

Improved team learning-based grey wolf optimizer for optimization tasks and engineering problems

Jingkai Cui^{1,2} · Tianyu Liu^{1,2} · Mingchao Zhu¹ · Zhenbang Xu^{1,2}

Accepted: 3 November 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Optimization refers to finding the optimal solution to minimize or maximize the objective function. In the field of engineering, this plays an important role in designing parameters and reducing manufacturing costs. Meta-heuristics such as the grey wolf optimizer (GWO) are efficient ways to solve optimization problems. However, the GWO suffers from premature convergence or low accuracy. In this study, a team learning-based grey wolf optimizer (TLGWO), which consists of two strategies, is proposed to overcome these shortcomings. The neighbor learning strategy introduces the influence of neighbors to improve the local search ability, whereas the random learning strategy provides new search directions to enhance global exploration. Four engineering problems with constraints and 21 benchmark functions were employed to verify the competitiveness of the TLGWO. The test results were compared with three derivatives of the GWO and nine other state-of-the-art algorithms. Furthermore, the experimental results were analyzed using the Friedman and mean absolute error statistical tests. The results show that the proposed TLGWO can provide superior solutions to the compared algorithms on most optimization tasks and solve engineering problems with constraints.

Keywords Meta-heuristics algorithms \cdot Grey wolf optimizer \cdot Benchmark functions optimization \cdot Constrained problems optimization

Abbreviations

GWO	Grey wolf optimizer
TLGWO	Team learning-based grey wolf optimizer
MVO	Multi-verse optimizer
TEO	Thermal exchange optimization

Zhenbang Xu xuzhenbang@ciomp.ac.cn

¹ Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China

² University of Chinese Academy of Sciences, Beijing 100049, China

GSA	Gravitational search algorithm
RO	Ray optimization algorithm
PSO	Particle swarm optimization
KH	Krill herd algorithm
DE	Differential evolution algorithm
ABC	Artificial bee colony algorithm
ALO	Ant lion optimizer
WOA	Whale optimization algorithm
BOA	Butterfly optimization algorithm
ELM	Extreme learning machine
MDM-GWO	Mutation-driven modified grey wolf optimizer
MsRwGWO	Multi-strategy random weighted grey wolf optimizer
CGWO	Gaze cues learning-based grey wolf optimizer
RBGWO	Randomized balanced grey wolf optimizer
SGWO	Society-based grey wolf optimizer
AGWO	Adaptive grey wolf optimizer
RNA-GWO	Grey wolf optimizer with RNA crossover operation
MCA	Min-conflict local search algorithm
HGWOP	Hybrid GWO with PSO
B-GWO	Balanced grey wolf optimization
SGWO-FH	Sparsity-based grey wolf optimization algorithm
HGWO	Hybrid grey wolf optimizer
EEGWO	Exploration-enhanced grey wolf optimizer
IGWO	Improved grey wolf optimizer
HPSO	Self-organizing hierarchical particle swarm optimizer
SADE	Self-adapting differential evolution algorithm
MABC	Modified artificial bee colony algorithm
DEKH	Hybrid krill herd algorithm
sinDE	Sinusoidal differential evolution algorithm
CMVO	Chaotic multi-verse optimizer
BMWOA	Associative learning-based exploratory whale optimizer
BBOA	Enhanced butterfly optimization algorithm
DALO	Improved antlion optimizer
MAE	Mean absolute error
IK	Inverse kinematics
DOF	Degrees of freedom

1 Introduction

Optimization is a broad field of research that refers to finding a set of optimal variables to minimize or maximize an objective function without violating constraints. In the field of engineering, optimization problems usually refer to searching for optimal parameters to minimize manufacturing costs, or designing controllers to minimize control errors. The optimization of complex systems usually

involves many difficulties, such as nonlinearity, non-differentiability, high computational cost, large solution space, and multimodality [1]. Conventional mathematical methods, such as exact or approximate algorithms, can no longer solve these problems efficiently [2].

In contrast to conventional methods, meta-heuristic algorithms have become a competitive alternative to solving complex optimization problems owing to their simplicity and flexibility. Some meta-heuristic algorithms originated from physical rules, such as the multi-verse optimizer (MVO) [3], thermal exchange optimization (TEO) [4], gravitational search algorithm (GSA) [5], and ray optimization algorithm (RO) [6]. However, most algorithms are inspired by nature, such as the particle swarm optimization (PSO) [7], the krill herd algorithm (KH) [8], the differential evolution algorithm (DE) [9], the artificial bee colony algorithm (ABC) [10], the ant lion optimizer (ALO) [11], the whale optimization algorithm (WOA) [12], and the butterfly optimization algorithm (BOA) [13].

Using mathematical methods to describe the group class and hunting mechanism of grey wolves, Mirjalili [14] proposed an innovative meta-heuristic algorithm: the grey wolf algorithm (GWO). The grey wolf society is divided into four classes. The top leader is the alpha, who has the most extensive experience and is responsible for directing the wolves to find and hunt prey. The middle class comprises the beta and delta. They obey alpha's leadership and convey alpha orders to their subordinate wolves. The remaining wolves form the lowest class, called the omega, and act in accordance with the instructions of the leaders. During a hunt, the best wolf replaces the original alpha as the new leader.

The hunting mechanism simulated by the GWO includes three steps: tracking, surrounding, and assaulting the prey. There are two changing parameters, a and C, that adaptively adjust the degree of exploitation and exploration performed by the algorithm. Many studies have shown that the GWO has strong competitiveness owing to its fewer control parameters and easy implementation. Moreover, the GWO has also been widely used in real-world optimization problems, such as feature selection [15], image segmentation [16], and economic dispatch [17]. This shows that the GWO has great research value and application potential. However, the GWO relies only on three chief wolves to update the population, which leads to limitations such as insufficient population diversity and premature convergence.

To improve the performance of the GWO and better solve optimization problems, we designed two learning strategies to propose a new optimizer called TLGWO. The proposed strategies mimic the learning behavior of the grey wolf, including learning from its neighbors, as well as learning from random wolves in the team. The main features and contributions of our work are shown as follows:

 Proposal of a neighbor learning strategy. This strategy defines the sensing distance for each search agent, and the other search agents within the sensing distance are its neighbors. Neighbors with higher fitness values attract the search agent, whereas neighbors with lower fitness values repel it. The comprehensive influence of neighbors on a certain wolf will prompt it to move closer to its prey, which accelerates the convergence of the GWO.

- Proposal of a random learning strategy. During the global search, this strategy allows a grey wolf to learn from random wolves in the team, which allocates some of the wolves to a new area. This random distribution provides grey wolves with the opportunity to search for better prey, which enhances the exploration ability of the GWO.
- Comparison of the proposed TLGWO with the classical version of the GWO, three upgraded variants of the GWO, and nine recent state-of-the-art algorithms on 21 benchmark functions. Four engineering design problems with constraints were also used for the evaluation. The Friedman and mean absolute error statistical tests were performed to analyze the experimental results.
- Enhanced and balanced exploitation and exploration of the GWO due to the proposed algorithm, which is conducive to solving the optimization problem of complex systems. The two proposed learning strategies are also applicable to other algorithms, which provides new ideas for the improvement in meta-heuristic algorithms.

The remainder of this paper is organized as follows. Section 2 summarizes related work, including the classical GWO and some GWO variants. Specific team learning strategies and the details of the proposed TLGWO are described in Sect. 3. Section 4 lists the benchmark functions, introduces the experimental conditions, and analyzes the experimental results. Section 5 describes the engineering design problems and presents the corresponding tests. Finally, Sect. 6 provides a summary statement and discusses future work.

2 Related work

In this section, we introduce the grey wolf optimizer (GWO), as well as some recently proposed GWO variants.

2.1 Grey wolf optimizer

The grey wolf optimizer (GWO) was originally proposed by Mirjalili et al. [14] in 2014. It uses mathematical methods to show the group hierarchy and hunting laws of grey wolves.

2.1.1 Social hierarchy

According to the description of the GWO, the grey wolf society consists of four classes: alpha (α), beta (β), delta (δ), and omega (ω). α , β , and δ are the most suitable, second most suitable, and third most suitable results of the current population, respectively. They represent the leaders of the wolf pack and have the best

understanding of the location of the prey. The other wolves are denoted by ω . Their positions are changed based on α , β , and δ in each iteration. Throughout the optimization process, ω wolves were guided by the leaders to capture the best prey (the optimal solution) in their hunting space.

2.1.2 Encircling prey

The first step of hunting is to encircle the prey. The following two equations are proposed to describe the encircling behavior:

$$D = \left| C \cdot X_p(t) - X(t) \right| \tag{1}$$

$$X(t+1) = X_p(t) - A \cdot D \tag{2}$$

where X_p and X indicate the position variables of the prey and grey wolf, respectively. *t* is the current iteration. A and C are coefficient variables that are calculated as follows:

$$A = 2a \cdot r_1 - a \tag{3}$$

$$C = 2 \cdot r_2 \tag{4}$$

where r_1 and r_2 are random variables with values in [0, 1]. *a* linearly decreases from 2 to 0 during the entire search and is calculated as follows:

$$a(t) = 2 - 2t / \text{MaxIter}$$
⁽⁵⁾

where MaxIter is the maximum number of iterations.

2.1.3 Attacking prey

The second step involves harassing and attacking prey. Assuming that the three wolves α , β , and δ have better knowledge of the location of their prey than the other wolves, the other wolves (ω) will follow them to get closer to the prey. The positions of the ω wolves are updated as follows:

$$X_1 = X_\alpha - A_1 \cdot \left| C_1 \cdot X_\alpha - X \right| \tag{6}$$

$$X_2 = X_\beta - A_2 \cdot \left| C_2 \cdot X_\beta - X \right| \tag{7}$$

$$X_3 = X_\delta - A_3 \cdot \left| C_3 \cdot X_\delta - X \right| \tag{8}$$

$$X(t+1) = \left(X_1(t) + X_2(t) + X_3(t)\right)/3 \tag{9}$$

☑ Springer

where X_{α} , X_{β} , and X_{δ} denote the positions of α , β , and δ , respectively. The GWO algorithm is presented in Algorithm 1.

Algorithm 1. Grey Wolf Optimizer (GWO)
1: Initialize the grey wolf population <i>X</i> and the algorithm's parameters <i>a</i> , <i>A</i> and <i>C</i>
2: The current iteration $t = 1$
3: while <i>t</i> < the maximum number of iterations <i>MaxIter</i> do
4: for $i = 1$ to the population size N do
5: Calculate the fitness of each search agent
6: Select the best-performing wolves α , β , and δ
7: end for
8: Calculate the parameter <i>a</i> using Eq. (5)
9: for $i = 1$ to <i>N</i> do
10: Calculate the parameters A and C using Eq. (3) and (4)
11: Update the position of the current search agent using Eq. (9)
12: end for
13: $t = t + 1$
14: end while

2.2 Recently proposed GWO variants

The search process of meta-heuristic algorithms includes two stages: exploration and exploitation. During exploration, the search agent investigates promising areas in the search space as widely as possible, which requires the search to be random and global. Exploitation refers to the ability of the search agent to find a better solution in a promising local search area.

In the GWO, the wolves are led by α , β , and δ to search the optimal solution in the search space. This behavior is good at exploitation but weakens exploration, so the algorithm may converge prematurely owing to insufficient exploration and fall into a local optimum. Another limitation is that when the population update is determined by only the three best wolves, the diversity of the population decreases, which is not conducive to finding the global optimal solution.

Many recent studies are devoted to overcoming the shortcomings of the GWO. Ma et al. [18] combined extreme learning machine (ELM) with the GWO and proposed the GWO-ELM algorithm to solve the optimization problem of composite beams (CBs). The experimental results showed that the GWO-ELM could determine the overall behavior of the CBs quickly and accurately. However, more tests such as optimization of benchmark functions are not used to verify the universality of the algorithm.

Shehata et al. [19] combined the autonomous group particle swarm algorithm (AGPSO) and the grey wolf optimizer to propose a hybrid optimizer called AGPSO-GWO. The application on the optimization of the current transmission systems verified the effectiveness of the hybrid algorithm. Since the AGPSO-GWO is a

hybrid version of two algorithms, the computational complexity of the algorithm is increased.

Singh and Bansal [20] designed a new search mechanism and a driven scheme to propose the mutation-driven modified grey wolf optimizer (MDM-GWO). The mutation mechanism based on Levy flight is used to enhance the global search ability of the algorithm. The experimental results show that the new strategies improve the convergence speed and exploration ability of the GWO.

Inac et al. [21] proposed the multi-strategy random weighted grey wolf optimizer (MsRwGWO) containing new strategies such as a boundary checking mechanism and a greedy selection mechanism to improve the performance of the GWO. However, the comparative results show that the MsRwGWO is less competitive on high-dimensional optimization problems.

Nadimi-Shahraki et al. [22] designed two new strategies called neighbor gaze cues learning (NGCL) and random gaze cues learning (RGCL). The NGCL strategy enhances the exploitation ability of the algorithm, and the RGCL strategy improves the population diversity. They applied these strategies in the GWO and proposed the gaze cues learning-based grey wolf optimizer (CGWO). These new strategies effectively enhance the competitiveness of the GWO.

Adhikary and Acharyya [23] proposed the randomized balanced grey wolf optimizer (RBGWO), which is inspired by the social hierarchy and random walk strategies. Unconstrained and constrained real-world optimization problems are used to test the performance of the algorithm. Experimental results show that the added strategies effectively improve the search efficiency of the algorithm.

Hosseini-Hemati [24] proposed the society-based grey wolf optimizer (SGWO) to optimize power dispatch problem. In the SGWO, the population is divided into several societies. Each society has an independent leader who leads other wolves closer to their prey. Moreover, a new mechanism for attacking prey was applied. The results show that the SGWO can solve the optimization problem of power system quickly and effectively.

In order to improve computational efficiency, Meidani et al. [25] proposed the adaptive grey wolf optimizer (AGWO). In the AGWO, the parameters are automatically adjusted according to a three-point fitness history, which effectively accelerates the convergence of the algorithm. However, the performance of the algorithm has not been further tested on real-world optimization problems with constraints.

Liu and Wang [26] designed a crossover operator according to the structure of RNA molecules. The proposed grey wolf optimizer with RNA crossover operation (RNA-GWO) is used for optimization problems of benchmark functions and wavelet neural networks. The results show that the RNA-GWO effectively improved the global search ability of the GWO. Optimization problems for complex systems with constraints have not been used to test the performance of the algorithm.

Makhadmeh et al. [27] combined the min-conflict local search algorithm (MCA) and the GWO to propose a new algorithm called GWO-MCA. The comparison results with other algorithms show that the GWO-MCA has great advantages in solving the power scheduling problem in smart home. Future work can be considered to modify the selection strategy of the MCA to further improve the quality of the solutions.

Based on the unique search advantages of the PSO and GWO, Zhang et al. [28] proposed the hybrid GWO with PSO (HGWOP). A poor-for-change strategy organically integrates the PSO and GWO to maximize the overall performance. The test results on benchmark functions show that the HGWOP has stronger universality. However, the proportion of the HGWOP ranking first on different functions is not large, and its application in practical problems is not considered.

In the field of terrestrial networks, Gupta et al. [29] proposed the balanced grey wolf optimization (B-GWO) algorithm to optimize the unmanned aerial vehicles deployment and power allocation. In this work, the iterative process of the GWO is divided into three stages and each stage has a unique parameter update strategy. The comparison results show that the B-GWO has superior performance in solving non-convex optimization problems.

Rajput [30] proposed the sparsity-based grey wolf optimization algorithm (SGWO-FH) to optimize the least square representation problem in face hallucination techniques. The concept of sparsity effectively improves the computational speed of the algorithm. Furthermore, a domain-specific prior is introduced to initialize the population. Compared with other methods, the SGWO-FH produces better super-resolved faces. However, the algorithm is easily affected by noise, and its robustness needs to be improved.

3 Team learning-based grey wolf optimizer

The proposed team learning-based grey wolf optimizer (TLGWO) contains two different strategies: neighbor learning and random learning.

3.1 Neighbor learning strategy

When chasing prey, grey wolves decide their actions based on the three leaders, as described in Eq. (9). In addition, the influence of neighbor wolves on an individual wolf cannot be ignored. This neighbor effect may be attractive or repulsive. If a neighbor is closer to the prey, an individual will be attracted to it. Otherwise, the individual stays away from the neighbors. Both cases motivate individuals to move toward the prey, so the exploitation of the algorithm is further enhanced. In this paper, we proposed a neighbor learning strategy to introduce the influence of neighbors.

The neighbor learning strategy is presented as follows:

$$\alpha_i^{\text{neighbor}} = \sum_{j=1}^M \hat{F}_{ij} \hat{X}_{ij} \tag{10}$$

$$\hat{F}_{ij} = \frac{F_i - F_j}{F^{\text{worst}} - F^{\text{best}}} \tag{11}$$

$$\hat{X}_{ij} = \frac{X_j - X_i}{\|X_j - X_i\| + \epsilon}$$
(12)

where *M* denotes the number of neighbors of the *i*th individual. F_i and F_j are the fitness of the *i*th individual and the *j*th neighbor, respectively. F^{worst} and F^{best} indicate the worst and the best fitness values in the population, respectively. *X* represents the position of an individual or neighbor. \hat{X}_{ij} is a unit vector, and a small positive number ε is added to avoid singularities. \hat{F}_{ij} is the normalized fitness value that determines whether the effect of the individual and the neighbor is attractive or repulsive.

We take the following example to further explain the neighbor learning strategy. Figure 1 shows a two-dimensional search space. The red point O represents the global optimum. The three orange concentric circles represent the contour lines of three fitness values F_1 , F_2 , and F_3 . Search agents on the same concentric circle have the same fitness value. The closer to the global optimum, the smaller the fitness value, so $F_1 < F_2 < F_3$. The yellow point A refers to an individual in the search agents. The points inside the green circle are the neighbors of A. In Fig. 1, B, C, and D are neighbors of A, but E is not. There are three types of neighbors of A: closer to the global optimum than A (like D), as far as A from the global optimum (like C), and farther from the global optimum than A (like B).

In Eq. (11), $\hat{F}_{ij} \in [-1, 1]$ is a parameter that determines whether neighbors attract or repel the individual. Since $F_1 < F_2$, for A and D, $\hat{F}_{AD} > 0$. Similarly, $\hat{F}_{AB} < 0$, $\hat{F}_{AC} = 0$. We define the coordinate X of each search agent to be a vector whose direction is from the origin O to itself. In Eq. (12), \hat{X}_{ij} is a unit vector and its direction is from *i* to *j*. For A and D, the direction of vector $v_{AD} = \hat{F}_{AD} \hat{X}_{AD}$ is from A to D. Similarly, the direction of vector v_{AB} is from B to A. In other words, D has an attractive effect on A, B has a repulsive effect on A, and C has no effect on A. In the next iteration, the amount of position change of A is determined by all its neighbors, as shown in Eq. (10). In this example, $\alpha_A^{\text{neighbor}} = v_{AB} + v_{AD}$. The direction of $\alpha_A^{\text{neighbor}}$ is toward the origin O, so the neighbor learning strategy

Fig. 1 Schematic of neighbor learning strategy



motivates A to move toward the global optimum. The above explanation continues to apply to search spaces with higher dimensions.

To select the neighbors of the *i*th individual, the sensing distance is defined:

$$d_i^{\text{sensing}} = \parallel X_i - \alpha_i^{\text{leader}} \parallel$$
(13)

where a_i^{leader} represents the position change calculated by the classical GWO, and its calculation method is shown in Eq. (9). If the Euclidean distance between two individuals is less than the sensing distance d_i^{sensing} , the individuals are neighbors.

3.2 Random learning strategy

Insufficient exploration is one of the main limitations of the GWO, which stems from the fact that all grey wolves move in relation to the three wolves α , β , and δ . In this paper, we introduce a random learning strategy, so that individuals are not only led by the leaders but are also affected by other random individuals in the population. The random learning is described as follows:

$$\alpha_i^{\text{random}} = \rho(X_m - X_n), \ m \neq n \neq i \tag{14}$$

where X_m and X_n are two different individuals randomly chosen from the population. ρ is a random scale factor that determines the walk distance and $\rho \in [0, 1]$.

Comparing Eq. (14) with Eq. (9), the significant difference is that the motion generated by Eq. (9) always forces the individual to move toward the current best solutions (X_{α} , X_{β} and X_{δ}), whereas the position update caused by Eq. (14) is completely random. This means that the motion induced by Eq. (14) may prompt individuals to escape from local optima. Therefore, the random learning strategy described by Eq. (14) provides enough randomness for exploration. Moreover, the random individuals X_m and X_n also increase the diversity of the population, which plays an important role in overcoming premature convergence.

Random learning provides individuals with the opportunity to avoid local optima, which greatly enhances the exploration of the algorithm. It mimics the scattered foraging of grey wolves in nature in response to food shortages. If the food found by the leaders is not sufficient to supply the entire wolf pack, some grey wolves may migrate to new areas with abundant food, which improves their survivability.

3.3 Proposed TLGWO

When the neighbor learning strategy and random learning strategy are combined, the position-updated operator of the GWO is rewritten as follows:

$$X_i(t+1) = B_1 \cdot \rho_1 \cdot \alpha_i^{\text{leader}} + \alpha_i^{\text{neighbor}} + B_2 \cdot \rho_2 \cdot \alpha_i^{\text{random}}$$
(15)

where $\alpha_i^{\text{leader}} = (X_1(t) + X_2(t) + X_3(t))/3$ is the same as in Eq. (9). α_i^{leader} is the movement of the *i*th individual under the influence of leaders α , β , and δ . $\alpha_i^{\text{neighbor}}$

and α_i^{random} are the movements caused by neighbors and random learning, respectively. $\rho_1, \rho_2 \in [0, 1]$ are random parameters that are used to determine the distance of the movements. B_1 and B_2 are defined as weighting factors to regulate the exploration and exploitation capabilities of the GWO, and they are calculated as follows:

$$B_1(t) = \left(\frac{\text{MaxIter} - t}{\text{MaxIter}}\right)^{\mu}$$
(16)

$$B_2(t) = 1 - B_1(t) \tag{17}$$

where $\mu = 1.5$ is set based on experience. A similar setting of μ can be found in [20].

 B_1 and B_2 are nonlinear variables of iteration *t*. At the beginning of the search, the individuals are dispersed. A larger B_1 is set to enhance the exploitation so that the algorithm quickly converges to the optimal solution. At the end of the search, individuals are concentrated near the optimal solution. The value of B_2 increases to enhance exploration and encourage more individuals to disperse to new search areas to find better solutions.

The TLGWO algorithm is presented in Algorithm 2. The neighbor learning strategy is good at local search and can help individuals find a precise solution faster. The random learning strategy encourages individuals to explore new areas, which is conducive to break through the local optimum.

Algorithm 2. Team Learning-based Grey Wolf Optimizer (TLGWO)
1: Initialize the grey wolf population <i>X</i> and the algorithm's parameters <i>a</i> , <i>A</i> , <i>C</i> , <i>B</i> ₁ , and <i>B</i> ₂
2: The current iteration $t = 1$
3: while <i>t</i> < the maximum number of iterations <i>MaxIter</i> do
4: for $i = 1$ to the population size N do
5: Calculate the fitness of each search agent
6: Select the best-performing wolves α , β , and δ
7: end for
8: Calculate the parameter a using Eq. (5)
9: for $i = 1$ to <i>N</i> do
10: Update the parameters A and C using Eq. (3) and (4)
11: Calculate the movement caused by leaders using Eq. (9)
12: Calculate the sensing distance using Eq. (13)
13: for $j = 1$ to N do
14: Select the neighbors of the <i>i</i> th individual
15: end for
16: for $k = 1$ to the number of neighbors M
17: Calculate the movement caused by neighbors using Eq. (10)
18: end for
19: Update the parameters B_1 and B_2 using Eq. (16) and (17)
20: Calculate the movement caused by random learning using Eq. (14)
21: Update the position of the current search agent using Eq. (15)
22: end for
23: $t = t + 1$
24: end while

4 Experimental results and analysis

In this section, the proposed TLGWO was tested on 21 commonly used benchmark functions. The test results were compared with the GWO and three GWO variants, as well as nine state-of-the-art algorithms. For a fair comparison, we chose improved versions of the state-of-the-art algorithms. Moreover, several statistical analysis methods were used to discuss the results.

The algorithms used for comparison are as follows: grey wolf optimizer (GWO) [14], hybrid grey wolf optimizer (HGWO) [17], exploration-enhanced grey wolf optimizer (EEGWO) [31], improved grey wolf optimizer (IGWO) [32], self-organizing hierarchical particle swarm optimizer (HPSO) [33], self-adapting differential evolution algorithm (SADE) [34], modified artificial bee colony algorithm (MABC) [35], hybrid krill herd algorithm (DEKH) [36], sinusoidal differential evolution algorithm (sinDE) [37], chaotic multi-verse optimizer (CMVO) [38], associative learning-based exploratory whale optimizer (BMWOA) [39], enhanced butterfly optimization algorithm (BBOA) [40], and improved antlion optimizer (DALO) [41].

Table 1 lists the parameter settings of the algorithms mentioned in this study. All parameters were set to the values in the original reference studies. All experiments were performed on a computer with an Intel® CoreTM i7-8750H CPU @ 2.20 GHz and 8.00 GB RAM in a Windows 10 environment. The stopping rule for all algorithms is that the number of iterations reaches the maximum number of iterations. In this paper, the maximum number of iterations was set to 500.

4.1 Benchmark functions

Twenty-one benchmark functions with dimensions D=10, 30, and 50 from a series of reference studies [14, 15, 20, 42] were applied to verify the superiority of the TLGWO. The selected benchmark functions are presented in Table 2. The

Algorithm	Year	Parameter settings
GWO	2014	$a \in [2, 0], r_1, r_2 \in [0, 1]$
HGWO	2016	$a \in [2,0], r_1, r_2 \in [0,1], W = 1, C_r \in [0,0.2]$
EEGWO	2017	$r_1, r_2 \in [0, 1], b_1 = 0.1, b_2 = 0.9, \mu = 1.5, a_{\text{initial}} = 2, a_{\text{final}} = 0$
IGWO	2021	$a \in [2, 0], r_1, r_2 \in [0, 1]$
HPSO	2004	$c_{1i} = 2.5, c_{1f} = 0.5, c_{2i} = 0.5, c_{2f} = 2.5$
SADE	2006	$\tau_1 = \tau_2 = 0.1, F_i = 0.1, F_u = 0.9$
MABC	2012	$\emptyset_{ij} \in [-1, 1]$, limit = 200, MR = 0.4, SF _i = 1
DEKH	2014	$N^{\text{max}} = 0.01, v_f = 0.02, C_t = 0.5, \text{FW} = 0.1, \text{CR} = 0.4$
sinDE	2014	$U_j \in [0, 1], \text{ freq} = 0.25$
CMVO	2019	WEP $\in [0.2, 1]$, TDR $\in [0.6, 1]$, $p = 6$
BMWOA	2020	$l \sim U(-1,1), p \sim U(0,1), \beta = 0.005, bw = 0.5$
BBOA	2020	$a \in [0.1, 0.3], r \in [0, 1], c = 0.01, p = 0.8$
DALO	2021	$J_r = 1, w_d = 8, w_r = 1.4$

 Table 1
 Parameter settings of algorithms

Table 2 Selected benchmark	unctions	
Name	Function	Search range
Sphere	$f_1(x) = \sum_{i=1}^n x_i^2$	[-100,100]
Schwefel 2.22	$f_2(x) = \sum_{i=1}^{n} x_i + \prod_{i=1}^{n} x_i $	[-10,10]
Rotated hyper-ellipsoid	$f_3(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j\right)^2$	[-100,100]
Schwefel 2.21	$f_4(x) = \max\left\{ \left x_i , 1 \le i \le n \right\} \right\}$	[-100,100]
Sum square	$f_5(x) = \sum_{i=1}^n i x_i^2$	[-10,10]
Quartic	$f_6(\mathbf{x}) = \sum_{i=1}^n i x_i^4$	[-1.28,1.28]
Noise	$f_{\mathcal{T}}(x) = \sum_{i=1}^{n} i x_i^4 + \text{random}[0, 1]$	[-1.28,1.28]
Sum power	$f_8(x) = \sum_{i=1}^n x_i ^{(i+1)}$	[-1,1]
Rosenbrock	$f_{9}(x) = \sum_{i=1}^{n-1} \left[100(x_{i+1} - x_{i}^{2})^{2} + (x_{i} - 1)^{2} \right]$	[-30,30]
Schwefel 2.26	$f_{10}(x) = 418.9828n - \sum_{i=1}^{n} x_i \sin\left(\sqrt{ x_i }\right)$	[-500,500]
Rastrigin	$f_{11}(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos\left(2\pi x_i\right) + 10 \right]$	[-5.12,5.12]

Improved team learning-based grey wolf optimizer for...

 $\underline{\textcircled{O}}$ Springer

Table 2 (continued)		
Name	Function	Search range
NCRastrigin	$f_{12}(x) = \sum_{i=1}^{n} \left[y_i^2 - 10\cos\left(2\pi y_i\right) + 10 \right]$	[-5.12,5.12]
	$y_i = \begin{cases} x_{j_1} x_j < 1/2\\ \frac{round(2x_j)}{2}, x_j \ge 1/2 \end{cases}$	
Griewank	$f_{13}(x) = \frac{1}{400} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[- 600,600]
Alpine	$f_{14}(x) = \sum_{i=1}^{n} \left x_i \sin \left(x_i \right) + 0.1 x_i \right $	[-10,10]
Inverted Cosine Mixture	$f_{15}(x) = 0.1n - \left(0.1 \prod_{i=1}^{n} \cos\left(5\pi x_i\right) - \prod_{i=1}^{n} x_i^2\right)$	[-1,1]
Zakharov	$f_{16}(x) = \sum_{i=1}^{n} x_i^2 + \left(\sum_{i=1}^{n} 0.5ix_i\right)^2 + \left(\sum_{i=1}^{n} 0.5ix_i\right)^4$	[-5,10]
Pathological	$f_{17}(\mathbf{x}) = \sum_{i=2}^{n} 0.5 + \frac{\sin^2 \left(\sqrt{100x_{i-1}^2 + x_i^2}\right) - 0.5}{1 + 0.001(x_{i-1}^2 - 2x_{i-1}x_i + x_i^2)^2}$	[- 100,100]

Table 2 (continued)		
Name	Function	Search range
Penalized 1	$f_{18}(x) = \frac{\pi}{n} \left\{ 10\sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 \left[1 + 10\sin^2(\pi y_{i+1}) \right] + (y_n - 1)^2 \right\}$	[-50,50]
	$+ \sum_{i=1}^{n} u(x_i, 10, 100, 4)$	
	$y_i = 1 + \frac{x_i + 1}{4}$	
	$u(x_{i}, a, k, m) = \begin{cases} k(x_{i} - a)^{m}, & x_{i} \ge a \\ 0, & -a < x_{i} < a \\ k(-x_{i} - a)^{m}, & x_{i} \le -a \end{cases}$	
Penalized 2	$f_{19}(x) = 0.1 \left\{ \sin^2 \left(3\pi x_1 \right) + \sum_{i=1}^{n-1} (x_i - 1)^2 \left[1 + \sin^2 \left(3\pi x_{i+1} \right) \right] + (x_n - 1)^2 \left[1 + \sin^2 \left(2\pi x_n \right) \right] \right\} + \sum_{i=1}^n u(x_i, 5, 100, 4)$	[-50,50]
Salomon	$f_{20}(x) = 1 - \cos\left(2\pi\sqrt{\sum_{i=1}^{n} x_i^2}\right) + 0.1\sqrt{\sum_{i=1}^{n} x_i^2}$	[-100,100]
Ackley	$f_{21}(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^{n} \cos\left(2\pi x_i\right)\right) + 20 + e$	[-32,32]

search space (2-D version) for some of these benchmark functions are shown in Fig. 2. In Fig. 2, the variables x_i (i=1, 2) and the function value f of the benchmark function with dimension D=2 constitute a three-dimensional surface. Below the surface is a contour plot, and the color of the surface changes according to the height of f.

The unimodal functions (f_1-f_9) were implemented to test the global search performance of the algorithms owing to the functions having only one optimal solution. Comparatively, the multimodal functions $(f_{10}-f_{21})$ have many local optima, which are helpful for examining the exploration ability of algorithms. The minimum value of all functions was 0.



Fig. 2 Search space (2-D version) for benchmark functions

4.2 Comparison with the GWO and its variants

To demonstrate the competitiveness of the TLGWO, the GWO and three popular GWO variants (HGWO, EEGWO, and IGWO) were used for comparison. The details of the algorithms are shown in Table 1. The population size and the maximum number of iterations were 30 and 500, respectively. For each benchmark function having 10, 30, and 50 dimensions, 30 runs were performed. The results of the comparison are presented in Tables 3 and 4, where "Mean" represents the average best value and "Std" refers to the standard deviation value. Wilcoxon's signed-rank test with a significance level of 5% was applied to compare the superiority of algorithms. The "+" indicates that the TLGWO performs better than this algorithm, "-" means the performance of the TLGWO is inferior to this algorithm, and "~" represents that the TLGWO is not significantly different from this algorithm. The best results are shown in bold.

Tables 3 and 4 show that the TLGWO achieves the best results for most functions. Compared with the GWO, the TLGWO achieves similar results on f_{15} and better results on all other functions. Compared with the HGWO, the TLGWO obtains better results for all functions except f_9 . On f_9 with D=10, the TLGWO loses to the HGWO, but as the dimensions increase, the performance of the TLGWO catches up to that of the HGWO. Similar behavior can also be observed in comparison with the IGWO on f_{10} and f_{19} .

Compared with the EEGWO, the TLGWO obtains better results on 13 functions and achieves the same optima on seven functions. On f_7 with D=10 and 30, the TLGWO and EEGWO show similar performances, but on f_7 with D=50, the EEGWO is even better. Compared with the IGWO, the TLGWO performs better on 16 functions, worse on f_9 and f_{18} , and similarly on f_{15} . According to the above analysis, the TLGWO is more competitive than other GWO variants for almost all unimodal and multimodal functions, and its performance does not decrease significantly with an increase in dimensionality.

Figure 3 shows the convergence curves of the six algorithms mentioned above with D=30 on f_4 , f_7 , f_{14} , and f_{17} . It can be seen that the TLGWO converges to the global optimum at a fast rate. Similar phenomena were observed for most other functions. Owing to space limitations, the resulting figures for the remaining functions are omitted.

4.3 Comparison with other state-of-the-art algorithms

The TLGWO was compared with nine other state-of-the-art algorithms. For the 21 benchmark functions, all algorithms were run independently 30 times. The population size was 30, and the best result was output after 500 iterations. Tables 5 and 6 show the average and standard deviation of the optimization results, and the best results are highlighted in bold.

From Tables 5 and 6, it can be observed that the TLGWO obtains the best results in all dimensions for all functions except functions f_9 , f_{10} , f_{18} , and f_{19} . On f_9 with

10 Ś Jo 1 6. . GWO fo LL P UMUI UMUaa OWOI1 ÷ , ц ٩

Table 3	Experiment	tal results of GV	WO, HGWO, EEC	3WO, IGWC	O, and TLGWO fo	or unimodal	functions $(D=10, 3)$	0,50)			
Fun	Dim	Index	GWO		HGWO		EEGWO		IGWO		TLGWO
f_1	10	Mean	2.62E-57	+	1.16E-28	+	1.08E - 190	+	2.79E-61	+	0.00E + 00
		Std	6.20E-57		2.46E-28		0.00E+00		6.97E-61		0.00E+00
	30	Mean	2.64E-27	+	5.05E-14	+	1.18E-186	+	3.64E-28	+	0.00E+00
		Std	5.56E-27		7.70E-14		0.00E+00		1.20E - 27		0.00E+00
	50	Mean	1.23E-19	+	1.07E - 09	+	2.38E-187	+	3.78E - 20	+	0.00E+00
		Std	1.65E-19		1.01E - 09		0.00E+00		3.57E-20		0.00E+00
f_2	10	Mean	1.12E-32	+	1.14E - 17	+	6.80E-97	+	1.95E - 36	+	1.24E-247
		Std	2.74E-32		1.60E-17		2.02E-96		2.91E-36		0.00E+00
	30	Mean	7.20E-17	+	2.22E-09	+	1.31E-95	+	7.50E-18	+	6.56E-246
		Std	3.02E-17		2.56E-09		3.36E-95		4.85E-18		0.00E+00
	50	Mean	2.40E-12	+	4.08E - 07	+	2.44E-95	+	4.45E-13	+	1.33E-245
		Std	1.08E-12		1.49E - 07		3.85E-95		3.49E-13		0.00E+00
f_3	10	Mean	6.93E-26	+	2.98E-16	+	3.88E-187	+	2.80E-27	+	0.00E+00
		Std	2.08E-25		8.74E-16		0.00E+00		8.52E-27		0.00E+00
	30	Mean	7.02E-06	+	1.18E - 02	+	1.85E-186	+	1.11E - 03	+	0.00E+00
		Std	1.87E-05		3.62E - 02		0.00E+00		1.94E - 03		0.00E+00
	50	Mean	8.59E-01	+	2.69E+01	+	4.34E-187	+	1.64E + 01	+	0.00E+00
		Std	1.40E + 00		3.26E+01		0.00E+00		2.40E+01		0.00E+00
f_4	10	Mean	1.37E-18	+	5.02E - 06	+	2.31E-96	+	7.79E-19	+	6.87E-247
		Std	1.82E-18		7.62E-06		6.16E-96		1.07E-18		0.00E+00
	30	Mean	1.04E - 06	+	1.45E+00	+	2.91E - 94	+	3.17E - 05	+	1.96E-245
		Std	9.08E-07		1.30E+00		1.07E-93		5.86E-05		0.00E + 00
	50	Mean	3.99E - 04	+	8.77E+00	+	3.21E - 94	+	1.39E - 02	+	2.38E-244
		Std	2.84E-04		3.61E+00		1.12E - 93		1.00E - 02		0.00E+00

J. Cui et al.

Table 3 (continued)										
Fun	Dim	Index	GWO		HGWO		EEGWO		IGWO		TLGWO
f_5	10	Mean	5.60E-59	+	2.18E-29	+	1.10E-191	+	2.06E-62	+	0.00E+00
		Std	1.61E-58		9.34E-29		0.00E+00		5.61E-62		0.00E+00
	30	Mean	1.32E-28	+	6.79E-15	+	5.56E-188	+	3.57E-29	+	0.00E+00
		Std	3.02E-28		8.30E-15		0.00E+00		7.06E-29		0.00E+00
	50	Mean	1.42E-20	+	1.74E-10	+	5.70E-183	+	8.61E-21	+	0.00E+00
		Std	1.04E - 20		2.03E-10		0.00E+00		8.82E-21		0.00E+00
f_6	10	Mean	1.18E - 102	+	8.58E-54	+	0.00E+00	u	2.70E-105	+	0.00E + 00
		Std	3.10E-102		3.64E-53		0.00E+00		1.47E - 104		0.00E + 00
	30	Mean	9.64E-52	+	7.66E-27	+	0.00E+00	u	2.11E-49	+	0.00E + 00
		Std	2.55E-51		2.22E-26		0.00E+00		6.01E - 49		0.00E + 00
	50	Mean	3.06E-37	+	1.54E-19	+	0.00E+00	u	3.43E-35	+	0.00E+00
		Std	5.03E-37		1.77E-19		0.00E+00		1.34E - 34		0.00E+00
f_7	10	Mean	8.63E-04	+	3.92E - 03	+	9.12E-05	u	7.91E-04	+	6.18E - 05
		Std	4.61E-04		2.45E-03		7.17E-05		4.29E-04		5.97E-05
	30	Mean	1.69E - 03	+	1.58E - 02	+	8.08E-05	u	3.47E-03	+	7.71E-05
		Std	8.68E-04		7.32E-03		9.07E-05		1.22E - 03		5.75E-05
	50	Mean	3.72E - 03	÷	3.17E-02	+	5.79 E - 05	I	7.04E - 03	+	8.63E-05
		Std	1.21E-03		9.92E - 03		5.02E - 05		3.05E - 03		6.79E-05

Table 3 ((continued)										
Fun	Dim	Index	GWO		HGWO		EEGWO		IGWO		TLGWO
f_8	10	Mean	2.88E-117	+	1.91E-61	+	5.47E-202	+	2.48E-124	+	0.00E+00
		Std	9.13E-117		8.90E-61		0.00E+00		1.26E-123		0.00E+00
	30	Mean	2.25E-98	+	1.34E-49	+	2.30E-197	+	7.59E-88	+	0.00E+00
		Std	6.84E-98		6.73E-49		0.00E+00		4.15E-87		0.00E+00
	50	Mean	4.49E-91	+	1.32E-40	+	4.14E-194	+	1.37E-78	+	0.00E+00
		Std	1.31E-90		7.12E-40		0.00E+00		7.37E-78		0.00E+00
f_9	10	Mean	8.91E+00	+	7.58E+00	I	8.92E+00	+	3.06E + 00	I	6.50E+00
		Std	1.98E - 02		1.26E+00		1.02E - 01		9.17E - 01		4.52E-01
	30	Mean	2.87E+01	+	2.83E+01	N	2.89E+01	+	2.42E+01	I	2.70E+01
		Std	1.67E-02		9.97E-01		3.62E - 02		5.57E - 01		7.59E-01
	50	Mean	4.85E+01	+	4.83E+01	N	4.89E + 01	+	4.53E+01	I	4.77E+01
		Std	2.93E-02		6.91E-01		2.14E - 02		1.33E+00		5.48E-01
Dim	10	≈/-/+	0/0/6		8/1/0		7/0/2		8/1/0		
	30	≈/-/+	0/0/6		8/0/1		7/0/2		8/1/0		
	50	$\approx / - / +$	0/0/6		8/0/1		7/1/1		8/1/0		

Table 4 E	xperimental	l results of GW(O, HGWO, EEG	WO, IGWC), and TLGWO foi	r multimod	A functions $(D=10,$	30, 50)			
Fun	Dim	Index	GWO		HGWO		EEGWO		IGWO		TLGWO
f_{10}	10	Mean	1.39E+03	+	1.56E+03	+	2.91E+03	+	2.49E+02	I	8.26E+02
		Std	2.17E+02		3.18E + 02		2.79E+02		1.85E + 02		3.42E+02
	30	Mean	6.41E+03	+	5.85E+03	+	1.05E+04	+	4.42E+03	u	3.68E + 03
		Std	7.37E+02		5.18E+02		4.27E+02		1.73E+03		8.06E + 02
	50	Mean	1.14E + 04	+	1.03E+04	+	1.79E+04	+	9.70E+03	+	6.95E+03
		Std	1.11E + 03		9.99E + 02		8.06E+02		3.40E + 03		9.39E + 02
f_{11}	10	Mean	4.07E-01	+	2.33E+01	+	0.00E + 00	u	4.00E+00	+	0.00E + 00
		Std	9.88E - 01		9.14E+00		0.00E + 00		3.12E + 00		0.00E + 00
	30	Mean	2.15E+00	+	1.17E+02	+	0.00E + 00	u	2.56E+01	+	0.00E + 00
		Std	3.29E+00		2.46E+01		0.00E + 00		2.28E+01		0.00E + 00
	50	Mean	3.38E+00	+	2.46E+02	+	0.00E + 00	u	5.46E+01	+	0.00E + 00
		Std	3.18E + 00		5.35E+01		0.00E + 00		$2.96E \pm 01$		0.00E + 00
f_{12}	10	Mean	5.02E+00	+	9.06E+00	+	0.00E + 00	N	7.02E+00	+	0.00E+00
		Std	3.33E+00		4.21E+00		0.00E + 00		1.82E + 00		0.00E + 00
	30	Mean	7.89E+00	+	1.22E+02	+	0.00E + 00	U	4.54E+01	+	0.00E+00
		Std	4.34E+00		3.28E + 01		0.00E + 00		2.90E+01		0.00E+00
	50	Mean	1.53E+01	+	2.55E+02	+	0.00E + 00	U	8.63E+01	+	0.00E+00
		Std	8.62E+00		5.34E+01		0.00E + 00		6.66E+01		0.00E+00
f_{13}	10	Mean	2.21E-02	+	2.12E-01	+	0.00E + 00	N	4.06E-02	+	0.00E + 00
		Std	2.13E-02		1.42E-01		0.00E + 00		5.19E - 02		0.00E+00
	30	Mean	5.38E - 03	+	1.21E-02	+	0.00E + 00	U	2.71E-03	+	0.00E+00
		Std	8.99E-03		1.81E - 02		0.00E + 00		5.91E-03		0.00E+00
	50	Mean	2.19E-03	+	1.01E-02	+	0.00E + 00	U	3.46E - 03	+	0.00E+00
		Std	5.00E - 03		2.05E-02		0.00E + 00		5.12E-03		0.00E+00

Table 4 🥡	continued)										
Fun	Dim	Index	GWO		HGWO		EEGWO		IGWO		TLGWO
f_{14}	10	Mean	1.84E-04	+	1.64E+00	+	3.83E-97	+	1.00E - 05	+	5.61E-249
		Std	2.69E-04		1.07E+00		1.55E-96		1.35E-05		0.00E+00
	30	Mean	7.31E-04	+	1.33E+01	+	3.00E - 97	+	3.29E-04	+	8.10E-248
		Std	8.13E-04		4.70E+00		4.95E-97		2.86E-04		0.00E+00
	50	Mean	1.05E-03	+	2.79E+01	+	1.28E-95	+	1.18E - 03	+	1.09E - 247
		Std	1.18E - 03		4.78E+00		4.57E-95		8.25E-04		0.00E + 00
f_{15}	10	Mean	0.00E + 00	N	1.28E-01	+	0.00E + 00	N	0.00E + 00	u	0.00E+00
		Std	0.00E + 00		2.40E-01		0.00E + 00		0.00E + 00		0.00E+00
	30	Mean	0.00E + 00	N	1.23E+00	+	0.00E + 00	N	0.00E + 00	N	0.00E + 00
		Std	0.00E + 00		7.33E-01		0.00E + 00		0.00E + 00		0.00E+00
	50	Mean	0.00E + 00	N	2.14E+00	+	0.00E + 00	N	5.92E-17	N	0.00E+00
		Std	0.00E + 00		1.17E+00		0.00E + 00		2.25E-16		0.00E+00
f_{16}	10	Mean	1.73E - 33	+	3.67E-22	+	1.43E-193	+	8.27E-38	+	0.00E + 00
		Std	3.51E-33		7.12E-22		0.00E + 00		1.58E-37		0.00E+00
	30	Mean	2.05E-07	+	7.01E-06	+	1.24E - 188	+	2.72E-07	+	0.00E+00
		Std	4.89E-07		1.15E-05		0.00E + 00		7.22E-07		0.00E+00
	50	Mean	9.37E-02	+	6.62E-02	+	1.00E - 186	+	7.86E-02	+	0.00E+00
		Std	8.20E-02		1.04E-01		0.00E+00		5.98E-02		0.00E+00

Table 4 (continued)										
Fun	Dim	Index	GWO		HGWO		EEGWO		IGWO		TLGWO
f_{17}	10	Mean	1.97E+00	+	2.89E+00	+	0.00E + 00	u	2.54E+00	+	0.00E+00
		Std	4.69E-01		3.99E-01		0.00E + 00		3.10E-01		0.00E+00
	30	Mean	1.09E+01	+	1.15E+01	+	0.00E + 00	u	1.19E+01	+	0.00E+00
		Std	9.32E-01		4.40E-01		0.00E + 00		3.87E-01		0.00E+00
	50	Mean	2.06E+01	+	2.08E+01	+	0.00E + 00	U	2.14E+01	+	0.00E+00
		Std	6.99E - 01		6.04E-01		0.00E + 00		2.99E-01		0.00E+00
f_{18}	10	Mean	9.83E-02	+	3.03E+00	+	7.46E-01	+	$3.07E{-10}$	I	6.69E-02
		Std	3.48E - 02		2.50E+00		2.16E-01		2.19E - 10		6.14E-02
	30	Mean	9.78E-02	+	5.99E+00	+	1.06E + 00	+	7.13E-03	I	6.72E-02
		Std	4.37E-02		2.77E+00		1.57E-01		2.62E-02		5.76E-02
	50	Mean	1.05E - 01	+	6.01E+00	+	1.12E+00	+	3.01E - 02	I	8.28E-02
		Std	4.82E-02		2.55E+00		1.13E-01		1.62E - 02		4.19E - 02
f_{19}	10	Mean	2.82E-01	+	6.47E-01	+	8.91E-01	+	3.73E - 08	I	1.59E-01
		Std	4.55E-02		2.44E-01		6.07E-02		5.34E - 08		7.34E-02
	30	Mean	6.28E-01	+	3.15E+00	+	2.98E+00	+	1.09E - 01	I	6.00E-01
		Std	2.10E-01		8.76E-01		1.32E-02		9.83E-02		2.11E-01
	50	Mean	2.15E+00	+	6.13E+00	+	4.99E+00	+	1.17E+00	N	1.00E+00
		Std	3.79E - 01		1.63E+00		9.73E-03		3.23E-01		2.92E - 01

Improved team learning-based grey wolf optimizer for...

Table 4 (c	continued)										
Fun	Dim	Index	GWO		HGWO		EEGWO		IGWO		TLGWO
f_{20}	10	Mean	9.98E-02	+	2.53E-01	+	1.59E-96	+	9.98E-02	+	0.00E+00
		Std	4.92E-10		5.07E-02		7.01E-96		1.78E-09		0.00E+00
	30	Mean	1.84E-01	+	5.09E-01	+	1.54E - 94	+	2.13E-01	+	0.00E+00
		Std	4.39E-02		1.09E-01		8.22E-94		3.45E - 02		0.00E+00
	50	Mean	2.09E-01	+	7.66E-01	+	2.08E-95	+	3.03E-01	+	0.00E+00
		Std	3.16E - 02		9.94E-02		6.81E-95		4.13E-02		0.00E+00
f_{21}	10	Mean	7.99E-15	+	1.69E+00	+	8.88E-16	N	4.91E-15	+	8.88E-16
		Std	2.36E-15		1.54E+00		0.00E + 00		1.22E-15		0.00E+00
	30	Mean	1.01E-13	+	1.33E+00	+	8.88E-16	N	5.92E-14	+	8.88E-16
		Std	1.97E-14		1.30E+00		0.00E + 00		1.19E-14		0.00E+00
	50	Mean	4.01E-11	+	8.25E-01	+	8.88E-16	N	3.20E-11	+	8.88E-16
		Std	1.95E-11		1.12E+00		0.00E + 00		2.10E-11		0.00E+00
Dim	10	≈/-/+	11/0/1		12/0/0		6/0/6		8/3/1		
	30	≈/-/+	11/0/1		12/0/0		6/0/6		8/2/2		
	50	≈/-/+	11/0/1		12/0/0		6/0/6		9/1/2		



Fig. 3 Convergence curves of five algorithms with D = 30 on f_4 , f_7 , f_{14} , and f_{17}

D=10, the SADE performs best, and when the dimension is increased to 30 and 50, the BMWOA achieves the best performance. However, the results obtained by the TLGWO are quite close to the best results, especially when the dimension is high (when D=30, the difference between the best result and that of the TLGWO is 2.8%, and when D=50, the difference is 0.62%).

On f_{10} with D = 10, the sinDE finds the optimal solution, and when D = 30 and 50, the BMWOA outperforms the other algorithms. On f_{18} with D = 10, 30, and 50, the best results are obtained by HPSO, sinDE, and BMWOA, respectively. The sinDE achieves a great advantage on f_{19} with D = 10 and 30, but when the dimension is 50, HPSO is best.

Figure 4 shows the convergence characteristics of the TLGWO and nine state-ofthe-art algorithms on some typical functions. According to Fig. 4, the TLGWO has a **Table 5** Experimental results of nine state-of-the-art algorithms and TLGWO for unimodal functions (*D*=10, 30, 50)

F	D	Index	HPSO	SADE	MABC	DEKH	sinDE	CMVO	BMWOA	BBOA	DALO	TLGWO
f_1	10	Mean	2.90E-27	1.82E-14	9.13E+00	4.49E-01	1.16E-21	2.09E-02	8.93E-51	6.46E-03	7.02E-09	0.00E+00
		Std	9.14E-27	5.26E-14	3.83E+00	1.09E - 01	1.02E-21	7.84E-03	2.24E-50	2.60E-03	4.79E-09	0.00E+00
	30	Mean	8.88E-03	1.37E-02	1.19E+04	3.88E+01	6.12E-05	1.93E+00	3.16E-43	1.56E-01	1.11E - 03	0.00E+00
		Std	1.75E-02	1.81E-02	4.35E+03	1.05E+01	4.81E-05	5.16E+01	1.00E-42	7.89E-03	8.16E-04	0.00E+00
	50	Mean	3.26E+00	1.50E+01	5.43E+04	1.03E+02	2.82E-01	1.56E+01	5.65E-51	1.72E-01	5.95E+00	0.00E + 00
		Std	2.95E+00	1.83E+01	5.46E+03	3.38E+01	1.69E-01	3.27E+00	1.40E - 50	5.52E-03	4.13E+00	0.00E + 00
f_2	10	Mean	1.95E-07	1.23E-09	6.28E-01	1.48E - 01	3.53E-13	4.78E-02	2.09E-34	1.15E-01	1.14E+00	1.85E-248
		Std	6.18E-07	3.48E-09	2.17E-01	2.29E-02	2.43E-13	1.72E-02	3.81E-34	1.44E-02	1.94E+00	0.00E+00
	30	Mean	5.09E-01	1.78E-02	4.18E + 01	2.47E+01	7.25E-04	6.62E+00	6.56E-33	4.98E-01	3.13E+01	1.13E-248
		Std	4.23E-01	2.27E-02	4.50E+00	2.26E+00	2.73E-04	7.06E+00	1.15E-32	5.41E-01	4.09E+01	0.00E+00
	50	Mean	8.56E+00	7.48E-01	1.25E+04	4.64E+02	1.48E - 01	1.42E+02	1.49E - 30	3.50E+05	1.36E+02	9.44E-248
		Std	3.47E+00	8.17E-01	1.71E+04	9.72E+02	2.55E-02	3.40E+01	4.66E-30	4.88E+04	8.20E+01	0.00E + 00
f_3	10	Mean	1.22E-05	8.37E-02	8.58E+02	1.59E+03	2.90E - 02	8.27E-02	1.81E+02	1.39E-01	1.08E-02	0.00E+00
		Std	3.20E-05	1.27E-01	3.39E+02	1.35E+03	4.02E-02	5.88E-02	2.03E+02	9.76E-03	1.65E-02	0.00E+00
	30	Mean	8.37E+01	1.04E+04	2.61E+04	7.73E+03	7.03E+03	1.55E+02	3.43E+04	1.81E - 01	4.72E+03	0.00E+00
		Std	1.62E+01	1.15E+04	4.85E+03	4.27E+03	2.53E+03	4.70E+01	2.05E+04	1.02E-02	2.34E+03	0.00E+00
	50	Mean	1.66E+03	4.49E+04	7.41E+04	4.56E+04	4.81E+04	5.72E+03	1.97E + 05	1.99E - 01	1.91E+04	0.00E+00
		Std	4.51E+02	3.03E+04	8.32E+03	1.15E+04	1.45E+04	2.20E+03	2.78E+04	8.97E-03	6.83E+03	0.00E + 00
f_4	10	Mean	1.45E-05	1.02E - 01	3.94E+00	2.59E+00	4.13E - 04	8.03E-02	1.23E+00	4.74E-01	1.30E - 03	1.68E - 247
		Std	1.32E-05	3.07E-01	1.20E+00	1.84E+00	1.28E - 03	2.98E-02	1.68E+00	2.04E - 02	6.48E-04	0.00E+00
	30	Mean	1.11E+00	1.75E+01	5.77E+01	3.17E+01	9.05E+00	1.85E+00	5.33E+01	5.58E-01	1.51E+01	2.46E-245
		Std	1.41E - 01	8.47E+00	4.70E+00	1.03E+01	6.22E+00	5.00E - 01	2.35E+01	2.72E-02	2.14E+00	0.00E+00
	50	Mean	4.36E+00	$3.64E \pm 01$	7.64E+01	6.15E+01	3.23E+01	1.52E+01	5.19E + 01	5.58E-01	2.44E+01	1.33E-246
		Std	9.67E-01	9.39E+00	3.55E+00	8.84E+00	5.99E+00	5.52E+00	3.00E+01	1.82E-02	4.07E+00	0.00E+00

Table 5	cont	inued)										
F	D	Index	OSHH	SADE	MABC	DEKH	sinDE	CMVO	BMWOA	BBOA	DALO	TLGWO
f_5	10	Mean	3.65E-20	5.07E-07	8.63E-11	3.58E+00	4.08E-23	1.63E-03	3.93E-82	8.29E-02	6.09E-07	0.00E+00
		Std	7.75E-20	1.60E-06	1.08E-10	4.18E+00	2.89E-23	1.16E-03	1.09E-81	4.96E-03	1.06E-06	0.00E + 00
	30	Mean	2.64E-03	3.01E-03	1.98E+00	1.83E+01	3.05E-06	1.41E+00	1.41E-78	1.27E-01	3.66E+00	0.00E + 00
		Std	3.54E-03	8.34E-03	6.04E-01	1.38E+01	1.82E-06	9.99E-01	4.04E-78	4.26E-03	4.82E+00	0.00E+00
	50	Mean	6.92E+00	2.19E+00	3.60E+02	9.58E+01	5.66E-02	2.81E+01	1.19E-74	1.44E-01	5.32E+01	0.00E+00
		Std	7.55E+00	2.26E+00	1.25E+02	3.86E+01	3.55E-02	1.47E+01	3.75E-74	5.14E-03	5.01E+01	0.00E + 00
f_6	10	Mean	3.23E-35	3.89E-16	4.52E-25	7.36E-04	6.99E-40	1.51E-11	1.55E-120	2.03E-03	3.15E-24	0.00E+00
		Std	8.28E-35	1.23E-15	7.09E-25	1.11E - 03	1.29E-39	1.46E-11	4.92E-120	1.50E - 04	3.31E-24	0.00E+00
	30	Mean	2.91E-05	6.03E-04	3.77E-05	1.18E - 01	3.17E-09	2.30E-07	4.08E-115	3.35E - 03	7.23E-09	0.00E+00
		Std	3.86E-05	1.25E-03	2.88E-05	7.49E-02	8.24E-09	1.78E-07	8.76E-115	9.62E-05	1.57E-08	0.00E+00
	50	Mean	7.65E-01	1.04E - 02	3.47E-01	1.86E-01	6.16E-05	1.26E-05	5.44E-110	4.02E - 03	9.45E-04	0.00E+00
		Std	8.51E-01	2.31E - 02	2.48E-01	1.58E-01	6.86E-05	6.30E-06	1.72E-109	1.34E - 04	8.28E-04	0.00E + 00
f_7	10	Mean	7.33E - 03	1.72E - 02	3.22E-02	7.81E-02	1.28E - 02	3.69 E - 03	1.71E-03	3.24E-04	1.99E - 02	7.20E - 05
		Std	3.52E - 03	5.67E-03	1.03E - 02	4.39E-02	6.16E-03	2.10E-03	1.51E-03	1.64E - 04	8.96E-03	4.74E-05
	30	Mean	1.75E-01	9.41E - 02	4.64E-01	5.11E-01	7.84E-02	3.79E-02	4.47E-03	4.19E - 04	2.76E-01	7.18E-05
		Std	7.71E-02	4.09E-02	7.25E–02	2.41E-01	2.15E-02	8.56E-03	4.01E - 03	2.03E-04	7.66E-02	6.03E-05
	50	Mean	1.96E+00	1.99E - 01	2.60E+00	9.37E-01	1.57E-01	1.12E-01	4.28E - 03	6.36E - 04	7.85E-01	6.23E-05
		Std	8.66E-01	7.94E-02	1.00E+00	1.42E - 01	3.74E - 02	2.30E - 02	3.91E - 03	5.71E-04	1.65E-01	6.66E-05

Table	5 (cont	inued)										
F	D	Index	OSHH	SADE	MABC	DEKH	sinDE	CMVO	BMWOA	BBOA	DALO	TLGWO
f_8	10	Mean	5.77E-34	3.33E-19	2.94E-18	3.66E-05	7.26E-50	2.86E-08	2.56E-107	9.86E-04	2.62E-07	0.00E+00
		Std	1.59E-33	7.08E-19	4.17E-18	3.21E-05	1.48E-49	3.68E-08	8.11E-107	4.65E-04	1.49E-07	0.00E+00
	30	Mean	4.18E-04	1.73E-08	3.13E - 08	1.33E-05	6.12E-18	5.67E-07	4.33E-110	9.93E-04	1.04E-06	0.00E+00
		Std	1.24E-03	5.45E-08	6.53E-08	9.05E-06	1.03E-17	4.05E-07	8.82E-110	2.91E-04	5.85E-07	0.00E+00
	50	Mean	7.50E-01	2.53E-08	4.84E-06	8.19E-06	1.09E - 09	1.03E-06	3.62E-108	1.22E-03	7.60E-07	0.00E+00
		Std	4.21E-01	7.98E-08	3.93E-06	6.55E-06	2.73E-09	6.11E-07	1.14E - 107	4.33E-04	3.83E-07	0.00E + 00
f_9	10	Mean	7.85E+00	5.06E + 00	8.33E+02	2.39E+01	6.47E+00	9.11E + 01	6.48E+00	5.09E+00	3.91E+02	8.90E+00
		Std	7.32E+00	2.04E + 00	3.96E+02	5.44E+00	1.01E+00	1.69E+02	1.33E+00	3.59E-01	5.96E+02	1.76E-02
	30	Mean	1.37E+02	8.62E+01	5.89E+03	5.31E+04	5.96E+01	1.11E+03	2.82E+01	2.87E+01	2.21E+02	2.87E+01
		Std	8.99E+01	3.81E+01	1.97E+03	2.12E+04	4.00E+01	1.13E + 03	1.95E - 01	5.82E-02	1.68E+02	2.34E-02
	50	Mean	9.49E+02	2.75E+03	6.90E+05	1.12E+05	6.47E+02	1.70E+03	4.84E+01	4.87E+01	4.16E+03	4.85E+01
		Std	6.05E+02	3.98E+03	2.80E+05	5.24E+04	7.42E+02	9.80E+02	1.25E - 02	4.73E-02	5.94E+03	4.03E-02

ומטוכט	ndva				argonanina and			(n - n) error	(00,00			
F	D	Index	OSdH	SADE	MABC	DEKH	sinDE	CMVO	BMWOA	BBOA	DALO	TLGWO
f_{10}	10	Mean	1.82E+03	4.74E+02	5.71E+02	1.36E+03	4.42E+02	1.11E+03	9.27E+02	2.66E+03	2.03E+03	9.05E+02
		Std	2.75E+02	2.96E+02	1.60E+02	3.31E+02	2.86E+02	2.91E+02	6.65E+02	1.95E+02	3.67E+02	3.79E+02
	30	Mean	8.50E+03	5.20E+03	3.69E+03	4.70E+03	3.78E+03	4.80E+03	2.65E+03	9.90E+03	7.07E+03	3.73E+03
		Std	7.01E+02	1.24E+03	3.57E+02	7.14E+02	7.24E+02	7.93E+02	1.75E+03	4.44E+02	6.73E+01	6.74E+02
	50	Mean	1.49E+04	1.21E+04	7.99E+03	8.27E+03	9.06E + 03	8.70E+03	3.76E + 03	1.78E+04	1.18E+04	7.22E+03
		Std	1.72E+03	1.11E+03	3.12E+02	2.20E+03	3.96E + 02	8.36E+02	2.90E+03	6.34E+02	5.35E+01	5.27E+02
f_{11}	10	Mean	4.87E+00	1.26E-01	1.14E-01	1.84E-01	1.06E - 02	1.47E-01	0.00E+00	4.43E+00	2.19E-01	0.00E+00
		Std	2.11E+00	3.95E-01	3.12E-01	9.69E+00	1.19E-02	4.68E + 00	0.00E + 00	1.34E+01	1.04E - 01	0.00E+00
	30	Mean	5.11E+01	8.99E+01	2.97E+01	4.87E+01	6.10E+01	1.14E+02	1.13E+00	2.14E-01	7.69E+01	0.00E+00
		Std	1.42E+01	2.66E + 01	3.86E+00	2.21E+01	1.16E+01	2.99E+01	3.59E+00	2.11E-02	3.03E+01	0.00E+00
	50	Mean	1.34E + 02	2.57E+02	1.23E+02	9.45E+01	1.95E+02	2.48E+02	0.00E+00	2.21E-01	1.36E+02	0.00E+00
		Std	2.92E+01	6.09E+01	1.16E+01	2.45E+01	2.70E+01	3.96E + 01	0.00E+00	1.15E-02	3.15E+01	0.00E+00
f_{12}	10	Mean	4.96E + 00	1.83E+00	5.40E-01	1.82E+01	8.10E-01	1.28E+01	0.00E+00	1.07E+01	2.36E+01	0.00E+00
		Std	1.00E+00	1.85E+00	4.91E-01	4.10E+00	5.03E-01	4.49E+00	0.00E+00	1.38E+01	7.30E+00	0.00E+00
	30	Mean	7.07E+01	6.32E+01	2.23E+01	6.06E + 01	6.03E+01	1.72E+02	0.00E+00	2.28E-01	9.47E+01	0.00E+00
		Std	1.81E + 01	1.98E+01	2.62E+00	1.84E+01	1.36E+01	3.01E+01	0.00E + 00	2.13E - 02	2.84E+01	0.00E+00
	50	Mean	2.11E+02	2.06E+02	8.69E+01	1.67E+02	1.98E+02	3.68E+02	0.00E+00	2.50E-01	2.10E+02	0.00E+00
		Std	6.68E + 01	5.42E+01	8.16E + 00	4.41E+01	1.82E+01	4.69E + 01	0.00E+00	4.35E-02	7.01E+01	0.00E+00
f_{13}	10	Mean	2.50E-01	3.22E-02	1.00E - 02	7.34E-01	6.01E-03	4.48E-01	4.70E-02	8.92E-01	1.98E-01	0.00E+00
		Std	4.11E - 01	3.27E-02	7.92E-03	3.33E-01	6.64E - 03	1.59E-01	9.92E-02	6.31E-02	8.40E-02	0.00E+00
	30	Mean	2.97E - 03	7.77E-02	1.11E+00	1.39E+00	2.13E-04	8.64E-01	1.75E-02	1.00E+00	5.46E-02	0.00E+00
		Std	6.35E-03	1.37E-01	5.47E-02	1.75E-01	4.24E-04	8.53E-02	5.54E-02	9.55E-03	2.87E-02	0.00E+00
	50	Mean	1.55E-02	9.59E-01	1.42E + 01	3.32E+00	2.69E - 01	1.09E+00	0.00E+00	$9.94E_{-01}$	1.04E+00	0.00E+00
		Std	9.84E - 03	1.69E-01	4.73E+00	9.98E-01	7.79E-02	2.58E-02	0.00E+00	9.84E - 03	1.16E-01	0.00E+00

Table 6 Experimental results of nine state-of-the-art algorithms and TLGWO for multimodal functions (D = 10, 30, 50)

Table 6	(conti	inued)										
F	D	Index	OSHH	SADE	MABC	DEKH	sinDE	CMVO	BMWOA	BBOA	DALO	TLGWO
f_{14}	10	Mean	4.59E-11	2.08E-05	2.36E-04	8.28E-02	2.01E-04	2.45E-01	1.66E+00	6.35E-02	3.07E-01	2.51E-251
		Std	6.71E-11	3.06E-05	9.91E-05	5.86E-02	1.76E-04	1.58E-01	2.48E+00	5.87E-03	3.61E-01	0.00E+00
	30	Mean	1.20E-02	3.05E-01	9.02E-01	1.07E+00	6.45E-03	7.19E+00	2.13E-46	1.14E-01	7.68E+00	3.73E-249
		Std	7.07E-03	5.98E-01	1.87E-01	6.62E-01	2.51E-03	2.21E+00	6.55E-46	1.37E-02	4.61E+00	0.00E+00
	50	Mean	1.11E+00	4.34E+00	8.72E+00	5.18E+00	4.41E-01	2.01E+01	4.37E-54	1.59E-01	1.69E+01	5.14E-247
		Std	7.17E-01	5.24E+00	1.75E+00	1.47E+00	4.31E-01	3.96E+00	8.90E-54	1.50E-02	6.15E+00	0.00E+00
f_{15}	10	Mean	0.00E+00	2.95E-02	7.72E-12	2.51E-01	0.00E+00	8.86E-02	0.00E+00	4.50E-02	4.28E-01	0.00E+00
		Std	0.00E+00	9.34E-02	1.02E-11	2.20E-01	0.00E+00	1.24E-01	0.00E + 00	1.47E-03	2.02E-01	0.00E+00
	30	Mean	4.46E-01	1.47E-01	1.14E-01	4.47E-01	8.26E-08	9.91E - 01	0.00E + 00	6.53E-02	1.50E+00	0.00E + 00
		Std	2.20E-01	3.75E-01	5.82E-02	2.79E-01	7.49E-08	4.18E-01	0.00E+00	1.57E-03	3.09E-01	0.00E+00
	50	Mean	2.00E+00	2.64E-01	1.64E+00	9.81E-01	1.83E-03	2.05E+00	0.00E+00	7.43E-02	3.16E+00	0.00E+00
		Std	4.73E-01	3.61E-01	3.17E-01	2.54E-01	2.06E-03	6.35E-01	0.00E+00	1.53E-03	8.91E-01	0.00E+00
f_{16}	10	Mean	2.36E-07	3.58E-04	2.64E+01	4.45E+01	1.32E-04	5.37E-04	9.20E+00	5.04E-02	1.75E-08	0.00E + 00
		Std	4.78E-07	6.72E-04	1.17E+01	2.40E+01	1.39E - 04	3.33E-04	8.98E+00	8.57E-03	1.78E-08	0.00E+00
	30	Mean	7.87E+01	1.80E+02	3.01E+02	4.07E+02	1.56E+02	5.35E-01	5.17E+02	7.92E-02	1.98E+02	0.00E+00
		Std	3.11E + 01	1.10E+02	4.52E+01	6.38E+01	5.14E+01	2.53E-01	1.37E + 02	3.67E-03	4.63E + 01	0.00E + 00
	50	Mean	6.96E+02	6.79E+02	6.38E+02	5.98E+02	6.78E+02	3.13E+01	8.47E+02	8.29E-02	6.02E+02	0.00E+00
		Std	2.81E+02	1.31E+02	6.95E+01	2.28E+02	7.52E+01	1.40E+01	8.03E+01	2.67E-03	1.58E+02	0.00E+00

Table 6	cont	tinued)										
F	D	Index	HPSO	SADE	MABC	DEKH	sinDE	CMVO	BMWOA	BBOA	DALO	TLGWO
f_{17}	10	Mean	1.92E+00	1.21E+00	1.08E+00	1.30E+00	1.52E+00	2.31E+00	1.46E-03	1.30E+00	5.00E-02	0.00E+00
		Std	6.86E-01	5.05E-01	1.64E-01	7.28E-01	3.71E-01	4.45E-01	4.57E-03	2.14E-01	1.56E-01	0.00E+00
	30	Mean	1.07E+01	9.79E+00	7.67E+00	3.33E+00	9.69E+00	1.09E+01	3.08E-01	8.37E+00	1.81E-01	0.00E+00
		Std	1.05E+00	4.45E-01	3.63E-01	8.08E-01	4.45E-01	5.95E-01	6.73E-01	3.91E-01	3.59E-01	0.00E+00
	50	Mean	1.98E+01	1.87E+01	1.64E+01	7.07E+00	1.84E+01	2.05E+01	1.07E-01	1.68E+01	7.51E-01	0.00E+00
		Std	6.06E-01	6.01E-01	4.15E-01	2.02E+00	6.31E-01	6.36E-01	3.14E-01	3.78E-01	8.80E-01	0.00E+00
f_{18}	10	Mean	3.03E-22	3.13E-15	2.35E-10	5.24E+00	8.94E-22	3.18E-02	6.47E-03	4.06E-01	1.79E+00	6.06E-02
		Std	6.41E-22	9.88E-15	2.95E-10	4.15E+00	1.76E-21	9.95E-02	7.99E-03	2.23E-01	1.64E+00	4.51E-02
	30	Mean	1.03E - 02	6.22E+04	4.10E - 01	2.57E+01	6.90E - 04	1.64E+00	2.15E-02	8.41E-01	1.47E+01	6.25E-02
		Std	3.27E-02	1.87E+05	1.78E-01	1.84E+01	2.11E-03	1.19E+00	1.52E-02	2.14E - 01	4.43E+00	4.48E-02
	50	Mean	1.17E-01	1.02E+06	8.51E+03	3.35E+01	5.77E-01	6.44E+00	2.67E-02	1.09E+00	1.91E+01	7.47E-02
		Std	1.31E - 01	3.21E+06	2.27E+04	9.95E+00	2.88E-01	2.82E+00	1.48E - 02	1.50E-01	4.78E+00	3.27E-02
f_{19}	10	Mean	3.23E-21	1.56E-10	1.20E-08	1.93E+01	9.66E-22	4.41E-03	5.14E-02	7.77E-01	5.49E-03	1.55E-01
		Std	6.92E-21	4.83E-10	2.14E - 08	3.44E+01	1.78E - 21	3.82E-03	5.42E-02	2.34E - 01	1.39E-02	7.43E-02
	30	Mean	3.36E - 03	3.29E+04	1.21E+00	2.90E+04	7.78E-05	2.04E - 01	5.25E-01	3.34E+00	2.26E+01	5.57E-01
		Std	5.31E - 03	1.04E+05	4.72E-01	6.02E+04	7.35E-05	1.05E-01	1.03E-01	3.30E-01	2.27E+01	2.23E-01
	50	Mean	1.63E - 01	1.05E+05	3.48E+05	5.25E+04	9.48E + 00	1.31E+01	1.31E+00	5.27E+00	9.94E+01	1.10E+00
		Std	8.22E-02	2.69E+05	5.73E+05	5.30E+04	1.13E+01	1.69E+01	4.35E-01	4.44E-01	2.21E+01	2.58E-01

Table 6	(conti	inued)										
F	D	Index	OS4H	SADE	MABC	DEKH	sinDE	CMVO	BMWOA	BBOA	DALO	TLGWO
f_{20}	10	Mean	1.29E-01	1.09E-01	6.20E-01	1.49E-01	9.98E-02	2.19E-01	1.59E-01	1.99E-01	2.09E-01	0.00E+00
		Std	4.83E-02	3.16E-02	1.13E-01	7.07E-02	4.71E-09	6.32E-02	8.43E-02	2.12E-07	3.16E-02	0.00E+00
	30	Mean	4.59E-01	8.11E-01	8.16E+00	1.20E+00	5.19E-01	7.99E-01	1.09E - 01	2.95E-01	2.87E+00	0.00E+00
		Std	6.99E-02	1.11E-01	4.98E-01	2.92E-01	4.21E-02	9.42E-02	9.93E-02	1.23E-02	9.71E-01	0.00E+00
	50	Mean	7.60E-01	3.00E+00	1.84E+01	4.03E+00	1.47E+00	1.74E+00	1.19E - 01	2.99E-01	7.33E+00	0.00E+00
		Std	1.17E-01	9.15E-01	8.93E-01	8.92E-01	1.70E-01	2.79E-01	1.03E - 01	5.86E-06	1.09E+00	0.00E+00
f_{21}	10	Mean	4.31E-11	7.75E-04	2.57E-04	3.27E-02	1.49E-11	2.70E-01	5.15E-15	6.28E-01	5.18E-01	8.88E-16
		Std	3.41E-11	2.45E-03	1.50E-04	8.71E-03	1.01E-11	4.72E-01	2.24E-15	3.75E-02	8.66E-01	0.00E+00
	30	Mean	4.18E - 01	4.88E-01	4.14E+00	1.81E+01	1.70E - 03	2.11E+00	5.50E-15	7.13E-01	5.33E+00	8.88E-16
		Std	7.00E-01	8.32E-01	3.84E - 01	5.59E+00	1.01E - 03	5.88E-01	1.71E-15	3.10E - 02	3.67E+00	0.00E+00
	50	Mean	1.72E+00	1.59E+00	1.05E+01	1.99E+01	2.93E-01	4.46E+00	4.79E-15	7.23E-01	9.78E+00	8.88E-16
		Std	4.00E - 01	3.15E+00	1.06E+00	4.90E-04	3.06E-01	5.45E+00	2.62E-15	2.71E-02	1.42E+00	0.00E+00



Fig. 4 Convergence curves of ten algorithms with D = 30 on f_4 , f_7 , f_{14} , and f_{17}

fast convergence rate, and its convergence result is closest to the global optimum for most functions.

4.4 Statistical analysis

Statistical tests can be applied in any experimental framework to determine whether a new method significantly improves existing methods. They can evaluate the performance of the algorithms in more aspects to get an objective ranking. In this section, we performed statistical analysis including the Friedman test and mean absolute error test to verify the superiority of the TLGWO. In addition, the computation time of all algorithms was tested and discussed.

4.4.1 Friedman test

The Friedman test [43] is a nonparametric analog used to detect significant differences between algorithms. The Friedman statistic F_f is defined as follows:

$$F_f = \frac{12n}{k(k+1)} \left[\sum_j R_j^2 - \frac{k(k+1)^2}{4} \right]$$
(18)

$$R_j = \frac{1}{n} \sum_i r_i^j \tag{19}$$

where *i* and *j* are the index numbers of the optimization problem and the algorithm, respectively. *n* is the total number of problems. The rank value r_i^j is from 1 (the best result) to *k* (the worst result). R_j is the average rank of the *j*th algorithm. When *n* and *k* are large enough (for example, n > 10 and k > 5), F_f is a χ^2 distribution with k - 1 degrees of freedom.

For the algorithms mentioned in this paper, the Friedman test results are shown in Table 7. It can be seen that the proposed TLGWO achieves the best ranking in all dimensions. The TLGWO achieved the best ranking because the team learning

Dim	Algorithn	n					
	GWO	HGWO	EEGWO	IGWO	HPSO	SADE	MABC
Average	e rank						
10	6.14	8.71	3.90	4.47	7.19	7.19	9.04
30	5.09	8.57	3.38	4.57	8.23	10.23	10.42
50	4.92	7.95	3.16	5.00	9.33	10.38	11.57
Overall	rank						
10	5	9	2	3	7.5	7.5	10
30	5	9	2	3	8	11	12
50	4	7	2	5	9	11	14
Dim	Algorithn	n					
_	DEKH	sinDE	CMVO	BMWOA	BBOA	DALO	TLGWO
Average	e rank						
10	11.85	5.76	10.09	6.33	10.66	10.42	2.57
30	11.61	6.85	10.04	5.04	8.00	10.85	2.04
50	10.71	8.19	10.14	4.26	7.19	10.42	1.73
Overall	rank						
10	14	4	11	6	13	12	1
30	14	6	10	4	7	13	1
50	13	8	10	3	6	12	1

Table 7 Friedman test results of 13 algorithms and TLGWO for 21 benchmark functions (D = 10, 30, 50)

Bold values indicate the best results

mechanism proposed in this study effectively enhanced and balanced the exploration and exploitation capabilities of the algorithm.

The proposed neighbor learning strategy promotes individuals to converge to the optimal solution through the influence of neighbors, which enhances the local search of the algorithm. Because of its superior exploitation capability, the TLGWO achieved the best results on unimodal functions (f_1-f_9) . In addition, the TLGWO obtained the best results on multimodal functions $(f_{10}-f_{21})$, which usually test the exploration strength of the algorithm. This is due to the proposed random learning strategy guiding the search agent to move in more directions through two random individuals, which improves the global search ability.

4.4.2 Mean absolute error test

Mean absolute error (MAE) is a statistic that shows the difference between the estimated value and the true value [32]. It is calculated as follows:

$$E = \frac{1}{N} \sum_{i=1}^{N} |f_i - f|$$
(20)

where N is the number of test functions, f_i is the optimization result obtained by the *i*th algorithm, and f is the global optimum. The mean absolute errors between the optimization results obtained by each algorithm and the global optimum of test functions are shown in Table 8.

When the dimension is 10, the mean absolute error of the IGWO is the smallest. The proposed TLGWO obtains the fourth smallest error. However, when the dimension is increased to 30 and 50, TLGWO has the smallest mean absolute errors. It can be seen that the TLGWO has strong competitiveness, especially in high-dimensional optimization problems. Generally, high-dimensional problems have more local optima. The random learning strategy in the TLGWO distributes some of the search agents to new areas for global search, which is beneficial to escape from local optima. Therefore, the TLGWO is better at solving high-dimensional optimization problems.

4.4.3 Computation time analysis

In this section, the computation time of all algorithms was tested. The population size was set to 30. The stopping condition was that the number of iterations reaches 500. For the 21 benchmark functions with dimension D=30, each algorithm was run 10 times. The average computation time (unit: second) is shown in Table 9. To get the rank of the computation time of each algorithm, we performed the Friedman test on the results in Table 9. The statistical test results are shown in Table 10.

From Table 9, it can be seen that most algorithms except the DALO are within 1 s of computation time on most benchmark functions. This shows that most of the algorithms involved in the study have a very fast computation speed, and the difference in computation time is small. The results in Table 10 show that the computation speed of the TLGWO ranks 9th among the 14 algorithms, whereas

Dim	Algorithm				
	GWO	HGWO	EEGWO	IGWO	HPSO
MAE					
10	6.70E+01	7.67E+01	1.39E+02	1.27E+01	8.76E+01
30	3.08E+02	2.93E+02	5.02E+02	2.16E+02	4.25E+02
50	5.47E+02	5.21E+02	8.55E+02	4.73E+02	8.86E+02
Rank					
10	7	8	13	1	9
30	5	3	8	2	6
50	4	3	6	2	7
Dim	Algorithm				
	SADE	MABC	DEKH	sinDE	CMVO
MAE					
10	2.30E+01	1.10E+02	1.46E+02	2.15E+01	5.80E+01
30	5.29E+03	2.29E+03	4.53E+03	5.32E+02	3.04E+02
50	5.65E+04	5.70E+04	1.05E+04	2.81E+03	8.11E+02
Rank					
10	3	10	14	2	6
30	14	12	13	9	4
50	13	14	11	9	5
Dim	Algorithm				
	BMWOA	BE	BOA	DALO	TLGWO
MAE					
10	5.37E+01	1.2	28E+02	1.17E+02	4.35E+01
30	1.79E+03	4.7	74E+02	5.95E+02	1.79E+02
50	9.61E+03	1.7	75E+04	1.73E+03	3.46E+02
Rank					
10	5	12		11	4
30	11	7		10	1
50	10	12		8	1

 Table 8
 Mean absolute errors of 13 algorithms and TLGWO for 21 benchmark functions (D=10, 30, 50)

Bold values indicate the best results

the HPSO obtains the best ranking. However, the average computation time of the TLGWO is 0.2889 s, which is only 0.1958s longer than the average computation time of the HPSO. In addition, it can be seen from Table 7 in Sect. 4.4.1 that the quality of the optimization results obtained by the TLGWO is much better than that of the HPSO.

The computation time is related to the structure of the algorithm. Due to the new team learning mechanism, the average computation time of the TLGWO increased by 0.1513 s compared to the classical GWO. Since the new mechanism greatly improves the performance, this time consumption is acceptable.

Table 9 C	omputation	ו time (unit	: second) of	13 algorith	ims and TL	GWO for 2	1 benchmark	functions (D = 30)					
F	OSdH	SADE	MABC	DEKH	sinDE	CMVO	BMWOA	BBOA	DALO	GWO	HGWO	EEGWO	IGWO	TLGWO
f_1	0.0438	0.0941	0.1069	0.2954	0.0887	0.1091	0.3805	0.0963	7.2562	0.0946	0.1480	0.0976	0.4633	0.2277
f_2	0.0415	0.0979	0.1056	0.3027	0.0949	0.0932	0.3335	0.0938	7.3059	0.0871	0.1712	0.0907	0.4864	0.2691
f_3	0.1589	0.4315	0.6843	1.0513	0.4431	0.2160	0.6640	0.5575	7.3143	0.2165	0.3860	0.4409	0.8084	0.3772
f_4	0.0415	0.0855	0.1088	0.3790	0.1024	0.1051	0.3287	0.0889	7.1721	0.0863	0.1334	0.0846	0.4537	0.2376
f_5	0.0461	0.1026	0.1342	0.3081	0.1011	0.1044	0.3863	0.1066	7.3609	0.0990	0.1592	0.1088	0.5217	0.2609
f_6	0.1030	0.2700	0.4023	0.6505	0.2683	0.1652	0.4642	0.3199	7.3460	0.1631	0.2677	0.2872	0.6700	0.3115
f_7	0.1038	0.2667	0.4107	0.6605	0.2759	0.1704	0.5056	0.3340	7.4836	0.1529	0.2610	0.2636	0.6523	0.3083
f_8	0.0978	0.2430	0.3638	0.6010	0.2488	0.1556	0.3799	0.3024	7.3635	0.1560	0.2372	0.2395	0.6293	0.3365
f_9	0.0528	0.1236	0.1584	0.3523	0.1312	0.1154	1.2276	0.1445	7.2151	0.0953	0.1651	0.1257	0.5015	0.2673
f_{10}	0.0671	0.1795	0.1781	0.4537	0.1366	0.1197	0.3770	0.1507	7.3018	0.1061	0.1646	0.1439	0.5494	0.2509
f_{11}	0.0526	0.1243	0.1579	0.3480	0.1254	0.1199	0.3521	0.1188	7.3062	0.0956	0.1655	0.1165	0.4852	0.2410
f_{12}	0.0628	0.1508	0.2003	0.4198	0.1500	0.1242	0.4654	0.1692	7.3083	0.1096	0.1801	0.1469	0.5767	0.2453
f_{13}	0.0873	0.1651	0.2535	0.4878	0.1666	0.1299	0.4674	0.2137	7.8037	0.1431	0.1992	0.1882	0.5647	0.2517
f_{14}	0.0470	0.0986	0.1814	0.3538	0.1222	0.1263	0.3504	0.1205	7.4217	0.0922	0.1633	0.1106	0.4554	0.2430
f_{15}	0.3137	0.7677	1.9007	1.7857	0.7362	0.4175	0.5021	1.2283	7.8054	0.3033	0.5178	1.0192	1.0008	0.4772
f_{16}	0.0548	0.1356	0.1877	0.3909	0.1277	0.1005	0.4460	0.1419	7.1676	0.1041	0.1720	0.1248	0.5036	0.2445
f_{17}	0.0642	0.1440	0.1978	0.4295	0.1498	0.1286	0.3783	0.1833	7.2686	0.1049	0.1695	0.1363	0.5187	0.2598
f_{18}	0.2162	0.5707	0.9281	1.2828	0.6080	0.2763	1.3177	0.7354	7.3217	0.2499	0.4793	0.5991	1.0265	0.3842
f_{19}	0.2100	0.5810	0.9436	1.3687	0.5841	0.2798	1.2754	0.7674	7.3529	0.2521	0.4704	0.5870	0.9871	0.3852
f_{20}	0.0392	0.0924	0.1095	0.3015	0.0935	0.1037	0.3222	0.0930	7.3425	0.0873	0.1348	0.0887	0.4601	0.2632
f_{21}	0.0507	0.1167	0.1713	0.3747	0.1157	0.1164	0.3436	0.1337	7.1063	0.0903	0.1436	0.1197	0.4767	0.2250
Average	0.0931	0.2305	0.3755	0.5999	0.2319	0.1561	0.5366	0.2905	7.3488	0.1376	0.2328	0.2438	0.6091	0.2889
Bold value	s indicate t	the best res	ults											

		r				
Algorithm						
GWO	HGWO	EEGWO	IGWO	HPSO	SADE	MABC
Average ra	nk					
2.33	7.09	5.28	12.47	1.04	5.38	9.14
Overall rar	ık					
2	7	4	13	1	5	10
Algorithm						
DEKH	sinDE	CMVO	BMWOA	BBOA	DALO	TLGWO
Average ra	nk					
11.76	5.66	3.90	11.23	7.09	14.00	8.57
Overall rar	ık					
12	6	3	11	8	14	9

Table 10 Friedman test results of computation time

Bold values indicate the best results

5 Engineering applications

To verify the effectiveness of the proposed TLGWO in practical applications, three constrained engineering design problems, namely tension/compression spring design [44], welded beam design [45], and pressure vessel design [46], were selected. In addition, the TLGWO and other algorithms involved in this paper were applied to solve the inverse kinematics (IK) problem of an 8-degree-of-freedom (DOF) serial robot. All algorithms have a population size of 100 and a maximum number of iterations of 1000.

5.1 Constraint handling method

In this paper, the constraint handling method proposed in [47] was applied. An optimization problem with constraints is usually described as follows:

Minimize
$$f(x)$$

Subject to $c_i(x) \ge 0$, $i = 1, 2, ..., I$
 $h_j(x) = 0$, $j = 1, 2, ..., J$
 $x_d^l \le x_d \le x_d^u$, $d = 1, 2, ..., D$
(21)

where f(x) is the fitness function. The dimension of the problem is *D*, and the value range of the *d*th variable is $[x_d^l, x_d^u]$. There are *I* inequality constraints $c_i(x)$ and *J* equality constraints $h_i(x)$.

In order to reduce the competitiveness of solutions that do not satisfy the constraints (infeasible solutions), a penalty function is used to modify the fitness of the solutions. The fitness function F(x) with penalty is designed as follows:

$$F(x) = \begin{cases} f(x), & \text{if } c_i(x) \ge 0 \forall i = 1, 2, \dots, I \\ f_{\max} + \sum_{i=1}^{I} [c_i(x)], & \text{otherwise} \end{cases}$$
(22)

where f_{max} is the fitness value of the worst solution. When the operand is positive, [] returns 0; when the operand is negative, [] returns the absolute value. Equation (22) means that when a solution does not violate any constraints, its fitness is maintained; otherwise, its fitness is modified according to the number and degree of constraints it violates, and the worst solution in the current population.

Figure 5 is an illustration of the constraint handling method. The six solid green dots represent the six solutions in the population, and the solid black line represents the modified fitness values. The fitness values of infeasible solutions are larger than that of feasible solutions, so the penalized solution loses an advantage in the competition to become the best solution.

5.2 Engineering problem I: Tension/compression spring design

The tension/compression spring design problem ("Appendix 1.1") is a minimization constraint problem. The goal of the optimization is to minimize the weight of the spring. The constraints include shear stress, surge frequency, and minimum deflection. This problem has three design variables (see Fig. 6): wire diameter d_w , mean coil diameter d_c , and number of active coils N.

Table 11 shows feasible solutions found by the TLGWO and the other 13 algorithms on the tension/compression spring design problem. The best solution, with a cost of 0.012724, was found by the GWO algorithm, and the proposed TLGWO algorithm found the second-best solution with a cost of 0.012809. In terms of tension/compression spring design, TLGWO demonstrates strong competitiveness.



🙆 Springer

Fig. 6 Tension/compression spring design



Table 11 Experimental results of 13 state-of-the-art algorithms	Algorithm	Optimal va	lues of design	variables	Optimal cost
and TLGWO for the tension/		$d_{ m w}$	$d_{\rm c}$	Ν	
problem	HPSO	0.055283	0.449400	7.387500	0.012893
	SADE	0.054604	0.408640	9.843800	0.014431
	MABC	0.053463	0.388107	10.82120	0.014223
	DEKH	0.057207	0.502590	6.399200	0.013815
	sinDE	0.054203	0.408439	10.43300	0.014920
	CMVO	0.057020	0.499000	6.124000	0.013180
	BMWOA	0.058918	0.556850	5.009500	0.013550
	BBOA	0.056240	0.475100	6.893200	0.013364
	DALO	0.055160	0.446100	7.485400	0.012875
	GWO	0.050116	0.319965	13.83240	0.012724
	HGWO	0.056058	0.471190	6.776500	0.012995
	EEGWO	0.055360	0.396100	11.88630	0.016858
	IGWO	0.051855	0.354521	11.97650	0.013324
	TLGWO	0.050000	0.317103	14.15790	0.012809

Bold values indicate the best results

5.3 Engineering problem II: welded beam design

As shown in "Appendix 1.2," the optimization goal of the welded beam design problem is to minimize the manufacturing cost of the welded beam. There are four design variables for this problem (see Fig. 7): the thickness of the weld h, height of the beam T, length of the attached part of the beam l, and thickness of the beam b. Relevant constraints include the shear stress τ , buckling load on the beam P_c , bending stress in the beam σ , and end deflection of the beam δ .

The experimental results of the TLGWO and the other 13 algorithms for the welded beam design problem are presented in Table 12. For the welded beam design problem, the HGWO achieved the best solution with a cost of 1.695400. The proposed TLGWO found a solution that was very close to the best result, with a cost of 1.727200.

Fig. 7 Welded beam design



 Table 12
 Experimental results of 13 state-of-the-art algorithms and TLGWO on the welded beam design problem

Algorithm	Optimal value	es of design variab	les		Optimal cost
	h	l	Т	b	
HPSO	0.197411	3.315061	10.00000	0.201395	1.820395
SADE	0.213430	6.198200	8.254900	0.257430	2.376900
MABC	0.194050	3.703600	9.753500	0.224880	2.022200
DEKH	0.320250	2.399500	7.638800	0.328250	2.250100
sinDE	0.170170	6.099200	9.047400	0.238460	2.281300
CMVO	0.205611	3.472103	9.040931	0.205709	1.725472
BMWOA	0.131940	8.471700	8.887200	0.212700	2.206600
BBOA	0.188140	4.185900	9.304800	0.204870	1.831500
DALO	0.150170	4.617600	9.036600	0.205730	1.780200
GWO	0.205678	3.471403	9.036964	0.205729	1.724995
HGWO	0.205670	3.254600	9.036700	0.205730	1.695400
EEGWO	0.287580	2.277400	9.367400	0.309460	2.478200
IGWO	0.190190	3.502300	9.319600	0.205130	1.749700
TLGWO	0.195150	3.588500	9.044800	0.205950	1.727200

Bold values indicate the best results

5.4 Engineering problem III: pressure vessel design

The goal of the pressure vessel design problem is to reduce the manufacturing cost of the vessel as much as possible. "Appendix 1.3" describes the objective functions and constraints of the problem. Figure 8 shows the four design variables of the problem, which are the thickness of the shell T_s , thickness of the head T_h , inner radius R, and length of the cylindrical section L.



Fig. 8 Pressure vessel design

Table 13 shows the experimental results of TLGWO and 13 state-of-the-art algorithms on pressure vessel design. As shown in Table 13, TLGWO achieves the best solution at $x^* = (0.785283, 0.388714, 40.56697, 197.6661)$ with an optimal cost of 5938.9395. In summary, TLGWO shows excellent performance and great competitiveness in solving constrained practical engineering problems.

5.5 Engineering problem IV: inverse kinematics solution

In this section, the TLGWO and other 13 algorithms were applied to solve the IK of an 8-DOF serial robot. Figure 9 shows the structure of the robot, which is composed of one translation joint and seven rotation joints. The IK problem can be described as finding a set of joint variables that make the pose of the robot end-effector

Algorithm	Optimal value		Optimal cost		
	$\overline{T_s}$	T_h	R	L	
HPSO	0.916845	0.453197	47.50493	119.4608	6167.1292
SADE	1.567434	2.479227	46.91671	133.3930	19,133.476
MABC	1.662900	0.954586	76.84130	10.28290	15,145.663
DEKH	3.251310	35.98050	65.51910	42.01280	295,354.72
sinDE	2.353231	2.153293	68.79089	164.9708	45,190.344
CMVO	0.787553	0.394033	40.76956	195.9078	5966.0172
BMWOA	0.931951	0.459752	47.32779	121.0679	6303.1592
BBOA	1.321390	0.532168	55.71410	60.98570	7998.8290
DALO	0.826773	0.408675	42.83796	167.7143	5974.4731
GWO	0.812500	0.434500	42.08918	176.7587	6051.5639
HGWO	0.910428	0.450058	47.16467	122.5616	6152.9859
EEGWO	2.804914	0.991835	53.06128	142.1000	29,950.623
IGWO	1.163076	0.723840	52.46822	102.0611	9264.9101
TLGWO	0.785283	0.388714	40.56697	197.6661	5938.9395

Bold values indicate the best results



Fig. 9 8-DOF serial robot

coincide with the pose of the target point. "Appendix 1.4" details the fitness function of this problem, as well as the forward kinematics of the robot.

Many studies [48–50] have shown that using meta-heuristic algorithms to solve IK problems can effectively avoid singularity and complex computation. The pseudocode for solving IK based on meta-heuristic is shown in Algorithm 3. In this work, we randomly selected a set of joint variables to calculate the pose matrix of the end-effector. This pose matrix contains the position and orientation information of the target point. The joint configuration is selected as $[0.5 \text{ m}, 30^\circ, 60^\circ, 90^\circ, 60^\circ, 30^\circ, 60^\circ, 90^\circ]$. The pose matrix of the target point is calculated by forward kinematics as follows:

$${}_{8}^{0}T = \begin{bmatrix} 0.3683 & 0.6998 & 0.6121 & 0.3326 \\ -0.8248 & -0.0580 & 0.5625 & 0.1533 \\ 0.4291 & -0.7120 & 0.5558 & 2.1010 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
(23)

The population size of all algorithms is set to 100. For each algorithm, the IK solution is output after 1000 iterations. Table 14 shows the results of the TLGWO and other 13 meta-heuristic algorithms for solving IK problems, where the unit of

Algo-	Optimal	values of join	t variables						Pose
rithm	$\overline{d_1}$	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8	error
HPSO	0.37583	-0.26874	1.44510	-0.16366	-1.47800	-0.52089	-0.67831	-0.96871	0.01496
SADE	0.00000	2.87980	1.54260	2.87980	-2.12930	-0.89002	-0.33170	-0.69845	0.14019
MABC	0.47298	-0.44856	0.20371	-0.09356	1.07630	2.87980	1.91990	1.69790	0.00824
DEKH	0.08324	-0.73903	1.22270	-2.87540	1.88010	1.53060	-0.12253	0.08721	0.05566
sinDE	0.42699	0.02820	0.15271	-2.66360	-1.28380	-1.02630	1.71670	1.32190	0.00320
CMVO	0.56116	-0.67054	0.40800	-0.44641	0.94484	0.10642	-1.86910	- 1.66930	0.11739
BMWOA	0.14843	-0.66928	1.30030	-2.82760	1.82500	-1.31030	0.42167	2.99730	0.00764
BBOA	0.11408	-2.11440	1.70920	1.68400	-2.12930	-1.43420	-0.30302	0.28462	0.09839
DALO	0.02552	2.63780	3.39220	0.70945	0.00973	-1.37990	0.45240	2.31190	0.18766
GWO	0.14501	1.89670	3.36830	2.07440	1.82480	0.49659	-1.67840	-0.63662	0.00845
HGWO	0.02828	1.74390	3.49070	1.59290	1.94660	-0.06333	-1.46510	-0.19934	0.09369
EEGWO	0.42803	1.12300	1.73380	1.81300	1.44810	2.36340	-0.94844	-0.87596	0.02932
IGWO	0.12232	-1.39710	0.13475	1.76320	- 1.85060	-2.33780	-1.45320	-0.31762	0.00958
TLGWO	0.50094	-2.74130	2.42500	-2.02320	1.03980	1.17890	1.27230	1.36730	0.00147

Table 14 Experimental results of 13 state-of-the-art algorithms and TLGWO on the IK problem

Bold values indicate the best results

 d_1 is meter and the units of $\theta_2 - \theta_8$ are radian. The IK of redundant robots (with more than 6 DOF) has multiple solutions. In other words, there are infinite joint configurations that enable the end-effector to reach the target point represented by Eq. (23). In Table 14, the pose error between the end-effector and the target point is used to evaluate and compare the solutions.

Algorithm 3. Meta-heuristic IK solver
1: Initialize the population and parameters of meta-heuristic algorithm
2: while stopping rule is not true do
3: for each search agent X_i do
4: Calculate the forward kinematics F_i of X_i
5: Calculate the fitness value $f(F_i)$
6: Update search agent X_i by meta-heuristic algorithm
7: Calculate the forward kinematics F_i' of X_i'
8: Calculate the fitness value $f(F_i)$
9: $\mathbf{if} f(F_i) \le f(F_i)$
10: Update search agent $X_i = X_i'$
11: end if
12: end for
13: end while
14: return the best search agent as the IK solution

It can be seen from Table 14 that the TLGWO obtains the best solution with a pose error of 0.00147. Therefore, the proposed TLGWO can effectively solve the IK problem of redundant robots and can find the best solution compared with other state-of-the-art algorithms.

6 Conclusions and future work

This paper proposed an improved team learning-based grey wolf optimizer (denoted as TLGWO) to enhance and balance the exploitation and exploration abilities of the grey wolf optimizer (GWO). The team learning strategies include the neighbor learning strategy and random learning strategy. The neighbor learning strategy assigns neighbors to each search agent. The influence of neighbors on the search agent may be attractive or repulsive, but both will prompt the search agent to move toward the optimal solution, which improves the local search ability of the algorithm. The random learning strategy selects two random individuals in the population to guide the movement of the search agent, which is beneficial to the global search. Furthermore, an adaptive parameter tuning method was proposed to balance the exploitation and exploration of the algorithm.

The proposed TLGWO was tested on optimization tasks and engineering problems. First, 21 benchmark functions were used to prove the superiority of the TLGWO. In a comparison of nine state-of-the-art algorithms and three GWOderived algorithms, the TLGWO achieved the best results in most cases. The Friedman test and mean absolute error statistical test were used to compare and discuss experimental results. Second, four engineering design problems with constraints were used to test the capabilities of the algorithm. The results revealed that the TLGWO provided the best or close to the best results. In summary, the TLGWO proposed in this study exhibits generally superior performance.

In the future, we will further investigate adaptive strategies to improve the performance of the GWO. In addition, the application of meta-heuristic algorithms will be extended to more fields, such as the inverse kinematics and trajectory optimization of robots. We will continue to refine the team learning-based strategy proposed in this paper and apply it to other algorithms.

Appendix 1: Engineering problems

Appendix 1.1: Tension/compression spring design problem

Consider $x = [x_1 x_2 x_3] = [d_w d_c N]$

Minimize $f(x) = (x_3 + 2)x_2x_1^2$ Subject to $x_1^3x_2$

$$g_{1}(x) = 1 - \frac{x_{2}x_{3}}{71785x_{1}^{4}} \le 0,$$

$$g_{2}(x) = \frac{4x_{2}^{2} - x_{1}x_{2}}{12566(x_{2}x_{1}^{3} - x_{1}^{4})} + \frac{1}{5108x_{1}^{2} - 1} \le 0,$$

$$g_{3}(x) = 1 - \frac{140.45x_{1}}{x_{2}^{2}x_{3}} \le 0,$$

$$g_{4}(x) = \frac{x_{1} + x_{2}}{1.5} - 1 \le 0$$

Variable range $0.05 \le x_1 \le 2.00$, $0.25 \le x_2 \le 1.30$, $2.00 \le x_3 \le 15.0$

Appendix 1.2: Welded beam design problem

Consider
$$x = [x_1 x_2 x_3 x_4] = [hlTb]$$

Minimize $f(x) = 1.10471x_1^2 x_2 + 0.04811x_3 x_4 (14.0 + x_2)$
Subject to
 $g_1(x) = \tau(x) - \tau_{max} \le 0,$
 $g_2(x) = \sigma(x) - \sigma_{max} \le 0,$
 $g_3(x) = \delta(x) - \delta_{max} \le 0,$
 $g_4(x) = x_1 - x_4 \le 0,$
 $g_5(x) = P - P_c(x) \le 0,$
 $g_6(x) = 0.125 - x_1 \le 0,$
 $g_7(x) = 1.10471x_1^2 + 0.04811x_3 x_4 (14.0 + x_2) - 5.0 \le 0$
where $\tau(x) = \sqrt{(\tau')^2 + 2\tau' \tau'' \frac{x_2}{2R} + (\tau'')^2},$

$$\begin{split} \tau' &= \frac{P}{\sqrt{2}x_1 x_2}, \ \tau'' = \frac{MR}{J}, \ M = P\left(L + \frac{x_2}{2}\right), \ R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2}, \\ J &= 2\left\{\sqrt{2}x_1 x_2 \left[\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\}, \ \sigma(x) = \frac{6PL}{x_3^2 x_4}, \ \delta(x) = \frac{6PL^3}{Ex_3^2 x_4}, \\ P_c(x) &= \frac{4.013E\sqrt{\frac{x_3^2 x_4^6}{36}}}{L^2} \left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right), \\ P &= 6000 \text{ lb}, \ L = 14 \text{ in.}, \ \delta_{\max} = 0.25 \text{ in.}, \\ E &= 30 \times 10^6 \text{ psi}, \ G = 12 \times 10^6 \text{ psi}, \ \tau_{\max} = 13, 600 \text{ psi}, \ \sigma_{\max} = 30,000 \end{split}$$

Variable range
$$0.1 \le x_1 \le 2.0$$
,
 $0.1 \le x_2 \le 10.0$,
 $0.1 \le x_3 \le 10.0$,
 $0.1 \le x_4 \le 2.0$.

Appendix 1.3: Pressure vessel design problem

Consider $x = [x_1 x_2 x_3 x_4] = [T_s T_h RL]$ Minimize $f(x) = 0.6224 x_1 x_3 x_4 + 1.7781 x_2 x_3^2 + 3.1661 x_1^2 x_4 + 19.84 x_1^2 x_3$ Subject to $g_1(x) = -x_1 + 0.0193 x_3 \le 0,$ $g_2(x) = -x_2 + 0.00954 x_3 \le 0,$ $g_3(x) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1,296,000 \le 0,$ $g_4(x) = x_4 - 240 \le 0$ Variable range $0 \le x_1 \le 99$

Variable range
$$0 \le x_1 \le 99$$
,
 $0 \le x_2 \le 99$,
 $10 \le x_3 \le 200$,
 $10 \le x_4 \le 200$

Appendix 1.4: Inverse kinematics problem

The solution of the IK problem can be expressed as

$$x = [x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8] = [d_1 \theta_2 \theta_3 \theta_4 \theta_5 \theta_6 \theta_7 \theta_8]$$

psi

where d_1 is the joint variable of the translational joint (unit: m), $\theta_2 \sim \theta_8$ are the joint variables of the rotational joints (unit: °).

The fitness function is designed as follows:

$$f(x) = k_1 P_{\text{err}} + k_2 O_{\text{err}}$$

where $k_1 = k_2 = 0.5$ are the weight coefficients. P_{err} is the position error between the robot end-effector and the target point, and O_{err} is the orientation error between the end-effector and the target point. They are calculated as follows:

$$P_{\text{err}} = \left\| P_e - P_t \right\|_2$$
$$O_{\text{err}} = \left\| \varphi_1 \rho_2 - \varphi_2 \rho_1 + \rho_1 \times \rho_2 \right\|_2$$

where P_e and P_t are the position vectors of the end-effector and the target point, respectively. $\{\varphi_1, \rho_1\}$ and $\{\varphi_2, \rho_2\}$ are quaternions corresponding to the orientation matrices of the end-effector and the target point, respectively. When the orientation of the end-effector coincides with the orientation of the target point, $O_{\rm err} = 0$; otherwise, $O_{\rm err} = 1$.

The position and orientation of the end-effector or target point are solved by forward kinematics, which is expressed as follows:

$$f(x) = \begin{bmatrix} R & P \\ 0 & 1 \end{bmatrix} = {}_{8}^{0}T = {}_{1}^{0}T_{2}^{1}T_{3}^{2}T_{4}^{3}T_{5}^{4}T_{6}^{5}T_{7}^{6}T_{8}^{7}T$$

$${}_{i}^{i-1}T = \begin{bmatrix} \cos\theta_{i} & -\sin\theta_{i} & 0 & a_{i-1} \\ \sin\theta_{i}\cos\alpha_{i-1} & \cos\theta_{i}\cos\alpha_{i-1} & -\sin\alpha_{i-1} & -\sin\alpha_{i-1}d_{i} \\ \sin\theta_{i}\sin\alpha_{i-1} & \cos\theta_{i}\sin\alpha_{i-1} & \cos\alpha_{i-1} & \cos\alpha_{i-1}d_{i} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where *R* is the orientation matrix and *P* is the position matrix. $_{i}^{i-1}T$ is the transformation matrix of the coordinate system $\{i\}$ relative to the coordinate system $\{i-1\}$. $_{i}^{i-1}T$ can be obtained from DH parameters of the robot, which are shown in Table 15.

Acknowledgements This research was funded by the National Natural Science Foundation of China under Grant 11672290, 11972343 and 62173047.

Author contributions All authors contributed to the study conception and design. The theoretical research and test experiments of the proposed algorithm were completed by JC. Material preparation, data collection, and analysis were performed by JC, TL, and MZ. The first draft of the manuscript was written by JC, and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Data availability All data generated or analyzed during this study are included in this published article.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Table 15 robot	DH parameters of the	Joint i	$\alpha_{i-1}(^{\circ})$	a_{i-1} (m)	$d_i(\mathbf{m})$	$\theta_i(^\circ)$	Range
		1	-90	0	d_1	-90	0 to 1 m
		2	180	1.525	0	θ_2	-165° to 165°
		3	90	0	0	θ_3 +90	-20° to 200°
		4	90	0	0.3635	$ heta_4$	-165° to 165°
		5	-90	0	0	θ_5	-122° to 122°
		6	90	0	0.418	θ_6	-165° to 165°
		7	-90	0	0	θ_7	-110° to 110°
		8	90	0	0.265	θ_8 +90	-175° to 175°
		-					

References

- Jones DF, Mirrazavi SK, Tamiz M (2002) Multi-objective meta-heuristics: an overview of the current state-of-the-art. Eur J Oper Res 137(1):1–9. https://doi.org/10.1016/S0377-2217(01)00123-0
- Wang GG, Cai XJ, Cui ZH, Min GY, Chen JJ (2020) High performance computing for cyber physical social systems by using evolutionary multi-objective optimization algorithm. IEEE Trans Emerg Top Comput 8(1):20–30. https://doi.org/10.1109/TETC.2017.2703784
- Mirjalili S, Mirjalili SM, Hatamlou A (2016) Multi-Verse optimizer: a nature-inspired algorithm for global optimization. Neural Comput Appl 27:495–513. https://doi.org/10.1007/s00521-015-1870-7
- Kaveh A, Dadras A (2017) A novel meta-heuristic optimization algorithm: thermal exchange optimization. Adv Eng Softw 110:69–84. https://doi.org/10.1016/j.advengsoft.2017.03.014
- Rashedi E, Nezamabadi PH, Saryazdi S (2009) GSA: a gravitational search algorithm. Inf Sci 179:2232–2248. https://doi.org/10.1016/j.ins.2009.03.004
- Kaveh A, Khayatazad M (2012) A new meta-heuristic method: ray optimization. Comput Struct 112:283–294. https://doi.org/10.1016/j.compstruc.2012.09.003
- Kennedy J, Eberhart R (1995) Particle swarm optimization. IEEE Int Conf Neural Netw Proc 1995:1942–1948. https://doi.org/10.1109/ICNN.1995.488968
- Gandomi AH, Alavi AH (2012) Krill herd: A new bio-inspired optimization algorithm. Commun Nonlinear Sci Numer Simul 17(12):4831–4845. https://doi.org/10.1016/j.cnsns.2012.05.010
- Storn R, Price K (1997) Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11(4):341–359. https://doi.org/10.1023/A:1008202821 328
- Dervis K, Celal O (2011) A novel clustering approach: Artificial Bee Colony (ABC) algorithm. Appl Soft Comput 11(1):652–657. https://doi.org/10.1016/j.asoc.2009.12.025
- 11. Mirjalili S (2015) The ant lion optimizer. Adv Eng Softw 83:80-98
- 12. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51-67. https://doi.org/10.1016/j.advengsoft.2016.01.008
- Arora S, Singh S (2019) Butterfly optimization algorithm: a novel approach for global optimization. Soft Comput 23(3):715–734. https://doi.org/10.1007/s00500-018-3102-4
- Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61. https:// doi.org/10.1016/j.advengsoft.2013.12.007
- Qiang T, Xue-Chen C, Xing-Cheng L (2019) Multi-strategy ensemble grey wolf optimizer and its application to feature selection. Appl Soft Comput 76:16–30. https://doi.org/10.1016/j.asoc.2018. 11.047
- 16. Khairuzzaman AK, Chaudhury S (2017) Multilevel thresholding using grey wolf optimizer for image segmentation. Expert Syst Appl 86:64–76. https://doi.org/10.1016/j.eswa.2017.04.029
- Jayabarathi T, Raghunathan T, Adarsh BR, Suganthan PN (2016) Economic dispatch using hybrid grey wolf optimizer. Energy 111:630–641. https://doi.org/10.1016/j.energy.2016.05.105
- Ma RQ, Karimzadeh M, Ghabussi A, Zandi Y, Baharom S, Selmi A, Maureira-Carsalade N (2021) Assessment of composite beam performance using GWO-ELM metaheuristic algorithm. Eng Comput. https://doi.org/10.1007/s00366-021-01363-1

- Shehata AAA, Tolba MAA, El-Rifaie AMM, Korovkin NV (2022) Power system operation enhancement using a new hybrid methodology for optimal allocation of FACTS devices. Energy Rep 8:217–238. https://doi.org/10.1016/j.egyr.2021.11.241
- Singh S, Bansal JC (2022) Mutation-driven grey wolf optimizer with modified search mechanism. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2021.116450
- Inac T, Dokur E, Yuzgec U (2022) A multi-strategy random weighted grey wolf optimizerbased multi-layer perceptron model for short-term wind speed forecasting. Neural Comput Appl. https://doi.org/10.1007/s00521-022-07303-4
- Nadimi-Shahraki MH, Taghian S, Mirjalili S, Zamani H, Bahreininejad A (2022) GGWO: gaze cues learning-based grey wolf optimizer and its applications for solving engineering problems. J Comput Sci. https://doi.org/10.1016/j.jocs.2022.101636
- Adhikary J, Acharyya S (2022) Randomized balanced grey wolf optimizer (RBGWO) for solving real life optimization problems. Appl Soft Comput. https://doi.org/10.1016/j.asoc.2022.108429
- 24. Hosseini-Hemati S, Beigvand SD, Abdi H, Rastgou A (2022) Society-based grey wolf optimizer for large scale combined heat and power economic dispatch problem considering power losses. Appl Soft Comput. https://doi.org/10.1016/j.asoc.2021.108351
- Meidani K, Hemmasian A, Mirjalili S, Farimani AB (2022) Adaptive grey wolf optimizer. Neural Comput Appl 34(10):7711–7731. https://doi.org/10.1007/s00521-021-06885-9
- Liu X, Wang N (2021) A novel gray wolf optimizer with RNA crossover operation for tackling the non-parametric modeling problem of FCC process. Knowl-Based Syst 216:106751. https:// doi.org/10.1016/j.knosys.2021.106751
- Makhadmeh SN, Khader AT, Al-Betar MA, Naim S, Abasi AK, Alyasseri ZAA (2021) A novel hybrid grey wolf optimizer with min-conflict algorithm for power scheduling problem in a smart home. Swarm Evol Comput 60:100793. https://doi.org/10.1016/j.swevo.2020.100793
- Zhang XM, Lin QY, Mao WT, Liu SW, Dou Z, Liu GQ (2021) Hybrid Particle Swarm and Grey Wolf Optimizer and its application to clustering optimization. Appl Soft Comput 101:107061. https://doi.org/10.1016/j.asoc.2020.107061
- Gupta A, Trivedi A, Prasad B (2022) B-GWO based multi-UAV deployment and power allocation in NOMA assisted wireless networks. Wirel Netw 28:3199–3211. https://doi.org/10.1007/ s11276-022-03045-2
- Rajput SS (2022) S-GWO-FH: sparsity-based grey wolf optimization algorithm for face hallucination. Soft Comput 26:9323–9338. https://doi.org/10.1007/s00500-022-07250-1
- Wen L, Jian-Jun J, Xi-Ming L, Ming-Zhu T (2018) An exploration-enhanced grey wolf optimizer to solve high-dimensional numerical optimization. Eng Appl Artif Intell 68:63–80. https://doi. org/10.1016/j.engappai.2017.10.024
- Mohammad H, Shahraki N, Taghian S, Mirjalili S (2021) An improved grey wolf optimizer for solving engineering problems. Expert Syst Appl 166:113917. https://doi.org/10.1016/j.eswa. 2020.113917
- Ratnaweera A, Halgamuge SK, Watson HC (2004) Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients. IEEE Trans Evol Comput 8(3):240–255. https://doi.org/10.1109/TEVC.2004.826071
- Brest J, Greiner S, Boskovic B, Mernik M, Zumer V (2006) Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. IEEE Trans Evol Comput 10(6):646–657. https://doi.org/10.1109/TEVC.2006.872133
- Akay B, Karaboga D (2012) A modified Artificial Bee Colony algorithm for real-parameter optimization. Inf Sci 192:120–142. https://doi.org/10.1016/j.ins.2010.07.015
- Wang GG, Gandomi AH, Alavi AH, Hao GS (2014) Hybrid krill herd algorithm with differential evolution for global numerical optimization. Neural Comput Appl 25:297–308. https://doi.org/ 10.1007/s00521-013-1485-9
- Draa A, Bouzoubia S, Boukhalfa I (2015) A sinusoidal differential evolution algorithm for numerical optimization. Appl Soft Comput 27:99–126. https://doi.org/10.1016/j.asoc.2014.11. 003
- Ewees AA, Abd El Aziz M, Hassanien AE (2019) Chaotic multi-verse optimizer-based feature selection. Neural Comput Appl 31(4):991–1006. https://doi.org/10.1007/s00521-017-3131-4
- Heidari AA, Aljarah I, Faris H, Chen HL, Luo J, Mirjalili S (2020) An enhanced associative learning-based exploratory whale optimizer for global optimization. Neural Comput Appl 32(9):5185– 5211. https://doi.org/10.1007/s00521-019-04015-0

- Sharma TK (2020) Enhanced butterfly optimization algorithm for reliability optimization problems. J Ambient Intell Hum Comput 12(7):7595–7619. https://doi.org/10.1007/s12652-020-02481-2
- Dong H, Xu YL, Li XP, Yang ZL, Zou CH (2021) An improved antlion optimizer with dynamic random walk and dynamic opposite learning. Knowl-Based Syst. https://doi.org/10.1016/j.knosys. 2021.106752
- Wei-Feng G, Liu S-Y, Ling-Ling H (2013) A novel artificial bee colony algorithm based on modified search equation and orthogonal learning. IEEE Trans Cybern 43(3):1011–1024. https://doi.org/ 10.1109/TSMCB.2012.2222373
- 43. Derrac J, Garcia S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol Comput 1(1):3–18. https://doi.org/10.1016/j.swevo.2011.02.002
- Belegundu AD, Arora JS (1985) A study of mathematical programming methods for structural optimization. Int J Numer Methods Eng 21(9):1583–1599. https://doi.org/10.1002/nme.1620210904
- 45. Coello CAC (2000) Use of a self-adaptive penalty approach for engineering optimization problems. Comput Ind 41(2):113–127. https://doi.org/10.1016/S0166-3615(99)00046-9
- Kannan BK, Kramer SN (1994) An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design. J Mech Des 116(2):405– 411. https://doi.org/10.1115/1.2919393
- Deb K (2000) An efficient constraint handling method for genetic algorithms. Comput Methods Appl 186(2–4):311–338. https://doi.org/10.1016/S0045-7825(99)00389-8
- Dereli S, Koker R (2020) A meta-heuristic proposal for inverse kinematics solution of 7-DOF serial robotic manipulator: quantum behaved particle swarm algorithm. Artif Intell Rev 53:949–964. https://doi.org/10.1007/s10462-019-09683-x
- Deng H, Xie C (2021) An improved particle swarm optimization algorithm for inverse kinematics solution of multi-DOF serial robotic manipulators. Soft Comput 25:13695–13708. https://doi.org/ 10.1007/s00500-021-06007-6
- Rokbani N, Mirjalili S, Slim M, Alimi AM (2022) A beta salp swarm algorithm meta-heuristic for inverse kinematics and optimization. Appl Intell 52:10493–10518. https://doi.org/10.1007/ s10489-021-02831-3

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

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.