. From the above outcomes, it can be observed that the improved Cauchy mutation chaotic COA with the introduction of the Chebyshev map performs better in terms of overall performance and approaches the comprehensive optimization results of the majority of functions. Similarly, comparing the outcomes of optimization algorithms enhanced by incorporating other chaotic maps in Table I with those in Table II can also lead to the same conclusion. The enhanced CICOA obtains the optimal values from optimization functions $f_2, f_3, f_7, f_{10}, f_{12}$ are 4.11E+02, 6.07E+02, 2.03E+03, 2.50E+03 and 2.87E+03, which are the minimum compared with the optimal values of other six types of algorithms such as GOA and IDMO in Table II. The enhanced GACOA obtains the average values from optimizing functions f_8, f_{10} and f_{11} are 2.22E+03, 2.55e +03 and 2.69E+03 respectively, which are the minimum compared with the average values obtained by the other six types of algorithms in Table II. The enhanced LOCOA obtains the variances from optimization functions f_5, f_8 and f_{11} are 9.43e +01, 1.81E+00 and 8.45E+01 respectively, which are also the minimum values compared with the variances obtained by other six types of algorithms in Table II. Therefore, it can be concluded that the enhanced COA for optimization purposes reliance on Cauchy mutation and ten types of chaotic maps have better optimization effects than other algorithms, and the enhancement of the enhanced algorithm is deemed feasible.



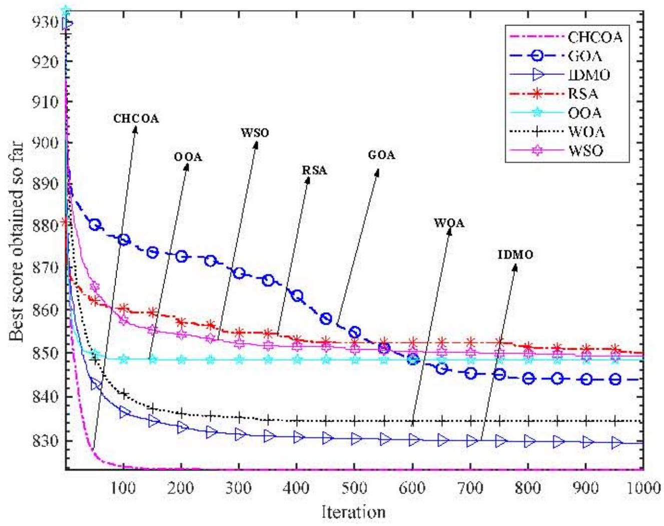
(1) f_1



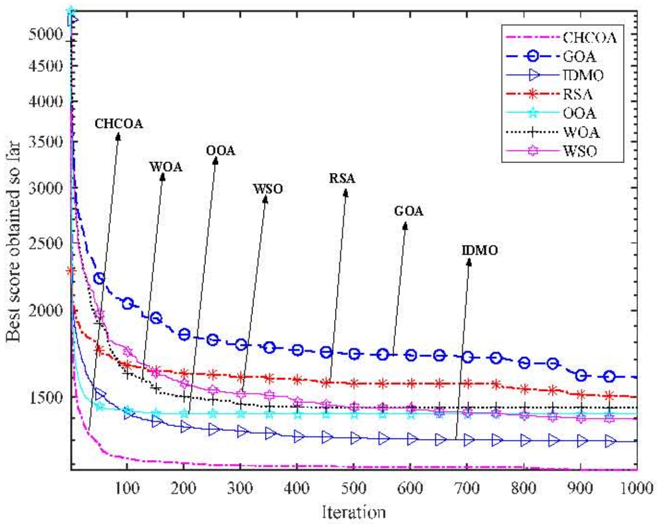
(2) f_2



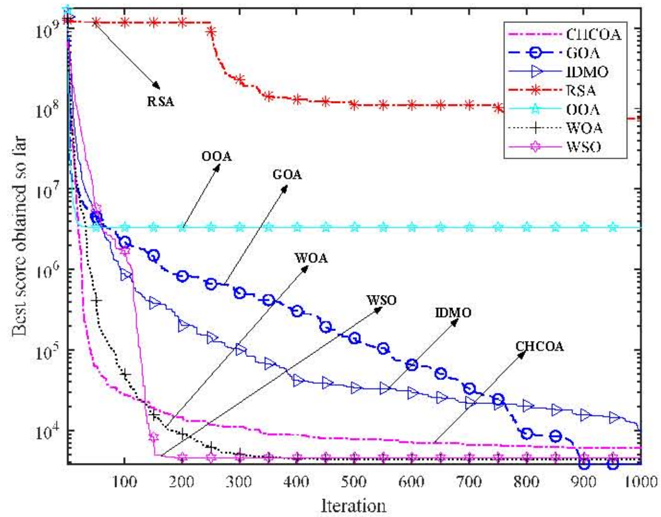
(3) f_3



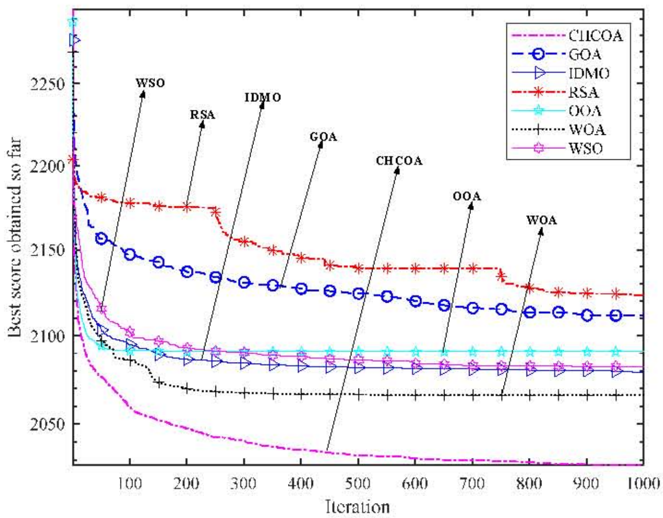
(4) f_4



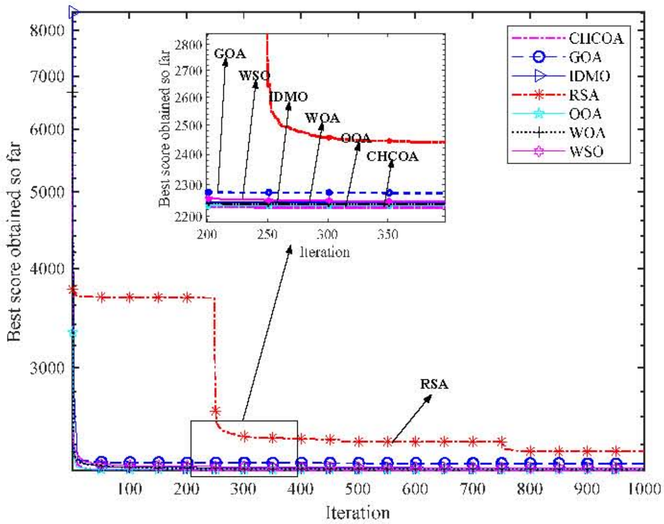
(5) f_5



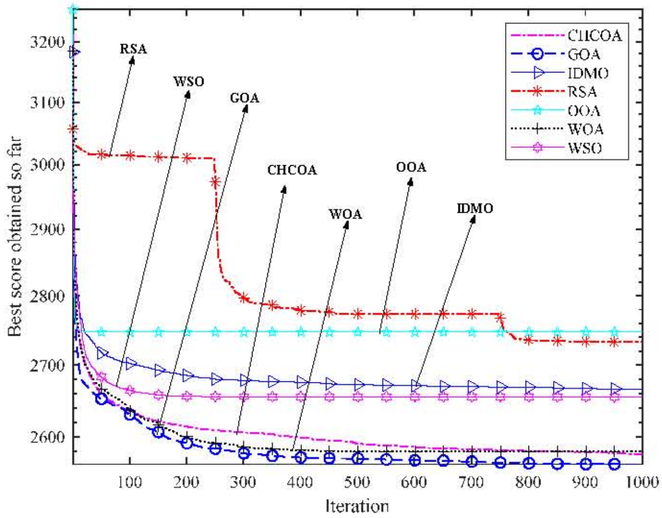
(6) f_6



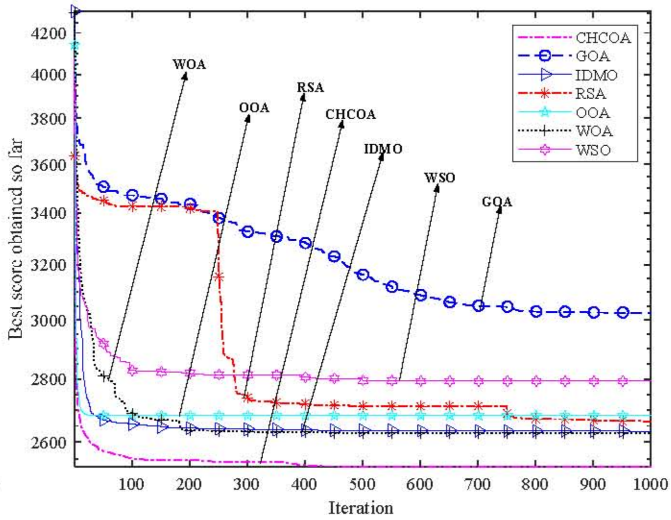
(7) f_7



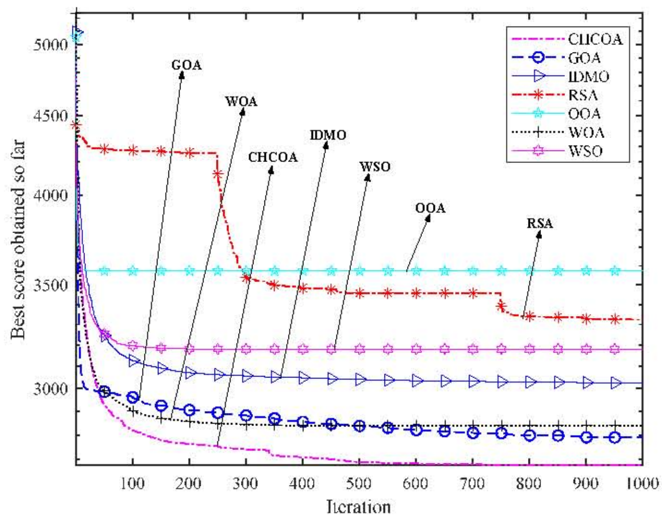
(8) f_8



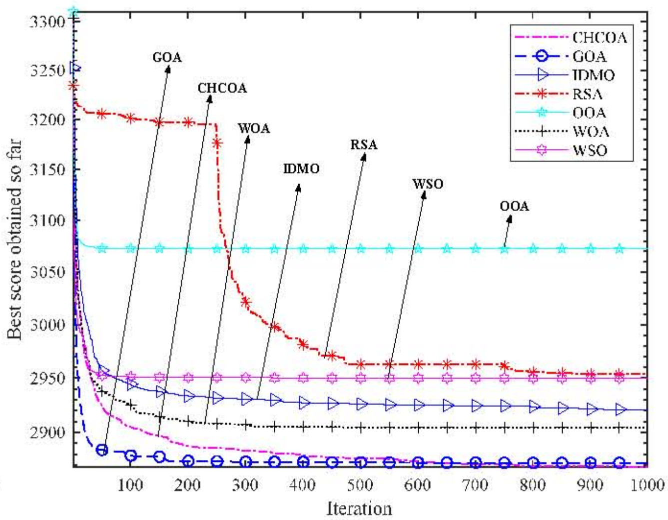
(9) f_9



(10) f_{10}



(11) f_{11}



(12) f_{12}

Fig. 4 The convergence graph of CEC-BC-2022 test function solved by enhanced Chebyshev map COA and other intelligent algorithms.

TABLE II. PERFORMANCE COMPARISON OUTCOMES OF CEC-2022 FUNCTION OPTIMIZATION

Function		CHCOA	GOA	IDMO	RSA	OOA	WOA	WSO
f_1	Best	2.82E+03	3.00E+02	**2.33E+03**	7.37E+03	8.16E+03	3.59E+04	4.64E+03
	Ave	**7.28E+02**	2.93E+03	2.79E+03	8.61E+03	8.51E+03	2.14E+04	7.11E+03
	Std	**8.70E+02**	6.51E+03	1.32E+03	2.81E+03	1.63E+03	1.30E+04	2.72E+03
f_2	Best	**4.01E+02**	4.12E+02	4.52E+02	6.23E+02	3.35E+03	4.73E+02	4.91E+02
	Ave	4.19E+02	**4.17E+02**	5.70E+02	1.02E+03	1.72E+03	4.49E+02	8.48E+02
	Std	4.02E+01	2.39E+01	**1.99E+02**	5.23E+02	8.32E+02	5.57E+01	5.62E+02
f_3	Best	**6.03E+02**	6.27E+02	6.14E+02	6.48E+02	6.46E+02	6.52E+02	6.53E+02
	Ave	**6.10E+02**	6.22E+02	6.35E+02	6.47E+02	6.44E+02	6.38E+02	6.40E+02
	Std	8.89E+00	1.51E+01	1.49E+01	**7.33E+00**	1.01E+01	1.36E+01	1.15E+01
f_4	Best	**8.21E+02**	8.37E+02	8.25E+02	8.52E+02	8.49E+02	8.45E+02	8.55E+02
	Ave	**8.23E+02**	8.44E+02	8.29E+02	8.50E+02	8.48E+02	8.34E+02	8.49E+02
	Std	**6.79E+00**	1.77E+01	7.39E+00	7.22E+00	1.02E+01	1.39E+01	1.27E+01
f_5	Best	1.17E+03	**9.00E+02**	1.13E+03	1.68E+03	1.11E+03	1.32E+03	1.17E+03
	Ave	**1.18E+03**	1.60E+03	1.29E+03	1.50E+03	1.42E+03	1.45E+03	1.39E+03

	Std	1.30E+02	7.98E+02	1.97E+02	**1.13E+02**	1.86E+02	4.09E+02	2.25E+02
	Best	8.65E+03	**2.01E+03**	1.83E+04	5.92E+07	1.23E+04	8.09E+03	5.52E+03
f_6	Ave	6.09E+03	**3.83E+03**	9.52E+03	7.53E+07	3.33E+06	4.32E+03	4.59E+03
	Std	2.01E+03	2.35E+03	7.80E+03	4.32E+07	5.54E+06	**1.97E+03**	2.27E+03
	Best	**2.03E+03**	2.04E+03	2.07E+03	2.09E+03	2.09E+03	2.04E+03	2.09E+03
f_7	Ave	**2.03E+03**	2.11E+03	2.08E+03	2.12E+03	2.09E+03	2.07E+03	2.08E+03
	Std	**9.06E+00**	6.32E+01	3.29E+01	2.53E+01	2.63E+01	2.39E+01	3.37E+01
	Best	**2.22E+03**	2.22E+03	2.23E+03	5.26E+03	2.26E+03	2.23E+03	2.23E+03
f_8	Ave	**2.22E+03**	2.27E+03	2.24E+03	2.35E+03	2.23E+03	2.24E+03	2.24E+03
	Std	**4.39E+00**	5.33E+01	4.89E+01	5.51E+02	9.48E+00	1.21E+01	2.69E+01
	Best	2.68E+03	2.53E+03	2.67E+03	2.67E+03	2.80E+03	**2.63E+03**	2.67E+03
f_9	Ave	**2.55E+03**	2.56E+03	2.67E+03	2.73E+03	2.75E+03	2.58E+03	2.66E+03
	Std	4.51E+01	5.18E+01	**3.44E+01**	4.79E+01	3.49E+01	5.04E+01	4.22E+01
	Best	2.62E+03	2.63E+03	**2.51E+03**	2.71E+03	2.59E+03	2.65E+03	2.76E+03
f_{10}	Ave	**2.53E+03**	3.03E+03	2.63E+03	2.66E+03	2.68E+03	2.63E+03	2.79E+03
	Std	5.31E+01	5.10E+02	2.22E+02	**1.18E+02**	1.61E+02	2.11E+02	4.61E+02
	Best	2.61E+03	**2.60E+03**	2.80E+03	2.91E+03	3.31E+03	2.66E+03	2.94E+03
f_{11}	Ave	**2.68E+03**	2.79E+03	3.02E+03	3.32E+03	3.57E+03	2.84E+03	3.18E+03
	Std	**1.02E+02**	1.96E+02	4.15E+02	4.30E+02	4.76E+02	1.57E+02	3.75E+02
	Best	**2.85E+03**	2.88E+03	2.93E+03	2.89E+03	2.97E+03	2.97E+03	2.92E+03
f_{12}	Ave	**2.87E+03**	2.87E+03	2.92E+03	2.95E+03	3.07E+03	2.90E+03	2.95E+03
	Std	**3.48E+00**	1.49E+01	3.78E+01	1.12E+02	7.59E+01	4.96E+01	4.96E+01

C. *Engineering Optimization Design Problems*

(1) Three-bar truss design problem

Three-bar truss design problem refers to the optimization of the structure under constraints to achieve minimal weight while satisfying specific performance requirements, thereby ensuring high mechanical performance and stability. A three-bar truss consists of three members and is commonly used in engineering structures such as bridges and towers. The model diagram of the three-bar truss design problem is depicted in Fig. 5.

Objective function: $f(X)=\left(2\sqrt{2}X_1+X_2\right)*l$

Constraints: $g_1(X)=\dfrac{\sqrt{2}X_1+X_2}{\sqrt{2}X_1+2X_1X_2}P-\sigma\le 0$

$$g_2(X)=\frac{X_2}{\sqrt{2}X_1+2X_1X_2}P-\sigma\le 0$$

$$g_3(X)=\frac{1}{\sqrt{2}X_2+X_1}P-\sigma\le 0$$

where, $0\le X_1,X_2\le 1$, $l=100$ cm, $P=2$ KN/cm, $\sigma=2$ KN/cm.

Owing to the difference in the output of the enhanced COA reliance on ten chaotic maps to solve the three-bar truss design problem is minimal among each other, a comparison was conducted between the aforementioned enhanced CHCOA, which exhibited the finest overall efficacy in solving twelve test functions in the CEC-BC-2022, and six other intelligent optimization algorithms, namely GOA, IDMO, RSA, OOA, WOA and WSO. The experimental outcomes of optimizing the most excellent solution to the three-bar truss design problem by using these selected algorithms are presented. For the sake of facilitating the observation of experimental outcomes, the upper limit of iterations for each algorithm was fixed at 100 generations. The optimal values, mean values and variances from 30 experimental runs were compiled in Table III, with the outstanding experimental data being highlighted in bold. The convergence graphs of the algorithms for enhancing the efficacy of addressing the truss design problem are depicted in Fig. 6. According to Table IV, the optimal and average values obtained by the enhanced CHCOA for optimizing the three-bar truss design problem are 2.64E+02 and 2.64E+02, respectively, which are the smallest among the optimal and average values obtained by other intelligent optimization algorithms in Table IV. From Fig. 6 and Table IV, it can be illustrated that the overall performance of the enhanced CHCOA for the three-bar truss design problem is the best.

(2) Cantilever beam design problem

The cantilever beam design problem involves achieving economic optimization while satisfying specific conditions and regulations, ensuring good performance and stability under various loads. The primary objectives are to enhance the beam's flexural and shear capacity while minimizing material usage and meeting predetermined design requirements. The model diagram of the cantilever beam design problem is depicted in Fig. 7.

Objective function:

$$f(X)=1.1047X_1^2X_2+0.04811X_3X_4\left(14.0+X_2\right)$$

Constraints: $g_1(X)=\tau(X)-\tau_{\max}\le 0$

$$g_2(X)=\sigma(X)-\sigma_{\max}\le 0$$
$$g_3(X)=\delta(X)-\delta_{\max}\le 0$$
$$g_4(X)=X_1-X_4\le 0$$
$$g_5(X)=P-P_c(X)\le 0$$
$$g_6(X)=0.125-X_1\le 0$$
$$g_7(X)=1.10471X_1^2+0.04811X_3X_4(14.0+X_2)-5.0\le 0$$

where, $\tau(X)=\sqrt{(\tau')^2+2\tau'\tau''\frac{X_2}{2R}+(\tau'')^2}$, $\tau'=\frac{P}{\sqrt{2}X_1X_2}$,

$\tau''=\frac{MR}{J}$, $M=P\left(L+\frac{X_2}{2}\right)$, $R=\sqrt{\frac{X_2^2}{4}+\left(\frac{X_1+X_3}{2}\right)^2}$,

$J=2\left\{\sqrt{2}X_1X_2\left[\frac{X_2^2}{4}+\left(\frac{X_1+X_3}{2}\right)^2\right]\right\}$, $\sigma(X)=\frac{6PL}{X_4X_3^2}$,

$\delta(X)=\frac{6PL^3}{EX_3^2X_4}$, $P_c(X)=\frac{4.013E\sqrt{\frac{X_3^2X_4^6}{36}}}{L^2}\left(1-\frac{X_3}{2L}\sqrt{\frac{E}{4G}}\right)$,

$P=6000$ lb, $L=14$ in., $\delta_{\max}=0.25$ in., $E=30\times 1^6$ psi, $G=12\times 10^6$ psi, $\tau_{\max}=13600$ psi, $\sigma_{\max}=30000$ psi, $0.1\le X_1\le 2$, $0.1\le X_2\le 10$, $0.1\le X_3\le 10$, $0.1\le X_4\le 2$.

Owing to the difference in the dominance of the enhanced COA reliance on ten chaotic maps to solve the cantilever beam design problem is minimal among each other, a comparison was conducted between the aforementioned enhanced CHCOA, which exhibited the finest overall efficacy in solving twelve test functions in the CEC-BC-2022, and six other intelligent optimization algorithms, namely GOA, IDMO, RSA, OOA, WOA and WSO. The experimental outcomes of optimizing the optimal solution of the cantilever beam design problem by using these selected algorithms. For the sake of facilitating the observation of experimental outcomes, the upper limit of iterations for each algorithm was fixed at 100 generations. The optimal values, mean values and variances from 30 experimental runs were compiled in Table V, with the outstanding experimental data being highlighted in bold. The convergence graphs of the algorithms for enhancing the efficacy of addressing the cantilever beam design problem are depicted in Fig. 8. According to Table VI, the average value and variance obtained by the enhanced CHCOA for the cantilever beam design problem is 2.42E+00 and 4.92E-01, which is the smallest among the average values and variances obtained by other intelligent optimization algorithms in Table VI. From Fig. 8 and Table VI, it can be illustrated that the enhanced CHCOA has achieved satisfactory outcomes for the cantilever beam design problem.

(3) Pressure vessel problem

A pressure vessel is a closed equipment capable of withstanding internal pressures, commonly used for storing and transporting gases, liquids and other media. Pressure vessel problem primarily involves the design, fabrication, operation, inspection and maintenance of these vessels. The model diagram of the pressure vessel problem is presented in Fig. 9.

Objective function:

$$f(X)=0.6224X_1X_3X_4+1.7781X_2X_3^2+3.1661X_1^2X_4+19.84X_1^2X_3$$

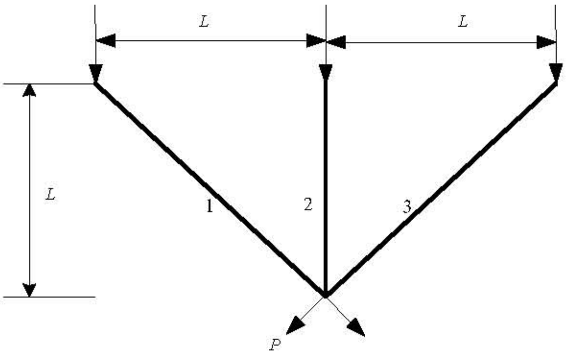


Fig. 5 A model drawing of a three-bar truss.

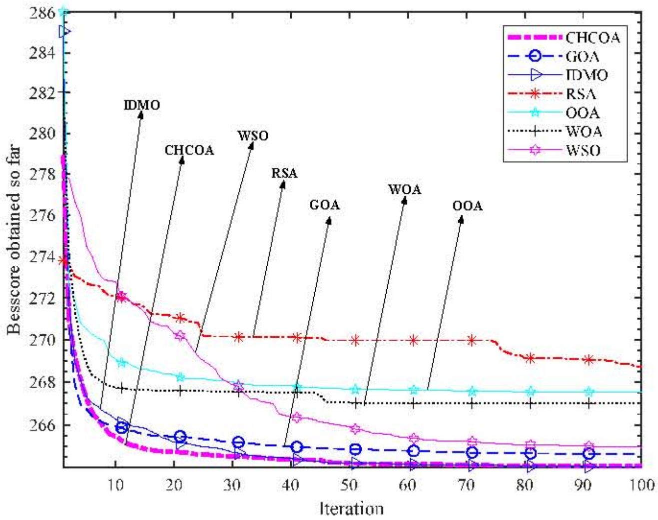


Fig. 6 The convergence diagram of the three-bar truss optimized by the enhanced CHCOA and other intelligent algorithms.

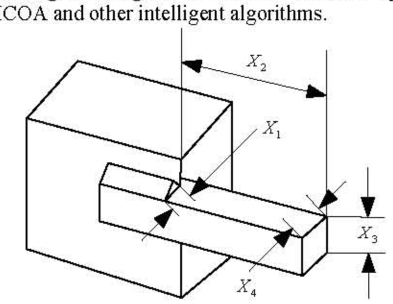


Fig. 7 A model drawing of a cantilever beam problem model drawing of a cantilever beam problem.

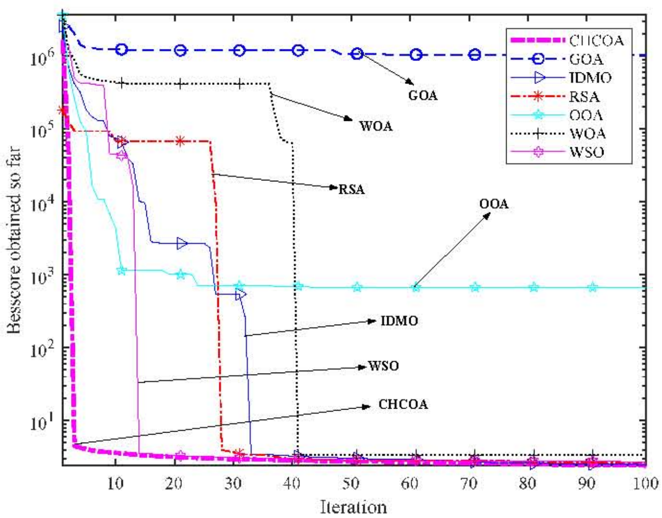


Fig. 8 The convergence diagram of the cantilever beam problem optimized by the enhanced CHCOA and other intelligent algorithms.

Constraints: $g_1(X) = 0.0193X_3 - X_1 \le 0$

$$g_2(X) = 0.00954X_3 - X_2 \le 0$$

$$g_3(X) = 1296000 - \pi X_3^2 X_4 - \frac{4}{3}\pi X_3^3 \le 0$$

$$g_4(X) = X_4 - 240 \le 0$$

where, $0.0625 \le X_1, X_2 \le 6.1875$.

Owing to the difference in the dominance of the enhanced COA reliance on ten chaotic maps for solving the pressure vessel problem is minimal among each other, a comparison was conducted between the aforementioned enhanced CHCOA, which exhibited the finest overall efficacy in solving twelve test functions in the CEC-BC-2022, and six other intelligent optimization algorithms, namely GOA, IDMO, RSA, OOA, WOA and WSO. The experimental results of enhancing the efficacy of addressing the optimal solution of the pressure vessel problem by using these selected algorithms. For the sake of facilitating the observation of experimental outcomes, the upper limit of iterations for each algorithm was fixed at 100 generations. The optimal values, mean values and variances from 30 experimental runs were compiled in Table VII, with the outstanding experimental data being highlighted in bold. The convergence graphs of the algorithms for enhancing the efficacy of addressing the pressure vessel problem are depicted in Fig. 10. According to Table VIII , the average value and variance obtained by the enhanced CHCOA for optimizing the pressure vessel problem are 6.58E+03 and 3.45E+02, respectively, which are the smallest among the average values and variances obtained by other intelligent optimization algorithms in Table VIII. From Fig. 10 and Table VIII, it can be illustrated that the overall performance of the enhanced CHCOA for the pressure vessel design problem is the best.

(4) Tension spring problem

The tension spring problem primarily concerns the design, fabrication, application and upkeep of springs. A tension spring is an elastic element that can undergo tensile deformation under external force, widely employed in various mechanical and electronic devices for force transmission, vibration dampening and compensation. The model diagram of the tension spring design problem is presented in Fig. 11.

Objective function: $f(X) = (X_3 + 2)X_2X_1^2$

Constraints: $g_1(X) = 1 - \frac{X_2^3 X_3}{71785X_1^4} \le 0$

$$g_2(X) = \frac{4X_2^2 - X_1X_2}{12566(X_2X_1^3 - X_1^4)} + \frac{1}{5108X_1^2} \le 0$$

$$g_3(X) = 1 - \frac{140.45X_1}{X_2^2 X_3} \le 0$$

$$g_4(X) = \frac{X_1 + X_2}{1.5} - 1 \le 0$$

where, $0.05 \le X_1 \le 2.00$, $0.25 \le X_2 \le 1.30$, $2.00 \le X_3 \le 15.0$.

Owing to the difference in the dominance of the enhanced COA reliance on ten chaotic maps for solving the tension spring problem is minimal among each other, a comparison was conducted between the aforementioned enhanced CHCOA, which exhibited the finest overall efficacy in solving twelve test functions in the CEC-BC-2022, and six other intelligent optimization algorithms, namely GOA, IDMO, RSA, OOA, WOA and WSO. The experimental results of optimizing the optimal solution of the tension spring problem by using these selected algorithms. For the sake of facilitating the observation of experimental outcomes, the upper limit of iterations for each algorithm was fixed at 100 generations. The optimal values, mean values and variances from 30 experimental runs were compiled in Table IX, with the outstanding experimental data being highlighted in bold. The convergence graphs of the algorithms for enhancing the efficacy of addressing the tension spring problem are depicted in Fig. 12. According to Table X, the optimal value, average value and variance obtained by the enhanced CHCOA for the design of tension springs are 1.20E-02, 1.21E-02 and 1.08E-04, respectively, which are the smallest among the optimal values, average values and variances obtained by other intelligent optimization algorithms in Table X. From Fig. 12 and Table X, it can be illustrated that the overall optimization effect of the CHCOA for the tension spring problem is the best.

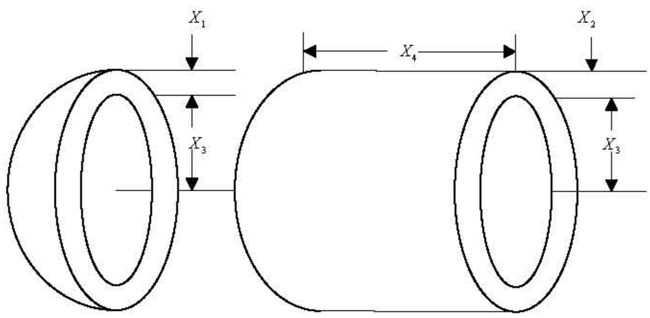


Fig. 9 A model drawing of a pressure vessel problem.

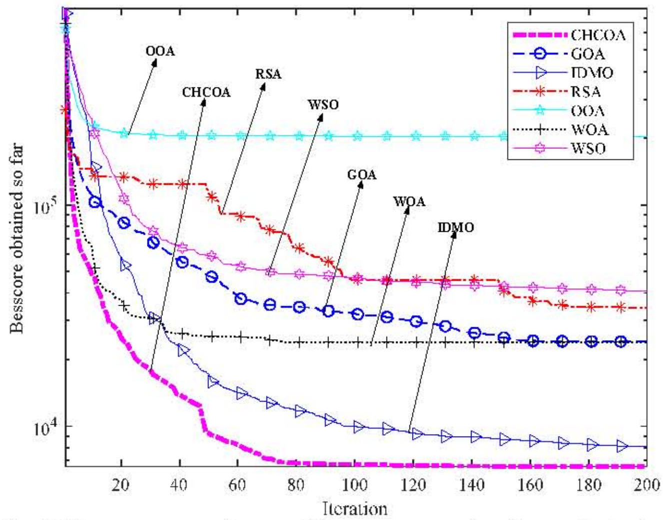


Fig. 10 The convergence diagram of the pressure vessel problem optimized by the enhanced CHCOA and other intelligent algorithms.

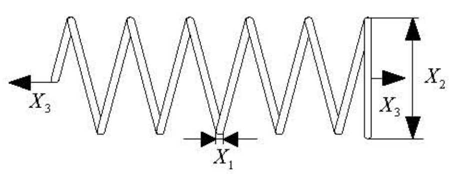


Fig. 11 A model drawing of a tension spring problem.

V. CONCLUSION

The experimental outcomes of four distinct categories of engineering optimization design problems demonstrate the good optimization effects of the Cauchy mutation chaotic COA. Compared with other introduced chaotic maps, the Cauchy mutation chaotic COA with Chebyshev map and Circle map have a good effect on optimal value. Among the thirty running outcomes, the Cauchy mutation chaotic COA enhanced by the Chebyshev map achieves the best average optimization. The variance effect of the Cauchy mutation chaotic COA enhanced by introducing the Chebyshev map, Iterative map, Singer map and Tent map are pretty good. The convergence effect of the Cauchy mutation chaotic COA enhanced by the Chebyshev map is pretty good, and the optimal values and average values of enhanced the efficacy of addressing the three-bar truss design problem and the tension spring problem obtained by the enhanced CHCOA are the best. For the three-bar truss design problem and tension spring problem, the optimal values obtained by the enhanced CHCOA are the best.

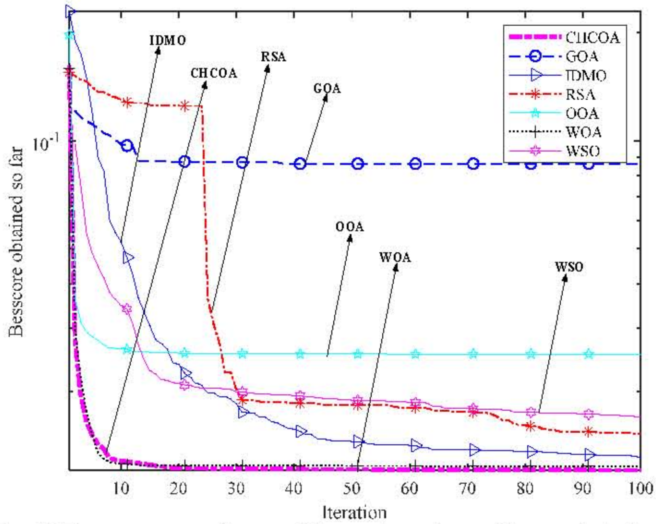


Fig. 12 The convergence diagram of the tension spring problem optimized by the enhanced CHCOA and other intelligent algorithms.

TABLE III. THE BEST SOLUTION OBTAINED FROM THE THREE-BAR TRUSS DESIGN PROBLEM

	CHCOA	GOA	IDMO	RSA	OOA	WOA	WSO
$f(X)$	2.64E+02	2.64E+02	2.64E+02	2.64E+02	2.70E+02	2.66E+02	2.65E+02
X_1	7.80E-01	7.69E-01	7.86E-01	7.88E-01	7.35E-01	7.52E-01	7.65E-01
X_2	4.34E-01	4.72E-01	4.17E-01	4.57E-01	5.95E-01	5.44E-01	4.85E-01

TABLE IV. THE OUTCOMES OBTAINED FROM THE THREE-BAR TRUSS DESIGN PROBLEM

	CHCOA	GOA	IDMO	RSA	OOA	WOA	WSO
Best	**2.64E+02**	2.64E+02	2.64E+02	2.64E+02	2.64E+02	2.64E+02	2.64E+02
Ave	**2.64E+02**	2.65E+02	2.64E+02	2.69E+02	2.68E+02	2.67E+02	2.65E+02
Std	2.24E-01	1.37E+00	**1.26E-01**	3.92E+00	2.53E+00	4.39E+00	1.79E+00

TABLE V. THE BEST SOLUTION OBTAINED FROM THE CANTILEVER BEAM PROBLEM

	CHCOA	GOA	IDMO	RSA	OOA	WOA	WSO
$f(X)$	3.23E+00	6.78E+04	2.40E+00	4.17E+00	5.12E+00	3.74E+00	**2.21E+00**
X_1	3.27E-01	5.69E-01	3.20E-01	2.72E-01	4.69E-01	3.61E-01	2.80E-01
X_2	3.10E+00	2.34E+00	3.19E+00	4.79E+00	3.66E+00	4.34E+00	4.40E+00
X_3	6.82E+00	5.30E+00	7.36E+00	8.39E+00	5.34E+00	6.20E+00	6.70E+00
X_4	4.05E-01	7.48E-01	4.01E-01	3.69E-01	6.60E-01	5.85E-01	4.59E-01

TABLE VI. THE OUTCOMES OBTAINED FROM THE CANTILEVER BEAM PROBLEM

	CHCOA	GOA	IDMO	RSA	OOA	WOA	WSO
Best	1.82E+00	1.95E+00	1.87E+00	1.95E+00	1.99E+00	1.95E+00	**1.78E+00**
Ave	**2.42E+00**	1.04E+06	2.50E+00	2.65E+00	6.71E+02	3.37E+00	2.71E+00
Std	**4.92E-01**	3.43E+06	6.13E-01	5.89E-01	3.65E+03	1.09E+00	6.18E-01

TABLE VII. THE BEST SOLUTION OBTAINED FROM THE PRESSURE VESSEL PROBLEM

	CHCOA	GOA	IDMO	RSA	OOA	WOA	WSO
$f(X)$	6.99E+03	2.04E+05	6.84E+03	7.27E+03	3.04E+05	3.68E+04	7.40E+04
X_1	1.07E+00	8.16E-01	1.06E+00	1.27E+00	7.32E+00	1.14E+00	1.56E+00
X_2	5.14E-01	6.69E+00	7.26E-01	2.06E+00	1.86E+01	1.29E+00	6.81E+00
X_3	5.66E+01	4.37E+01	5.30E+01	4.86E+01	5.65E+01	5.27E+01	5.40E+01
X_4	5.79E+01	1.66E+02	9.19E+01	1.29E+02	5.66E+01	9.01E+01	7.94E+01

TABLE VIII. THE OUTCOMES OBTAINED FROM THE PRESSURE VESSEL PROBLEM

	CHCOA	GOA	IDMO	RSA	OOA	WOA	WSO
Best	5.91E+03	**5.74E+03**	6.36E+03	6.60E+03	1.02E+04	6.23E+03	6.29E+03
Ave	**6.58E+03**	2.41E+04	8.08E+03	3.41E+04	2.04E+05	2.40E+04	4.08E+04
Std	**3.45E+02**	4.72E+04	1.15E+03	4.61E+04	1.48E+05	3.76E+04	8.70E+04

TABLE IX. THE BEST SOLUTION OBTAINED FROM THE TENSION SPRING PROBLEM

	CHCOA	GOA	IDMO	RSA	OOA	WOA	WSO
$f(X)$	1.22E-02	1.45E-02	1.25E-02	1.92E-02	4.80E-02	1.20E-02	2.31E-02
X_1	5.65E-02	1.02E-01	6.09E-02	6.55E-02	5.73E-02	5.79E-02	6.01E-02
X_2	3.13E-01	6.08E-01	4.11E-01	4.84E-01	3.24E-01	3.48E-01	3.94E-01
X_3	7.40E+00	6.96E+00	5.22E+00	5.45E+00	4.15E+00	6.92E+00	4.89E+00

TABLE X. THE OUTCOMES OBTAINED FROM THE TENSION SPRING PROBLEM

	CHCOA	GOA	IDMO	RSA	OOA	WOA	WSO
Best	**1.20E-02**	1.37E-02	1.21E-02	1.25E-02	1.22E-02	1.20E-02	1.20E-02
Ave	**1.21E-02**	8.63E-02	1.31E-02	1.53E-02	2.55E-02	1.24E-02	1.70E-02
Std	**1.08E-04**	8.36E-02	7.83E-04	1.63E-03	1.47E-02	6.66E-04	7.78E-03

REFERENCES

[1] P. Trojovsk, and M. Dehghani, "Pelican Optimization Algorithm: A Novel Nature-Inspired Algorithm for Engineering Applications," *Sensors*, vol. 22, no. 3, pp. 855-855, 2022.

[2] F. S. Gharehchopogh, I. Maleki, and Z. A. Dizaji, "Chaotic Vortex Search Algorithm: Metaheuristic Algorithm for Feature Selection," *Evolutionary Intelligence*, vol. 15, no. 3, pp. 1777-1808, 2022.

[3] I. Naruei, and F. Keynia, "Wild Horse Optimizer: A New Meta-heuristic Algorithm for Solving Engineering Optimization Problems," *Engineering with Computers*, vol. 38, no. 4, pp. 1-32, 2021.

[4] W. G. Zong, J. H. Kim, and G. V. Loganathan, "A New Heuristic Optimization Algorithm: Harmony Search," *Simulation*, vol. 76, no. 2, pp. 60-68, 2001.

[5] V. P. Hugo, V. Nodar, F. H. Ángel, and J. H. Alberto, "A Multi-Phase Method for Euclidean Traveling Salesman Problems," *Axioms*, vol. 11, no. 9, pp.439-439, 2022.

[6] S. Segura, R. Romero, and M. J. Rider, "Efficient Heuristic Algorithm Used for Optimal Capacitor Placement in Distribution Systems," *International journal of electrical power & energy systems*, vol. 32, no. 1, pp. 71-78, 2010.

[7] M. S. Puga, and J. S. Tancrez, "A Heuristic Algorithm for Solving Large Location–inventory Problems with Demand Uncertainty," *European Journal of Operational Research*, vol. 259, no. 2, pp. 413-423, 2017.

[8] L. Abualigah, and A. Diabat, "A Comprehensive Survey of the Grasshopper Optimization Algorithm: Outcomes, Variants, and Applications," *Neural Computing and Applications,* vol. 32, no. 19, pp. 15533-15556, 2020.

[9] S. Saremi, S. Mirjalili, and A. Lewis, "Biogeography-based Optimisation with Chaos," *Neural Computing and Applications,* vol. 25, no. 5, pp. 1077-1097, 2014.

[10] A. R. Jordehi, "Particle Swarm Optimisation for Dynamic Optimisation problems:a review," *Neural Computing and Applications,* vol. 25, pp. 1507-1516, 2014.

[11] F. Glover, M. Laguna, "Tabu Search," *Handbook of Combinatorial Optimization, Springer*, vol. 106, no. 2, pp. 2093-2229, 1998.

[12] J. H. Holland, "Genetic Algorithms," *Scientific American*, vol. 267, no. 1, pp. 66-73, 1992.

[13] K. Zervoudakis, and S. Tsafarakis, "A Mayfly Optimization Algorithm," *Computers & Industrial Engineering*, vol. 145, pp. 1-23, 2020.

[14] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper Optimisation Algorithm: Theory and application," *Advances in Engineering Software*, vol. 105, pp. 30-47, 2017.

[15] J. O. Agushaka, A. E. Ezugwu, and L. Abualigah, "Dwarf Mongoose Optimization Algorithm," *Computer Methods in Applied Mechanics and Engineering*, vol. 391, no. 1, pp. 1-38, 2022.

[16] X. S. Yang, "Firefly Algorithm, Stochastic Test Functions and Design Optimisation," *International Journal of Bio-Inspired Computation*, vol. 2, no. 2, pp. 78-84, 2010.

[17] M. Braik, A. Hammouri, J. Atwan, M. A. Al-Betar, and M. A. Awadallah, "White Shark Optimizer: A Novel Bio-inspired Meta-heuristic Algorithm for Global Optimization Problems," *Knowledge Based Systems*, vol. 243, no. 11, pp. 108475-108475, 2022.

[18] S. Mirjalil, and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software*, vol. 95, pp. 51-67, 2016.

[19] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, "Marine Predators Algorithm: A Nature-inspired Metaheuristic," *Expert Systems With Applications*, vol. 152, no. 15, pp. 113377-113377, 2020.

[20] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime Mould Algorithm: A New Method for Stochastic Optimization," *Future Generation Computer Systems*, vol. 111, pp. 300-323, 2020.

[21] L. Abualigah, M. A. Elaziz, P. Sumari, W. G. Zong, and A. H. Gandomi, "Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer," *Expert Systems with Applications*, vol. 191, no. 1, pp. 116158-116158, 2022.

[22] M. Dehghani, and P. Trojovský, "Osprey Optimization Algorithm: A New Bio-inspired Metaheuristic Algorithm for Solving Engineering Optimization Problems," *Frontiers in Mechanical Engineering*, vol. 8, no. 12, pp.1126450-1126450, 2022.

[23] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: A Gravitational Search Algorithm," *Information Sciences*, vol. 179, no. 13, pp. 2232-2248, 2009.

[24] K. Chen, L. Chen, and G. Hu, "PSO-Incorporated Hybrid Artificial Hummingbird Algorithm with Elite Opposition-Based Learning and Cauchy Mutation: A Case Study of Shape Optimization for CSGC–Ball Curves," *Biomimetics*, vol. 8, no. 4, pp. 377-377, 2023.

[25] Y. Y. Bao, X. Cheng, J. S. Wang, X. R. Zhao, X. Y. Zhang, and Y. Zheng, "Improved Teaching-Learning-Based Optimization Algorithm with Cauchy Mutation and Chaotic Operators," *Applied Intelligence*, vol. 53, no. 18, pp. 21362-21389, 2023.

[26] W. Xing, Q. Liu, and L. Zhang,. "An Adaptive Sand Cat Swarm Algorithm Based on Cauchy Mutation and Optimal Neighborhood Disturbance Strategy," *Biomimetics*, vol. 8, no. 2, pp. 191-229, 2023.

[27] H. Lu, H. Zhang, and L. Ma, "A New Optimization Algorithm Relianced on Chaos," *Journal of Zhejiang University-Science A*, vol. 7, no. 4, pp. 539-542, 2006.

[28] B. Alatas, "Chaotic Bee Colony Algorithms for Global Numerical Optimization," *Expert Systems with Applications*, vol. 37, no. 8, pp. 5682-5687, 2010.

[29] Z. Cheng, L, Zhang, X. Li, and X. Wu, "Chaotic Hybrid Particle Swarm Optimization Algorithm Based on Tent Map," *System Engineering and Electronics Technology*, vol. 1, pp. 103-106, 2007.
[30] N. Wang, L. M. Liu, and L. L. Liu, "Chaotic Genetic Algorithm," *Operations Research Bulletin*, vol. 3, pp. 1-10, 2001.
[31] X. Y. Xu, "Application of Chaotic Simulated Annealing Algorithm in Numerical Function Optimization," *Computer and Digital Engineering*, vol. 38, no. 3, pp. 37-40, 2010.
[32] J. Huang, "Research Status and Application of Chaotic Neural Network," *Journal of Information Engineering*, vol. 16, no. 2, pp. 25-32, 1997.
[33] B. M. Beisiegel, and W. Mantovani, "Habitat. Use, Home Range and Foraging Preferences of the Coati Nasua Nasua in a Pluvial Tropical Atlantic Forest Area," *Journal of Zoology*, vol. 269, no. 1, pp. 77-87, 2006.
[34] A. Naanaa, "Fast Chaotic Optimization Algorithm Based on Spatiotemporal Maps for Global Optimization," *Applied Mathematics and Computation*, vol. 269, no. 15, pp. 402-411, 2015.