



OPEN

A black-winged kite optimization algorithm enhanced by osprey optimization and vertical and horizontal crossover improvement

Yancang Li^{1,2}, Binli Shi¹✉, Weitao Qiao³ & Zunfeng Du⁴

This paper addresses issues of inadequate accuracy and inconsistency between global search efficacy and local development capability in the black-winged kite algorithm for practical problem-solving by proposing a black-winged kite optimization algorithm that integrates the Osprey optimization algorithm and Crossbar enhancement (DKCBKA). Firstly, the adaptive index factor and the fusion Osprey Optimization Algorithm approach are incorporated to enhance the algorithm's convergence rate, and the probability distribution factor is updated throughout the attack stage. Second, the stochastic difference variant method is implemented to prevent the method from entering the local optima. Lastly, the longitudinal and transversal crossover technique is incorporated to enhance the algorithm's convergence accuracy and to dynamically alter the population's global and individual optimal solutions. Fifteen benchmark functions are chosen to test the effectiveness of the enhanced algorithm and to compare the optimization efficiency of each technique. Simulation experiments are performed on the CEC2017 and CEC2019 test sets, revealing that the DKCBKA algorithm surpasses five standard swarm intelligence optimization methods and six improved optimization algorithms regarding solution accuracy and convergence speed. The superiority in meeting real optimization challenges is further demonstrated by the optimization of three real engineering optimization problems by DKCBKA, with optimization capabilities 18.222%, 99.885% and 0.561% higher than BKA, respectively.

Keywords Black-Winged Kite algorithm, Dynamic exponential factor, Osprey optimization algorithm, Stochastic differential variance strategy, Vertical and horizontal crossover strategy

Optimization challenges are becoming significant in disciplines such as engineering, economics, bioinformatics, and artificial intelligence, and others as human civilization, science, and technology improve. Historically, optimization approaches have been widely utilized across numerous fields, including medical problems^{1,2}, engineering optimization³, machine learning⁴, image processing⁵, offshore wind generation⁶, and many more. However, many real-world issues are highly nonlinear, multimodal, and unpredictable⁷, making it challenging to solve them using classic optimization strategies.

Meta-heuristic algorithms are categorized into four types according to their sources of motivation: group intelligence algorithms, evolution-based algorithms, human-based algorithms, and physics and chemistry-based algorithms. Optimization algorithms employing population intelligence are utilized to emulate the behavioral traits of biological populations to get the global optimal solution. In accordance with this algorithm, every group is a biological group that can complete tasks that are impossible for individuals to do due to the synergistic behavior of its members. Its features include excellent robustness, ease of implementation, and a simple structure^{8–10}. For example, Spider-Tailed Horned Viper Optimization (STHVO)¹¹, Orangutan Optimization Algorithm (OOA)¹², Pied Kingfisher Optimizer (PKO)¹³, Fossa Optimization Algorithm (FOA)¹⁴, Secretary Bird optimization algorithm (SBOA)¹⁵, Addax Optimization Algorithm (AOA)¹⁶.

The study of metaheuristic algorithms is an area that is always changing to meet the demands of practical applications and ever-more-complex global optimization issues. The principle of “there is no free lunch” posits that not one algorithm may achieve optimal performance across all optimization and search tasks. Thus,

¹School of Civil Engineering, Hebei University of Engineering, Handan 056038, Hebei, China. ²Hebei GEO University, Shijiazhuang 050031, Hebei, China. ³Jizhong Energy Fengfeng Group Company Limited, Handan 056201, Hebei, China. ⁴School of Civil Engineering, Tianjin University, Tianjin 300354, China. ✉email: shibinli2001@163.com

many researchers are now concentrating on improving the approach of metaheuristic algorithms. To address the inadequate optimization efficacy of the conventional Beluga Whale Optimization Algorithm in complex multidimensional issues, a mixed multi-strategy improved Beluga Whale Optimization Algorithm was developed in the literature¹⁷. Similarly, literature¹⁸ presented a mixed multi-strategy enhanced Sparrow Search Algorithm (MISSA) to resolve the traveling salesman problem, addressing deficiencies in its fundamental Sparrow Search Algorithm (SSA), such as delayed convergence, local optimum, and convergence speed, among other issues. literature¹⁹ proposed a hybrid cuckoo search algorithm (LHCS) utilizing a linearly declining population to enhance the system's local search capabilities and expedite convergence. A novel sparrow search method (IBSSA), founded on beetle antenna searches, was proposed in the literature²⁰ to enhance the efficacy for the sparrow search algorithm in tackling intricate optimization issues. In order to solve problems like reduced population diversity and To address issues such as diminished varied populations and the inclination to converge towards local optima towards the conclusion of algorithm iterations, a multi-strategy amalgamation of the enhanced Osprey Optimization Algorithm (IOOA) was developed in the literature²¹. To rectify the algorithm's flaws, including its constrained accuracy, slow convergence rate, and susceptibility to local optima, and performance that depends on parameter selection during the optimization process, literature²² developed an enhanced seagull optimization method with multi-strategy merging.

In practical engineering, swarm intelligent optimization algorithms have been used. For example, Jin et al.²³ combined the Boids model inspired by the bird swarm algorithm with deep reinforcement learning in the UAV pursuit task. Zhu et al.²⁴ proposed an IM algorithm based on Phase Evaluation Enhancement (PHEE) for social network analysis. Sun et al.²⁵ employed a BFC deployment optimization algorithm that is based on breadth-first search to determine the shortest path between the source node and the destination node. They subsequently verified that the algorithm is optimized in terms of end-to-end latency and bandwidth resource consumption.

Asha et al.²⁶ modified the parameters of the recurrent neural network (RNN) utilizing the enhanced Honey Badger Algorithm (SA-HBA) to forecast the optimal efficacy of the RNN. And Gai et al.²⁷ developed a modified YOLO-V4 deep learning method for the detection of cherry fruits. Saravanan et al.²⁸ suggested an improved technique for scheduling efficiency to resolve the job scheduling issue in cloud computing.

The main structure of this paper is as follows: Chapter 2 is a literature review, Chapter 3 introduces the basic BKA optimization process, and chapter 4 introduces the enhanced DKCBKA method. As part of the improvement strategy, 1) update the probability factor P_s , introduce dynamic exponential factor, and integrate Osprey optimization algorithm (OOA)²⁹ to improve the attack behavior; 2) Stochastic differential variance strategy is introduced in the migration stage to improve the overall optimization ability; 3) The accuracy and convergence speed of the algorithm are improved by using crossbar method. Horizontal crossing improves the global search ability of the method, while vertical crossing enables the system to escape local optimization. In Chapter 5, the optimization performance of several improvement strategies is evaluated and compared using a total of 15 benchmark functions. The test assesses performance metrics including ideal values, mean values, standard deviations, and convergence curves. In Chapter 6, using the CEC 2017 test function set with 19 functions and the CEC 2019 test function set with 10 functions, the effectiveness of the DKCBKA method is evaluated by comparing it with 6 enhancement algorithms and 5 swarm intelligence optimization algorithms. In the engineering application part of Chapter 7, three traditional engineering situations are used to evaluate the feasibility of the proposed algorithm DKCBKA in the actual engineering environment. Chapter 8 is the discussion and chapter 9 is the conclusion.

Literature review

Overview of existing research

The Black-winged Kite Algorithm (BKA)³⁰, a novel swarm intelligence optimization technique, was initially introduced by Jun Wang et al. and drew inspiration from the natural behaviors and hunting strategies of black-winged kite, and is characterized by strong adaptability, few adjustable parameters, and high convergence accuracy. Despite combining the leader approach and the Cauchy variation method, BKA performs well in determining the best function optimization when compared to other intelligent optimization algorithms³¹. BKA still has a lot of flaws, though. First, there are issues like low population variety, a lackluster ability to search globally, and a tendency to rely too much on local optimization in the end, among other things.

To address the aforementioned issues, Zhang et al.³² suggested a method that integrates a population initiated by logistic chaos map with the osprey optimization algorithm, which altered the initialized population's random distribution, enhanced population diversity, and sped up the program's rate of convergence; Xue et al.³³ integrated the BKA approach with the artificial rabbit optimization algorithm to maximize the advantages of the two algorithms for collaborative search. They utilized the master-slave model technique and incorporated effective point sets to initialize the population, hence enhancing the algorithm's search efficiency and optimizing its performance; Fu et al.³⁴ introduced an improved black-winged kite optimization method (IBKA) by substituting the Gompertz growth model for the attack phase parameter n . This algorithm achieves a balanced approach between local and global search, while exhibiting a reduced rate of step decay; Mu et al.³⁵ used enhanced Circle mapping, fused hierarchical reverse learning, and introduced the Nelder—Mead method to improve the BKA, which improved the optimization performance of the original algorithm; Zhao et al.³⁶ proposed an improved black-winged kite algorithm based on chaotic mapping and adversarial learning, which improves the optimization speed and accuracy of the original algorithm.

This work suggests a black-winged kite approach that integrates multi-strategy improvement (DKCBKA) to address these drawbacks of BKA.

A comparative analysis of this study and recent developments

In this section, we compare and analyze the algorithm DKCBKA proposed in this paper with recent relevant studies, including IM algorithm based on phase evaluation enhancement (PHEE), DRL method of pursuit based on Boids model (BOIDS-PE), Improved SADEKTS algorithm (ISADEKTS)³⁷, optimization algorithm based on breadth-first search (SFCDO) and an improved heuristic algorithm combining Clarke–Wright saving algorithm and adaptive large neighborhood search, and introducing Q learning to adjust operator weights (IHA)³⁸. The comparison dimensions are algorithm improvement strategies, application problems, advantages and limitations respectively, and the results are shown in Table 1.

Black-winged kite optimization algorithm

Black-winged kites nest in wide fields with trees and bushes, farmland, sparse woodlands, and grasslands. They eat field mice, insects, small birds, rabbits, and reptiles. Their hunting style consists primarily of perching on telephone poles and tall trees, waiting for passing birds and insects, and then diving down to catch them. Another strategy is to silently circle, glide, and monitor the ground for hours in the skies before diving down to catch their prey when it appears. Black-winged kites migrate to their breeding areas in the spring (April–May) and leave in the autumn (October–November). An algorithm model inspired by black-winged kites’ hunting abilities and migrating behaviors has been developed.

Initialization stage

The Black-winged Kite algorithm, like other methods, uses a random initialization mechanism.

$$X_i = BK_{lb} + rand \left(BK_{ub} - BK_{lb} \right)$$
 (1)

where rand represents an arbitrary value within the range of [0, 1], i is an integer within the range of 1 to N, X_i is the starting black-winged kite set, and BK_{lb} and BK_{ub} denote the lower and higher limits of the i -th black-winged kite in the j -th dimension, respectively.

Attacking behavior

Birds typically act aggressively because they need to defend their area and for self-defense. They might squawk, peck, and flap their wings in an aggressive manner when they feel threatened. During flight, black-winged kites modify the angle of their wings and tails to suit the speed of the wind. They then hover silently to examine their prey before swiftly diving and attacking. The black-winged kite’s assault behavior can be mathematically modelled as follows:

When $p < r$, at a great pace, the black-winged kite charges at its target, and the position update equation is

$$X_{t+1} = X_t + n \left(1 + \sin \left(r \right) \right) \times X_t$$
 (2)

When $p > r$, the most recent formula for the black-winged kite’s attack state position when it is hovering in midair is

$$X_{t+1} = X_t + n \left(2r - 1 \right) \times X_t$$
 (3)

$$n = 0.05 \times e^{-2 \times \left(\frac{t}{T} \right)^2}$$
 (4)

Contrast dimension	DKCBKA	PHEE	Boids-PE	ISADEKTS	SFCDO	IHA
Improvement technique	Integration of Osprey optimization algorithm, random difference variation and crossbar	Evolutionary algorithm based on random range partitioning, fast convergence strategy	Combine the advantages of bio-inspired algorithms and swarm intelligence of deep reinforcement learning	Knowledge transfer technique	Service function chain deployment optimization algorithm based on breadth-first search	Clarke–Wright saving algorithm and Q Learning dynamic adjustment
Application problem	High dimensional nonlinear constraints	Social network	Drone tracking problem	Inflatable wing problem	Optimization of service function deployment with low latency and high resource efficiency in network function virtualization	Dynamic pickup and delivery problem
Advantage	The convergence speed is fast and the convergence precision is high when dealing with practical problems	The convergence speed is fast and the ability to jump out of local optimal is enhanced	Improvement of swarm intelligence algorithm combined with deep reinforcement learning	Improve optimization efficiency and design accuracy	High resource efficiency, network load balancing, improve network reliability	The improved algorithm has efficient initial solution and great potential for practical application
Limitation	The computational cost increases slightly in higher dimensional problems	Overreliance on the neighbor structure of a vertex in social networking applications	Computation costs are high in high dimensional problems	Parameter sensitivity analysis was not performed	The algorithm complexity increases significantly as the network scale increases	Model extensibility limitation

Table 1. Comparison of DKCBKA with recent studies.

where the locations of the black-winged kite in the t -th and $(t + 1)$ -th iterations are indicated by X_t and X_{t+1} , respectively, p is an integer valued at 0.9, and r is a stochastic constant inside the interval $[0, 1]$. T represents the maximum amount of iterations, whereas t is the amount of iterations that have been completed to date.

Migration behavior

Bird migration represents an instinctual behavior in which birds navigate according to environmental cues. The migration behavior of BKA integrates Cauchy's mutation with the leader's strategy: The leader relinquishes their position and integrates into the migrating population if the current group's fitness value is inferior to that of a random population. The current population will maintain its leadership and guidance until it reaches its objective, provided its fitness value exceeds that of the random population. The following mathematical model illustrates the BKA's dynamic migration conduct:

$$X_{t+1} = \begin{cases} X_t + C(0,1) \times (X_t - L_t) & F_i < F_{ri} \\ X_t + C(0,1) \times (L_t - m \times X_t) & \text{else} \end{cases} \quad (5)$$

$$m = 2 \times \sin\left(r + \frac{2}{\pi}\right) \quad (6)$$

L_t symbolizes the commander of the black-winged kite during the t -th iteration to date. X_t and X_{t+1} reflect the location of the black-winged kite during the t -th and $(t + 1)$ -th iterations, respectively. F_i denotes the fitness score of an individual in the t -th iteration. F_{ri} reflects the fitness value of a black-winged kite during the t -th iteration. $C(0, 1)$ symbolizes the Cauchy mutation.

Two parameters define the continuous probability distribution known as the one-dimensional Cauchy distribution. The density function of probability of the a one-dimensional Cauchy distribution is shown by the given formula.

$$f(x, \delta, \mu) = \frac{1}{\pi} \frac{\delta}{\delta^2 + (x - \mu)^2}, -\infty < x < +\infty \quad (7)$$

The probability distribution function acquires standard form when $\delta = 1$ and μ equals 0. The following are the formulas that follow:

$$f(x, \delta, \mu) = \frac{1}{\pi} \frac{\delta}{x^2 + 1}, -\infty < x < +\infty \quad (8)$$

Enhanced optimization algorithm for the black-winged kite

The basic black-winged kite optimization algorithm exhibits limitations in its attack phase, where the dynamic selection strategy does not fully enhance adaptive capability. Furthermore, the algorithm's global search ability is insufficient, resulting in a propensity to converge on local optima. Additionally, the position update formula for alternative members of the population generates new candidates in proximity to the current individual and the best individual, potentially diminishing population diversity and contributing to local optima entrapment and reduced convergence efficiency. Building on previous research, this work suggests a fusion multi-strategy black-winged kite optimization methods to improve the analytical algorithm's optimization efficiency. The particular strategies are delineated in the following order:

Dynamic exponential factoring and fusion osprey algorithm strategy

The black-winged kite optimization technique employs a dynamic selection strategy for global search during the attack phase; however, it encounters difficulties in properly balancing worldwide search capabilities with local exploitation, rendering it susceptible to local optimization. To augment the method's improvement efficacy for seeking, this work integrates the Osprey algorithm's method for updating the black-winged kite's place with a dynamic index factor and selection strategy.

The nonlinear probability element ω is introduced to modify the original linear segmentation factor ω , as expressed below:

$$\omega = -e^{\left(1 - \frac{t}{T}\right)^3} \quad (9)$$

The dynamic exponential factor is crucial in optimizing the objective function, as an appropriate dynamic factor can enhance algorithm convergence and effectively balance domestic and international search. In the initial phase of iteration, a dynamic exponential factor is incorporated into the positional iteration formula to improve the targeted search capabilities of the method. This adjustment facilitates effective exploration of the ideal outcome, compensates for the initial limitations in local optima, and improves overall search efficacy in subsequent stages. The formula for the dynamic exponential factor is presented below:

$$\alpha = \left(e^{\left(1 - \left(\frac{t}{T}\right)^2\right)} \right)^{2kt} \quad (10)$$

If k represents a random variable in an exponential distribution, t represents the current number of iterations, and T represents the maximum number of iterations.

During the attack phase, the exploration strategy of the osprey optimization method can expand the target region, inhibit the technique from converging on an individual optimum, and augment the global search capabilities of the BKA.

$$X_{i,j}^{p1} = X_{i,j} + r_{i,j} \cdot (SF_{i,j} - I_{i,j} \cdot X_{i,j}) \quad (11)$$

where $SF_{i,j}$ is the fish chosen from i -th osprey during the j -th dimension, $r_{i,j}$ is an arbitrary value in $[0, 1]$, $I_{i,j}$ is a random value in the collection $\{1, 2\}$, and $X_{i,j}^{p1}$ is the new position of the i -th osprey in the j -th dimension.

A dynamic exponential factor is incorporated into the assault phase of the black-winged kite, which enhances the swarm intelligence algorithm's robustness, global search capability, and convergence speed by dynamically modifying the parameters to prevent premature convergence. Meanwhile, fusing the osprey optimization algorithm in the attack phase improves the fusion of the two swarm intelligence algorithms, which can combine the advantages of each and make up for the shortcomings of a single algorithm, thus enhancing the overall effectiveness. The Black-winged Kite algorithm's precision in searching in the local area can be enhanced by the fusion, which also concentrates on fine search during the attack phase of the osprey optimization algorithm, thereby facilitating the discovery of a superior solution.

The subsequent formula delineates the position upgrade for the enhanced algorithm during the attack phase.

$$X_{t+1} = \begin{cases} \alpha \times X_t + n(1 + \sin(r)) \times X_t & \omega < r \\ X_t + r_{i,j} \cdot (SF_{i,j} - I_{i,j} \cdot X_t) & \text{else} \end{cases} \quad (12)$$

Differential variation randomized

An examination of the relocation process of the black-winged kite reveals that the positional adjustment of individuals within the population occurs in proximity to the current individual X_t and the optimal individual L_t , indicating that other individuals are directed to migrate towards the optimal area represented by L_t . When L_t represents a locally optimal solution, the population will likely converge around this solution as iterations advance, resulting in diminished population variety, which may induce untimely convergence and decrease the precision with the algorithm's convergence. To ensure convergence, the population intelligence algorithm implements an optimal preservation strategy following each iteration. To address this challenge, it is customary to implement mutation operations to enhance population variety and prevent convergence to local optima. Literature³⁹ introduced an enhanced whale optimization algorithm utilizing stochastic differential mutation, drawing inspiration from the differential evolutionary algorithm.

This idea is adopted in this paper by incorporating a stochastic differential mutation strategy during the migration phase. This technique employs the present individual, the most effective private, and randomly selected individuals from the community to generate a new individual. By incorporating this strategy into the migration phase of the black-winged kite algorithm, the search diversity can be enhanced, the global search capability can be enhanced, and the algorithm can be prevented from entering the local optimal state, as illustrated by the subsequent expression:

$$X_{t+1} = r \times (L_t - X_t) + r \times (X'_p(t) - X_t) \quad (13)$$

where X_{t+1} represents a novel entity acquired through stochastic differential evolution, and r is a variable constrained inside the interval of 0 to 1. L_t is the best person's present location. $X'_p(t)$ designates the position of a randomly chosen member from a cohort.

Vertical and horizontal crossover strategy

The black-winged kites in the population exhibit a tendency to congregate around optimal individuals as the amount of repetitions grows. The method's ability to find the best solution is hampered by this trend, which gradually reduces population variety. This paper presents a crossover strategy that incorporates both vertical and horizontal elements, specifically detailing horizontal and vertical crossover methods. In the literature^{40,41}, the vertical and horizontal crossover technique serves to enhance the method's convergence efficiency and international inquiry capabilities while reducing local optimality. The horizontal crossover operation facilitates an expansion of the search range and diminishes search blind spots, thereby further improving the algorithm's international inquiry ability. Premature convergence in many population-based intelligent search algorithms frequently results from the stagnation of certain dimensions within the population. By making it easier to activate these stagnant dimensions, vertical crossover enables the algorithm to circumvent local optimal solutions.

Crossover strategy can provide a better balance between exploration and development of algorithms, and effectively improve the convergence accuracy. Applying it after the migration phase can help the algorithm explore new regions while also digging deeper into areas of potential that have been found. In addition, the crossover operation effectively enhances population diversity. The vertical and horizontal crossover strategy incorporates a competitive mechanism that facilitates population updates by evaluating the relative strengths and weaknesses of offspring compared to their parents. Sequential execution of horizontal and vertical crossover improves the algorithm's convergence speed and solution correctness. A quick explanation of these two crossover processes is given below.

A transverse crossover is an arithmetic crossover that involves two separate persons across all dimensions. Members of the population are originally partnered randomly, followed by the execution of a transverse crossover between the two paired individuals, as represented by the following equation.

$$X_{i,d}^{hc} = a_1 * X(i, O) + (1 - a_1) * X(j, O) + c_1 * (X(i, O) - X(j, O)) \quad (14)$$

$$X_{j,d}^{hc} = a_2 * X(j, O) + (1 - a_2) * X(i, O) + c_2 * (X(j, O) - X(i, O)) \quad (15)$$

where a_1, a_2 are $[0,1]$ random numbers, c_1, c_2 are $[-1,1]$ random numbers, $X(i, O)$ and $X(j, O)$ are the parent individuals after pairing of $X(i)$ and $X(j)$ in the d -dimension, and $X_{i,d}^{hc}$ and $X_{j,d}^{hc}$ stand for the d -dimensional $X(i, O)$ and $X(j, O)$ produced by lateral crossover, respectively. offspring. Individuals with lower objective function values are kept after the produced children are compared to their parents.

A longitudinal crossover refers to a crossover variant involving all individuals across two distinct dimensions. The BKA algorithm may demonstrate a propensity to converge to a local optimal value in subsequent iterations, typically resulting from the formation of a local optimum in a specific dimension during the update process. Each individual executes a longitudinal crossover to modify a single dimension while maintaining the integrity of the other dimensions. This approach allows the stagnant dimension to escape a local optimum without compromising the potential of another dimension that may also represent a local optimum. Two distinct dimensions, d_1 and d_2 , are randomly chosen for longitudinal crossover to produce offspring using the subsequent formula:

$$X_{i,D_1}^{vc} = b * X(i, D_1) + (1 - b) * X(j, D_2) \quad (16)$$

where b is a stochastic variable inside the interval $[0,1]$ and X_{i,D_1}^{vc} is the parent generating offspring by longitudinal crossover in D_1 and D_2 dimensions. Individuals exhibiting lower objective function values are retained following the comparison of the offspring with their progenitors.

DKCBKA implementation procedure and pseudo-code

The pseudo-code and complete implementation procedure of DKCBKA are detailed below, and Fig. 1 illustrates the algorithm's flowchart.

Step 1 Initialize the location of individual black-winged kites, within the search range, randomly generate an $N \times D$ dimensional matrix to store the place information of the population; set the algorithm related parameters, assigning values to the population size N , the maximum iterations T , the dimensionality D , and the population search boundaries lb , ub , and so on.

Step 2 Compute and order the fitness X_t of every individual within the population. and record the current best fitness value noted as L_t .

Step 3 update the parameters ω and α according to Eqs. (9) and (10).

When $\omega < \text{rand}$, a dynamic exponential factor is introduced to update the black-winged kite position throughout the attack stage according to the improved position update formula (12).

When $\omega > \text{rand}$, the position update formula (12) of the fundamental osprey optimization algorithm is used to revise the location of the black-winged kite; the fitness value of the black-winged kite is computed, and a determination is made regarding the revision of the desired position and target fitness score.

Step 4 Update the black-winged kite migration stage position in accordance with the judgment condition of Eq. (5).

Step 5 Update the black-winged kite position in accordance with the random difference variant of Eq. (13); calculate the black-winged kite fitness value and determine whether to update the target position and target fitness value.

Step 6 Select random positional crossover mutations to update black-winged kite positions according to the longitudinal crossover Eqs. (14)–(16).

Step 7 Ascertain the necessity of updating the optimal position by calculating the fitness score.

Step 8 Evaluate whether the algorithm satisfies the termination criterion; if it does, terminate the main iteration and display the target place and target value; if not, back to step 3.

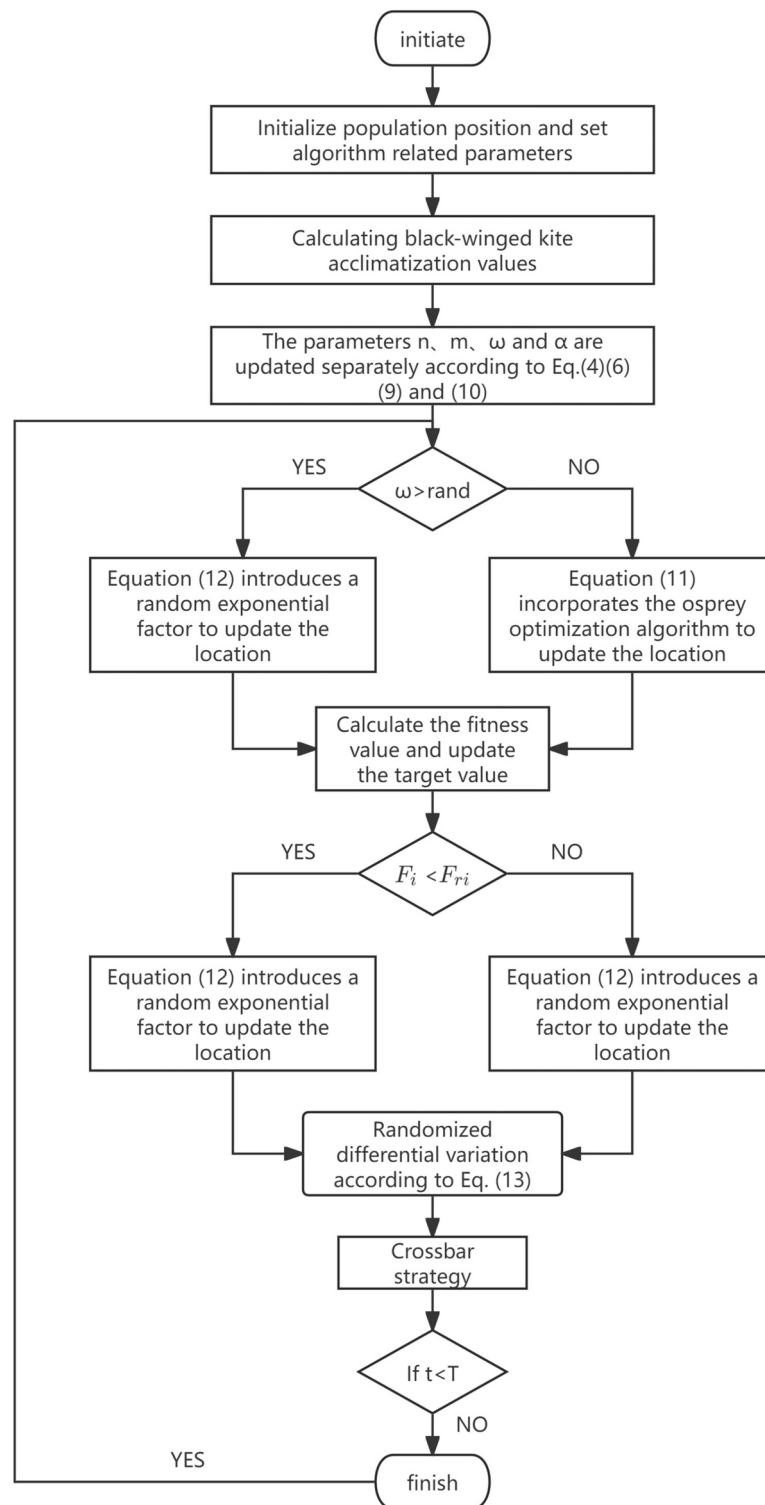


Fig. 1. DKCBKA flow chart.

Input: The variable dimension D , maximum number of iterations T , and population size N

Output: Best Fitness BKA, Best Position BKA, Convergence Curve

```

1: Initialize the black-winged kite population
2: Determine each black-winged kite's fitness value.
3: Initialize  $\omega, \alpha, n, m$ 
4: While ( $t < T$ ) execute
5:   if  $\omega < r$ 
6:     Adjust the position in accordance with Eq. (12)
7:   else if do
8:     Adjust the position in accordance with Eq. (11)
9:   end if
10:  if ( $F_t < F_{rt}$ ) do
11:    Adjust the position in accordance with Eq. (5).
12:  else if do
13:    Adjust the position in accordance with Eq. (5).
14:  if  $\text{rand} > 0.5$ 
15:    Update the position in accordance with Eq. (13)
16:  end if
17:  In accordance with Equations (14), (15), and (16), the crossbar approach is implemented.
18: end for
19: if  $X_t < L_t$  do
20:   Update the optimal black-winged kite
21: end if
22:  $t = t + 1$ 
23: End while
24: Provide  $X_{best}$  and  $F_{best}$ 

```

Algorithm 1. Improved the pseudocode of black-winged kite algorithm

Analysis of the DKCBKA algorithm's time complexity

This paper examines the temporal complexity of the DKCBKA algorithm. Population initialization, location updates, fitness assessment during iterations, and vertical and horizontal crossover operations are the main parts for the method. The initialization process exhibits a temporal challenge of $O(N \times D)$, with N representing the group's size and D denoting the problem's dimension. In each iteration, the time complexity for computing the fitness values within the population is $O(N \times D)$. The dynamic exponential factor in the attack strategy is enhanced based on the original linear factor. The original attack phase formula is replaced by the Osprey optimization algorithm's global search technique. Both modifications maintain the algorithm's temporal challenge at $O(N \times D)$. Additionally, the temporal challenge of the stochastic difference variant during the migration phase is $O(N \times D)$. The longitudinal and transversal crossover operation is segmented into two components, horizontal and vertical, with respective time complexities of $O(N/2 \times D)$ and $O(N \times D)$. The overall temporal challenge for each iteration is $O(2.5 \times N \times D)$. The algorithm executes T iterations, resulting in a temporal challenge of $O(2.5 \times T \times N \times D)$. The simplified time challenge is typically expressed as $O(T \times N \times D)$, disregarding the constant factor. The DKCBKA algorithm described in this study has a time complexity that is comparable to the BKA algorithm.

Algorithm abbreviation

The acronyms for each method and comparison algorithm used in this study are displayed in Table 2.

Analysis of experimental simulation results

Experimental environment

The computer configuration employed for the simulation tests in this study is an Intel(R) Core(TM) i5-10210U processor operating at 1.60 GHz, with 12.0 GB of RAM, running the Windows 11 operating system, and utilizing the Matlab2020(b) computational environment.

Short form of algorithm	Name of the algorithm or its policy
DBKA	BKA with the addition of dynamic exponential factor and fusion of osprey algorithm strategy
KBKA	BKA with the addition of stochastic differential variance strategy
CBKA	BKA with the addition of longitudinal and transversal crossover strategy
DKCBKA	Black-Winged Kite Optimization Algorithm Enhanced by Osprey Optimization and Vertical and Horizontal Crossover Improvement
GSABO	Subtractive optimizer algorithm incorporating golden sines
ECWOA	Whale optimization algorithm based on elite dyadic and cross-optimization
EOSMICOA	Multi-strategy chimpanzee optimization algorithm and its application to engineering problems
IGWO	Nonlinear parametric grey wolf optimization algorithm based on elite learning
CWXSCSO	Sand cat swarm optimization algorithm combining elite decentralization and crossover strategy and its application
IWKGJO	Improved golden jackal optimization algorithm based on hybrid strategy
GO	Goose algorithm
NOA	Nutcracker optimization algorithm
PO	Parrot optimization algorithm
SOA	Seagull optimization algorithm
SSA	Sparrow search algorithm

Table 2. Algorithm abbreviation table.

Introduction to test functions

This study evaluates the algorithm’s optimization capabilities and efficacy by conducting simulated tests that employ 15 established benchmark test functions. The standard test functions’ names, dimensions, value ranges, and theoretically ideal values are presented in “[Supplementary information](#)”. The single-peak evaluation functions F1–F7 assess the algorithm’s precision and convergence rate. The multi-peak evaluation functions F8–F13 are designed to evaluate the algorithm’s global search capability. The composite evaluation functions F14 and F15, with a singular global optimum and several local optima, the assessment of the algorithm’s stability and local development capabilities is conducted. N = 50 is the population size, T = 1000 is the maximum amount of iterations, the dimension is D = 30, and 30 iterations are carried out to determine the ideal value, standard deviation, and mean for comparison analysis.

Comparison of optimization performance of each improvement strategy

Three improvement strategies—BKA (DBKA) with the addition of dynamic exponential factor and fusion of osprey algorithm strategy, BKA (KBKA) with the addition of stochastic differential variance strategy, and BKA (CBKA) with the addition of longitudinal and transversal crossover strategy—are employed in this paper to confirm the accuracy and efficacy of the algorithms. BKA, DBKA, KBKA, CBKA, and DKCBKA were evaluated using 15 benchmark functions exhibiting various optimization characteristics. To assess how well the aforementioned algorithms find the best solution, the population size is set at N = 50, the maximum iteration at T = 1000, the dimensionality of a search space is D = 30. The experiment is conducted independently 30 times, with results presented as optimal value, mean, and standard deviation.

The ideal value, the average, and deviation from the mean of DKCBKA at 30 dimensions are substantially superior to those of BKA, according to Table 3. DKCBKA demonstrates convergence to the ideal value in theory in both simple single-peak functions F1–F6 and multiple-peak functions F9 and F11. The mean outcomes closely align with the theoretical desired values of the corresponding functions, exhibiting a standard deviation of 0. This indicates that DKCBKA exhibits superior optimization performance and enhanced robustness. While DKCBKA does not attain convergence to the hypothetical optimal value in F7–F8 and F14–F15, it has the utmost convergence precision when compared to other techniques. The complex multi-peak functions F12 and F13 indicate that the desired value, mean, deviation from the mean of DKCBKA and CBKA outperform the original approach by more than 26 orders of magnitude. This means that the addition of the vertical and horizontal crossover techniques significantly enhances the method’s ability to ascertain this ideal result and avoid neighborhood optima. DKCBKA surpasses the fundamental algorithm regarding optimization capability. The addition of a dynamic index component, together with the osprey optimization algorithm method and stochastic difference variant strategy, significantly enhances both method rate and convergence precision. Additionally, a balance between regional investigation capabilities and worldwide inquiry is made possible by the introduction of longitudinal and transversal crossover tactics. When these three tactics are combined, the original algorithm’s performance in finding the best answers is significantly improved.

Figure 2 illustrates the linear image iteration curves produced by the DKCBKA for functions F1–F2, F5–F7, F9, F11–F13, and F15. This demonstrates that the fitness values derived from the enhanced method at the initial stages of iteration are closely aligned with the optimal fitness values achieved at final convergence. The convergence curves of functions F4, F10, and F14 exhibit rapid convergence, albeit with slight fluctuations. The DKCBKA convergence curve in the F8 function shows several inflection points, suggesting that the suggested strategy can more easily exit the local most effective than the original technique. The global most effective is reached at the quickest rate in the F3 function image, and the DKCBKA’s convergence rate is noticeably higher than the basic BKA’s. The DKCBKA clearly shows the best optimization performance out of all the function images in Fig. 2.

Function	Algorithm	Optimal value	Mean value	Standard deviation
F1	BKA	6.5980E-217	3.4498E-193	0
	DBKA	0	0	0
	KBKA	0	2.9596E-86	1.1463E-85
	CBKA	0	0	0
	DKCBKA	0	0	0
F2	BKA	8.3634E-109	2.1247E-92	8.2290E-92
	DBKA	5.9450E-306	4.2284E-263	0
	KBKA	0	3.4177E-56	9.0245E-56
	CBKA	0	0	0
	DKCBKA	0	0	0
F3	BKA	2.7378E-216	1.5361E-180	0
	DBKA	0	0	0
	KBKA	0	2.4868E-114	9.6024E-114
	CBKA	0	0	0
	DKCBKA	0	0	0
F4	BKA	1.2149E-107	5.4117E-96	2.0959E-95
	DBKA	0	3.8028E-268	0
	KBKA	0	1.1303E-55	3.8704E-55
	CBKA	3.4314E-102	1.4802E-96	4.0434E-96
	DKCBKA	0	0	0
F5	BKA	2.4200E+01	2.6490E+01	1.4657E+00
	DBKA	2.8840E+01	2.8873E+01	2.2495E-02
	KBKA	2.5149E+01	2.6451E+01	1.1641E+00
	CBKA	0	0	0
	DKCBKA	0	0	0
F6	BKA	3.4484E-05	4.5658E-01	1.3162E+00
	DBKA	3.3608E+00	4.6133E+00	7.2642E-01
	KBKA	4.0095E-05	1.8233E-01	5.3615E-01
	CBKA	0	0	0
	DKCBKA	0	0	0
F7	BKA	1.0525E-05	8.9049E-05	8.0599E-05
	DBKA	1.3999E-06	2.8804E-05	2.1090E-05
	KBKA	7.9768E-06	1.9858E-04	2.9379E-04
	CBKA	1.7795E-06	7.2265E-06	4.9318E-06
	DKCBKA	4.9317E-07	2.1098E-06	1.7472E-06
F8	BKA	-1.1260E+04	-9.0892E+03	1.4812E+03
	DBKA	-8.9139E+03	-7.1840E+03	1.1355E+03
	KBKA	-1.2316E+04	-1.0478E+04	1.0622E+03
	CBKA	-1.2569E+04	-1.2569E+04	1.8828E-12
	DKCBKA	-1.2569E+04	-1.2569E+04	1.8828E-12
F9	BKA	0	0	0
	DBKA	0	0	0
	KBKA	0	0	0
	CBKA	0	0	0
	DKCBKA	0	0	0
F10	BKA	4.4409E-16	4.4409E-16	0
	DBKA	4.4409E-16	4.4409E-16	0
	KBKA	4.4409E-16	4.4409E-16	0
	CBKA	4.4409E-16	4.4409E-16	0
	DKCBKA	4.4409E-16	4.4409E-16	0
F11	BKA	0	0	0
	DBKA	0	3.4793E-04	1.8921E-03
	KBKA	0	0	0
	CBKA	0	0	0
	DKCBKA	0	0	0
Continued				

Function	Algorithm	Optimal value	Mean value	Standard deviation
F12	BKA	2.3466E-06	4.5571E-02	9.2764E-02
	DBKA	1.0802E-01	3.5247E-01	1.1289E-01
	KBKA	2.1041E-06	3.9427E-02	6.7684E-02
	CBKA	1.5705E-32	1.5705E-32	2.8330E-48
	DKCBKA	1.5705E-32	1.5705E-32	2.8330E-48
F13	BKA	2.3359E-01	1.2791E+00	8.1901E-01
	DB.KA	1.4705E+00	2.3089E+00	4.1883E-01
	KBKA	1.1051E-02	1.2931E+00	8.0404E-01
	CB.KA	1.3498E-32	1.3498E-32	2.8330E-48
	DKCBKA	1.3498E-32	1.3498E-32	2.8330E-48
F14	BKA	9.9800E-01	9.9800E-01	5.9344E-17
	DBKA	9.9800E-01	2.1193E+00	1.6218E+00
	KBKA	9.9800E-01	9.9800E-01	1.4536E-16
	CBKA	9.9800E-01	9.9800E-01	0
	DKCBKA	9.9800E-01	9.9800E-01	0
F15	BKA	3.0749E-04	3.6853E-04	2.3643E-04
	DBKA	3.0749E-04	4.0160E-04	3.0265E-04
	KBKA	3.0749E-04	5.7583E-04	4.1267E-04
	CBKA	3.0749E-04	3.0749E-04	9.8264E-20
	DKCBKA	3.0749E-04	3.0749E-04	9.5006E-20

Table 3. Compares the optimization outcomes of 15 benchmark functions (F1–F15) utilizing different approaches to improvement. Significant values are displayed in bold.

In conclusion, DKCBKA offers numerous advantages for identifying the best possible answer of a function and has superior performance in convergence and optimization search efficiency.

Comparison of DKCBKA with other improved algorithms

The DKCBKA algorithm is evaluated against the initial BKA algorithm and six superior enhanced optimization techniques.to further validate its effectiveness: the sand cat swarm optimization algorithm combining elite decentralization and crossover strategy and its application (CWXS�SO)⁴², the whale optimization algorithm based on elite dyadic and cross-optimization (ECWOA)⁴³, the multi-strategy chimpanzee optimization algorithm and its application to engineering problems (EOSMICOA)⁴⁴, the nonlinear parametric grey wolf optimization algorithm based on elite learning (IGWO)⁴⁵, the subtractive optimizer algorithm incorporating golden sines (GSABO)⁴⁶, and the improved golden jackal optimization algorithm based on hybrid strategy (IWKGJO)⁴⁷. In order to fairly examine the performance of DKCBKA in high dimensions, the total population size for every method in this experiment is established at N=50, iteration T=1000, and dimensions are 30 and 100 dimensions, respectively. Each experiment is carried out independently 30 times. Table 4 displays the parameter configurations of the algorithms that are the subject of this paper.

CEC2017 test function information

Nineteen of the twenty-nine functions in the CEC2017 test set were chosen to assess DKCBKA’s optimization performance. The algorithm’s speed and efficiency in identifying a singular optimal solution can be accurately assessed, as F1 and F3 are single–peak operates with one ideal solution. Conversely, the straightforward multi-peak functions F4–F10, that have numerous local optimum answers, are employed in evaluating the method’s capacity to evade local optima. The algorithm’s ability to address more complex optimization challenges is evaluated using the hybrid functions F11–F20, which combine the characteristics of single–peak and multi-peak functions. These functions complicate the optimization of algorithms and evaluate their flexibility and resilience across many contexts. “Supplementary information” presents the essential information regarding the CEC2017 test function.

Analysis of experimental results

Table 5 contains the optimization indexes, which are denoted as min., avg., and std., the optimal value, average value, and standard deviation, respectively. Examining the comparative statistics presented in Table 5 at a dimension of 100 shows that DKCBKA outperforms the original BKA algorithm and performs better regarding optimization than the other six improved optimization techniques. The DKCBKA shows better optimal and average fitness values in functions C1 and C3, having an average deviation ranking somewhat lower than those of the CWXS�SO and GSABO. However, in functions C4, C11–C15, and C17–C19, the DKCBKA outperforms comparative algorithms, including IWKGJO, across all three optimization indices. The standard deviation of the algorithm described in this study is lower than that of the GSABO in C5, C7, C8, and F20, and it ranks second to the CWXS�SO in C9 and C11, and third in C10. Furthermore, in C16, the IWKGJO outperforms DKCBKA. The proposed approach attains the highest optimum value and a mean fitness score among the given functions. The

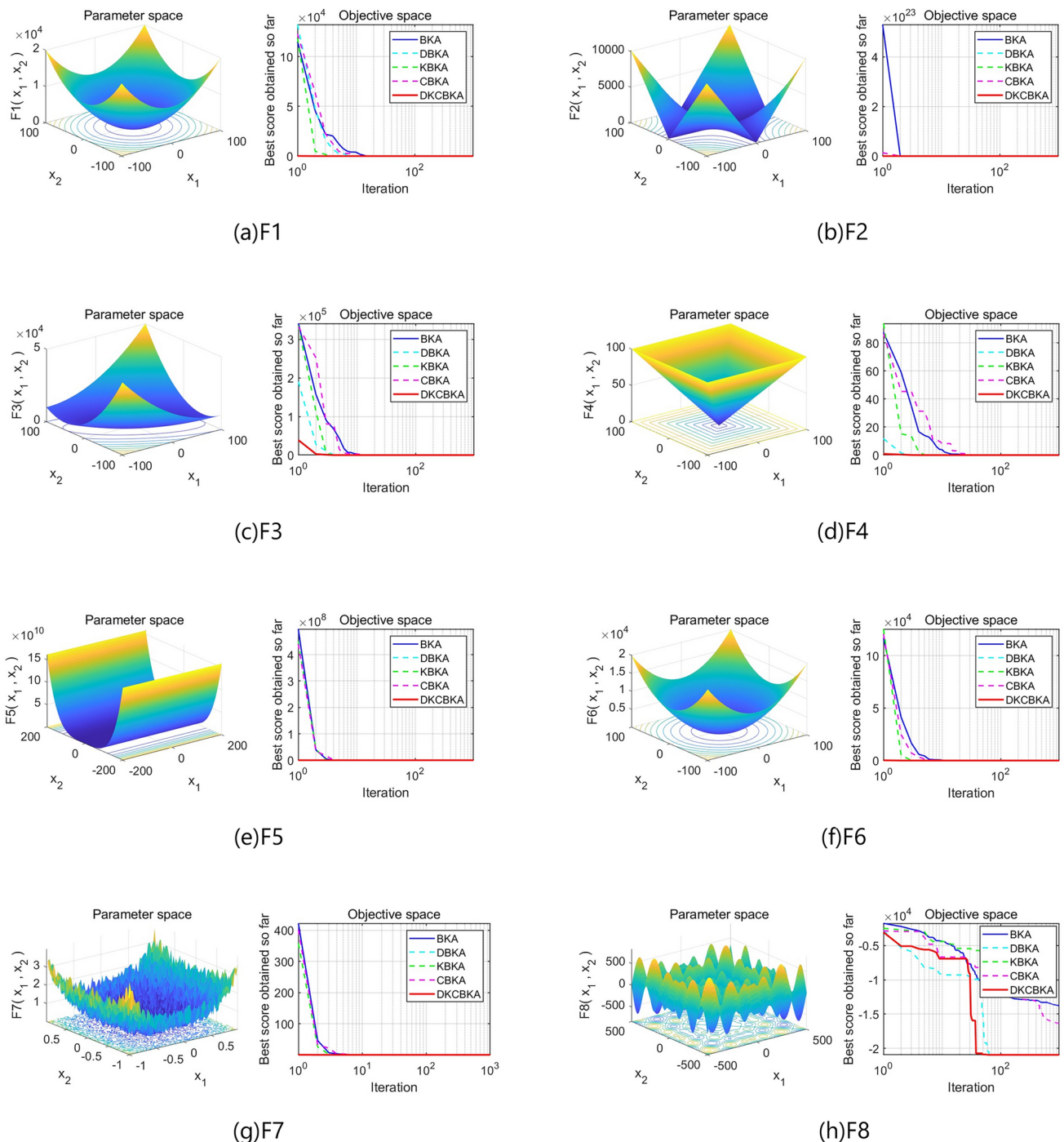
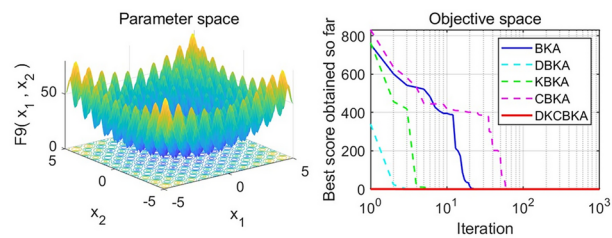


Fig. 2. Convergence graph for comparison between strategies.

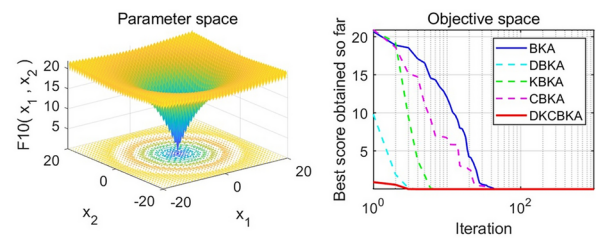
efficiency of the DKCBKA is subpar compared to the ECWOA concerning function C6. The DKCBKA exhibits a comparative advantage, signifying its capacity to tackle intricate optimization challenges in high-dimensional spaces, as well as remarkable adaptability and resilience in diverse contexts.

The 100-dimensional convergence image in Fig. 3 demonstrates that the DKCBKA has greater convergence accuracy compared to advanced methods such as the IWKGJO, particularly in functions C1–C20. In contrast to previous comparison algorithms and the initial method, the DKCBKA can converge rapidly before 200 iterations in C5–C10, C13, C15–C17, and C19–C20. Additionally, each function's convergence curves experience slight oscillations during the pre-convergence phase, demonstrating the algorithm's capacity to diverge from its regional optimum.

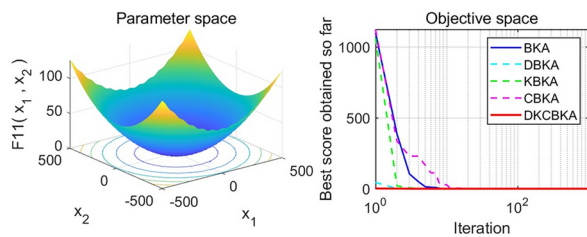
The optimization results of the DKCBKA on 19 CEC2017 test functions demonstrate enhanced performance relative to the initial method and six other advanced sophisticated algorithms for solution precision, convergence velocity, and stability.



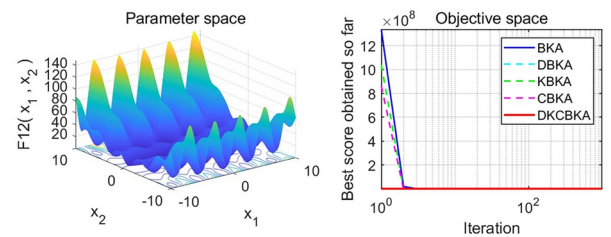
(i)F9



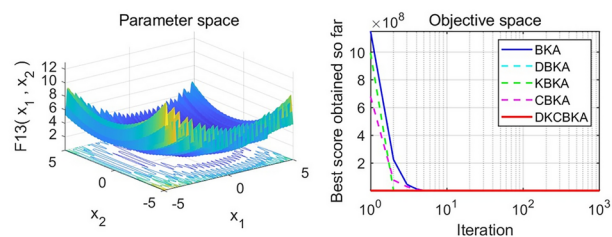
(j)F10



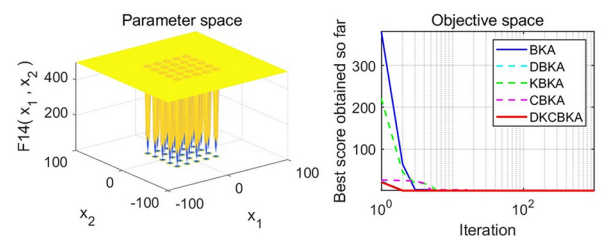
(k)F11



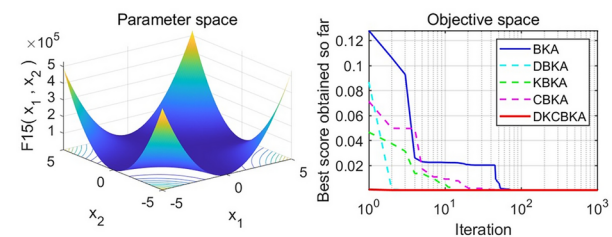
(l)F12



(m)F13



(n)F14



(o)F15

Figure 2. (continued)

Algorithm	Basic parameter
CWXSCSO	$S = 2$
ECWOA	$a_{min} = 0, a_{max} = 2, b = 1$
EOSMICOA	$b_1 = 0.5, b_2 = 0.5$
IGWO	$a_{initial} = 2, a_{final} = 0$
GSABO	$a = -\pi, b = \pi, r \in [0, 1]$
IWKGJO	$b = 1$

Table 4. Algorithmic parameter.

Function	optimization indicator	DKCBKA	BA	GSABO	ECWOA	EOSMICOA	HMSWHO	CWXSCSO	IWKGJO
C1	Min.	5.7951E+04	5.5498E+10	2.5547E+11	1.5042E+06	1.0947E+11	2.4602E+11	1.0711E+06	1.1164E+09
	Avg.	1.3115E+06	7.9798E+10	2.6020E+11	2.5424E+06	1.2533E+11	2.6731E+11	1.8218E+06	2.0713E+09
	Std.	2.0955E+06	1.7476E+10	2.8484E+09	9.8531E+05	1.1213E+10	8.4158E+09	5.4074E+05	5.4704E+08
C3	Min.	1.4213E+05	1.6284E+05	3.5034E+05	3.0830E+05	3.5591E+05	3.0250E+05	1.9499E+05	2.1841E+05
	Avg.	1.6799E+05	2.0378E+05	3.5230E+05	5.8768E+05	4.6775E+05	3.8320E+05	2.3631E+05	2.6055E+05
	Std.	1.7924E+04	5.8775E+04	1.4734E+03	1.5686E+05	7.0277E+04	1.2905E+05	2.5165E+04	2.4387E+04
C4	Min.	5.4445E+02	4.7989E+03	9.2752E+04	7.0011E+02	1.3416E+04	9.4196E+04	6.9452E+02	1.1980E+03
	Avg.	6.1451E+02	1.5664E+04	1.0321E+05	8.2081E+02	1.7246E+04	1.0701E+05	7.9724E+02	1.5897E+03
	Std.	5.6346E+01	1.7244E+04	5.4770E+03	7.4968E+01	2.6537E+03	7.5160E+03	6.1346E+01	2.5155E+02
C5	Min.	1.1391E+03	1.2912E+03	2.0476E+03	1.2014E+03	1.6856E+03	2.0858E+03	1.2513E+03	1.1503E+03
	Avg.	1.2798E+03	1.4177E+03	2.1386E+03	1.2967E+03	1.7812E+03	2.1535E+03	1.3512E+03	1.3253E+03
	Std.	8.3463E+01	6.9428E+01	3.0610E+01	7.7548E+01	5.6996E+01	4.0107E+01	4.0851E+01	1.1937E+02
C6	Min.	6.0428E+02	6.6814E+02	7.1035E+02	6.0059E+02	6.8182E+02	7.1288E+02	6.0463E+02	6.4647E+02
	Avg.	6.1315E+02	6.7900E+02	7.1619E+02	6.0112E+02	6.8880E+02	7.1789E+02	6.2846E+02	6.5705E+02
	Std.	6.4317E+00	1.2915E+01	2.8643E+00	4.0744E-01	3.8940E+00	2.5452E+00	1.6038E+01	8.4869E+00
C7	Min.	1.8639E+03	3.0630E+03	4.0462E+03	2.1235E+03	2.8761E+03	3.9468E+03	2.4509E+03	2.1602E+03
	Avg.	2.2650E+03	3.2932E+03	4.1078E+03	2.5306E+03	3.1442E+03	4.0499E+03	2.9843E+03	2.4538E+03
	Std.	2.6565E+02	2.0484E+02	2.7317E+01	2.5237E+02	1.4846E+02	4.9442E+01	2.3062E+02	1.8999E+02
C8	Min.	1.4658E+03	1.7562E+03	2.5986E+03	1.4828E+03	2.0195E+03	2.5701E+03	1.6233E+03	1.4936E+03
	Avg.	1.6240E+03	1.8995E+03	2.6452E+03	1.6909E+03	2.0841E+03	2.6270E+03	1.7779E+03	1.6832E+03
	Std.	8.6512E+01	1.5930E+02	1.7472E+01	9.0720E+01	4.7750E+01	2.6628E+01	7.7706E+01	1.4195E+02
C9	Min.	1.8326E+04	2.6471E+04	6.5870E+04	2.0510E+04	5.5637E+04	8.0407E+04	2.1569E+04	2.6893E+04
	Avg.	2.1976E+04	3.2320E+04	7.6333E+04	2.5591E+04	6.9348E+04	8.6990E+04	2.4018E+04	4.6272E+04
	Std.	1.4822E+03	9.8866E+03	7.7284E+03	5.3322E+03	5.3106E+03	3.6927E+03	1.3674E+03	1.3555E+04
C10	Min.	1.0816E+04	1.6176E+04	3.1288E+04	1.2038E+04	3.1183E+04	3.0706E+04	1.2856E+04	1.6953E+04
	Avg.	1.3602E+04	2.0200E+04	3.3536E+04	1.6206E+04	3.2291E+04	3.3348E+04	1.4706E+04	2.3465E+04
	Std.	1.2045E+03	4.4992E+03	1.0575E+03	1.6756E+03	4.7514E+02	7.6274E+02	1.0086E+03	4.7599E+03
C11	Min.	2.4238E+03	1.1282E+04	1.8305E+05	1.3849E+04	7.6453E+04	1.7023E+05	7.5182E+03	2.2173E+04
	Avg.	3.6577E+03	4.7694E+04	2.1070E+05	3.5413E+04	1.0664E+05	2.3386E+05	1.6219E+04	3.2109E+04
	Std.	8.2261E+02	4.7586E+04	3.2602E+04	1.2814E+04	1.7334E+04	7.4458E+04	4.9685E+03	7.7774E+03
C12	Min.	2.1157E+06	5.3289E+09	2.2186E+11	2.3018E+07	2.7477E+10	1.8340E+11	1.2960E+07	3.9690E+08
	Avg.	5.9175E+06	2.7325E+10	2.2946E+11	5.4718E+07	4.3306E+10	2.0307E+11	3.6590E+07	9.6555E+08
	Std.	3.6502E+06	4.2645E+10	5.1416E+09	3.1603E+07	7.9921E+09	9.4418E+09	2.0226E+07	3.9644E+08
C13	Min.	2.0417E+03	1.1404E+07	6.0791E+10	2.3405E+04	5.0955E+09	4.5482E+10	1.3819E+04	3.4793E+06
	Avg.	8.5128E+03	7.9331E+08	6.0821E+10	1.7311E+05	8.1272E+09	4.9799E+10	3.8894E+04	4.3847E+07
	Std.	7.5248E+03	1.2390E+09	7.2619E+07	1.7568E+05	2.5803E+09	2.7300E+09	1.8915E+04	5.0201E+07
C14	Min.	4.7733E+04	3.9883E+05	4.7182E+07	2.3018E+06	8.2636E+06	4.7307E+07	1.4147E+06	4.2037E+06
	Avg.	1.5461E+05	6.1716E+06	7.4098E+07	5.8920E+06	1.1828E+07	1.1334E+08	3.1199E+06	7.6766E+06
	Std.	8.8617E+04	1.0467E+07	5.6269E+07	2.7451E+06	2.7100E+06	5.1557E+07	2.0119E+06	2.4732E+06
C15	Min.	1.9276E+03	6.8046E+05	3.4475E+10	1.1941E+04	1.3816E+09	2.1316E+10	3.7900E+03	1.3426E+06
	Avg.	2.7193E+03	2.8822E+07	3.4631E+10	6.7574E+04	2.6049E+09	2.6081E+10	9.2354E+03	1.2882E+07
	Std.	9.1225E+02	6.8354E+07	5.2971E+07	8.0745E+04	9.5503E+08	2.3933E+09	4.7474E+03	2.4127E+07
C16	Min.	4.2833E+03	6.2721E+03	2.8890E+04	5.2096E+03	1.0014E+04	2.1728E+04	5.5377E+03	6.0961E+03
	Avg.	5.8844E+03	9.9701E+03	2.9707E+04	6.0861E+03	1.1290E+04	2.5561E+04	6.4615E+03	6.7031E+03
	Std.	9.7145E+02	3.8002E+03	5.5213E+02	6.8498E+02	8.1946E+02	1.9287E+03	7.6697E+02	5.0500E+02
C17	Min.	4.0405E+03	5.4653E+03	1.1495E+07	4.4435E+03	8.4989E+03	2.6041E+06	4.4477E+03	5.1891E+03
	Avg.	5.2425E+03	7.6013E+03	1.5185E+07	5.6376E+03	1.0398E+04	1.3192E+07	5.4570E+03	6.2111E+03
	Std.	5.4133E+02	2.1872E+03	6.3936E+06	7.0652E+02	1.4604E+03	7.4668E+06	6.3148E+02	6.5756E+02
C18	Min	1.2777E+05	3.9399E+05	1.5478E+08	1.7932E+06	1.0635E+07	1.2534E+08	1.6758E+06	2.5230E+06
	Avg.	2.3828E+05	8.1834E+06	3.1918E+08	5.0694E+06	2.0665E+07	2.9345E+08	3.5746E+06	7.7768E+06
	Std.	8.9920E+04	1.7738E+07	5.9644E+07	3.1753E+06	6.2207E+06	1.0235E+08	2.4767E+06	4.0074E+06
C19	Min.	2.2445E+03	2.6556E+06	3.6291E+10	1.5149E+04	8.7306E+08	2.0858E+10	4.6965E+03	3.1202E+06
	Avg.	7.8480E+03	1.5053E+07	3.8718E+10	8.2186E+04	1.5580E+09	2.6498E+10	1.3231E+04	1.3180E+07
	Std.	6.6878E+03	1.5078E+07	6.7150E+08	1.2587E+05	5.5380E+08	2.8511E+09	8.0032E+03	8.4490E+06
Continued									

Function	optimization indicator	DKCBKA	BKA	GSABO	ECWOA	EOSMICOA	HMSWHO	CWXSCSO	IWKGJO
C20	Min.	2.0720E+03	2.3193E+03	3.1418E+03	2.1815E+03	2.7771E+03	2.9359E+03	2.2672E+03	2.2062E+03
	Avg.	2.3369E+03	2.5300E+03	3.4628E+03	2.6324E+03	3.0006E+03	3.2541E+03	2.6143E+03	2.4382E+03
	Std.	1.2362E+02	1.6188E+02	9.7950E+01	1.9973E+02	1.3677E+02	1.8025E+02	1.7979E+02	1.7186E+02

Table 5. 100-dimensional optimization metrics for CEC2017.

CEC2017 Wilcoxon rank sum test

The effectiveness and precision of different approaches are illustrated by executing the ideal fitness values, typical fitness values, and average deviations of the seven compared methods independently for 30 iterations. However, this does not confirm whether DKCBKA and the seven algorithms above differ significantly in their capacity to resolve intricate optimization challenges. This section evaluates the optimization effectiveness of the BKA, GSABO, ECWOA, EOSMICOA, IGWO, CWXSCSO, and IWKGJO by applying the DKCBKA to the Wilcoxon rank-sum test for the 19 test functions in the test set CEC2017 in different dimensions. Relevant Wilcoxon rank sum test results and values for P are shown in Table 6. For hypothesis testing, 5% is the crucial amount, which is the significance level. The two comparison methods are thought to differ significantly if the P-value is less than 5%; if it is larger than 5%, the distinction between DKCBKA and the comparison algorithm with regard to of optimization results is not immediately evident. The performance of the DKCBKA is “far superior than,” “the same to,” or “lower to” other comparison algorithms, as shown by the symbols “+”, “=”, and “-”.

From Table 6, comparing the DKCBKA to the GSABO, IGWO, and EOSMICOA, it has a 95% chance of producing a significant difference in the overall optimization results. In C20, it performs marginally inferior than the BKA and IWKGJO algorithms, and in C5 and C8, it performs inferior than the ECWOA and IWKGJO, respectively. The ECWOA and CWXSCSO outperform the DKCBKA in C16 and C17, respectively.

Analysis of CEC2019 test function results

For thoroughly assess the optimization effectiveness of the DKCBKA, we selected 10 CEC2021 test functions characterized by distinct optimization features. “[Supplementary information](#)” displays the fundamental details of the CEC2019 test function. The DKCBKA was compared with five traditional optimization algorithms: GOOSE Algorithm (GO)⁴⁸, Nutcracker Optimization Algorithm (NOA)⁴⁹, Parrot Optimization Algorithm (PO), Seagull Optimization Algorithm (SOA)⁵⁰, and Sparrow Search Algorithm (SSA)⁵¹. The following settings are set: population size N=50, maximum iterations T=1000, and dimensionality D=10. Table 7 shows the parameter settings of the comparison algorithm.

The optimization indexes in Table 8 are respectively the optimal value, average value and standard deviation represented by min, avg and std. Table 8 demonstrates it, relative to six other advanced optimization algorithms, the DKCBKA exhibits enhanced performance in FC1–FC10 and achieves the optimal value. Across all 10 functions, the enhanced method, DKCBKA, has a significantly higher convergence accuracy than the comparison approach. Additionally, the average fitness value ranks first. However, in functions FC2 and FC10, the DKCBKA performs somewhat worse than the SSA approach regarding standard deviation. In FC1, FC3, FC5, and FC9, the performance is notably robust. The SSA outperforms the DKCBKA in FC2 and FC10. The NOA in FC4, FC6, FC7, and FC8 is slightly more effective than the enhanced technique described in this study.

Figure 4 illustrates that the DKCBKA has enhanced optimization performance relative to the other methods. In FC1-FC3, the DKCBKA demonstrates a rapid convergence speed, facilitating its attainment of the global optimum with efficiency. In the FC6, FC9, and FC10 test routines, the initial value of the DKCBKA at the start of the iteration is nearly equivalent to the worldwide ideal In comparison to the other algorithms, the DKCBKA demonstrates superior optimization accuracy, as evidenced by the results from FC4, FC7, and FC8.

In conclusion, the CEC2019 test set demonstrates that the DKCBKA outperforms five traditional optimization algorithms.

CEC2019 Wilcoxon rank sum test

The rank sum check is a quantitative statistical method that is used to determine whether there is a significant variation in the distribution locations of two distinct samples. The rank sum test is effective in assessing performance differences among various algorithms on specific functions in optimization studies of method performance comparison. This section of the article compares the results of the CEC2019 test benchmark. The optimization efficacy of the GO, NOA, PO, SOA, and SSA algorithms is statistically assessed against the performance of the DKCBKA algorithm employing the Wilcoxon rank sum test.

Table 9 indicates that the DKCBKA outperforms both the NOA and SOA, but DKCBKA is inferior to PO and SSA algorithm in FC1, and inferior to PO and GO in the functions FC2 and FC10, respectively, but it surpasses the comparison technique utilized in the other functions, and the optimization performance is substantial. The findings indicate that DKCBKA possesses a substantial optimization superiority compared to the five swarm intelligence methods.

Application examples

Mathematical models and mechanical optimization problems are strongly related, determining the design variables, objective functions, and constraints is essential to developing the optimal design mathematical model. In this section, DKCBKA is chosen to be run 30 times for comparative analysis with WOA⁵², HHO⁵³, GWO⁵⁴,

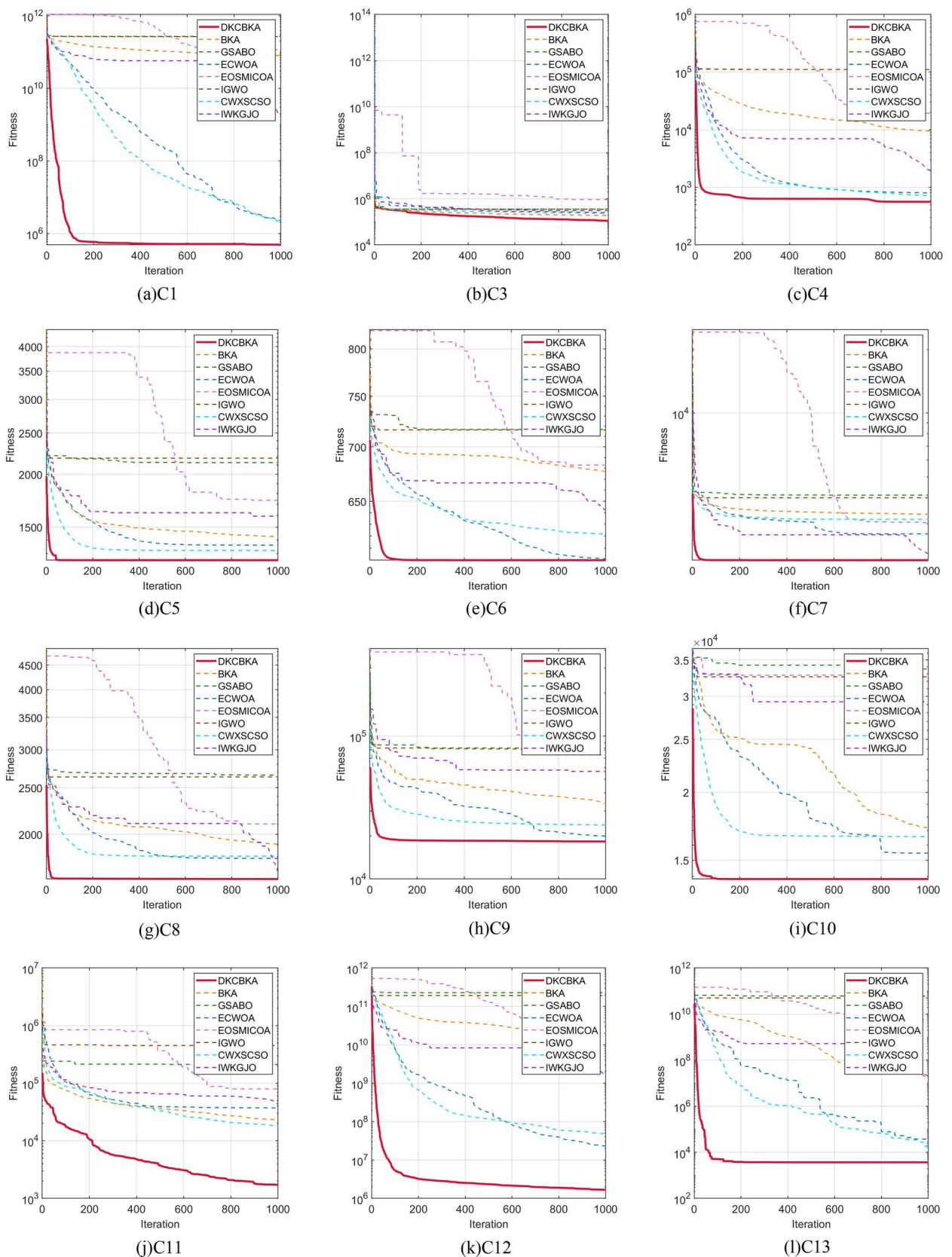


Fig. 3. Convergence curve of the 100-dimensional function for CEC2017.

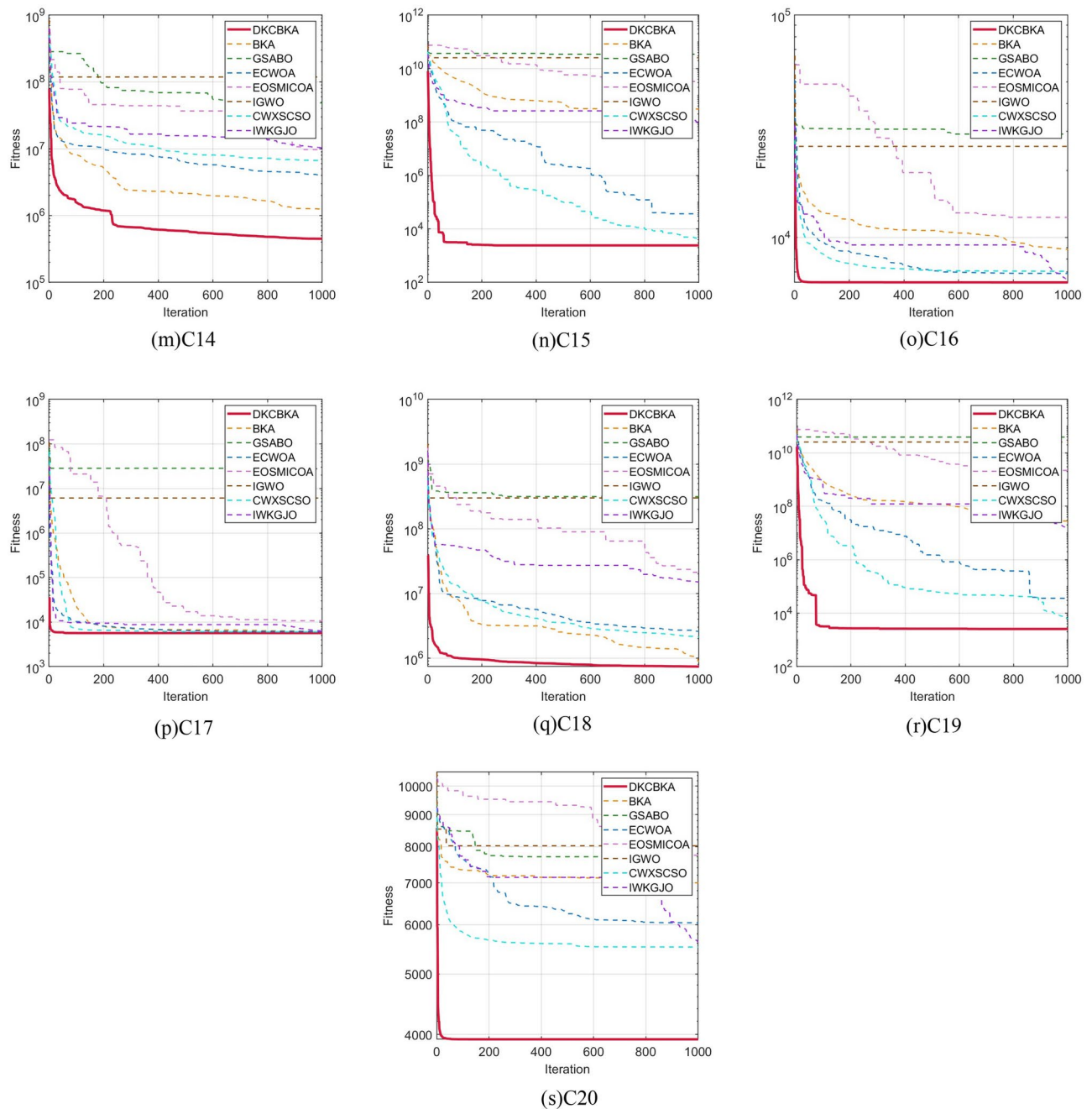


Figure 3. (continued)

OOA, SO⁵⁵, and BKA for three common engineering problems: the pressure vessel design problem, the triple-bar truss design problem, and the speed reducer design problem. To assess the optimization efficacy of DKCBKA in engineering applications, the group size is established at 30, and the total number of iterations is fixed at 1000.

Pressure vessel design issues

The pressure vessel is designed as efficiently as possible to reduce the overall production cost. Shell thickness (T_s), head thickness (T_h), inner radius (R), and cylinder length (L) are four of the optimization variables. Figure 5 illustrates the convergence diagram of its optimization of structure. The upper and lower limits of the above size parameters are set in DKCBKA, and the penalty function method is used, when the design exceeds these limits, the algorithm will give a large penalty. The cost function is considered as part of the optimization problem, and the cost is controlled by minimizing $\min f(\vec{x})$ by the objective function. The following is the pressure vessel design's mathematical model:

Variant

Function	DKCBKA-BKA	DKCBKA-GSABO	DKCBKA-ECWOA	DKCBKA-EOSMICOA	DKCBKA-IGWO	DKCBKA-CWXSCSO	DKCBKA-IWKGJO
C1	3.0199E-11	3.0199E-11	4.5726E-09	3.0199E-11	3.0199E-11	7.2208E-06	3.0199E-11
C3	1.2860E-06	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	5.4941E-11	3.0199E-11
C4	3.0199E-11	3.0199E-11	1.9568E-10	3.0199E-11	3.0199E-11	5.4620E-06	3.0199E-11
C5	9.7555E-10	3.0199E-11	1.9073E-01	3.0199E-11	3.0199E-11	1.3367E-05	5.0842E-03
C6	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	9.2113E-05	3.0199E-11
C7	6.6955E-11	3.0199E-11	5.8282E-03	3.8202E-10	3.0199E-11	7.5991E-07	1.6955E-02
C8	8.8910E-10	3.0199E-11	6.9724E-03	3.0199E-11	3.0199E-11	2.0283E-07	3.0418E-01
C9	3.0199E-11	3.0199E-11	1.0576E-03	3.0199E-11	3.0199E-11	2.8389E-04	3.0199E-11
C10	3.0199E-11	3.0199E-11	9.0632E-08	3.0199E-11	3.0199E-11	6.5486E-04	3.0199E-11
C11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11
C12	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11
C13	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	5.4941E-11	3.0199E-11
C14	2.5101E-02	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	4.5043E-11	3.3384E-11
C15	3.0199E-11	3.0199E-11	4.9752E-11	3.0199E-11	3.0199E-11	3.5923E-05	3.0199E-11
C16	9.9186E-11	3.0199E-11	9.9258E-02	3.0199E-11	3.0199E-11	4.2067E-02	5.3221E-03
C17	4.1997E-10	3.0199E-11	2.6077E-02	3.0199E-11	3.0199E-11	3.4783E-01	2.5721E-07
C18	2.0023E-06	3.0199E-11	3.3384E-11	3.0199E-11	3.0199E-11	3.6897E-11	3.0199E-11
C19	3.0199E-11	3.0199E-11	4.0772E-11	3.0199E-11	3.0199E-11	9.5332E-07	3.0199E-11
C20	6.1001E-01	3.0199E-11	1.5014E-02	3.0199E-11	3.0199E-11	8.6844E-03	1.2967E-01
+ / = / -	19/0/1	20/0/0	18/0/2	20/0/0	20/0/0	19/0/1	18/0/2

Table 6. Wilcoxon rank-sum test results for the 100-dimensional problem in CEC2017.

Algorithm	Basic parameter
GO	-
NOA	$\alpha = 0.05,$
PO	$\beta = 1.5$
SOA	$f_c = 2$
SSA	$ST = 0.6, PD = 0.7, SD = 0.2$

Table 7. Algorithmic parameter.

$\vec{x} = [x_1x_2x_3x_4] = [T_sT_hRL]$

Function

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

Subject to

$g_1(\vec{x}) = -x_1 + 0.0193x_3 \leq 0$
 $g_2(\vec{x}) = -x_3 + 0.00954x_3 \leq 0$
 $g_3(\vec{x}) = -\pi x_3^2 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0$
 $g_4(\vec{x}) = x_4 - 240 \leq 0$

design variable

$x_{1,2} \in [0.1,99]$
 $x_{3,4} \in [10,200]$

Figure 5 illustrates the optimal convergence rate and precision of the DKCBKA. Table 10 indicates that the total cost of the optimized design, achieved by DKCBKA is 6037.613549, which is 15.619% less than the base BKA and lowers the production cost by 18.222% when compared to WOA, HHO, GWO, OOA, and SO algorithms by 18.222%, 12.146%, 10.951%, 96.328%, and 11.488%, respectively. According to the results, DKCBKA has clear advantages over existing algorithms and can reduce the overall cost of pressure container design.

Function	Optimization indicator	DKCBKA	GO	NOA	PO	SOA	SSA
FC1	Min.	1	1	2.1637E+08	1	1.0306E+00	1
	Mean	1	4.2238E+08	4.7517E+08	1	3.8301E+06	1
	Std.	0	4.7827E+08	1.9054E+08	5.9344E-17	6.0493E+06	0
FC2	Min.	4.1848E+00	4.2841E+00	1.4020E+04	4.2235E+00	5	5.0000E+00
	Mean	4.3750E+00	1.6764E+04	1.9023E+04	4.5012E+00	3.0909E+03	5.0000E+00
	Std.	1.4536E-01	1.5260E+04	3.2965E+03	3.1097E-01	2.7167E+03	5.0944E-09
FC3	Min.	1.000000017	1.4096E+00	1.0823E+01	1.4143E+00	4.5051E+00	1.4092E+00
	Mean	1.3955E+00	8.6849E+00	1.1833E+01	4.2123E+00	6.9551E+00	2.1849E+00
	Std.	7.4697E-02	2.5938E+00	3.8518E-01	1.7249E+00	1.5707E+00	1.0621E+00
FC4	Min.	8.9600E+00	2.5874E+01	1.0161E+02	1.3553E+01	4.5664E+01	3.1844E+01
	Mean	2.2897E+01	7.7498E+01	1.1893E+02	3.7823E+01	7.2076E+01	6.5937E+01
	Std.	1.1907E+01	2.9727E+01	1.1409E+01	1.2806E+01	1.9509E+01	2.3839E+01
FC5	Min.	1.1348E+00	1.1403E+00	5.0126E+01	1.2627E+00	3.1495E+01	1.9790E+00
	Mean	1.4864E+00	2.8990E+00	1.0989E+02	2.1977E+00	6.6455E+01	6.6793E+00
	Std.	2.4535E-01	2.1391E+00	2.8961E+01	9.7002E-01	2.2127E+01	3.3662E+00
FC6	Min.	2.2865E+00	9.5083E+00	1.0996E+01	3.3344E+00	7.7423E+00	8.2595E+00
	Mean	4.3633E+00	1.2117E+01	1.3398E+01	6.4015E+00	1.0332E+01	1.0544E+01
	Std.	1.1171E+00	1.1399E+00	8.8722E-01	1.4872E+00	1.1462E+00	1.3750E+00
FC7	Min.	2.4169E+02	1.0588E+03	2.1076E+03	3.5538E+02	1.3400E+03	7.8192E+02
	Mean	6.3606E+02	1.5035E+03	2.4035E+03	1.0227E+03	2.1270E+03	1.2890E+03
	Std.	2.5670E+02	2.6591E+02	1.6866E+02	3.7526E+02	3.0874E+02	3.1111E+02
FC8	Min.	2.5185E+00	4.1815E+00	4.9836E+00	3.4864E+00	4.0875E+00	3.8335E+00
	Mean	3.5087E+00	5.0728E+00	5.2594E+00	4.2373E+00	4.7497E+00	4.7602E+00
	Std.	4.4290E-01	3.6765E-01	1.1538E-01	2.7424E-01	2.4012E-01	4.0788E-01
FC9	Min.	1.0869E+00	1.1173E+00	3.3288E+00	1.1177E+00	1.2978E+00	1.1080E+00
	Mean	1.2575E+00	1.3269E+00	4.4102E+00	1.3016E+00	2.9348E+00	1.3460E+00
	Std.	1.2710E-01	1.9621E-01	5.1402E-01	1.3761E-01	1.1256E+00	1.4708E-01
FC10	Min.	1.0090E+00	2.0985E+01	2.1391E+01	2.1017E+01	2.1456E+01	2.0983E+01
	Mean	1.7389E+01	2.1006E+01	2.1760E+01	2.1136E+01	2.1590E+01	2.0985E+01
	Std.	7.5180E+00	2.5630E-02	2.0506E-01	1.0135E-01	8.5954E-02	1.4984E-03

Table 8. CEC2019 different function optimization indicators.

Gear train design problem

In mechanical engineering, designing gear trains is a complex combinatorial optimization challenge. The main goal is to minimize the gear ratio—the ratio of input shaft rotation to output wheel speed—to reduce overall transmission cost. This involves determining the optimal number of teeth for each gear. Specifically, $N_a(x_1)$, $N_b(x_2)$, $N_d(x_3)$, and $N_f(x_4)$ represent the number of teeth on gears A, B, D, and F. The geometric constraint problem is that the size and shape of the gear need to meet specific geometric standards, so a reasonable range of gear parameters should be set in the optimization model, and the penalty function or constraint optimization algorithm is used to deal with these constraints. Finding the best combination of these variables is crucial for achieving the lowest gear ratio while maintaining functionality and efficiency.

Function

$$\min f(x) = \left(\frac{1}{6.931} - \frac{x_2x_3}{x_1x_4}\right)^2$$

Value range

$$12 \leq x_i \leq 60, i = 1, 2, 3, 4$$

The optimization challenge for the gear train design problem is effectively addressed by DKCBKA, demonstrating strong stability and convergence, as illustrated in Table 11 and Fig. 6. According to the information provided in Table 11, the adaptation value determined by the DKCBKA calculation is the lowest. This lowers construction costs to varying degrees in comparison to the comparative algorithms, demonstrating the validity and potential usefulness of DKCBKA in real-world engineering applications.

Reducer design issues

The speed reducer design issue is a mechanical optimization challenge aimed at minimizing the weight of the speed reducer by optimizing seven parameters while adhering to constraints related to shaft stresses, bending

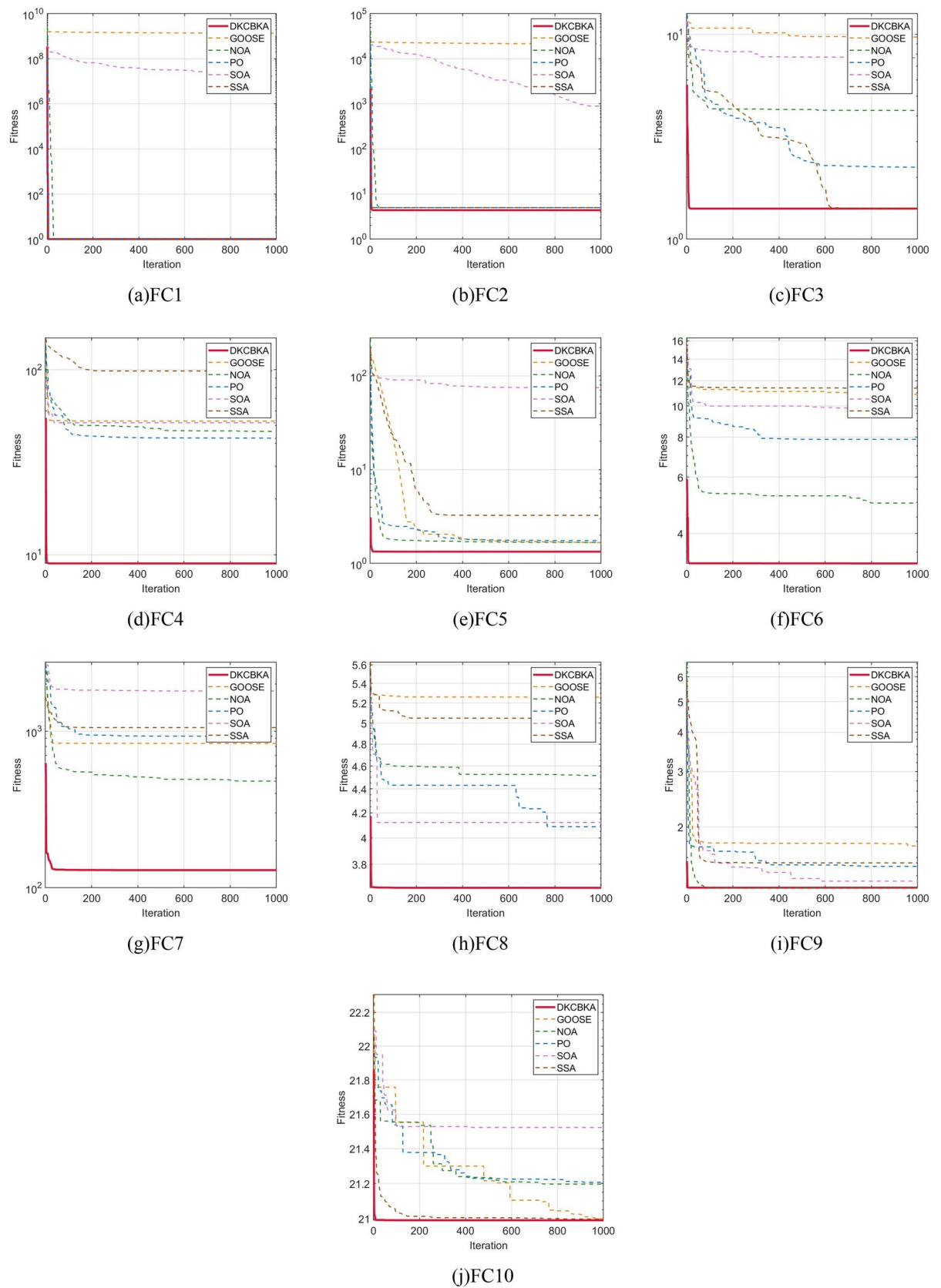


Fig. 4. CEC2019 function convergence curves.

Function	DKCBKA-GO	DKCBKA-NOA	DKCBKA-PO	DKCBKA-SOA	DKCBKA-SSA
FC1	6.8662E-07	6.8662E-07	3.5065E-01	1.2118E-12	1
FC2	2.8790E-06	3.0199E-11	3.0394E-01	3.0199E-11	6.4789E-12
FC3	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11
FC4	9.7555E-10	3.0199E-11	2.1327E-05	8.9934E-11	6.5183E-09
FC5	2.4913E-06	3.0199E-11	5.9706E-05	3.0199E-11	3.3384E-11
FC6	3.0199E-11	3.0199E-11	2.3800E-03	3.0199E-11	4.0772E-11
FC7	4.1825E-09	3.0199E-11	7.5991E-07	3.0199E-11	8.9934E-11
FC8	3.3384E-11	3.0199E-11	3.5708E-06	4.5043E-11	3.4971E-09
FC9	7.9590E-03	3.0199E-11	5.5699E-03	4.1997E-10	2.0058E-04
FC10	4.1191E-01	3.0199E-11	1.4733E-07	3.0199E-11	8.8411E-07
+/-/-	9/0/1	10/0/0	8/0/2	10/0/0	9/0/1

Table 9. CEC2019 Wilcoxon rank-sum test results.

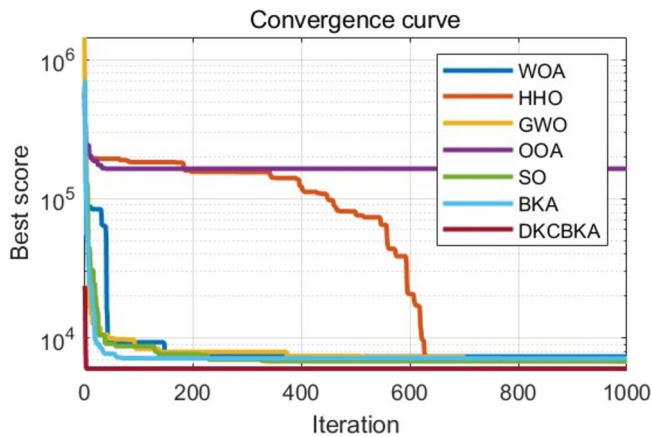


Fig. 5. Optimization convergence plot for pressure vessel design problem.

Algorithm	$T_s(x_1)$	$T_h(x_2)$	$R(x_3)$	$L(x_4)$	Result
WOA	1.1251	0.5076	49.7372	100.4441	7382.9426
HHO	1.1405	0.5554	57.9750	45.4366	6872.3033
GWO	1.1224	0.5550	58.1403	44.6336	6780.0796
OOA	8.0048	11.7142	56.5120	53.8409	164,444.3135
SO	1.1360	0.5615	58.8584	40.6022	6821.2336
BKA	1.2209	0.6036	63.2603	18.7373	7155.1961
DKCBKA	0.8586	0.4244	44.4884	149.1131	6037.6135

Table 10. Optimization results for pressure vessel design problems.

Algorithm	x_1	x_2	x_3	x_4	Result
WOA	52.8649	14.9602	25.5397	50.5625	2.3078E-11
HHO	52.7881	20.2659	13.1996	34.1497	2.3078E-11
GWO	23.1951	12	13.4163	47.1591	9.9216E-10
OOA	39.0758	12.0225	23.1911	49.2269	2.1808E-08
SO	50.9577	15.4184	26.3346	53.1509	2.3078E-11
BKA	60	15.0019	14.8132	26.3449	2.3576E-09
DKCBKA	42.8681	18.8402	15.9834	48.8304	2.7009E-12

Table 11. Optimization results of the gear train design problem.

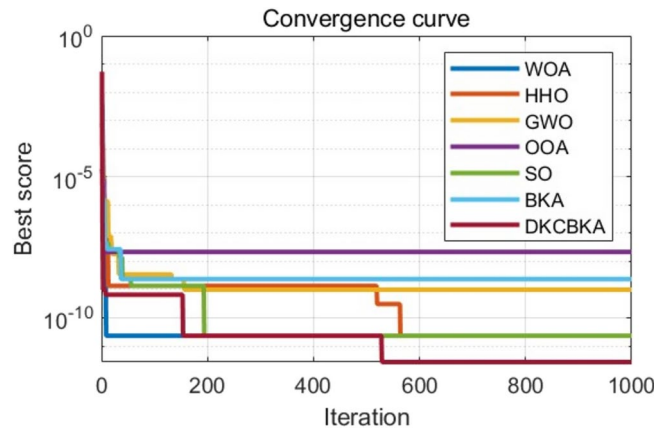


Fig. 6. Optimization convergence plot for gear train design problem.

stresses on gear teeth, transverse shaft deflections, and surface pressures. The seven design variables are facial breadth (b), quantity of gear teeth (p), tooth component (m), length of the initial shaft between the bearings (l_1), length of the secondary shaft between the bearings (l_2), diameter of the initial shaft (d_1), and diameter of the secondary shaft (d_2) denoted as x_1 – x_7 . In order to ensure that the transmission ratio and efficiency of the reducer meet the design requirements, the target range of transmission ratio and efficiency is set in the optimization model, and the optimization algorithm is used to find the design that meets these conditions. The following delineates the mathematical treatment of the challenge:

Variant

$$\vec{x} = [x_1 x_2 x_3 x_4 x_5 x_6 x_7] = [bmpl_1 l_2 d_1 d_2]$$

Function

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

Subject to

$$\begin{aligned} g_1(x) &= \frac{27}{x_1x_2^2x_3} - 1 \leq 0 \\ g_2(x) &= \frac{397.5}{x_1x_2^2x_3} - 1 \leq 0 \\ g_3(x) &= \frac{1.93x_5^3}{x_1x_6^4x_3} - 1 \leq 0 \\ g_4(x) &= \frac{1.93x_5^3}{x_2x_7^4x_3} - 1 \leq 0 \\ g_5(x) &= \frac{\left[\left(745 \left(\frac{x_4}{x_2x_3} \right) \right)^2 + 16.9 \times 10^6 \right]^{1/2}}{110x_6^3} - 1 \leq 0 \\ g_6(x) &= \frac{\left[\left(745 \left(\frac{x_5}{x_2x_3} \right) \right)^2 + 157.5 \times 10^6 \right]^{1/2}}{85x_7^3} - 1 \leq 0 \\ g_7(x) &= \frac{x_2x_3}{40} - 1 \leq 0 \\ g_8(x) &= \frac{5x_2}{x_1} - 1 \leq 0 \\ g_9(x) &= \frac{5x_1}{12x_2} - 1 \leq 0 \\ g_{10}(x) &= \frac{1.5x_6 + 1.9}{x_4} - 1 \leq 0 \\ g_{11}(x) &= \frac{1.1x_7 + 1.9}{x_5} - 1 \leq 0 \end{aligned}$$

Algorithm	x_1	x_2	x_3	x_4	x_5	x_6	x_7	Result
WOA	3.5600	0.7000	17.0000	7.3000	8.2322	3.3502	5.4592	3143.1675
HHO	3.5848	0.7000	17.0000	7.4582	7.8696	3.3505	5.3091	3046.9514
GWO	3.5019	0.7000	17.0000	7.3429	7.8420	3.3513	5.2888	3000.0164
OOA	3.5537	0.7493	22.2984	8.2393	8.1938	3.8287	5.2843	3553.6520
SO	3.5000	0.7000	17.0000	7.3000	7.7153	3.3502	5.2867	2994.4711
BJA	3.5103	0.7000	17.0049	7.8174	8.0035	3.3543	5.2868	3011.3531
DKCBKA	3.5000	0.7000	17.0000	7.3000	7.7153	3.3502	5.2867	2994.4711

Table 12. Optimization results of the gear train design problem.

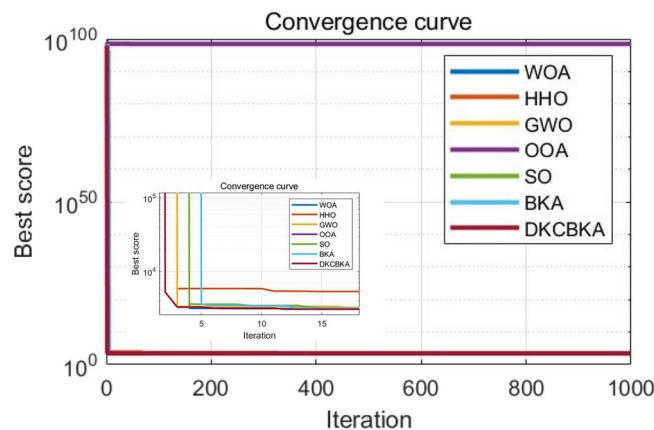


Fig. 7. Optimization convergence plot for the reducer design problem.

Value range

$$2.6 \leq x_1 \leq 3.6$$

$$0.7 \leq x_2 \leq 0.8$$

$$17 \leq x_3 \leq 28$$

$$7.3 \leq x_4$$

$$x_5 \leq 8.3$$

$$2.9 \leq x_6 \leq 3.9$$

$$5.0 \leq x_7 \leq 5.5$$

Table 12 displays the experimental findings for resolving the gearbox design's minimum weight issue. The gearbox design problem's optimization convergence plot is displayed in Fig. 7. According to Table 8's data, the DKCBKA algorithm's minimum weight is 2994.4711, indicating a good optimization accuracy and a 0.561% reduction over the traditional BKA. The convergence graph in Further evidence that DKCBKA is more capable of solving mechanical optimization problems comes from Fig. 7, which shows that it has a higher convergence efficiency than other methods.

Discussions

Improvements in a black-winged kite algorithm that incorporates a osprey optimization algorithm with a longitudinal crossover strategy can be attributed to the following. First, the black-winged kite algorithm's assault phase is enhanced by the dynamic exponential factor, and the osprey optimization algorithm is incorporated to enhance the algorithm's global search capability and convergence speed, as well as to prevent premature convergence. Following the migration phase, a stochastic differential mutation strategy is implemented to enhance the diversity of the population and balance the algorithm's global search and local exploration capabilities. This strategy is based on the mutation behavior in the differential evolution algorithm. Lastly, the algorithm's convergence accuracy is enhanced by the inclusion of the vertical and horizontal algorithm. This strategy includes both horizontal and vertical crossover.

Secondly, the superiority between the strategies was compared through 15 functions in the benchmark test set, the superiority of DKCBKA in comparison with 6 good improved algorithms was tested through the

CEC2017 test set as well as the performance of DKCBKA in comparison with 5 swarm intelligence algorithms through the CEC2019 test set.

Lastly, the efficacy of DKCBKA in real-world problems is evaluated using three real-world engineering examples: pressure vessel design issues, gear train design problems, and reducer design issues.

The above experimental analysis can be obtained:

- (1) The improved algorithm DKCBKA proposed in this paper occupies a relative advantage over the 15 compared algorithms, and the optimization performance improves significantly and has a good overall performance.
- (2) DKCBKA shows an absolute advantage in the 15 benchmark functions, and the optimization performance is greatly improved compared to the original algorithm. The optimization accuracy of DKCBKA is known to be better than 6 improved optimization algorithms in 95% of the cases by 20 functions in CEC2017, and the optimization accuracy of DKCBKA is better than other 5 swarm intelligence algorithms in all 10 functions in CEC2019. And the nonparametric statistical verification of the simulation results by Wilcoxon rank sum test shows that DKCBKA dominates the disadvantage in absolutely few cases. Finally three engineering examples show that the optimization performance of DKCBKA is improved by 18.222%, 99.885% and 0.561% respectively compared to the original algorithm.

Although the algorithm proposed in this paper significantly enhances the optimization performance of the original algorithm, it still has certain constraints. Specifically, using the maximum number of iterations to test functional performance may lack fairness and can be used instead of using the objective function to evaluate the number of iterations. The application of DKCBKA in practical engineering can be more novel.

Future research may apply the DKCBKA to novel engineering examples, including offshore wind turbine structural foundation optimization and other related issues, while further verifying the algorithm's performance through practical applications. Other test sets, such as CEC2011, can also be used to evaluate algorithm performance in the future. CEC2011, CEC2017, and CEC2019 test sets can be used to test the performance of the improved algorithm, including unimodal, multi-modal, composite problems, and support multiple dimensional testing. The difference is that CEC2011 has fewer problems and less complexity, which is suitable for preliminary algorithm verification; CEC2017 adds more composite and rotation problems, significantly improving the complexity, suitable for medium-complexity algorithm evaluation; CEC2019 further increases the complexity of the problem, making it more challenging and suitable for high-complexity algorithm evaluation.

Conclusion

This study provides a BKA that enhances the optimization power by integrating the Osprey optimization process with vertical and horizontal crossover techniques. The algorithm first alters the position update equation in the assault phase by integrating the dynamic index factor and the Osprey optimization technique. This approach enhances the method's precision and convergence rate. The program employs the mutation principle from the differential evolution algorithm to improve population variety and promote movement away from local optima via random differential variation during the migration phase. A vertical and horizontal crossover technique is introduced for improving group variety and boost the accuracy of algorithm convergence. To assess the optimization efficacy using the improved technique, the DKCBK is evaluated against the original algorithm across 15 benchmark functions from CEC2005. This is further validated by analyzing 19 functions from CEC2017 alongside 10 functions from CEC2019, utilizing six improved algorithms, including IWKGJO, and five swarm intelligence optimization algorithms, including GO. Practical illustrations in the design of pressure tanks, gear train, and reducers exemplify DKCBKA's engineering efficiency proficiency. The results show that the proposed DKCBKA has higher convergence speed and robustness, and the global search and local exploration capabilities are improved.

Data availability

The datasets utilized and/or analyzed in the current study are available from the corresponding author upon reasonable request.

Received: 29 December 2024; Accepted: 14 February 2025

Published online: 25 February 2025

References

1. Lian, J. et al. Parrot optimizer: Algorithm and applications to medical problems. *Comput. Biol. Med.* **172**, 108064. <https://doi.org/10.1016/j.combiomed.2024.108064> (2024).
2. Fatahi, A., Nadimi-Shahraki, M. H. & Zamani, H. An improved binary quantum-based avian navigation optimizer algorithm to select effective feature subset from medical data: A COVID-19 case study. *J. Bion. Eng.* **21**(1), 426–446. <https://doi.org/10.1007/s42235-023-00433-y> (2023).
3. Barua, S. & Merabet, A. Lévy Arithmetic Algorithm: An enhanced metaheuristic algorithm and its application to engineering optimization. *Expert Syst. Appl.* **241**, 122335 (2024).
4. Zhou, J., Zhang, Y. & Qiu, Y. State-of-the-art review of machine learning and optimization algorithms applications in environmental effects of blasting. *Artif. Intell. Rev.* **57**, 5 (2024).
5. Braik, M. Hybrid enhanced whale optimization algorithm for contrast and detail enhancement of color images. *Cluster Comput.* **27**, 231–267 (2024).
6. Wang, J. et al. Multi-objective optimization design for a 15 MW semisubmersible floating offshore wind turbine using evolutionary algorithm. *Appl. Energy* **377**, 124533 (2025).
7. Li, T., Su, J., Liu, W. & Corchado, J. M. Approximate Gaussian conjugacy: Parametric recursive filtering under nonlinearity, multimodality, uncertainty, and constraint, and beyond. *Front. Inf. Technol. Electron. Eng.* **18**, 1913–1939 (2017).

8. He, W. et al. Progress in prediction of remaining useful life of hydrogen fuel cells based on deep learning. *Renew. Sustain. Energy Rev.* **192**, 114193 (2024).
9. Nayak, J., Swapnarekha, H., Naik, B., Dhiman, G. & Vimal, S. 25 years of particle swarm optimization: Flourishing voyage of two decades. *Arch. Comput. Methods Eng.* **30**, 1663–1725 (2023).
10. Chen, W., Liu, J., Guo, H. & Kato, N. Toward Robust and Intelligent drone swarm: Challenges and future directions. *IEEE Netw.* **34**, 278–283 (2020).
11. Hamadneh, T. et al. Spider-tailed horned viper optimization: An effective bio-inspired metaheuristic algorithm for solving engineering applications. *Int. J. Intell. Eng. Syst.* (2025).
12. Hamadneh, T. et al. Orangutan optimization algorithm: An innovative bio-inspired metaheuristic approach for solving engineering optimization problems. *Int. J. Intell. Eng. Syst.* (2025).
13. Bouaouda, A., Hashim, F. A., Sayouti, Y. & Hussien, A. G. Pied kingfisher optimizer: a new bio-inspired algorithm for solving numerical optimization and industrial engineering problems. *Neural Comput. Appl.* <https://doi.org/10.1007/s00521-024-09879-5> (2024).
14. Hamadneh, T. et al. Fossa optimization algorithm: A new bio-inspired metaheuristic algorithm for engineering applications. *Int. J. Intell. Eng. Syst.* (2024).
15. Fu, Y., Liu, D., Chen, J. & He, L. Secretary bird optimization algorithm: A new metaheuristic for solving global optimization problems. *Artif. Intell. Rev.* **57**, 123 (2024).
16. Addax Optimization Algorithm. A novel nature-inspired optimizer for solving engineering applications. *IJIES* **17**, 732–743 (2024).
17. Huang, J. & Hu, H. Hybrid beluga whale optimization algorithm with multi-strategy for functions and engineering optimization problems. *J. Big Data* **11**, 3 (2024).
18. Li, W. et al. A multimixed strategy improved sparrow search algorithm and its application in TSP. *Math. Probl. Eng.* **2022**, 1–26 (2022).
19. Ouyang, C. et al. A multi-strategy hybrid cuckoo search algorithm with specular reflection based on a population linear decreasing strategy. *Int. J. Mach. Learn. Cyber.* **15**, 5683–5723 (2024).
20. Zhao, X., Yang, C., Zhu, D. & Liu, Y. A hybrid algorithm based on multi-strategy elite learning for global optimization. *Electronics* **13**, 2839 (2024).
21. Wen, X. et al. IOOA: A multi-strategy fusion improved osprey optimization algorithm for global optimization. *Electron. Res. Arch.* **32**(3), 2033–2074. <https://doi.org/10.3934/era.2024093> (2024).
22. Li, Y., Li, W., Yuan, Q., Shi, H. & Han, M. Multi-strategy improved seagull optimization algorithm. *Int. J. Comput. Intell. Syst.* **16**, 154 (2023).
23. Jin, W. et al. Enhanced UAV pursuit-evasion using boids modelling: A synergistic integration of bird swarm intelligence and DRL. *CMC-Comput. Mat. Contin.* **80**, 3523–3553 (2024).
24. Zhu, E., Wang, H., Zhang, Y., Zhang, K. & Liu, C. PHEE: Identifying influential nodes in social networks with a phased evaluation-enhanced search. *Neurocomputing* **572**, 127195 (2024).
25. Sun, G. et al. Low-latency and resource-efficient service function chaining orchestration in network function virtualization. *IEEE Internet Things J.* **7**, 5760–5772 (2020).
26. Asha, A., Arunachalam, R., Poonguzhali, I., Urooj, S. & Alelyani, S. Optimized RNN-based performance prediction of IoT and WSN-oriented smart city application using improved honey badger algorithm. *Measurement* **210**, 112505 (2023).
27. Gai, R., Chen, N. & Yuan, H. A detection algorithm for cherry fruits based on the improved YOLO-v4 model. *Neural Comput. Appl.* **35**, 13895–13906 (2023).
28. Saravanan, G., Neelakandan, S., Ezhumalai, P. & Maurya, S. Improved wild horse optimization with levy flight algorithm for effective task scheduling in cloud computing. *J. Cloud Comput. Adv. Syst. Appl.* **12**, 24 (2023).
29. Dehghani, M. & Trojovský, P. Osprey optimization algorithm: A new bio-inspired metaheuristic algorithm for solving engineering optimization problems. *Front. Mech. Eng. Switzerland* **8**, 1126450 (2023).
30. Wang, J., Wang, W., Hu, X., Qiu, L. & Zang, H. Black-winged kite algorithm: a nature-inspired meta-heuristic for solving benchmark functions and engineering problems. *Artif. Intell. Rev.* **57**, 98 (2024).
31. Das Gupta, S., Van Parys, B. P. G. & Ryu, E. K. Branch-and-bound performance estimation programming: a unified methodology for constructing optimal optimization methods. *Math. Program.* **204**, 567–639 (2024).
32. Zhang, Z., Wang, X. & Yue, Y. Heuristic optimization algorithm of black-winged kite fused with osprey and its engineering application. *Biomimetics* **9**, 595 (2024).
33. Xue, R. et al. Multi-strategy Integration Model Based on Black-Winged Kite Algorithm and Artificial Rabbit Optimization. in *Advances in Swarm Intelligence, PT I, ICSI 2024* (eds. Tan, Y. & Shi, Y.) vol. 14788, pp. 197–207 (Springer-Verlag Singapore Pte Ltd, Singapore, 2024).
34. Fu, J., Song, Z., Meng, J. & Wu, C. Prediction of lithium-ion battery state of health using a deep hybrid kernel extreme learning machine optimized by the improved black-winged kite algorithm. *Batteries-Basel* **10**, 398 (2024).
35. Mu, G. et al. MSBKA: A multi-strategy improved black-winged kite algorithm for feature selection of natural disaster tweets classification. *Biomimetics* **10**, 41 (2025).
36. Zhao, M., Su, Z., Zhao, C. & Hua, Z. Improved black-winged kite algorithm based on chaotic mapping and adversarial learning. *J. Phys. Conf. Ser.* **2898**, 012040 (2024).
37. Ma, N., Meng, J., Luo, J. & Liu, Q. Optimization of thermal-fluid-structure coupling for variable-span inflatable wings considering case correlation. *Aerosp. Sci. Technol.* **153**, 109448 (2024).
38. Xu, X. & Wei, Z. Dynamic pickup and delivery problem with transshipments and LIFO constraints. *Comput. Ind. Eng.* **175**, 108835 (2023).
39. Tan, X. & Long, W. An improved whale optimization algorithm based on stochastic differential mutation. *China Sciencepap.* **13**, 937–942 (2018).
40. Duan, Y., Liu, C., Li, S., Guo, X. & Yang, C. An automatic affinity propagation clustering based on improved equilibrium optimizer and t-SNE for high-dimensional data. *Inf. Sci.* **623**, 434–454 (2023).
41. Wang, X., Dong, X., Zhang, Y. & Chen, H. Crisscross Harris Hawks optimizer for global tasks and feature selection. *J. Bionic Eng.* **20**, 1153–1174 (2023).
42. Trojovský, P. & Dehghani, M. Subtraction-average-based optimizer: A new swarm-inspired metaheuristic algorithm for solving optimization problems. *Biomimetics* **8**, 149 (2023).
43. Liu, K., Zhao, L. & Wang, H. Whale optimization algorithm based on elite opposition-based and crisscross optimization. *J. Chin. Comput. Syst.* **41**, 2092–2097 (2020).
44. Huang, Q., Liu, S., Li, M. & Guo, Y. Multi-strategy chimp optimization algorithm and its application of engineering problem. *Comput. Eng. Appl.* **58**, 174–183 (2022).
45. Lu, M., He, D. & Qu, L. Grey wolf optimization algorithm based on elite learning for nonlinear parameters. *J. Guangxi Normal Univ. (Nat. Sci. Edn.)* **39**, 55–67 (2021).
46. Li, Y., Yu, Q. & Du, Z. Sand cat swarm optimization algorithm and its application integrating elite decentralization and crossbar strategy. *Sci. Rep.* **14**, 8927 (2024).
47. Li, Y., Yu, Q., Wang, Z., Du, Z. & Jin, Z. An improved golden jackal optimization algorithm based on mixed strategies. *Mathematics* **12**, 1506 (2024).

48. Hamad, R. K. & Rashid, T. A. GOOSE algorithm: A powerful optimization tool for real-world engineering challenges and beyond. *Evol. Syst.* **15**, 1249–1274 (2024).
49. Abdel-Basset, M., Mohamed, R., Jameel, M. & Abouhawwash, M. Nutcracker optimizer: A novel nature-inspired metaheuristic algorithm for global optimization and engineering design problems. *Knowledge-Based Syst.* **262**, 110248 (2023).
50. Dhiman, G. & Kumar, V. Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems. *Knowledge-Based Syst.* **165**, 169–196 (2019).
51. Xue, J. & Shen, B. A novel swarm intelligence optimization approach: Sparrow search algorithm. *Syst. Sci. Control Eng.* **8**, 22–34 (2020).
52. Mirjalili, S. & Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **95**, 51–67 (2016).
53. Heidari, A. A. et al. Harris hawks optimization: Algorithm and applications. *Futur. Gener. Comp. Syst.* **97**, 849–872 (2019).
54. Mirjalili, S., Mirjalili, S. M. & Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **69**, 46–61 (2014).
55. Hashim, F. A. & Hussien, A. G. Snake Optimizer: A novel meta-heuristic optimization algorithm. *Knowledge-Based Syst.* **242**, 108320 (2022).

Author contributions

S.B. 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/s41598-025-90660-6>.

Correspondence and requests for materials should be addressed to B.S.

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.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025