scientific reports

OPEN

Sand cat swarm optimization algorithm and its application integrating elite decentralization and crossbar strategy

Yancang Li[1,3], Qian Yu[1] & Zunfeng Du[2,3]

Sand cat swarm optimization algorithm is a meta-heuristic algorithm created to replicate the hunting behavior observed by sand cats. The presented sand cat swarm optimization method (CWXSCSO) addresses the issues of low convergence precision and local optimality in the standard sand cat swarm optimization algorithm. It accomplished this through the utilization of elite decentralization and a crossbar approach. To begin with, a novel dynamic exponential factor is introduced. Furthermore, throughout the developmental phase, the approach of elite decentralization is incorporated to augment the capacity to transcend the confines of the local optimal. Ultimately, the crossover technique is employed to produce novel solutions and augment the algorithm's capacity to emerge from local space. The techniques were evaluated by performing a comparison with 15 benchmark functions. The CWXSCSO algorithm was compared with six advanced upgraded algorithms using CEC2019 and CEC2021. Statistical analysis, convergence analysis, and complexity analysis use statistics for assessing it. The CWXSCSO is employed to verify its efficacy in solving engineering difficulties by handling six traditional engineering optimization problems. The results demonstrate that the upgraded sand cat swarm optimization algorithm exhibits higher global optimization capability and demonstrates proficiency in dealing with real-world optimization applications.

Keywords Sand cat swarm optimization algorithm, Dynamic exponential factor, Elite decentralization strategy, Crossbar strategy, Engineering application

The rapid development of industry and the evolving landscape have given rise to an assortment of engineering applications. Our objective is to enhance the efficiency of these technical challenges within a designated period. Each engineering application presents unique solutions, and it is evident that no singular optimization technique possesses the capability to effectively tackle all optimization challenges effectively. Hence, this proliferation of applications has introduced additional challenges to the field of optimization. Historically, optimization methods have been widely employed across a diverse range of applications, such as path planning[1,2], location problem[3], production shop scheduling[4,5], power generation prediction[6] and a multitude of additional issues. In the present era, as engineering issues become more intricate and challenging to simulate, it is crucial to prioritize accelerating the creation of superior optimization algorithms. Hence, the process of optimizing algorithms is still far.

Scholars have developed a range of optimization algorithms, drawing inspiration from biology, nature, and society. These optimization techniques have been refined and extensively employed to address diverse intricate engineering challenges. Optimization algorithms can be divided: swarm, evolutionary, physical, and human. Optimization algorithms for populations aim to replicate the social behavior observed in populations. Examples of optimization algorithms that replicate the predatory behavior of creatures include the whale optimization algorithm (WOA)[7], the Harris Eagle algorithm (HHO)[8], and the chimpanzee optimization algorithm (COA)[9]. Additionally, the Black tern algorithm (STOA)[10] emulates the migratory and aggressive life patterns observed in black tern groups. Darwinian evolution inspires for evolutionary optimization techniques. One of the algorithms that falls within this category is the backtracking search optimization algorithm (BSA)[11] and the differential evolution algorithm (DE)[12]. Physical optimization techniques are derived based on fundamental principles of physics. The principle of simulated annealing is based on the simulated annealing algorithm (SA)[13]. The proposed

[1]School of Civil Engineering, Hebei University of Engineering, Handan 056038, Hebei, China. [2]School of Civil Engineering, Tianjin University, Tianjin 300354, China. [3]These authors contributed equally: Yancang Li and Zunfeng Du. email: 15879223963@163.com

technique, known as the gravitational search algorithm (GSA)[14], draws inspiration from the law of gravitation. The method known as the black hole algorithm (BHBO)[15] is derived from the inherent characteristics of black holes. The algorithms known as human optimization algorithms are derived from the study of human behavior. One instance of an optimization algorithm is the brainstorming optimization method (MSA)[16], which utilizes of human behavior to address optimization problems. Another algorithm, Human Learning Optimization (HLO)[17], originated from a simplified model of human learning. Optimization challenges possess a notable capacity to effectively address engineering problems.

With a growing variety of proposed optimization methods, numerous researchers have enhanced these algorithms. An enhanced self-adaptive beneficial factor-based SOS (SaISOS) with adaptive beneficial factors was proposed by Nama et al.[18]. The researcher incorporated a three-way mutualism phase into the model, along with the introduction of a random-weighted reflection coefficient and a novel control operator. In a sequence of tests, the enhanced algorithm demonstrates a significant superiority over its competitors. Nama et al.[19] introduced a refined backtracking search technique known as GQR-BSA. The client updates the coordinate structure of BSA by implementing quasi-reflection, quantum Gaussian mutation, adaptive parameter execution, and leaping based on quasi-reflection. This permits the system to transition from the local optimal to the global optimal. In their study, Nama et al.[20] presented an enhanced symbiosis algorithm known as I-SOS. The setting up of a balance between the core of exploration and activity is achieved by employing adjusted return factors, modified parasite stages, and search strategies that rely on random weights. The results of the benchmark function test demonstrate that the implementation of I-SOS translates to a boost in search performance. Nama et al.[21] blended the SMA algorithm with the quasi-reflectomy-based learning mechanism (QRBL), resulting in the facilitation of population diversity early development, improved convergence, and elimination of local optimizations. Luo et al.[22] proposed a multi-objective balance optimizer slime mold algorithm (MOEOSMA), which uses dynamic coefficients, an elite filing mechanism, a crowding distance method and an equalization pool strategy to enhance the algorithm's capability. The test findings illustrate MOEOSMA's intense competition. Yin et al.[23] introduced a multi-objective EOSMA (MOEOSMA). The equilibrium optimizer's concentration update operator is applied. After optimizing the value using the greedy technique, the random difference mutation operator is added. MOEOSMA has a lower solution time and an improved convergence accuracy, according to the equivalent results.

Scholarly improvements to algorithms render them more suitable for complex engineering optimization issues, other than to be applicable to simple examples like trusses. Zhang et al.[24] put forward a search algorithm for bald eagles based on bionic polar coordinates (PBES). To improve the algorithm, the initialization is modified, and parameters and polar coordinates are introduced. Upon conducting tests, it has been determined that the enhanced algorithm exhibits a favorable impact on the approximation of curves. With the goal to ascertain the active earth pressure of retaining walls supporting C-backfill, for instance, Nama et al.[25] developed a novel improved backtracking search optimization algorithm (IBSA) based on adaptive control parameters. The findings demonstrated that it had a positive impact. Chakraborty et al.[26] introduced an enhanced symbiotic search method called NMSOS. It proved that the algorithm improved on all outcomes and could use a multistage threshold method with varying thresholds to segment COVID-19 chest X-ray pictures. The original algorithm's search capability is altered by integrating the development potential of SOS with the searching potential of SQI. This method establishes the shallow strip foundation's seismic bearing capacity under pseudodynamic conditions and prepares it for numerical analysis.

In the current study, a novel swarm intelligence optimization algorithm is chosen. In 2022, Seyyedabbasi et al.[27] presented the Sand Cat swarm optimization technique. The primary function of this program is to imitate the sand cat's hunting habits. Because it's hard to get food in the desert, sand cats choose to spend the day underground and hunt at night. The way the sand cat hunts is also really fascinating. They may detect prey moving underground because they are highly sensitive to sound frequencies and can hear sounds with low frequencies. The SCSO exhibits notable benefits in terms of enhanced mining capacity and reliable performance in addressing real-world challenges. Nevertheless, the weaknesses of SCSO are obvious. During the sophisticated phase of the SCSO algorithm, the individual sand cat has a tendency to become trapped in a local optima, resulting in lack of ability to identify a more favorable position.

Numerous researchers have conducted research with the objective of strengthening the comprehensive the ability of the SCSO algorithm. The authors Wu et al.[28] provided a modified approach for tackling limited engineering optimization problems by introducing an improved sand cat swarm optimization problem. Attempting to boost the mobility of sand cats and improve their worldwide discovery ability, the researchers implemented the modified sand cat swarm optimization algorithm (MSCSO) with a loitering strategy. In an attempt to improve the overall performance of the algorithm and expedite convergence, a shot-based reverse learning method is incorporated. A power transformer defect diagnosis approach was proposed by Lu et al.[29], which utilizes an improved sand Cat swarm optimization algorithm and a unit with a bidirectional gated cycle. The conventional sand Cat swarm algorithm was enhanced through the incorporation of logical chaotic mapping, a water wave dynamic component, adaptive weighting, and a gold sine approach. The superiority of ISCSO in terms of optimization precision and quickness of convergence has been demonstrated. As a result, a fault diagnosis technique utilizing L-Isomap and ISSO-BigRU has been created. The adaptive sand cat swarm optimization algorithm (COSCSO) was proposed by Wang et al.[30]. This approach is based on Nonlinear adaptive parameters、the Cauchy variation and the optimal neighborhood perturbation strategy. The enhanced algorithm has the capability of minimizing the existence of local optima within the population, expediting the rate of convergence, promoting the efficiency of search, and promoting population biodiversity. In their study, Jia et al.[31] introduced a Sand Cat swarm optimization algorithm that incorporates quasi-reverse learning strategies. This algorithm effectively converts the three-dimensional path planning problem into an objective function derived from a mathematical model. Consequently, the algorithm facilitates the identification of the optimal path while considering the security constraint. Empirical evidence demonstrates that the enhanced algorithm can effectively identify the most advantageous route in various obstacle-laden conditions. The reactive power optimization approach for storage-distribution networks with wind and wind energy was suggested by Shang et al.[32]. The approach incorporates the multi-objective MOSCSO algorithm. The control variable in this research is the energy storage facility, and the simulation experiment runs using MOSCSO. In the exploration and development stages, Jiang et al.[33] introduced the Cauchy mutation mechanism and Gaussian mutation mechanism, respectively. They additionally created the ISCSO algorithm and enhanced the engine failure detection technique of SDAE using the improved algorithm. The usefulness of the proposed strategy in enhancing average diagnostic accuracy while minimizing average time has been verified. The previously mentioned investigators primarily concentrate on developing the sand cat swarm optimization algorithm through adaptive weights and local variation. These tactics not only strengthen the algorithm's efficiency in various ways but also have potential applications in engineering.

Given the restrictions of conventional SCSO, along with the enhanced approaches put out by other researchers, this study introduces a novel technique called CWXSCSO, which combines elite decentralization and crossbar Sand Cat swarm optimization. The paper's fundamental framework is laid out as follows: The initial portion presents the fundamental optimization process of SCSO. The subsequent section introduces an original dynamic exponential element and implements elite decentralization during the developmental phase. Simultaneously, the algorithm incorporates the crossbar approach to disrupt the previous optimal solution. In the third portion, an aggregate of 15 benchmark test functions is employed to evaluate and compare different improvement strategies. The optimized performance of each strategy is assessed and compared. In Sect. "Comparison between CWXSCSO and different swarm intelligence algorithms", the efficacy of the CWXSCSO approach is assessed using the identical set of CEC2019 test functions and CEC2021 test functions. This evaluation involves assessing several performance metrics such as the ideal value, median value, standard deviation, convergence curve, and the results of the Wilcoxon rank sum test. Section "Engineering application" of the study employs six classical engineering cases to evaluate the viability of the new algorithm in real-world engineering scenarios. To summarize, when comparing the conventional SCSO algorithm with CWXSCSO, it can be observed that CWXSCSO exhibits a certain degree of effectiveness in extinguishing the algorithm from local optima. Additionally, CWXSCSO strengthens the pace of convergence and solution accuracy of SCSO.

Sand cat swarm optimization algorithm

The optimization algorithms of sand cat swarms draw inspiration from their capacity to identify low-frequency noise. Sand cats inhabit challenging habitats characterized by sandy and stony deserts, such as the Sahara in Central Asia, the Sahara in Africa, and the Arabian Peninsula. They engage in daily relaxation and nocturnal hunting activities. They participate in prey detection by sensing low-frequency sounds, whether it is above or below ground. When prey is subterranean, they promptly identify it and excavate it. Based on the behavioral patterns exhibited by sand cats, the process of foraging can be delineated into two distinct phases: prey detection and prey predation. The SCSO algorithm places emphasis on two distinct phases, namely exploration and development, with a particular focus on maintaining a balance between these phases.

Initialize

The sand cat in a D-dimensional optimization problem is a one-dimensional array that operates as a representation of the solution to the problem. Every variable value $(x_1, x_1, \ldots, x_n)$ represents a node. Additionally, every X must be positioned within the limits of the upper and lower limits.

Initially, an initialization matrix is generated based on the issue size, denoted as $\left(N_{pop} \times N_d\right)$, $\left(pop = 1, \ldots, n\right)$. Furthermore, the solution that corresponds to the given input is generated in every iteration. If the subsequent output value exhibits higher efficiency, the current approach will be substituted. The solution for the next iteration is not stored if a superior solution is not obtained. Each sand cat's fitness value was established via the fitness function.

$$SandCat_i = \{SC_1, SC_2, \ldots, SC_n\};\ 1 < i \leq n$$

$$Act_i^j = \{X_{11}, X_{12}, \ldots, X_{nm}\};\ 1 < i \leq n;\ 1 < j \leq d$$

$$Fitness = f(SandCat) = f(SC_1, SC_2, \ldots, SC_n);\ \forall x_i$$

Hunt for prey

The vector R is derived from Eq. (2). The adaptive parameter R enhances the equilibrium between the transition and development of the two phases.

The parameter $\overrightarrow{r_G}$ defines a general sensitivity that exhibits a linear reduction from 2 to 0. Furthermore, the variable r symbolizes the sensitivity span exhibited by each cat.

The expression iter$_c$ denotes the present proportion of iterations. The notation iter$_{Max}$ denotes the upper limit of iterations. The S_M number is derived from the acoustic attributes of the sand cat, hence its assumed value is 2.

$$r_G = s_M - \left(\frac{S_M * \text{iter}_c}{\text{iter}_{Max}}\right) \tag{1}$$

$$R = 2 * r_G * \text{rand}(0,1) - r_G \tag{2}$$

$$r = r_G * rand(0, 1) \tag{3}$$

Every individual sand cat adjusts its location based on its most ideal position (X_b), present position (X_c), and its sensitivity range (r). Hence, the sand cat possesses the capability to identify alternative ideal prey areas, as determined by Eq. (4).

$$X(t + 1) = r * (X_b(t) - \text{rand}(0, 1) * X_c(t)) \tag{4}$$

Attack prey

The formula (5) denotes the gap X_{rnd} between the sand cat and the prey, representing the simulation of the sand cat's attack on the target. Assuming a circular sensitivity span for the sand cat, a position is produced randomly from the best position (X_b) and the present position (X_c). Subsequently, a random angle is picked using the roulette method, and the assault is executed using formula (6). The utilization of randomly generated angles can effectively mitigate the risk of the algorithm succumbing to local optima.

$$X_{rnd} = |\text{rand}(0, 1) * X_b(t) - X_c(t)| \tag{5}$$

$$X(t + 1) = X_b(t) - r * X_{rnd} * cos(\theta) \tag{6}$$

Exploration and development

The utilization of adaptive values for the $\overrightarrow{r_G}$ and R parameters facilitates the process of exploration and development, enabling the SCSO to smoothly transition between the two stages. The values of the R argument are considered to be well balanced when the contents of the $\overrightarrow{r_G}$ argument are spread out in an equitable manner. In a nutshell, the R value is a stochastic value within the range $[-2\,\overrightarrow{r_G}, 2\,\overrightarrow{r_G}]$, where $\overrightarrow{r_G}$ is reduced from 2 to 0 in each iteration. Consequently, the parameter R is a randomized value within the bounds Report Phrase of $[-4, 4]$.

If the quantity of |R| is less than 1, the sand cats are directed to engage in prey assault. Conversely, if |R| is greater than 1, the cats are assigned the responsibility of identifying a novel potential solution over the entire region.

$$X(t + 1) = \begin{cases} r * (X_b(t) - rand(0, 1) * X_c(t));\ |R| > 1;\ \textit{exploration} \\ X_b(t) - X_{rnd} * \cos\alpha * r;\ |R| \le 1;\ \textit{exploitation} \end{cases} \tag{7}$$

Improved sand cat swarm optimization algorithm

This study presents novel approaches to address the issues of slow convergence and susceptibility to local optima in the SCSO algorithm. Specifically, it introduces a dynamic exponential factor, an elite decentralization technique, and a crossbar strategy as potential enhancements to the SCSO algorithm. The subsequent part contains a detailed introduction to three improvement strategies.

Dynamic exponential factor

The parameter of the weight factor holds significant importance. With a relatively high weight factor, the algorithm exhibits robust global search capabilities, enabling it to enhance population variety and cover a vast area. When the comparison is small, the algorithm has a robust local search capability, enabling it to efficiently explore the ideal solution and expedite convergence. Local optimization is a process in which the sand cat participates in local search, as described by formula (4). According to formula (6), as the sand cat swarm approaches the local solution, it is restricted to approaching the solution that is deemed to be locally optimal and lacks the ability to achieve superior local optimization. A novel dynamic exponential factor is suggested as a solution to this challenge, drawing inspiration from existing work. This factor can compensate for the limited capacity for local exploitation during the initial phase and improve the overall search capability during the latter phase, so preventing the population from prematurely settling in the local optimal. Equation (8) illustrates an equation for the exponential factor.

$$\omega = \left(e^{\left(1 - \left(\frac{t}{T}\right)^2\right)}\right)^{kt} \tag{8}$$

In the given context, k represents an optimization factor that adheres to an exponential distribution. The variable t depicts the present quantity of iterations, while T indicates the ultimate amount of iterations.

Elite decentralization strategy

In the desert, the sand cat's power to update its individual location is limited to relying on the guidance of random individuals within the community, resulting in a weak global search capability. This study shows a novel approach to elite decentralization, aiming toward improving the proximity of individual sand cats to elite individuals and bolstering the local development capabilities of the sand cat population. The ultimate goal is to expedite and improve the sand cat population's ability to identify optimal solutions with greater speed and accuracy.

The calculation of fitness for each individual sand cat is performed, followed by the replication of the individual with the highest level of fitness into n copies, so building the elite matrix. When the probability of |R| is equal to 1, various probabilities h are introduced to update the location. Specifically, when h is equal to or less than 1/3, the population is searched worldwide. This approach addresses the limitation of the original algorithm, which is vulnerable to local optima, and prevents the sand cat population from evolving into "precocious". When the value of h exceeds 0.5, the elite matrix is included, ensuring that each dimension of the sand cat individual is in close alignment to the elite individual. This allows for the rapid identification of the best value.

Please revise formula (6) to update its position to the following formula:

When $h \leq 1/3$:

$$X(t+1) = \omega * X_b(t) - C_1 * \cos(C_2)|r * X_{rnd} * cos(\theta)| + X_E \tag{9}$$

When $h > 1/3$:

$$X(t+1) = \omega * |X_b(t) - X(t)| + X_E \tag{10}$$

among:

$$C_1 = 2 * \pi * rand(0,1) \tag{11}$$

$$C_2 = 2 * \left(1 - \frac{t}{T}\right) * \cos\left(\frac{3*\pi*t}{2}\right)^3 \tag{12}$$

Crossbar strategy

With a boost in the total amount of iterations, the sand cats within the population tend to cluster around the optimal individuals. This may give rise to the phenomenon of population diversity decline and hinder the algorithm's ability to develop the global optimal solution. This paper incorporates the horizontal crossover strategy into the sand cat swarm optimization algorithm to mitigate the occurrence of local optima within the algorithm. The horizontal crossover is utilized to cross-search the population, thus minimizing search blind spots and addressing the global optimization problem. The vertical crossover operation is executed on the optimal solution for the purpose of addressing early convergence of the algorithm, hence enabling the algorithm to transcend the local optima and enhance the population's variety. The Crossbar technique, as described in reference[34], has the potential to improve the worldwide search functionality for addressing intricate optimization issues. Consequently, this can result in better precision in solving the algorithm and accelerated convergence speed.

Transverse crossing

Horizontal crossover refers to the process of conducting crossover operations across all dimensions of a population for the purpose to facilitate reciprocal learning among distinct individuals. Prior to the implementation of the horizontal crossing approach, all participants of the sand cat population are randomly paired together without any repetition. Subsequently, an arithmetic crossover is conducted, with the probability P_h typically assigned an estimate of 1. The offspring are formed through the process of crossing the parent generation, as indicated by formula (13), (14):

$$M^{hc}_{i,d} = q_1 * X(i,d) + \left(1 - q_1\right) * X\left(j,d\right) + c_1 * \left(X(i,d) - X\left(j,d\right)\right) \tag{13}$$

$$M^{hc}_{j,d} = q_2 * X\left(j,d\right) + \left(1 - q_2\right) * X(i,d) + c_2 * \left(X\left(j,d\right) - X(i,d)\right) \tag{14}$$

where: q_1 and q_2 are random numbers [0,1]; Both c_1 and c_2 are random numbers [-1, 1]. $X(i,d)$ and $X\left(j,d\right)$ are the parents of d dimension $X(i)$ and $X\left(j\right)$ respectively. $M^{hc}_{i,d}$ and $M^{hc}_{j,d}$ represent the D-dimensional progeny of $X(i,d)$ and $X\left(j,d\right)$ by horizontal crossing, respectively. The generated offspring compete with their parents to retain the best fitness individual.

Longitudinal crossing

The SCSO algorithm exhibits a tendency to encounter local optima in subsequent iterations, a phenomenon frequently attributed to the occurrence of local optima in specific dimensions during the updating procedure. The vertical crossover is a type of arithmetic crossover that is applied to all individuals between two distinct dimensions. It involves updating only a specific dimension, so facilitating the evasion of a dimension that is imprisoned in a local optima. The vertical intersection of the d_1 and d_2 dimensions of individual i yields the descendant individuals, as calculated via Eq. (15).

$$M^{vc}_{i,d_1} = q * X(i,d_1) + \left(1 - q\right) * X\left(j,d_2\right) \tag{15}$$

where: q is the random number on [0,1]: M^{vc}_{i,d_1} is the child of parent $X(i)$ generated by vertical crossing in d_1 and d_2 dimensions. The offspring individuals produced by longitudinal crossing compete with their parents to retain the individuals with better fitness.

Implementation of CWXSCSO algorithm

The modified sand cat swarm optimization technique suggested in this research follows the basic flow based on the aforementioned enhancement methods.

Step 1 Establish the initial position of the population and ascertain the parameters;

Step 2 The objective is to determine the fitness of a sand cat colony by calculating the current ideal location and target value.

Step 3 The position update formula of the basic SCSO algorithm (4) is utilized to update the current position of the sand cat when the value of |R| is greater than 1.

When the quantity of R is less than 1, the process of choosing the hunting approach is selected based on a random probability h. When h is less than or equal to 1/3, the spatial arrangement of the sand cat is updated using the improved position formula (9). Conversely, when h is greater than or equal to 1/3, the spot of the sand cat is updated using the location formula (10).

Step 4 The horizontal cross operation is performed to cross all dimensions, and the position is updated based on Eqs. (13) and (14).

Step 5 Based on Formula (15), the longitudinal crossover operation is executed, followed by a comparison of fitness standards, and ultimately, the ideal reserve is selected.

Step 6 Assess the extent to which the algorithm satisfies the stop condition. If affirmative, exit the primary loop and display the desired location and value; else, revert back to Step 3.

Step 7 Terminate the program and display the optimal outcome.

Improved pseudocode of sand cat group optimization algorithm

```
input：Initialize the population structure $\vec{x} = \{\overrightarrow{x_1}, \overrightarrow{x_2}, \ldots, \overrightarrow{x_n}\}, t, n$
output：Global optimal solution $\overrightarrow{x_k} = \{\overrightarrow{x_{1k}}, \overrightarrow{x_{2k}}, \ldots, \overrightarrow{x_{nk}}\}$
Initialize the sand cat population
The fitness of N individuals was calculated
Initialize $r$，$r_g$和$R$
While $t < T$
  $For\ each\ x_i, i = 1,2, \ldots, N$
  Choose a random Angle based on the roulette wheel ($0^\circ \le \alpha \le 360^\circ$)
    $if\ (R < 1)$
      $if\ (h \le 1/3)$
        Update the position according to equation (9)
      $else$
        Update the position according to equation (10)
      $end\ if$
    $if\ (R \ge 1)$
      Update the position according to equation (4)
    $end\ if$
   According to equation (13), (14) and (15), the crossbar strategy is carried out
$end\ for$
$t = t + 1$
$end\ while$
```

Algorithm 1. Improved the pseudocode of sand cat swarm optimization algorithm.

CWXSCSO flow chart

Figure 1 describes the flow chart of the improved algorithm in detail, as shown below:

Computational complexity

The numerical representation of the time complexity of an algorithm is commonly denoted as O. The CWXSCSO algorithm primarily comprises the dynamic exponential factor, elite decentralization approach, and crossbar strategy. The dimension of the search space is given as m when the population size is N, and the greatest amount of iterations is T, the time complexity analysis of the algorithm in this paper is as follows: The time complexity of initialization is $O(m)$; The time complexity of calculating the fitness value is the dynamic exponent $O(N + N \times logN)$. The dynamic index is improved on the basis of the original linear weight, and the time complexity is still $O(N \times m)$ The elite decentralization approach has a temporal complexity of $O(N \times m)$. The temporal complexity of the crossbar technique can be expressed as $O(N \times m/2 + m/2) = O(N \times m + m)$, and other calculations are small and negligible. Thus, the CWXSCSO method exhibits an overall computational complexity of $O\left(m + T \times N\left(1 + m + \log N\right)\right)$, aligning with the computational complexity of the conventional sand cat swarm optimization algorithm.

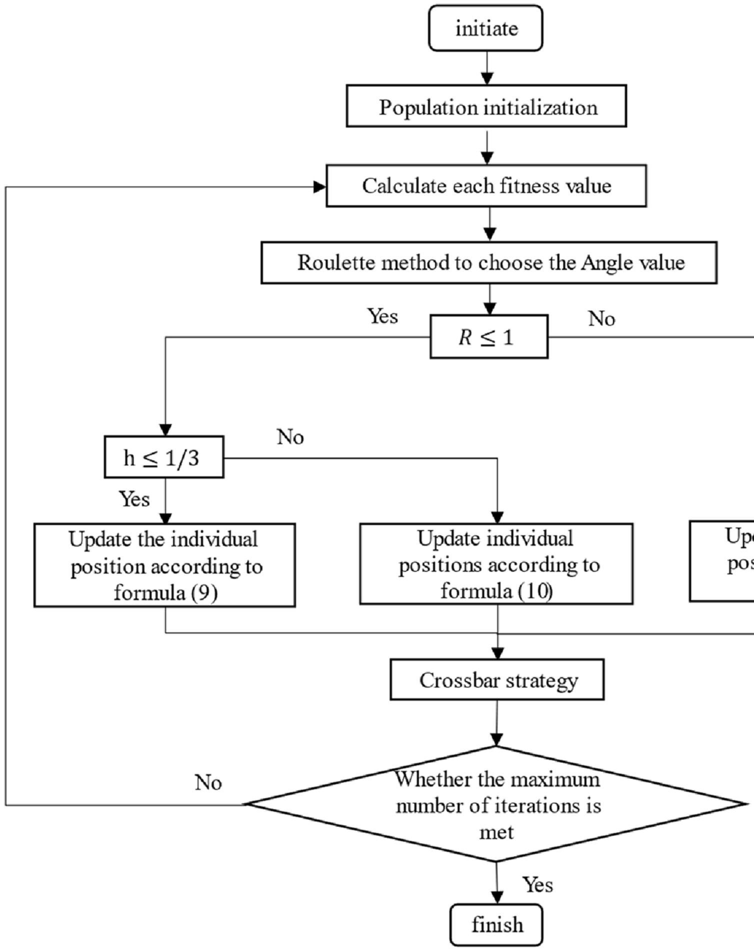

initiate
Population initialization
Calculate each fitness value
Roulette method to choose the Angle value
Yes
$R \leq 1$
No
$h \leq 1/3$
No
Yes
Update the individual position according to formula (9)
Update individual positions according to formula (10)
Update the individual position according to formula (4)
Crossbar strategy
No
Whether the maximum number of iterations is met
Yes
finish

Figure 1. CWXSCSO flow chart.

Simulation experiment and result analysis

Benchmark function

To assess the efficacy and enhancement of CWXSCSO, a set of 15 benchmark functions was chosen, as outlined in Appendix A1. The function F1-F7 is unimodal, meaning it has just one global ideal and no local optimal. This characteristic allows for a more accurate evaluation of the algorithm's convergence time and optimization accuracy. The function F8-F13 exhibits multidimensionality and multimodality. Multiple local extreme values are frequently employed to assess the algorithm's performance in preventing local optima and facilitating search worldwide. The fixed-dimensional multimodal functions F14 and F15 are being referred to. Multimodal functions exhibit numerous local extrema.

Experimental results and analysis of reference function

To enhance the verification of the efficacy of each enhanced method in CWXSCSO, three strategies are examined individually. The concept can be categorized into three distinct strategies: dynamic factor (JSCSO), elite decentralization strategy (ZSCSO), and Crossbar strategy (XSCSO). The length of the query space for F1-F15 is fixed at 30, with a total size of $N = 50$. Additionally, the upper limit for the number of iterations, denoted as T_{max}, is limited to 1000. The optimum value, average value, and standard deviation are obtained by executing each function 30 times.

The statistics shown in Tables 1 and 2 demonstrate that the ideal value, median value, and standard deviation of CWXSCSO exhibit improved performance in comparison with SCSO when considering a dimensionality of 30. CWXSCSO, JSCSO, and ZSCSO algorithms achieve the theoretical best value for every round of the F1-F4 function for simple unimodal functions. In the F5 algorithm, the ideal value and mean value of CWXSCSO exhibit inferior performance compared to XSCSO, while the normal deviation is inferior to that of JSCSO. Nevertheless, CWXSCSO still possesses certain advantages when compared to the basic algorithm. The convergence of the method to the theoretical best value for the complex unimodal function F6 is frequently challenging. However, the inclusion of the crossbar method has resulted in improved optimization accuracy for the algorithm. The modified algorithm exhibits a 15-fold increase in accuracy compared to its original counterpart. The CWXSCSO strategy in F7 has superior performance in terms of ideal value, average value, and

Function	Algorithm	Optimal value	Mean value	Standard deviation
F1	SCSO	1.6505E−248	1.7794E−232	0
	JSCSO	0	0	0
	ZSCSO	0	0	0
	XSCSO	6.3662E−255	4.7234E−239	0
	CWXSCSO	**0**	**0**	**0**
F2	SCSO	4.9174E−128	7.4487E−122	1.6613E−121
	JSCSO	0	0	0
	ZSCSO	0	0	0
	XSCSO	2.9000E−132	3.9944E−127	2.0088E−126
	CWXSCSO	**0**	**0**	**0**
F3	SCSO	4.6337E−219	1.2671E−203	0
	JSCSO	0	0	0
	ZSCSO	0	0	0
	XSCSO	1.0331E−213	1.2787E−196	0
	CWXSCSO	**0**	**0**	**0**
F4	SCSO	2.3733E−110	3.4932E−103	1.5215E−102
	JSCSO	0	0	0
	ZSCSO	0	0	0
	XSCSO	3.3619E−109	4.5124E−101	2.4371E−100
	CWXSCSO	**0**	**0**	**0**
F5	SCSO	4.6159E+01	4.7947E+01	8.7684E−01
	JSCSO	4.8102E+01	4.8839E+01	**2.1074E−01**
	ZSCSO	4.8070E+01	4.8755E+01	2.4136E−01
	XSCSO	**4.4000E+01**	**4.4501E+01**	2.5036E−01
	CWXSCSO	4.5664E+01	4.6590E+01	6.9594E−01
F6	SCSO	1.9686E+00	3.5213E+00	7.0735E−01
	JSCSO	1.0616E+01	1.1043E+01	2.4024E−01
	ZSCSO	9.2714E+00	1.0675E+01	4.8528E−01
	XSCSO	9.9686E−06	1.6590E−05	4.0816E−06
	CWXSCSO	**2.7082E−15**	**1.7763E−13**	**3.4775E−13**
F7	SCSO	9.6579E−07	4.9265E−05	5.5692E−05
	JSCSO	6.5727E−07	1.8145E−05	1.4117E−05
	ZSCSO	1.0667E−07	1.7231E−05	1.7336E−05
	XSCSO	1.1360E−06	1.4352E−05	1.4999E−05
	CWXSCSO	**2.2485E−07**	**4.8359E−06**	**5.1410E−06**

Table 1. The optimization outcomes of the single-peak test function (F1-F6) were compared using various improvement methodologies. Significant values are in bold.

variance compared to SCSO and other methodologies. CWXSCSO exhibits the maximum optimization accuracy and the lowest standard deviation for multi-modal functions F8, F12, and F13. In conjunction with CWXSCSO, XSCSO exhibits superior optimization efficacy, hence indicating the advantageous nature of employing the crossbar technique to facilitate the algorithm's departure from local optima. All strategies in the F9-F11 function attain the theoretical ideal value, suggesting that the incorporated strategies exhibit favorable stability. Table 2 demonstrates that the optimization accuracy of CWXSCSO is preferable to that of SCSO and other processes for the multi-modal function F14-F15 with fixed dimensions. In simple terms, the revised algorithm incorporates dynamic factors and elite decentralization strategies to enhance its development performance. Additionally, the crossbar technique facilitates the algorithm's ability to transcend local optima.

The iteration rules for functions exhibit a high degree of similarity, as depicted in Fig. 2. The iterative curve of CWXSCSO on the F1-F7 function exhibits a nearly linear pattern, suggesting that the enhanced approach outperforms the original technique with regard to both velocities of convergence and optimization precision. The F8 curve clearly demonstrates that the CWXSCSO algorithm achieved the optimal value after 110 iterations, but the previous algorithm did not reach the optimal value within 1000 iterations. Moreover, the image has multiple inflection points, indicating that the improved algorithm not only has a high fitness value, but also has a good ability to jump out of the local optimal. The curve between F11 and F13 has a linear pattern, suggesting that while the ideal value can be identified, the enhanced method demonstrates superior speed. The convergence speed of F9, F10, F14, and F15 exhibits rapid convergence towards the initial optimal value. Additionally, there are instances where the enhanced algorithm demonstrates the ability to surpass local optima and achieve speedy convergence.

Function	Algorithm	Optimal value	Mean value	Standard deviation
F8	SCSO	−1.3399E+04	−1.1025E+04	1.0451E+03
	JSCSO	−5.0405E+03	−3.7293E+03	4.9570E+02
	ZSCSO	−1.1647E+04	−9.5069E+03	1.2672E+03
	XSCSO	−2.0930E+04	2.0477E+04	2.3510E+02
	CWXSCSO	**−2.0949E+04**	**−2.0949E+04**	**9.3256E−09**
F9	SCSO	0	0	0
	JSCSO	0	0	0
	ZSCSO	0	0	0
	XSCSO	0	0	0
	CWXSCSO	0	0	0
F10	SCSO	4.4409E−16	4.4409E−16	
	JSCSO	4.4409E−16	4.4409E−16	
	ZSCSO	4.4409E−16	4.4409E−16	
	XSCSO	4.4409E−16	4.4409E−16	
	CWXSCSO	4.4409E−16	4.4409E−16	
F11	SCSO	0	0	0
	JSCSO	0	0	0
	ZSCSO	0	0	0
	XSCSO	0	0	0
	CWXSCSO	0	0	0
F12	SCSO	4.9159E−02	1.0704E−01	4.2033E−02
	JSCSO	8.6098E−01	9.8243E−01	7.4996E−02
	ZSCSO	7.4167E−01	9.8563E−01	1.3681E−01
	XSCSO	3.2517E−07	5.8252E−07	1.5196E−07
	CWXSCSO	**9.7303E−17**	**2.5354E−15**	**5.5692E−15**
F13	SCSO	2.9328E+00	4.4606E+00	3.1602E−01
	JSCSO	4.7350E+00	4.8159E+00	3.3593E−02
	ZSCSO	4.1989E+00	4.6829E+00	1.7204E−01
	XSCSO	1.0363E−05	1.8981E−05	4.7085E−06
	CWXSCSO	**1.5743E−20**	**1.2104E−18**	**1.8608E−18**
F14	SCSO	9.9800E−01	1.3948E+00	8.8732E−01
	JSCSO	2.9821E+00	4.5383E+00	3.4798E+00
	ZSCSO	2.9821E+00	8.0322E+00	4.6754E+00
	XSCSO	9.9800E−01	9.9800E−01	1.5260E−14
	CWXSCSO	**9.9800E−01**	**9.9800E−01**	**1.5701E−16**
F15	SCSO	3.0749E−04	4.0386E−04	2.7901E−04
	JSCSO	4.5646E−04	7.4476E−04	1.4641E−04
	ZSCSO	3.1404E−04	9.0459E−04	1.4293E−03
	XSCSO	3.0749E−04	3.3801E−04	1.6718E−04
	CWXSCSO	**3.0749E−04**	**3.3062E−04**	**4.2702E−05**

Table 2. The optimization outcomes of multi-peak and fixed-dimension test functions (F8-F15) were compared using various improvement methodologies. Significant values are in bold.

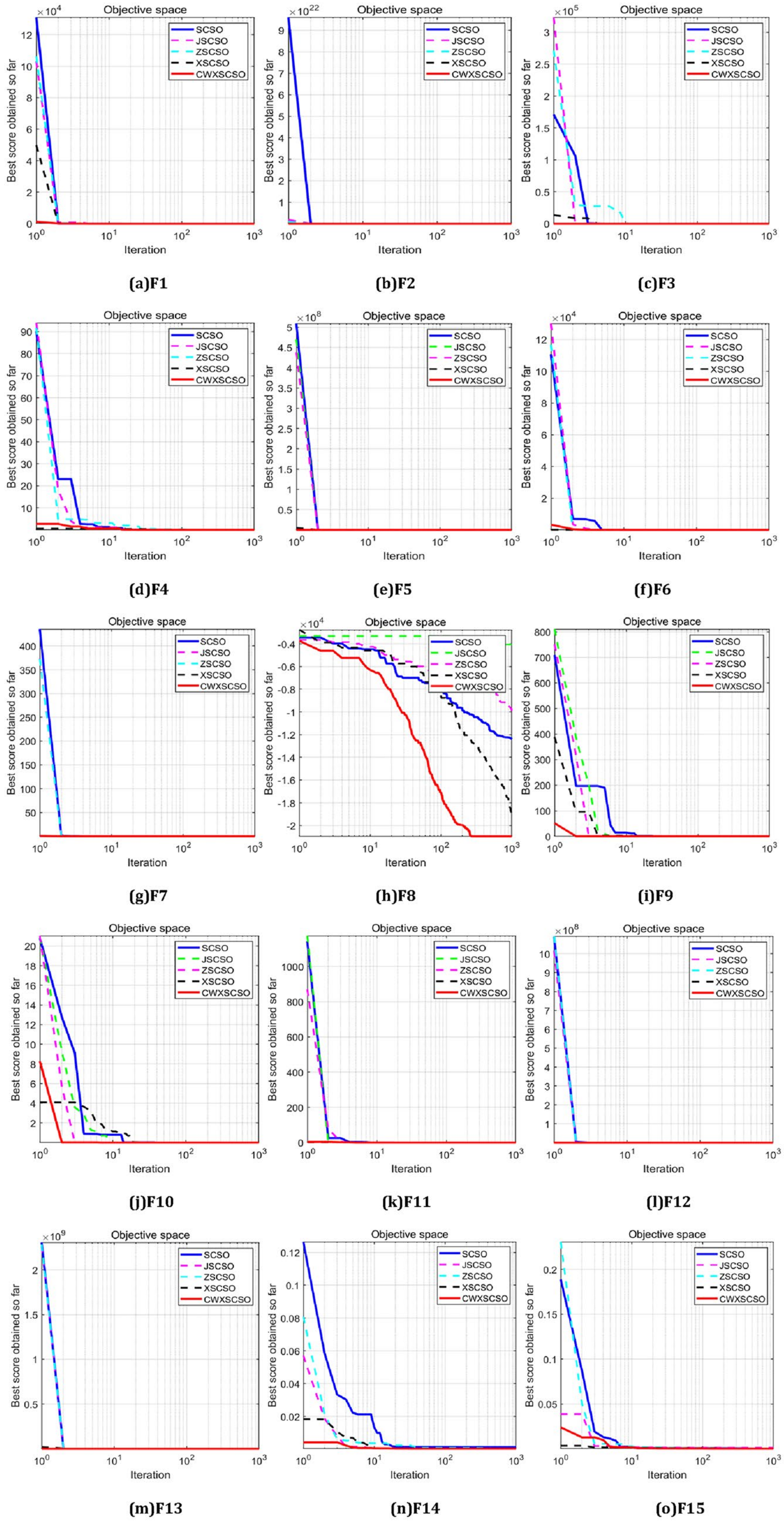
The numerical simulation results, as shown in Tables 1 and 2, along with the convergence diagram depicted in Fig. 2, demonstrate that the CWXSCSO algorithm exhibits superior optimization accuracy, faster convergence acceleration, and better robustness compared to the alternative approach. These findings provide empirical evidence supporting the effectiveness of the algorithm under consideration.

Comparison between CWXSCSO and different swarm intelligence algorithms

Test on CEC2019 benchmark function

The effectiveness analysis of the improved algorithm

This section presents a selection of six new upgraded algorithms for the purpose of comparing the optimization performance of CWXSCSO with other optimization algorithms. They are Salp swarm algorithm based on craziness and adaptive (CASSA)[35], Subtraction-Average-Based Optimizer (GSABO)[36], Grey Wolf Optimization Algorithm Based on Elite Learning for Nonlinear Parameters (IGWO)[37], Whale optimization algorithm based on chaotic search strategy (CWOA)[38], Whale Optimization Algorithm Based on Elite Opposition-based and



(a)F1

(b)F2

(c)F3

(d)F4

(e)F5

(f)F6

(g)F7

(h)F8

(i)F9

(j)F10

(k)F11

(l)F12

(m)F13

(n)F14

(o)F15

Figure 2. Convergence curve for comparison between strategies.

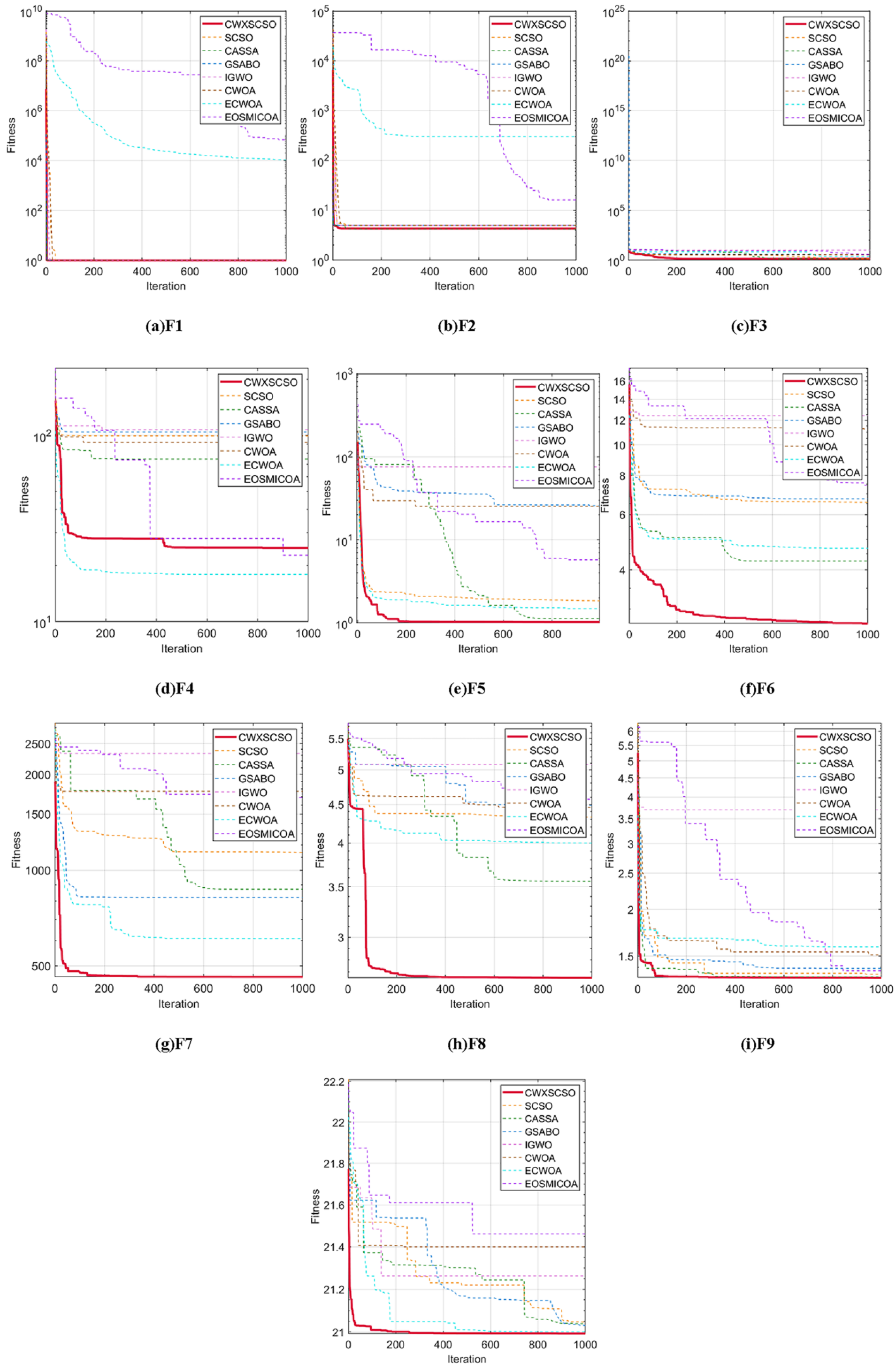
Crisscross Optimization (ECWOA)[34] and Multi-Strategy Chimp Optimization Algorithm and Its Application of Engineering Problem (EOSMICOA)[39]. The solution to the benchmark function in Appendix A2 has been obtained. The experiment had a population size of $N = 50$, with 1000 iterations. Each experiment was completed independently 30 times.

The analysis of Table 3 reveals that the enhanced CWXSCSO algorithm exhibits superior optimization accuracy compared to the other six intelligent optimization algorithms. Furthermore, the improved CWXSCSO algorithm demonstrates greater efficacy on the chosen test functions. The optimization accuracy of CWXSCSO meets the theoretical optimal value of 1 while solving the function F1-F2. Additionally, the average and standard deviations of CWXSCSO are preferable to those of the comparison algorithm. The results produced from the F3, F5, F6, F8, and F10 functions exhibit superior performance compared to the comparison algorithm. The standard deviation of CWXSCSO in function F4 is inferior to that of ECWOA, with just a slight improvement. However, in the F7 function, the ideal value and mean exhibit inferior performance compared to ECWOA, but the standard deviation demonstrates greater accuracy in comparison to ECWOA. In the F9 function, the CWXSCSO algorithm exhibits higher precision comparable to other algorithms. However, it is worth noting that the median value and standard deviation of CWXSCSO are comparatively weaker to those of other algorithms. Hence, it can be ascertained that while certain values of the enhanced algorithm presented in this study may not align with the theoretical ideal value, it generally exhibits superiority over other algorithms and possesses advantages in the realm of function optimization problems. This observation further underscores the efficacy of the proposed method.

The convergence properties of CWXSCSO can be easily observed in Fig. 3, in comparison to other methods. The figure clearly demonstrates that in functions F1 and F2, the curve exhibits rapid convergence and can swiftly approach the global optimal value. In function F3, the population rapidly attains an ideal value, which exhibits superior accuracy compared to alternative algorithms. Despite the relatively low solution accuracy in the F4 and F5 images, the function nevertheless offers several advantages beyond the original approach and other techniques. The optimization of F6-F10 exhibits several nuanced inflection points, suggesting that the algorithm demonstrates an exceptional capacity to overcome local optima and achieve stronger convergence

		CASSA	GSABO	IGWO	CWOA	ECWOA	EOSMICOA	SCSO	CWXSCSO
F1	Best	1	1	1	1	3474.3991	6260.3105	1	**1**
	Mean	1	1	1	1	10,000.6681	10,000.6681	1	**1**
	Std	0	0	0	0	8518.0766	477,985.0910	0	**0**
F2	Best	1.4091	1.00E+20	8.5056	4.0767	1.4099	1.4933	2.3151	**1.4091**
	Mean	2.9881	1.00E+20	9.2246	5.1725	4.5412	2.1452	2.7594	**1.4092**
	Std	2.2330	1.7639	1.0168	1.5497	4.4283	0.9219	0.6284	**1.53E−05**
F3	Best	1.1230	11.4670	75.4821	25.3538	1.4307	3.2677	1.5556	**1.0207**
	Mean	1.1526	18.5796	96.4431	31.0024	1.4526	4.4711	1.6968	**1.0672**
	Std	0.2248	0.0815	7.6634	20.8279	0.1307	12.1507	1.3174	**0.0715**
F4	Best	2.5058	10.2528	10.2574	9.1796	2.3958	6.4279	6.2361	**2.1589**
	Mean	3.8835	10.3381	10.8924	10.0072	3.9861	7.0716	7.2697	**3.6835**
	Std	1.9484	0.1206	0.8980	1.1704	2.2490	0.9103	1.4617	2.1561
F5	Best	1.2141	1.1329	77.7605	23.5512	1.2593	3.2219	1.4557	**1.1203**
	Mean	1.3501	1.3484	94.1279	28.1548	1.4679	3.9688	1.6677	**1.2474**
	Std	0.2108	0.2175	14.1900	4.4841	0.2427	1.0345	0.2204	**0.1262**
F6	Best	3.1971	3.3326	5.0071	4.1673	3.4876	3.8973	3.2628	**3.0976**
	Mean	3.9949	4.0924	5.2635	4.6559	3.9699	4.5614	4.0700	**3.7678**
	Std	0.5302	0.4634	0.3457	0.3097	0.3837	0.4100	0.7014	**0.2903**
F7	Best	1.1795	3.3630	3.8032	1.4946	**1.1436**	1.2602	1.2684	1.2625
	Mean	1.3022	3.4388	3.9093	1.5758	**1.2892**	1.2708	1.3057	1.3136
	Std	0.1735	0.1071	0.1500	0.1149	0.2058	**0.0150**	0.0528	0.0723
F8	Best	21	21.3679	21.5129	21.3529	21.0038	21.3571	21.0295	**21**
	Mean	21.0483	21.3932	21.5931	21.4019	21.0045	21.4181	21.0904	**21.0002**
	Std	0.0683	0.0358	0.1134	0.0694	9.58E−04	0.0862	0.0861	**2.90E−04**
F9	Best	1.1473	1.2329	2.9577	1.5190	1.1462	1.1723	1.2199	**1.1180**
	Mean	**1.2021**	1.3221	3.4002	1.5839	1.3918	1.2529	1.3176	1.3344
	Std	0.0955	0.1036	0.3092	**0.0593**	0.2111	0.0625	0.0751	0.1853
F10	Best	20.9196	20.9814	21.3684	21.3097	21.0005	21.3421	21.0485	**20.9039**
	Mean	21.0430	21.0153	21.4347	21.3790	21.0073	21.4387	21.0491	**20.9908**
	Std	0.1121	0.0556	0.0501	0.0672	0.0091	0.0909	0.0982	**0.0085**

Table 3. Compare the results with his intelligent algorithm. Significant values are in bold.



(a)F1 (b)F2 (c)F3

(d)F4 (e)F5 (f)F6

(g)F7 (h)F8 (i)F9

(j)F10

Figure 3. The fitness curves of each algorithm are compared.

accuracy as opposed to other techniques. In a nutshell judging on the CEC2019 test, CWXSCSO outperforms other algorithms.

Wilcoxon rank sum test

The Wilcoxon rank sum test is a non-parametric statistical test that can be utilized irrespective of the distribution of the subject under investigation and the availability of information regarding the distribution. Hence, it is frequently employed to assess the data distribution of two sets of autonomous samples that deviate from a normal distribution. This approach uses the rank of the sample as a substitute for the sample value in order to facilitate data comparison, thereby mitigating the impact of a single value within the sample on the entire sample. Hence, this approach can provide a more scientific representation of the algorithm's optimization performance contrasted to the median value and standard deviation. This section presents a comparison and analysis of the findings obtained from the CEC2019 test. The optimization outcomes of CASSA, GSABO, IGWO, CWOA, ECWOA, and EOSMICOA were compared using the Wilcoxon rank sum test, based on the CWXSCSO method. The hypothesis test employs a significance level of $p = 5\%$ as the criterion for judgment. At a significance level of 5%, it can be concluded that there is a substantial difference between the two groups of samples.

In Table 4, the presence of *NaN* indicates that there lacks a statistically noteworthy disparity between the two groups of data. Additionally, the symbols "+", "=", and " − " are used to indicate that the outcomes of the CWXSCSO algorithm outperforms, is equivalent to, or falls short of the comparison algorithm, appropriately. The findings indicate that CWXSCSO exhibits a notable advantage over both the novel method and existing enhanced optimization techniques.

Test on CEC2021 benchmark function

Validity analysis on CEC2021 function

With the aim to comprehensively evaluate the optimization capabilities of the upgraded method, a set of 10 CEC2021 test functions with distinct optimization features were chosen (as illustrated in Appendix A3). The CWXSCSO is computed and thereafter compared to the previous six enhancements. The parameters are uniformly set to the total size $N = 50$, the upper limit number of iterations $T_{max} = 1000$, and the dimension $d = 10$. Each function was simulated 30 times to obtain the optimum value, the average value, and standard deviation of the result. A plot is generated to depict the convergence curves of ten functions.

Based on the analysis of the data presented in Table 5, it turns out apparent CWXSCSO has higher performance relative to other algorithms in terms of both the ideal value and average value over the ten functions. The C3 function exhibits superior minimum and average values compared to GSABO and IGWO, while its standard deviation surpasses that of IGWO and EOSMICOA. This observation indicates that the enhanced algorithm lacks robust flexibility to the C3 function. The lowest and average values of functions C7 and C9 are optimal. However, the standard deviation is suboptimal. The algorithm exhibits a wide variety of fluctuations, although it possesses the capability to get the ideal theoretical value. The CWXSCSO algorithm outperforms other algorithms in functions C1, C2, C4-C6, C8, and C10, exhibiting superior values with a narrow range of fluctuations and excellent precision.

Figure 4 clearly demonstrates that all images of functions C1-C10 exhibit significant slopes, suggesting that the enhanced technique achieves faster convergence compared to existing optimization strategies. Photos C1, C4-C6, and C9 exhibit several twists in the CWXSCSO lines, suggesting that the enhanced CWXSCSO possesses the capability to transcend local optima. Contrary to CWXSCSO, other comparison algorithms like EOSMICOA have the capacity to exit the local optimal at numerous iterations. However, their rates of convergence rapidity and precision are not as great. CWXSCSO demonstrates rapid convergence to the global optimal in photos C3, C7, C8, and C10. In images, C3 and C8, the optimal fitness value of CWXSCSO convergence are comparable to that of the original method and CASSA algorithm, but it exhibits the fastest convergence speed.

To summarize, the CEC2021 function test demonstrates that the CWXSCSO algorithm outperforms competing algorithms, hence confirming its effectiveness.

Wilcoxon rank sum test of CEC2021 test function

This section presents a comparison and analysis of the findings obtained from the CEC2021 test, with the aim of enhancing the evaluation of the algorithm's optimization performance. The optimization outcomes of CASSA, GSABO, IGWO, CWOA, ECWOA, and EOSMICOA were compared using the Wilcoxon rank sum comparison test, based on the CWXSCSO method. According to the analysis of the results in Table 6, it indicate that CWXSCSO has no significant disparity with the comparison algorithm across several functions. However, none

Function	CWXSCSO-CASSA	CWXSCSO-GSABO	CWXSCSO-IGWO	CWXSCSO-CWOA	CWXSCSO-ECWOA	CWXSCSO-EOSMICOA
F1	1.2118e − 12	1.2118e − 12	1.2118e − 12	*NaN*	1.2118e − 12	1.2118e − 12
F2	1.2118e − 12	1.2118e − 12	1.2118e − 12	*NaN*	1.2118e − 12	1.2118e − 12
F3	2.9727e − 2	3.0199e − 11	3.0199e − 11	3.0199e − 11	2.0095e − 1	2.3715e − 10
F4	3.0199e − 11	3.0199e − 11	3.0199e − 11	3.0199e − 11	3.0199e − 11	3.0199e − 11
F5	9.7555e − 10	3.0199e − 11	5.8737e − 4	3.5638e − 4	9.7555e − 10	3.0199e − 11
F6	2.2798e − 11	2.2798e − 11	2.2798e − 11	2.2798e − 11	2.443e − 3	2.2798e − 11
F7	3.0199e − 11	3.0199e − 11	3.0199e − 11	3.0199e − 11	3.0199e − 11	3.0199e − 11
F8	3.0199e − 11	3.0199e − 11	3.0199e − 11	3.0199e − 11	3.0199e − 11	3.0199e − 11
F9	5.4991e − 3	4.5356e − 09	2.5464e − 11	2.5464e − 11	8.5026e − 09	2.5464e − 11
F10	2.7071e − 1	2.9972e − 11	3.3384e − 11	1.5292e − 05	3.0339e − 3	3.0199e − 11
+ / = / −	9/0/1	10/0/0	10/0/0	8/2/0	9/0/1	10/0/0

Table 4. Results of Wilcoxon rank sum test for CEC2019 functions.

		CASSA	GSABO	IGWO	CWOA	ECWOA	EOSMICOA	SCSO	CWXSCSO
C1	Best	1.2770E + 02	1.2993E + 09	7.0927E + 09	1.1002E + 09	1.1900E + 03	1.0502E + 08	2.7518E + 03	**1.2365E + 02**
	Mean	4.4886E + 03	2.5211E + 09	1.0700E + 10	3.8009E + 09	9.4817E + 03	2.6645E + 08	3.6662E + 07	**1.4211E + 03**
	Std	4.0543E + 03	1.1013E + 09	2.2716E + 09	1.8719E + 09	7.5948E + 03	2.0107E + 08	1.1572E + 08	**2.0781E + 03**
C2	Best	1.7667E + 03	2.3346E + 03	3.2079E + 03	2.3391E + 03	1.6261E + 03	2.2988E + 03	1.4653E + 03	**1.4790E + 03**
	Mean	1.9982E + 03	2.4861E + 03	3.3776E + 03	2.5867E + 03	1.7865E + 03	2.4907E + 03	1.9857E + 03	**1.7264E + 03**
	Std	1.6752E + 02	1.5764E + 02	1.5839E + 02	2.2368E + 02	2.1225E + 02	1.4459E + 02	3.7318E + 02	**1.3608E + 02**
C3	Best	7.7204E + 02	7.9987E + 02	7.9485E + 02	**7.2364E + 02**	7.4722E + 02	7.3684E + 02	7.2538E + 02	7.7204E + 02
	Mean	8.1813E + 02	8.2776E + 02	8.0825E + 02	**7.3987E + 02**	7.5915E + 02	7.6649E + 02	7.4936E + 02	8.1813E + 02
	Std	1.8206E + 01	1.4039E + 01	**9.3858E + 00**	1.7625E + 01	1.2128E + 01	2.1164E + 01	2.0668E + 01	1.8206E + 01
C4	Best	2.1189E + 03	2.7864E + 03	2.9433E + 04	2.1943E + 03	2.5204E + 03	2.1458E + 03	1.9325E + 03	**1.9023E + 03**
	Mean	4.3398E + 03	6.0417E + 03	7.4329E + 05	6.1336E + 05	1.1093E + 04	1.3695E + 04	7.5499E + 03	**2.6462E + 03**
	Std	2.4632E + 03	2.5662E + 03	6.9340E + 05	7.1498E + 05	9.6835E + 03	6.9745E + 03	5.7211E + 03	**1.6531E + 03**
C5	Best	6.4462E + 03	3.4701E + 05	7.5719E + 04	2.0987E + 04	6.2107E + 03	2.9823E + 03	2.3850E + 03	**1.7081E + 03**
	Mean	1.8034E + 04	6.8947E + 05	1.6888E + 05	5.7084E + 04	5.7894E + 04	4.4089E + 03	7.4505E + 03	**1.7026E + 03**
	Std	1.1607E + 04	2.0154E + 05	1.0430E + 05	2.7900E + 04	4.2677E + 04	8.3703E + 02	4.0438E + 03	**2.5760E + 00**
C6	Best	1.6069E + 03	1.7253E + 03	2.0057E + 03	1.9540E + 03	1.6012E + 03	1.7437E + 03	1.6206E + 03	**1.6012E + 03**
	Mean	1.8896E + 03	2.0374E + 03	2.3068E + 03	2.0543E + 03	1.7799E + 03	1.8695E + 03	1.7903E + 03	**1.6613E + 03**
	Std	8.1860E + 01	1.7762E + 02	1.4360E + 02	8.1340E + 01	1.4419E + 02	1.0423E + 02	1.6925E + 02	**7.7536E + 01**
C7	Best	6.3967E + 03	3.8404E + 03	1.2623E + 04	6.9373E + 03	3.5546E + 03	5.5198E + 03	2.5484E + 03	**2.5220E + 03**
	Mean	9.9662E + 03	1.5493E + 04	2.3631E + 04	1.1691E + 04	1.0403E + 04	7.7588E + 03	7.5426E + 03	**6.8049E + 03**
	Std	3.5773E + 03	8.6727E + 03	9.2201E + 03	4.7531E + 03	4.7816E + 03	**1.4491E + 03**	5.9916E + 03	3.5373E + 03
C8	Best	2.2247E + 03	2.3359E + 03	2.7625E + 03	2.5119E + 03	2.2358E + 03	3.5616E + 03	2.3015E + 03	**2.2203E + 03**
	Mean	2.2989E + 03	2.5472E + 03	3.0783E + 03	2.6354E + 03	2.3003E + 03	3.8722E + 03	2.3120E + 03	**2.2976E + 03**
	Std	3.3494E + 01	6.1475E + 02	2.1883E + 02	5.1482E + 01	5.3854E + 00	1.4566E + 02	8.1098E + 00	**4.2554E + 00**
C9	Best	2.7391E + 03	2.5728E + 03	2.8215E + 03	2.7478E + 03	2.7469E + 03	2.7703E + 03	2.5007E + 03	**2.5000E + 03**
	Mean	2.7529E + 03	2.7102E + 03	2.9502E + 03	2.8016E + 03	2.7720E + 03	2.7751E + 03	2.6671E + 03	**2.6512E + 03**
	Std	1.8449E + 01	1.3534E + 02	7.9333E + 01	4.8917E + 01	3.2469E + 01	4.4290E + 00	1.2739E + 02	1.3806E + 02
C10	Best	2.8978E + 03	2.9234E + 03	3.2134E + 03	2.9804E + 03	2.8981E + 03	2.9151E + 03	2.9070E + 03	**2.8978E + 03**
	Mean	2.9466E + 03	3.1055E + 03	3.3319E + 03	3.1321E + 03	2.9294E + 03	2.9573E + 03	2.9387E + 03	**2.9317E + 03**
	Std	2.3832E + 01	1.4414E + 02	5.2824E + 01	8.8272E + 01	2.6473E + 01	2.6135E + 01	2.6639E + 01	**2.3378E + 01**

Table 5. Compared optimization results of different intelligent algorithms. Significant values are in bold.

of the functions is inferior to the algorithm, and almost all of the functions demonstrate superior performance in comparison to alternative algorithms. Hence, it may be inferred that CWXSCSO has excellent results compared to both the new method and existing enhanced optimization techniques.

Engineering application

Six engineering challenges have been selected for this part for the purpose to assess how well CWXSCSO performs when used to engineering optimization problems. The sine and cosine optimization algorithm (SCA)[40], frost and ice optimization algorithm (RIME)[41], butterfly optimization algorithm (BOA)[42], Harris Eagle Optimization algorithm (HHO)[8], and Osprey optimization algorithm (OOA)[43] were chosen as the primary three technical applications. The whale optimization algorithm (WOA)[7], the locust optimization algorithm (GOA)[44], the gray wolf optimization algorithm (GWO)[45], the marine predator optimization algorithm (MPA)[46], and the frost and ice optimization algorithm (RIME) were used to compare the final three technical applications. Every algorithm in the experiment has a population of 30 and an upper limit of 1000 iterations.

Pressure vessel design problems

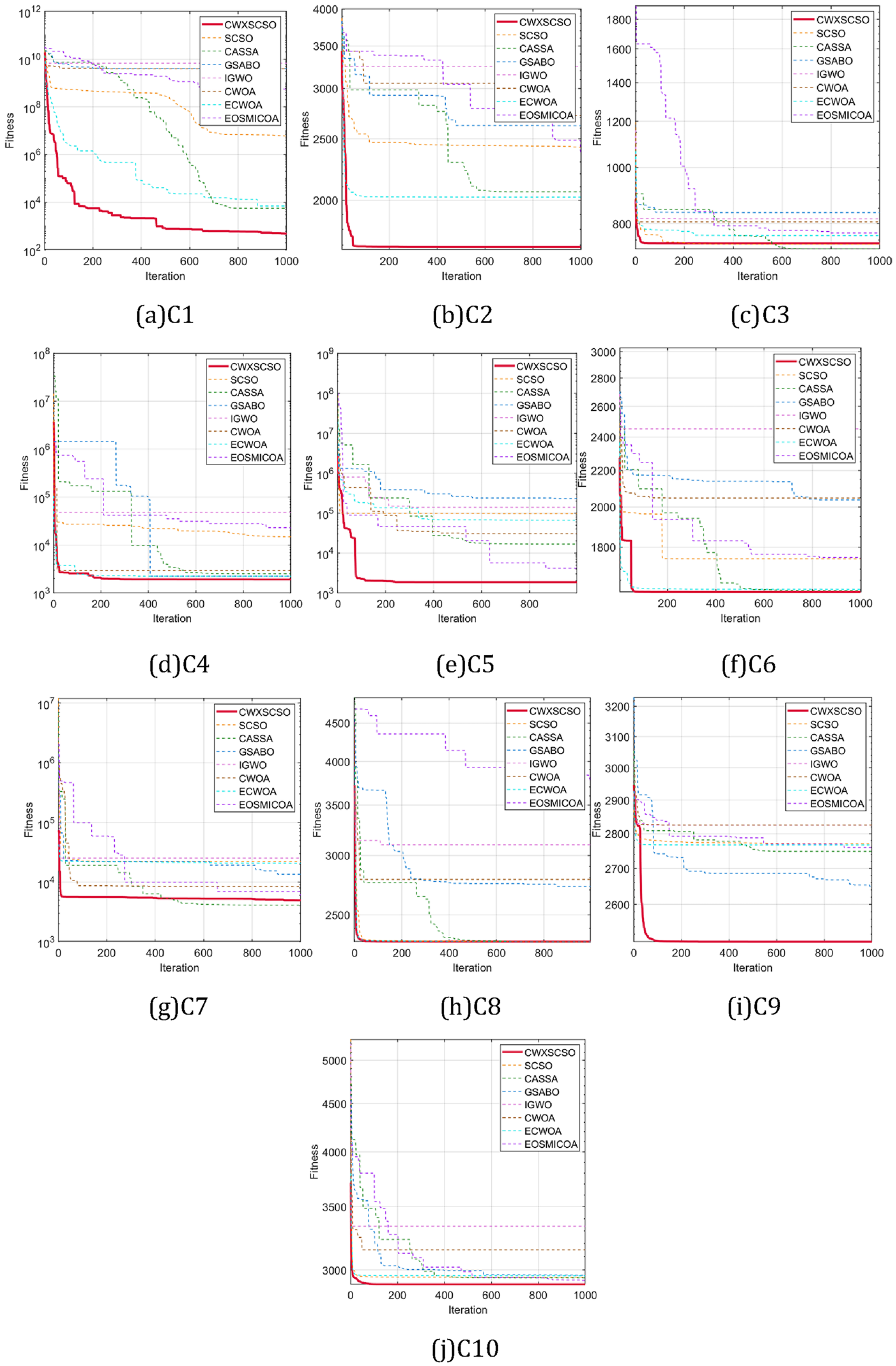
The performance of the modified algorithm pair gets assessed using pressure vessel design issues in this research. The main objective of the pressure vessel design challenge is to decrease the production expenses associated with the pressure vessel. This problem contains the selection of four optimization variables, namely shell thickness T_S, head thickness (T_h), inner radius (R), and length of cylinder section without head (L). The mathematical description of the pressure vessel design problem is as follows:

variable:

$$\overrightarrow{x} = [x_1 x_2 x_3 x_4] = [T_S T_h R L]$$

Function:

$$f\left(\overrightarrow{x}\right) = 0.6224 x_1 x_3 x_4 + 1.7781 x_2 x_3^2 + 3.1661 x_1^2 x_4 + 19.84 x_1^2 x_3$$



(a)C1

(b)C2

(c)C3

(d)C4

(e)C5

(f)C6

(g)C7

(h)C8

(i)C9

(j)C10

Figure 4. Convergence curves of different algorithms.

Function	CWXSCSO-CASSA	CWXSCSO-GSABO	CWXSCSO-IGWO	CWXSCSO-CWOA	CWXSCSO-ECWOA	CWXSCSO-EOSMICOA
C1	1.2118e − 12	1.2118e − 12	*NaN*	*NaN*	1.2118e − 12	1.2118e − 12
C2	2.9343e − 05	*NaN*	*NaN*	*NaN*	4.7899e − 06	1.3205e − 4
C3	1.9457e − 09	*NaN*	*NaN*	*NaN*	*NaN*	1.6572e − 11
C4	*NaN*	*NaN*	*NaN*	*NaN*	*NaN*	*NaN*
C5	1.2118e − 12	1.2118e − 12	*NaN*	*NaN*	1.2118e − 12	1.2118e − 12
C6	1.2118e − 12	1.2118e − 12	*NaN*	1.7016e − 08	1.2118e − 12	1.2118e − 12
C7	1.2118e − 12	1.2118e − 12	1.608e − 2	3.4526e − 07	1.2118e − 12	1.2118e − 12
C8	1.6082e − 12	*NaN*	*NaN*	*NaN*	*NaN*	*NaN*
C9	1.1364e − 11	2.9785e − 12	2.9009e − 07	7.8379e − 08	3.4571e − 12	1.271e − 11
C10	2.6286e − 11	2.6286e − 11	2.6286e − 11	2.6286e − 11	2.6286e − 11	2.6286e − 11
+ / = / −	9/1/0	6/4/0	3/7/0	4/6/0	7/3/0	8/2/0

Table 6. Results of Wilcoxon rank sum test for CEC2021 test.

Constraint condition:

$$g_1\left(\overrightarrow{x}\right) = -x_1 + 0.0193x_3 \leq 0$$

$$g_2\left(\overrightarrow{x}\right) = -x_3 + 0.00954x_3 \leq 0$$

$$g_3\left(\overrightarrow{x}\right) = -\pi x_3^2 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0$$

$$g_4\left(\overrightarrow{x}\right) = x_4 - 240 \leq 0$$

Variable interval:

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

The experimental findings of CWXSCSO and the comparison algorithm are presented in Table 7. The CWXSCSO yields a value of 5886.05. When compared to alternative algorithms, this particular algorithm exhibits a superior competitive advantage in terms of maintaining the proper functioning of the pressure vessel while simultaneously minimizing costs. Benefits in guaranteeing the operation of the pressure vessel while reducing expenses. The updated method demonstrates rapid convergence to the ideal value with the best convergence accuracy, as depicted in Fig. 5. In turn, the CWXSCSO facility exhibits exceptional engineering optimization capabilities.

Welding beam design problems

The issue at hand is the Welded Beam Design (WBD), which involves the utilization of an optimization method to minimize the production cost associated with the design. The optimization problem can be boiled down to the identification of four design variables that meet the constraints of shear stress (τ), bending stress (θ), beam bending load (P_c), end deviation (δ), and boundary conditions, namely beam length (l), height (t), thickness (b), and weld thickness (h). The objective is to minimize the manufacturing cost of welded beams. The problem of welded beams is a common example of a nonlinear programming problem. The mathematical description of the welded beam design problem is as follows:

Variable:

Algorithm	$T_S(x_1)$	$T_h(x_2)$	$R(x_3)$	$L(x_4)$	Result
BOA	1.7058	2.1120	63.6295	37.6427	21,767.48
HHO	1.0975	0.5495	56.2523	55.3661	6774.49
OOA	4.4124	13.0999	54.7594	64.5801	104,691.28
SCA	0.7830	0.3883	40.5700	196.6224	5899.53
RIME	1.2599	0.6233	65.2280	10.0000	7331.08
SCSO	0.8063	0.3986	41.7742	180.7003	5936.00
CWXSCSO	0.7783	0.3847	40.3262	199.9175	5886.05

Table 7. Optimization results of pressure vessel design problems.

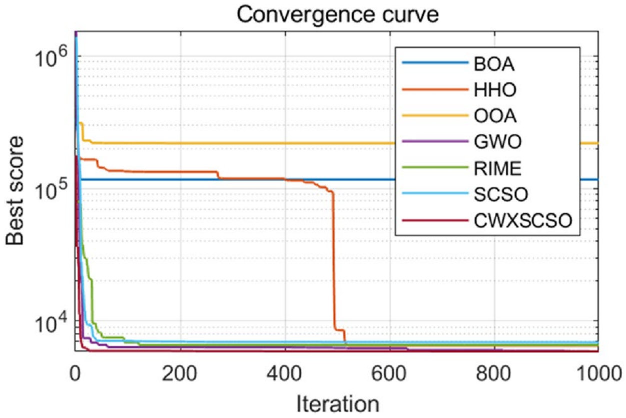


Figure 5. Optimization convergence diagram of pressure vessel design problem.

$$\overrightarrow{x} = [x_1 x_2 x_3 x_4] = [hltb]$$

Function:

$$f\left(\overrightarrow{x}\right) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2)$$

Constraint condition:

$$g_1\left(\overrightarrow{x}\right) = \tau\left(\overrightarrow{x}\right) - \tau_{max} \le 0$$

$$g_2\left(\overrightarrow{x}\right) = \sigma\left(\overrightarrow{x}\right) - \sigma_{max} \le 0$$

$$g_3\left(\overrightarrow{x}\right) = \delta\left(\overrightarrow{x}\right) - \delta_{max} \le 0$$

$$g_4\left(\overrightarrow{x}\right) = x_1 - x_4 \le 0$$

$$g_5\left(\overrightarrow{x}\right) = P - P_c\left(\overrightarrow{x}\right) \le 0$$

$$g_6\left(\overrightarrow{x}\right) = 0.125 - x_1 \le 0$$

$$g_7\left(\overrightarrow{x}\right) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2) - 5.0 \le 0$$

Variable interval:

$$0.1 \le x_1 \le 2, 0.1 \le x_2 \le 10, 0.1 \le x_3 \le 10, 0.1 \le x_4 \le 2$$

As can be seen from Table 8, the final result of CWXSCSO is 1.6935. As can be seen in Fig. 6, the initial fitness value of the improved algorithm is already very good, and there are several subtle turns later, indicating that it has the ability to jump out of the local optimal. The improved algorithm achieves the purpose of reducing the manufacturing cost, and the cost of manufacturing welded beams is minimal compared with other algorithms.

Algorithm	*h*	*l*	*t*	*b*	Result
BOA	0.1682	8.0508	8.8515	0.2168	2.2875
HHO	0.2087	3.1115	9.1852	0.2113	1.7476
OOA	0.7460	2.4806	4.7436	0.7467	4.3336
SCA	0.2071	3.1690	9.2701	0.2095	1.7547
RIME	0.2157	3.5682	7.9583	0.2654	1.9687
SCSO	0.2038	3.2750	9.0367	0.2057	1.6954
CWXSCSO	0.2058	3.2359	9.0342	0.2058	1.6935

Table 8. Optimization results of welding beam design problems.

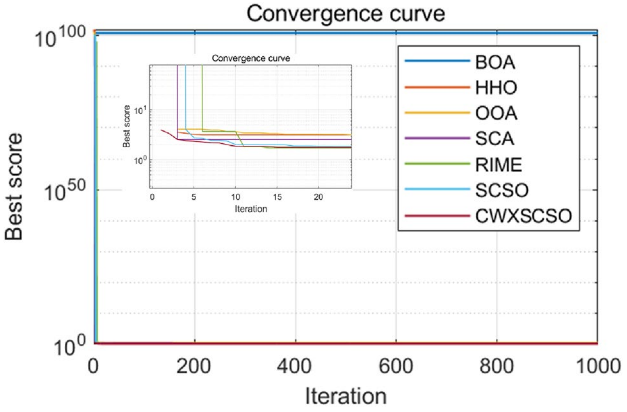


Figure 6. Optimization convergence diagram of welding beam design problem.

Reducer design problem

The reducer holds an important place within mechanical systems as a crucial component of the gear box, serving a diverse range of applications. The primary aim of this challenge is to diminish the overall weight of the reducer through the optimization of the seven parameter variables. They are the tooth surface width $b(=x_1)$, the gear module $m(=x_2)$, the tooth count in the pinion $z(=x_3)$, the measurement of the initial shaft distance between bearings. $l_1(=x_4)$, the distance between the bearings of the second shaft $l_2(=x_5)$, the diameter of the initial shaft $d_1(=x_6)$ and the measurement of the diameter of the second shaft $d_2(=x_7)$. The mathematical description of the speed reducer design problem is as follows:

Variable:

$$\overrightarrow{x}=[x_1x_2x_3x_4x_5x_6x_7]=[bmzl_1l_2d_1d_2]$$

Function:

$$f\left(\overrightarrow{x}\right)=0.7854x_1x_2^2\left(3.3333x_3^2+14.9334x_3-43.0934\right)-1.508x_1\left(x_6^2+x_7^2\right)+7.4777\left(x_6^3+x_7^3\right)+0.7854(x_4x_6^2+x_5x_7^2)$$

Constraint condition:

$$g_1\left(\overrightarrow{x}\right)=\frac{27}{x_1x_2^2x_3}-1\le 0$$

$$g_2\left(\overrightarrow{x}\right)=\frac{397.5}{x_1x_2^2x_3^2}-1\le 0$$

$$g_3\left(\overrightarrow{x}\right)=\frac{1.93x_4^3}{x_2x_3x_6^4}-1\le 0$$

$$g_4\left(\overrightarrow{x}\right)=\frac{1.93x_5^3}{x_2x_3x_7^4}-1\le 0$$

$$g_5\left(\overrightarrow{x}\right)=\frac{\sqrt{\left(\frac{745x_4}{x_2x_3}\right)^2+16.9\times 10^6}}{110.0x_6^3}-1\le 0$$

$$g_6\left(\overrightarrow{x}\right)=\frac{\sqrt{\left(\frac{745x_4}{x_2x_3}\right)^2+157.5\times 10^6}}{85.0x_6^3}-1\le 0$$

$$g_7\left(\overrightarrow{x}\right)=\frac{x_2x_3}{40}-1\le 0$$

$$g_8\left(\overrightarrow{x}\right)=\frac{5x_2}{x_1}-1\le 0$$

$$g_9(\overrightarrow{x}) = \frac{x_1}{12x_2} - 1 \leq 0$$

$$g_{10}(\overrightarrow{x}) = \frac{1.5x_6 + 1.9}{x_4} - 1 \leq 0$$

$$g_{11}(\overrightarrow{x}) = \frac{1.1x_7 + 1.9}{x_5} - 1 \leq 0$$

Variable interval:

$$2.6 \leq x_1 \leq 3.6, 0.7 \leq x_2 \leq 0.8, 17 \leq x_3 \leq 28, 7.3 \leq x_4 \leq 8.3, 7.8 \leq x_5 \leq 8.3,$$

$$2.9 \leq x_6 \leq 3.9, 5.0 \leq x_7 \leq 5.5$$

Table 9 and Fig. 7 demonstrate that the modified method is adept at minimizing the weight of the reducer under 11 boundaries. It suggests that the enhancement is effective and may be more effectively utilized in mechanical systems.

Step cone pulley problem

This engineering project aims to create a 4-step cone pulley with a minimal weight by looking at 5 design elements. Four variables represent the diameter of individual step of the pulley, denoted as $d_i(i = 1, 2, 3, 4)$, while the final variable represents the magnitude of the pulley's breadth, denoted as w. There are 8 nonlinear constraints and 3 linear constraints in the problem. The restriction is to maintain uniformity in the belt length C_i, tension ratio R_i, and belt transfer power P_i throughout all steps. The mathematical description of the step cone pulley problem is as follows:

Function:

$$f(x) = \rho\omega\left[d_1^2\left\{1 + \left(\frac{N_1}{N}\right)^2\right\} + d_2^2\left\{1 + \left(\frac{N_2}{N}\right)^2\right\} + d_3^2\left\{1 + \left(\frac{N_3}{N}\right)^2\right\} + d_4^2\left\{1 + \left(\frac{N_4}{N}\right)^2\right\}\right]$$

Constraint condition:

Algorithm	x_1	x_2	x_3	x_4	x_5	x_6	x_7	Result
BOA	3.3994	0.7088	19.6705	7.8068	8.2178	3.4333	5.2689	1.9119E+98
HHO	3.5827	0.7000	17.0000	8.2264	7.7636	3.4083	5.2867	3051.49
OOA	3.5411	0.7082	25.5903	7.7667	7.7708	3.5589	5.2860	4990.53
SCA	3.6000	0.7000	17.0000	7.6626	8.3000	3.6003	5.3054	3130.57
RIME	3.5003	0.7000	17.0000	7.3000	7.7227	3.3506	5.2868	2994.95
SCSO	3.5001	0.7000	17.0000	7.6154	8.1498	3.3510	5.2868	3007.17
CWXSCSO	3.5003	0.7000	17.0000	7.3000	7.7172	3.3503	5.2867	2994.66

Table 9. Optimization results of reducer design problems.

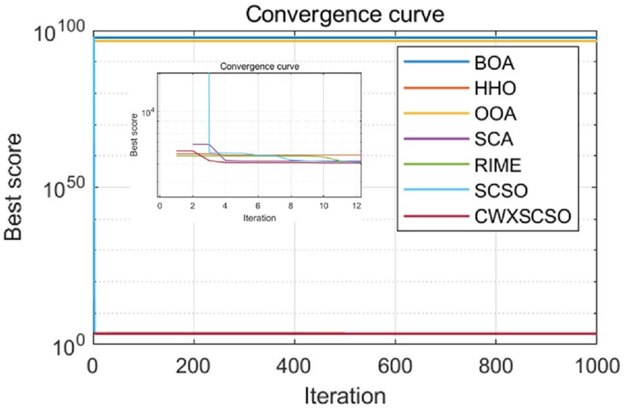


Figure 7. Reducer design optimization convergence curve.

$$h_1(x) = C_1 - C_2 = 0, h_2(x) = C_1 - C_3 = 0, h_3(x) = C_1 - C_4 = 0$$

$$g_{1,2,3,4}(x) = R_i \geq 2, g_{5,6,7,8}(x) = P_i \geq (0.75 * 745.6998)$$

where:

$$C_i = \frac{\pi d_i}{2}\left(1 + \frac{N_i}{N}\right) + \frac{\left(\frac{N_i}{N} - 1\right)^2}{4a} + 2ai = (1, 2, 3, 4)$$

$$R_i = exp\left[\mu\left\{\pi - 2\sin^{-1}\left\{\left(\frac{N_i}{N} - 1\right)\frac{d_i}{2a}\right\}\right\}\right]i = (1, 2, 3, 4)$$

$$P_i = stw\left[1 - exp\left[-\mu\left\{\pi - 2\sin^{-1}\left\{\left(\frac{N_i}{N} - 1\right)\frac{d_i}{2a}\right\}\right\}\right]\right]\frac{\pi d_i N_i}{60}i = (1, 2, 3, 4)$$

$$\rho = 7200kg/m^3, a = 3m, \mu = 0.35, s = 1.75MPa, t = 8mm$$

Variable interval:

$$0 \leq d_1, d_2 \leq 60, 0 \leq d_3, \omega \leq 90$$

Table 10 clearly demonstrates that the MPA method outperforms the CWXSCSO algorithm, but it still possesses certain advantages over other algorithms. Figure 8 illustrates that while the precision of convergence in CWXSCSO is less than that of MPA, its convergence speed beats that of MPA. Despite lacking MPA for the stepping cone pulley problem, CWXSCSO still has the benefit of rapid convergence speed.

Planetary gear train design optimization problem

In power mechanical systems, the design of a planetary gear train presents a limited optimization problem. The issue encompasses three optimization variables, specifically the quantity of gear teeth $(N_1, N_2, N_3, N_4, N_5, N_6)$, gear modulus (m_1, m_2), and the figure of merit (p). The primary aim of the issue is to limit the maximum error associated with the transmission ratio employed in automotive production. The issue at hand encompasses a

Algorithm	d_1	d_2	d_3	d_4	w	Result
WOA	41	56	75	90	85	1.6659E + 89
GOA	34	47	54	85	90	9.7914E + 97
GWO	41	56	74	89	89	2.2103E + 92
MPA	39	54	72	86	90	3.0726E + 81
RIME	39	54	72	87	90	2.0414E + 90
SCSO	40	56	74	89	86	5.3743E + 90
CWXSCSO	40	55	73	88	88	1.2490E + 89

Table 10. Optimization results of step cone pulley problem.

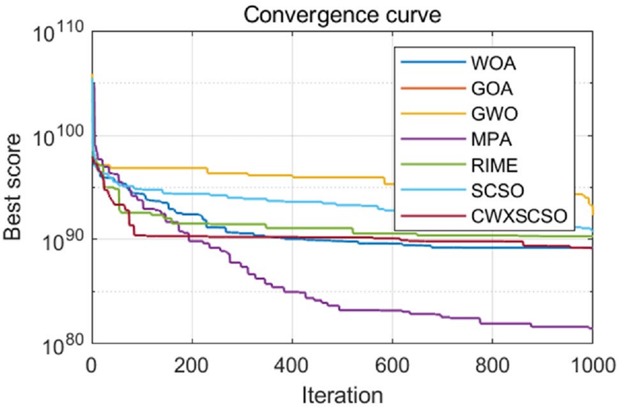


Figure 8. Optimization convergence diagram of step cone pulley problem.

total of six integer variables, three discrete variables, and eleven distinct geometric and assembly restrictions. The mathematical description of the planetary gear train design optimization problem is as follows:

Variable:

$$x = (x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9) = \left(N_1, N_2, N_3, N_4, N_5, N_6, m_1, m_2, p\right)$$

Function:

$$f(x) = max|i_k - i_{ok}|, k = \{1, 2, \ldots, R\}$$

where:

$$i_1 = \frac{N_6}{N_4}, i_{o1} = 3.11, i_2 = \frac{N_6(N_1N_3 + N_2N_4)}{N_1N_3(N_6 + N_4)}, i_{OR} = -3.11, I_R = -\frac{N_2N_6}{N_1N_3}, i_{O2} = 1.84$$

Constraint condition:

$$g_1(x) = m_2(N_6 + 2.5) - D_{max} \leq 0$$

$$g_2(x) = m_1(N_1 + N_2) + m_1(N_2 + 2) - D_{max} \leq 0$$

$$g_3(x) = m_2(N_4 + N_5) + m_2(N_5 + 2) - D_{max} \leq 0$$

$$g_4(x) = |m_1(N_1 + N_2) - m_1(N_6 + N_3)| - m_1 - m_2 \leq 0$$

$$g_5(x) = -(N_1 + N_2)\sin\left(\frac{\pi}{p}\right) + N_2 + 2 + \delta_{22} \leq 0$$

$$g_6(x) = -(N_6 - N_3)\sin\left(\frac{\pi}{p}\right) + N_3 + 2 + \delta_{33} \leq 0$$

$$g_7(x) = -(N_4 + N_5)\sin\left(\frac{\pi}{p}\right) + N_5 + 2 + \delta_{55} \leq 0$$

$$g_8(x) = (N_3 + N_5 + 2 + \delta_{35})^2 - (N_6 - N_3)^2 - (N_4 + N_5)^2 + 2(N_6 - N_3)(N_4 + N_5)\cos\left(\frac{2\pi}{p} - \beta\right) \leq 0$$

$$g_9(x) = N_4 - N_6 + 2N_5 + 2\delta_{56} + 4 \leq 0$$

$$g_{10}(x) = 2N_3 - N_6 + N_4 + 2\delta_{34} + 4 \leq 0$$

$$h_1(x) = \frac{N_6 - N_4}{p} = integer$$

where:

$$\delta_{22} = \delta_{33} = \delta_{55} = \delta_{35} = \delta_{56} = 0.5$$

$$\beta = \frac{cos^{-1}\left((N_4 + N_5)^2 + (N_6 - N_3)^2 - (N_3 + N_5)^2\right)}{2(N_6 - N_3)(N_4 + N_5)}$$

$$D_{max} = 220$$

Variable interval:

$$P = (3, 4, 5), m_1, m_2 = (1.75, 2.0, 2.25, 2.5, 2.75, 3.0), 17 \leq N_1 \leq 96,$$

$$14 \leq N_2 \leq 54, 14 \leq N_3 \leq 51, 17 \leq N_4 \leq 46, 14 \leq N_5 \leq 51, 48 \leq N_6 \leq 124$$

Based on the data shown in Fig. 9 and Table 11, it is evident that CWXSCSO continues to outperform other methods in terms of convergence accuracy and convergence speed. This illustrates the potential for widespread implementation and utilization of the upgraded algorithm in power machinery.

Robot clamping optimization problem

The issue of robot hand claws is a complex challenge within the field of mechanical structure engineering. The goal of the robot clamping optimization is to minimize the disparity between the highest and lowest magnitudes

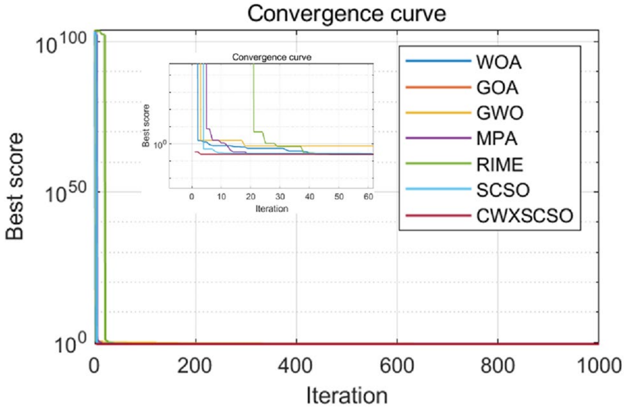


Figure 9. Convergence curve of planetary gear train design optimization problem.

Algorithm	N_1	N_2	N_3	N_4	N_5	N_6	m_1	m_2	p	Result
WOA	26	24	19	24	20	69	1	1	1	0.2422
GOA	17	16	14	17	14	50	2	1	1	0.2513
GWO	43	35	19	27	15	78	1	3	1	0.2379
MPA	30	24	22	32	22	92	3	1	1	0.2355
RIME	18	16	17	22	14	64	4	1	1	0.2398
SCSO	26	20	16	24	14	69	2	1	1	0
CWXSCSO	17	16	14	17	14	50	3	2	1	0

Table 11. Results of planetary gear train design optimization problem.

of forces. The challenge of robot grippers encompasses a total of seven continuous design variables– the three connecting rods (a, b, c), the vertical displacement of the linkages (d), the vertical distance separating the initial node of the robotic arm from the end of the actuator (e), the displacement in the horizontal direction between the actuator end and the linkages node (f), and the angle of the second and third linkages in a geometric context (ρ). There appear a total of seven distinct limitations. The mathematical description of the robot clamping optimization problem is as follows:

Variable:

$$x = (x_1, x_2, x_3, x_4, x_5, x_6, x_7) = \big(a, b, c, d, e, f, p\big)$$

Function:

$$f(x) = -\min_z F_k(x, z) + \max_z F_k(x, z)$$

Constraint condition:

$$g_1(x) = -Y_{min} + y((x), Z_{max}) \leq 0$$

$$g_2(x) = -y((x), Z_{max}) \leq 0$$

$$g_3(x) = Y_{max} - y((x), 0) \leq 0$$

$$g_4(x) = y((x), 0) - Y_G \leq 0$$

$$g_5(x) = l^2 + d^2 - (a + b)^2 \leq 0$$

$$g_6(x) = b^2 - (a - d)^2 - (l - Z_{max})^2 \leq 0$$

$$g_7(x) = Z_{max} - f$$

where:

$$\alpha = cos^{-1}\left(\frac{a^2 + g^2 - b^2}{2ag}\right) + \Phi, g = \sqrt{f^2 + (z - f)^2}$$

$$\beta = cos^{-1}\left(\frac{b^2 + g^2 - a^2}{2ag}\right) - \Phi, \Phi = tan^{-1}\left(\frac{d}{f - z}\right)$$

$$y(x, z) = 2\left(l + d + c\sin\left(\beta + p\right)\right)$$

$$F_k = \frac{Pb\sin(\alpha + \beta)}{2c\cos(\alpha)}, Y_{min} = 50, Y_{max} = 100, Y_G = 150, P = 100$$

Variable interval:

$$0 \le d \le 50, 100 \le c \le 200, 10 \le e, a, b \le 150, 1 \le p \le 3.14, 100 \le f \le 300$$

The data presented in Table 12 indicates that the CWXSCSO has the smallest disparity between its maximum force and minimum force. The curve convergence accuracy of CWXSCSO is the highest, as depicted in Fig. 10. Thus, the modified algorithm exhibits commendable competitive prowess within the field of mechanical engineering.

Conclusion

The article introduces a sand cat swarm optimization algorithm which enhances the optimization power by including elite decentralization and crossbar technique. The algorithm that has been enhanced introduces a novel dynamic exponential factor. The position formula is updated using the elite decentralization technique, followed by the introduction of the crossbar strategy to accelerate the rate of convergence and improve the precision of search results. All strategies undergo testing using an assortment of 15 benchmark functions. CWXSCSO demonstrates superior search efficiency and stability, along with improved local search power and optimization accuracy, when compared to SCSO. Among the many strategies, elite decentralization and crossbar strategies have been found to be advantageous in enabling the algorithm beyond local optima. Simulation experiments were conducted on 10 test functions of CEC2019 and 10 test functions of CEC2021 using CWXSCSO and 6 other optimization algorithms. The results show that CWXSCSO outperforms the other 6 optimization algorithms in terms of optimization results. Additionally, CWXSCSO has the capacity to produce global optimal solutions for

Algorithm	*a*	*b*	*c*	*d*	*e*	*f*	*ρ*	Result
WOA	100	38	100	0	10	100	1	1.4554E − 16
GOA	92	30	100	0	11	100	1	2.6670E − 16
GWO	100	38	199	0	125	100	2	7.3214E − 17
MPA	100	38	200	0	71	100	2	7.2741E − 17
RIME	100	38	200	0	101	100	2	7.3241E − 17
SCSO	100	38	200	0	14	100	1	7.0141E − 21
CWXSCSO	94	32	124	0	10	100	1	0

Table 12. Results of robot clamping optimization problem.

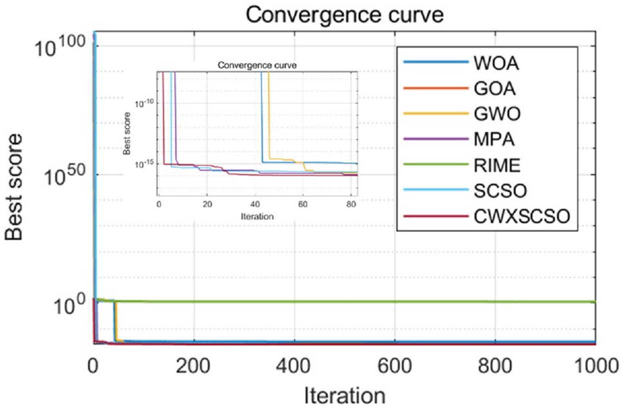


Figure 10. Convergence diagram of robot clamping optimization problem.

certain functions. The feasibility of the modified algorithm in practical engineering issues is further proven by the design of pressure vessel, welding beam, reducer, stepping cone pulley, planetary gear train design optimization, and robot clamping optimization. consequently, the primary purpose of this stage is to utilize it in order to address extensive, intricate multi-objective optimization problems and real engineering implementations.

Data availability

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Received: 16 February 2024; Accepted: 12 April 2024
Published online: 18 April 2024

References

1. Yuhang, R. & Liang, Z. An Adaptive evolutionary multi-objective estimation of distribution algorithm and its application to multi-UAV path planning. *IEEE Access* **11**, 50038–50051 (2023).
2. Yuchun, S., Sishan, W., Leyan, T., Shaohui, C., Yangju, C. Sishan, W. Research on path planning optimization algorithm based on loss function weight adaptation. *Comput. Eng. Appl.* (2024).
3. Xi, Y., Tao, F. & Brooks, S. Optimization of carton recycling site selection using particle swarm optimization algorithm considering residents' recycling willingness. *PeerJ Comput. Sci.* **9**, e1519 (2023).
4. Mzili, T., Mzili, I., Riffi, M. E. & Dhiman, G. Hybrid genetic and spotted hyena optimizer for flow shop scheduling problem. *Algorithms* **16**, 265 (2023).
5. Li, J. *et al.* Flexible job shop scheduling optimization for green manufacturing based on improved multi-objective wolf pack algorithm. *Appl. Sci.* **13**, 8535 (2023).
6. Qingwei, T. *et al.* Wind farm power transfer forecasting method based on CNN–LSTM. *Adv. Eng. Sci.* https://doi.org/10.15961/j.jsuese.202201165 (2024).
7. Mirjalili, S. & Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **95**, 51–67 (2016).
8. Heidari, A. A. *et al.* Harris hawks optimization: Algorithm and applications. *Fut. Gener. Comput. Syst.* **97**, 849–872 (2019).
9. Khishe, M. & Mosavi, M. R. Chimp optimization algorithm. *Expert Syst. Appl.* **149**, 113338 (2020).
10. Dhiman, G. & Kaur, A. STOA: A bio-inspired based optimization algorithm for industrial engineering problems. *Eng. Appl. Artif. Intell.* **82**, 148–174 (2019).
11. Civicioglu, P. Backtracking search optimization algorithm for numerical optimization problems. *Appl. Math. Computat.* **219**, 8121–8144 (2013).
12. Rocca, P., Oliveri, G. & Massa, A. Differential evolution as applied to electromagnetics. *IEEE Antennas Propag. Mag.* **53**, 38–49 (2011).
13. Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. Optimization by simulated annealing. In *Readings in Computer Vision* (eds Fischler, M. A. & Firschein, O.) 606–615 (Morgan Kaufmann, 1987). https://doi.org/10.1016/B978-0-08-051581-6.50059-3.
14. Rashedi, E., Nezamabadi-pour, H. & Saryazdi, S. G. S. A. A gravitational search algorithm. *Inf. Sci.* **179**, 2232–2248 (2009).
15. Bouchekara, H. R. E. H. Optimal design of electromagnetic devices using a black-hole-based optimization technique. *IEEE Trans. Magn.* **49**, 5709–5714 (2013).
16. Shi, Y. Brain storm optimization algorithm. In *Advances in Swarm Intelligence* (eds Tan, Y. *et al.*) 303–309 (Springer, 2011). https://doi.org/10.1007/978-3-642-21515-5_36.
17. Wang, L., Ni, H., Yang, R., Fei, M. & Ye, W. A simple human learning optimization algorithm. In *Computational Intelligence, Networked Systems and Their Applications* (eds Fei, M. *et al.*) 56–65 (Springer, 2014). https://doi.org/10.1007/978-3-662-45261-5_7.
18. Nama, S., Saha, A. K. & Sharma, S. A novel improved symbiotic organisms search algorithm. *Computat. Intell.* **38**, 947–977 (2022).
19. Nama, S., Sharma, S., Saha, A. K. & Gandomi, A. H. A quantum mutation-based backtracking search algorithm. *Artif. Intell. Rev.* **55**, 3019–3073 (2022).
20. Nama, S. A modification of I-SOS: Performance analysis to large scale functions. *Appl. Intell.* **51**, 7881–7902 (2021).
21. Nama, S. A novel improved SMA with quasi reflection operator: Performance analysis, application to the image segmentation problem of Covid-19 chest X-ray images. *Appl. Soft Comput.* **118**, 108483 (2022).
22. Luo, Q. *et al.* Multi-objective equilibrium optimizer slime mould algorithm and its application in solving engineering problems. *Struct. Multidisc. Optim.* **66**, 114 (2023).
23. Yin, S., Luo, Q., Zhou, G., Zhou, Y. & Zhu, B. An equilibrium optimizer slime mould algorithm for inverse kinematics of the 7-DOF robotic manipulator. *Sci. Rep.* **12**, 9421 (2022).
24. Zhang, Y., Zhou, Y., Zhou, G., Luo, Q. & Zhu, B. A curve approximation approach using bio-inspired polar coordinate bald eagle search algorithm. *Int. J. Comput. Intell. Syst.* **15**, 30 (2022).
25. Nama, S., Saha, A. K. & Ghosh, S. Improved backtracking search algorithm for pseudo dynamic active earth pressure on retaining wall supporting c-Φ backfill. *Appl. Soft Comput.* **52**, 885–897 (2017).
26. Chakraborty, S., Nama, S. & Saha, A. K. An improved symbiotic organisms search algorithm for higher dimensional optimization problems. *Knowl.-Based Syst.* **236**, 107779 (2022).
27. Seyyedabbasi, A. & Kiani, F. Sand Cat swarm optimization: A nature-inspired algorithm to solve global optimization problems. *Eng. Comput.* **39**, 2627–2651 (2023).
28. Wu, D. *et al.* Modified sand cat swarm optimization algorithm for solving constrained engineering optimization problems. *Mathematics* **10**, 4350 (2022).
29. Lu, W., Shi, C., Fu, H. & Xu, Y. A power transformer fault diagnosis method based on improved sand cat swarm optimization algorithm and bidirectional gated recurrent unit. *Electronics* **12**, 672 (2023).
30. Wang, X., Liu, Q. & Zhang, L. An adaptive sand cat swarm algorithm based on cauchy mutation and optimal neighborhood disturbance strategy. *Biomimetics* **8**, 191 (2023).
31. Heming, J., Zhuo, W., Changsheng, W., Honghua, R. & Yuanyuan, S. UAV 3D path planning based on improved sand cat swarm optimization algorithm. *J. Ningde Teach. Coll. (Nat. Sci.)* **35**, 171–179 (2023).
32. Liqun, S., Shaoqiang, Z., Jiangshan, L. Reactive power optimization of wind/solar power storage and distribution network based on multi-objective sand cat swarm algorithm. *J. Nanjing Univ. Inf. Sci. Technol.* doi:https://doi.org/10.13878/j.cnki.jnuist.20230621001.
33. Kaizheng, J. & Liping, L. Optimization of fault diagnosis for stacked denoising auto encoder' s engine by improved sand cat swarm optimization algorithm. *J. Mach. Des.* **40**, 56–62 (2023).
34. Kun, L., Lulu, Z. & Hui, W. Whale optimization algorithm based on elite opposition-based and crisscross optimization. *J. Chin. Comput. Syst.* **41**, 2092–2097 (2020).
35. Trojovský, P. & Dehghani, M. Subtraction-average-based optimizer: A new swarm-inspired metaheuristic algorithm for solving optimization problems. *Biomimetics* **8**, 149 (2023).

36. Damin, Z., Zhongyun, C., Ziyun, X., Huiyun, Z. & Wei, Y. Salp swarm algorithm based on craziness and adaptive. *Control Decis.* **35**, 2112–2120 (2020).
37. Miao, L., Dengxu, H. & Liangdong, Q. Grey wolf optimization algorithm based on elite learning for nonlinear parameters. *J. Guangxi Normal Univ. (Nat. Sci. Edn.)* **39**, 55–67 (2021).
38. Jianhao, W. *et al.* Whale optimization algorithm based on chaotic search strategy. *J. Control Decis.* **34**, 1893–1900 (2019).
39. Qian, H., Sheng, L., Mengmeng, L. & Yuxin, G. Multi-strategy chimp optimization algorithm and its application of engineering problem. *Comput. Eng. Appl.* **58**, 174–183 (2022).
40. Mirjalili, S. SCA: A Sine Cosine Algorithm for solving optimization problems. *Knowl.-Based Syst.* **96**, 120–133 (2016).
41. Su, H. R. I. M. E. A physics-based optimization. *Neurocomputing* **532**, 183–214 (2023).
42. Arora, S. & Singh, S. Butterfly optimization algorithm: A novel approach for global optimization. *Soft Comput.* **23**, 715–734 (2019).
43. Dehghani, M. & Trojovský, P. Osprey optimization algorithm: A new bio-inspired metaheuristic algorithm for solving engineering optimization problems. *Front. Mech. Eng.* **8**, 1126450 (2023).
44. Saremi, S., Mirjalili, S. & Lewis, A. Grasshopper optimisation algorithm: Theory and application. *Adv. Eng. Softw.* **105**, 30–47 (2017).
45. Mirjalili, S., Mirjalili, S. M. & Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **69**, 46–61 (2014).
46. Faramarzi, A., Heidarinejad, M., Mirjalili, S. & Gandomi, A. H. Marine predators algorithm: A nature-inspired metaheuristic. *Expert Syst. Appl.* **152**, 113377 (2020).

Author contributions

Yu Qian wrote the main manuscript text as well as all the diagrams. All the authors reviewed the manuscript.

Funding

This work was funded by the National Natural Science Foundation of China (grant no. 52278171), and Natural Science Foundation of Hebei Province (grant no. E2020402079).

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-024-59597-0.

Correspondence and requests for materials should be addressed to Q.Y.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.