KSCE Journal of Civil Engineering (2020) 24(12):3703-3713
DOI 10.1007/s12205-020-0504-5

pISSN 1226-7988, eISSN 1976-3808
www.springer.com/12205

Information Technology

# Modified Whale Optimization Algorithm Based on Tent Chaotic Mapping and Its Application in Structural Optimization

**Yancang Li**[a], **Muxuan Han**[a], and **Qinglin Guo**[a]

[a]*College of Civil Engineering, Hebei University of Engineering, Handan 056038, China*

**ARTICLE HISTORY**

Received 24 March 2020
Revised 1st 7 July 2020
Revised 2nd 24 July 2020
Accepted 27 July 2020
Published Online 13 October 2020

**KEYWORDS**

Whale optimization algorithm
Tent chaotic mapping
Selection strategy
Standard function
0-1 Knapsack Problem
Combinatorial optimization

**ABSTRACT**

In this paper, a novel modified whale optimization algorithm based on Tent chaos map and tournament selection strategy (MWOA) was presented. The aim of the improved algorithm is to reduce the possibility of the standard whale algorithm falling into local optimal. During the initialization of the population, in order to increase population diversity and randomness, MWOA cites the Tent chaos map. In the optimization process, in order to improve the development ability of the standard algorithm, the tournament selection strategy was employed to improve the algorithm accuracy. Numerical simulation and example calculation results show that the improved algorithm is superior to the standard WOA algorithm. The improved method provides a new method for truss structure optimization.

## 1. Introduction

The field of computational intelligence is advancing rapidly, and this makes the methods and algorithms of structural optimization design are constantly improving. In recent years, many scholars have worked on optimization algorithms for optimizing structures. Naderi et al. (2020) proposed a new meta-heuristic optimization algorithm called ToP-Up Technique for the size and shape optimization of truss structures. Kalemci et al. (2020) based on ACI 318-05 code and Rankine earth pressure theory, designed the retaining wall as an optimization problem. They designed a low-weight cantilever reinforced concrete retaining wall with shear key by using an optimization algorithm, which is programmed in MATLAB. Pelusi et al. (2020) used genetic algorithm to optimize the geometry of a concrete arch bridge with deck slab. Based on the volume of material, they put forward a finite element numerical model to represent the arch structure, so as to minimize the cost.

In 2016, Australian researcher Mirjalili (Mirjalili and Lewis, 2016) proposed a highly efficient heuristic Algorithm: the Whale Optimization Algorithm (WOA), inspired by the feeding behavior of marine animals such as whales. Many scholars have done a lot of research on this new algorithm. Kaur et al. (2018) introduced chaos mapping into the original algorithm. Elaziz and Mirjalili (2019) used the differential evolution algorithm (DE) to automatically select a chaotic map and a part of the population to alleviate the time-consuming shortcomings of the standard WOA. Jain et al. (2020) proposed a new whale optimization algorithm based on social network (SNWOA). Agrawal et al. (2020) proposed to integrate the quantum concept with the whale optimization algorithm (QWOA). Hao et al. (2020) proposed a hybrid strategy improved whale optimization algorithm. Zheng et al. (2019) used levy flight method to replace the random selection of parameters of the traditional WOA. Liu and He (2019) introduced the global exploration and local development capability of the adaptive probability threshold coordination algorithm. Wu et al. (2020) designed a convergence factor based on exponential function to coordinate the exploration and development ability of whale optimization algorithm. All these researches have effectively promoted the development of the algorithm, but it is still necessary to make further efforts to improve the efficiency of the algorithm and overcome the premature defects of the algorithm.

**CORRESPONDENCE** Yancang Li liyancang@hebeu.edu.cn College of Civil Engineering, Hebei University of Engineering, Handan 056038, China

This paper aims to improve the modified Whale Optimization Algorithm (MWOA) to make up for the shortcomings of the standard Whale Optimization Algorithm. Firstly, the Tent map was used to initialize the population to make it evenly distributed and expand the search range of whales. Secondly, adaptive dynamic adjustment of chaotic search space was adopted to improve the search accuracy. Finally, a tournament selection strategy was introduced to screen the better solution, so as to better balance the global search and local development capabilities. MWOA is proved to be robust and intelligent by the standard function and the test to solve the 0-1 Knapsack Problem.

## 2. Whale Optimization Algorithm

WOA is developed by simulating the hunting behavior of humpback whales. Humpback whales have a distinctive hunting method – bubble net foraging (Bozorgi and Yazdani, 2019; Memarzadeh et al., 2020). It forages through the mechanisms of continuous contraction and encircling, spiral position updating and random hunting (Jiang et al., 2020), as shown in Figs. 1 and 2.

The specific location update process is as follows:

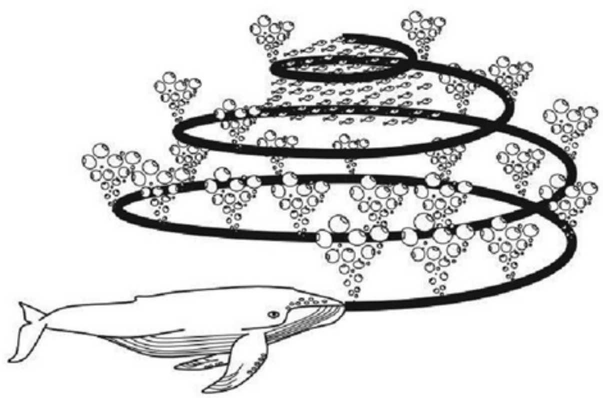

**Fig. 1.** Spiral Predation Mechanism

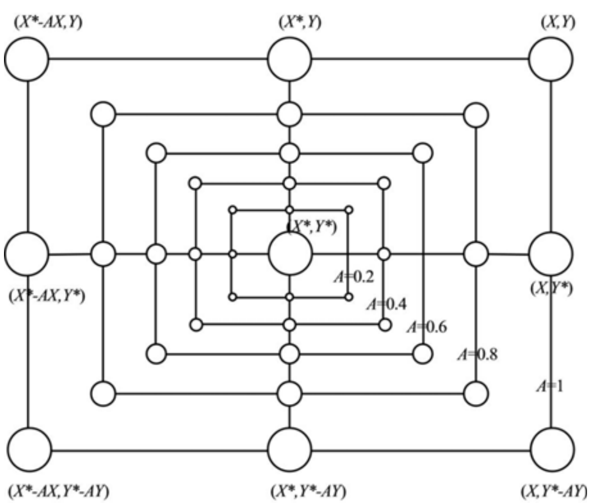


**Fig. 2.** Contraction Wrapping Mechanism

### 2.1 Wandering for Food

In the process of hunting, whales need to determine the location of prey to surround and capture prey, but the location of prey in the search space is usually unknown. Individuals were randomly selected from the whale population for positional updates.

$$D = |C \cdot X_{rand}(t) - X(t)| \tag{1}$$

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

where $X_{rand}$ represents the individual position of any whale, $X(t)$ represents the current whale position, $A \cdot D$ is the step size of the whale position update, and parameter $C$ is a disturbance quantity.

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

$$C = 2r \tag{4}$$

where $r$ represents the random variable between 0 and 1, and parameter $a$ represents the variable linearly reduced from 2 to 0.

### 2.2 Spiral Predation

When $|A| > 1$, the WOA algorithm does not select the target prey to update its position, but randomly selects a search body in the population to replace the target prey.

$$D = |C \cdot X_p(t) - X(t)| \tag{5}$$

$$X(t+1) = X_p(t) - A \cdot D \tag{6}$$

where $X_p(t)$ represents the position of the current optimal whale individual.

### 2.3 Enveloping Contraction

Based on the predation behavior of humpback whale bubble net, the WOA algorithm designs two strategies: contraction loop and spiral update position. The contraction is realized by reducing the value of convergence factor $a$ in Eq. (3), in which the value of $a$ is a random vector in the $[-a, a]$ interval. If $|A| \le 1$, the whale individual after the updated position approaches the target prey position from the original position, and completes the contraction encircle of the prey according to Eq. (2).

$$D' = |X_p(t) - X(t)| \tag{7}$$

$$X(t+1) = D' \cdot e^{bt} \cdot \cos(2\pi l) + X_p(t) \tag{8}$$

where $D'$ represents the length of the distance between individual $X$ and the optimal individual $X_p$ before the position update, $b$ is the spiral constant, generally 1, and $l$ is a random number on the interval $[-1, 1]$.

As the whale moves along the spiral path, its circle shrinks. In order to simulate this synchronization process, WOA chooses the same probability $p$ to update the position of contraction circle and spiral.

$$X(t+1) = \begin{cases} X_p(t) - A \cdot D & p < 0.5 \\ D' \cdot e^{bt} \cdot \cos(2\pi l) + X_p(t) & p \ge 0.5 \end{cases} \tag{9}$$

where $p$ represents a random number between [0, 1].

## 3. Modified Whale Optimization Algorithm

Gharehchopogh and Gholizadeh (2019) gives an overview of WOA applications used to solve various optimization problems. For example, Rajeshkumar and Kousalya (2017) combines back propagation neural network with WOA, and the method has fast convergence speed and high precision, which solves the local minimum capture problem that affects the solution quality. Kaur et al. (2018) hybridized WOA in order to design CWOA chaos theory. In CWOA, the parameter $p$ was adjusted by using chaos mapping to improve the convergence speed of the algorithm. Sayed (2018) used chaos to control WOA's random parameter values, and the exploration operators with chaos and modification performed best. In this paper, we use tent map to initialize the population and adopt the adaptive dynamic adjustment of chaotic search space to improve the search accuracy.

### 3.1 Chaotic Sequence Initialization

#### 3.1.1 Chaotic Mapping

Chaotic mapping (Batista and Viana, 2020; Mansouri and Wang, 2020; Shahna and Mohamed, 2020) is a complex dynamic behavior of nonlinear system. Chaos is unpredictable and aperiodic, which can be used to improve the optimization efficiency of the algorithm (Manjit et al., 2020). The basic idea is to map the optimization variable linearly to the chaos variable through the chaos map, then carry out the optimization search according to the ergodicity and randomness of the chaos, and finally convert the obtained solution linearly to the optimization variable space (Samar et al., 2020).

In this paper, the Tent map is used to optimize the search by using the randomness, ergodicity and regularity of chaotic variables, which makes the algorithm jump out of the local optimum, keeps the diversity of groups and improves the global searching ability. After Bernoulli shift transformation, the mathematical expression is

$$x_{n+1} = (2x_n) mod 1 . \tag{10}$$

The basic steps to generate Tent mapping chaotic sequence in the feasible domain are as follows:

Step 1: generate the initial value $x_0$ randomly (pay attention to avoid $x_0$ falling into a small period), denoted as the flag group $z$, $z_1 = x_0$, $i = j = 1$;

Step 2: iterate according to Eq. (10), each time self-increment 1, it will produce an $x$ sequence;

Step 3: if $t \geq T$, go to step 5; On the contrary, if it falls into the unstable period point, it goes to step 2;

Step 4: change the initial value of the iteration by $x_i = z_{j+1} = z_j + \varepsilon$, where, $\varepsilon$ is a random number, $i = j + 1$, and turn to step 2;

Step 5: at the end of the run, save the $x$ sequence produced by the iteration.

#### 3.1.2 Adaptive Chaotic Search

In this paper, Tent traversal uniformity is used to propose a whale algorithm based on Tent chaotic mapping, and its chaotic search idea is as follows (Kuang et al., 2014):

(1) Tent chaotic sequence is used to initialize the population.

The random distributed initial population was generated by Tent mapping to ensure the randomness of individuals in the initial population. As a result, the whales' search area becomes larger and the diversity of their group position increases.

(2) Adaptive dynamic adjustment of chaotic search space is adopted.

This paper assumes that fall into local optimal value is $X_k = (x_{k1}, ..., x_{kD})$, $x_{ki} \in [X^i_{min}, X^i_{max}]$. All the food sources of the current evolution $k$ generation were sorted from large to small according to the adaptive value. Take $n$ whales (accounting for 60% of the total number of whales), and calculate the minimum value $X^i_{min}$ and maximum value $X^i_{max}$ of the ith dimension of the $n$ whales respectively as the chaotic search space. Tent chaotic sequence is generated from the current optimal solution as the basic solution, and the optimal solution in the sequence is used to update the position of the food source to make it jump out of the local optimal. The main steps of Tent chaos search are as follows:

Step 1: use Eq. (11) to map $x_k$ to (0, 1), where $k = 1, \cdots, n$, $i = 1, \cdots, D$;

$$z^0_{ki} = (x_{ki} - X^i_{min})/(X^i_{max} - X^i_{min}) \tag{11}$$

Step 2: substitute the above formula into the Eq. (10) for iteration to generate the mixed variable sequence $z^m_{ki}$, ($m = 1, 2, \cdots, C_{max}$), and $C_{max}$ is the maximum iteration number of chaos search;

Step 3: use Eq. (12) to generate new solution $V_k$ by transferring $z^m_{ki}$ carrier into the neighborhood of the original solution space;

$$V_k = x_{ki} + (X^i_{max} - X^i_{min})/2 \times (2z^m_{ki} - 1) \tag{12}$$

Step 4: calculate the $F(V_k)$ and retain the best solution;

Step 5: determine whether the maximum number of chaotic searches is reached. If so, the chaos search will end; otherwise, turn to step 2.

### 3.2 Tournament Selection Strategy

In evolutionary algorithms, greedy selection strategy and tournament selection strategy (Chang and Zhao, 2018) are the two most commonly used selection strategies. The tournament selection strategy has many successful examples of algorithm improvement. Kılıç (Kılıç and Yüzgeç, 2019a) introduced the tournament selection strategy into the ant colony algorithm. Through ten standard function tests, it was proved that the improved algorithm could solve the problems of long running time and premature convergence, and improved the optimization performance of the algorithm. Fu (Fu et al., 2013) used the championship selection strategy to sort the randomly selected mutation vectors and select the basis vectors to improve the convergence rate and maintain the diversity of the population.

The tournament selection strategy (Chu et al., 2018; Kılıç and Yüzgeç, 2019b) can be specifically described as: $q$ individuals were randomly selected from the population for comparison according to the probability $P_i(t)$, and the individuals with large fitness were regarded as the optimal individuals. The algorithm in this paper uses this strategy to select feeding sources for whales, and takes $q = 2$, and adds 1 point to the individuals with high fitness, and repeats this process for all individuals, with the highest final score and the largest weight.

$$P_i(t) = c_i(t) / \sum_{i=1}^{N} c_i(t) \tag{13}$$

where $c_i(t)$ is the score of each individual.

### 3.3 Execution of MWOA

The specific process of MWOA is as follows:

Step 1: parameter initialization: set $N$ and $T$;

Step 2: initialize the population and set the initial value of times of substitution $t = 0$. The Tent chaotic mapping steps in section 2.1.1 are used to initialize the population;

Step 3: the whales used Eqs. (1) – (4) for foraging;

Step 4: update the individual position according to Eq. (9);

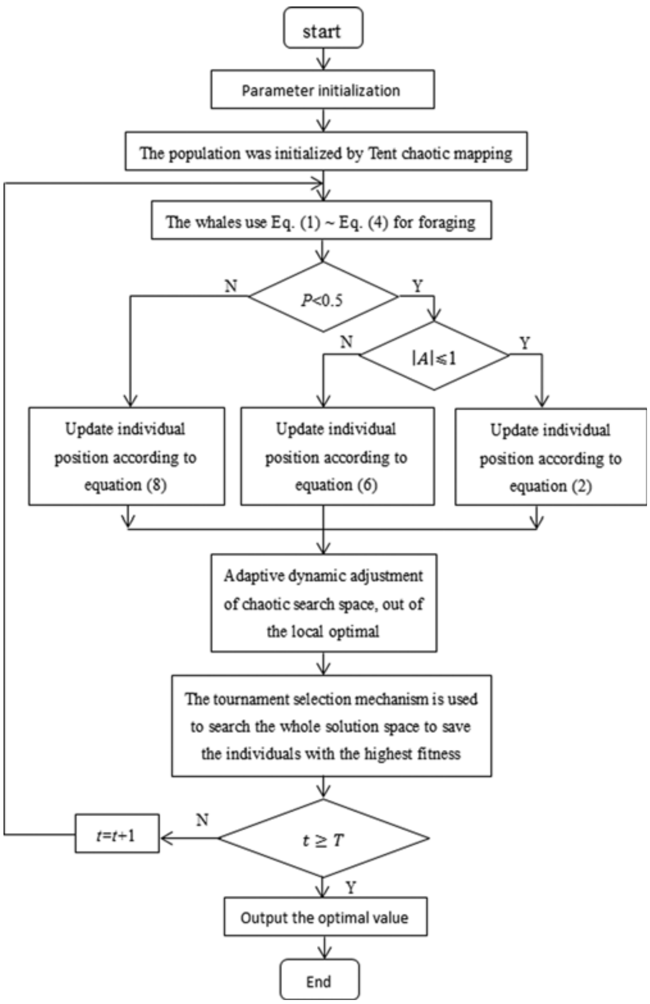


**Fig. 3.** MWOA Iterative Graph

Step 5: adaptive dynamic adjustment of chaotic search space according to section 2.1.2 to jump out of local optimization;

Step 6: search the whole solution space according to the tournament selection mechanism in section 2.2, and save the individuals with the highest fitness;

Step 7: decide whether $t \geq T$ is true or not. If so, the calculation will be stopped. Otherwise, $t = t + 1$, and step 3.

The MWOA flow chart is shown in Fig. 3.

### 3.4 Complexity Analysis of MWOA

In WOA, the time complexity is mainly affected by population size $N$, maximum number of iterations $T_{max}$ and search space dimension $D$, so the complexity is $O(T_{max} \cdot N \cdot D)$. The MWOA is improved from the standard WOA. According to the MWOA flow chart, the introduced Tent chaotic map increases the computation of $O(T_{max} \cdot N \cdot D)$. The tournament selection strategy introduced to avoid algorithms falling into local optimality increases the computation of $O(T_{max} \cdot N)$. Therefore, the time complexity of MWOA is $O(2T_{max} \cdot N \cdot D + T_{max} \cdot N)$ higher than that of standard WOA. When the spatial dimension of optimization problem is high, the time complexity of MWOA is approximately $O(T_{max} \cdot N \cdot D)$, which is consistent with the standard WOA. In addition, the space complexity is mainly affected by population size $N$ and search space dimension $D$, and the space complexity of both algorithms is $O(N \cdot D)$.

## 4. Simulation Tests

In this paper, six standard functions and 0-1 Knapsack Problem are selected to test MWOA to prove the effectiveness of its improvement (Li and Wang, 2019). Initialization parameters: population size $N = 200$, maximum number of iterations $T = 1{,}000$.

### 4.1 Solve the 0-1 Knapsack Problem

The 0 – 1 Knapsack Problem is a typical discrete optimization problem. The mathematical model is as follows:

$$max\ f(x) = \sum_{i=1}^{n} p_i \cdot x_i, \tag{14}$$

$$s.t. \begin{cases} \sum_{i=1}^{n} v_i \cdot x_i < V_{max} \\ x_i \in \{0,1\},\ i = 1, 2, \cdots, n, \end{cases} \tag{15}$$

where $n$ is the number of items, $x_i$ is the state, $v_i$ is the volume, $V_{max}$ is the maximum capacity of the backpack, $p_i$ is the value. 0 means not being loaded into the backpack, 1 means being loaded into the backpack. Partial loading and repeated loading are not allowed.

The basic parameter settings of the example are shown in Table 1.

The comparative results of 0-1 Knapsack Problem are shown in Table 2, which proves the superiority of MWOA.

**Table 1.** Comparison Table of Basic Parameters of 0-1 Knapsack Problem

| Parameter | The parameter value |
|---|---|
| Goods number | $n = 50$ |
| Backpack capacity | $V_{max} = 1{,}000$ |
| Goods value | $P$ = [220,208,198,192,180,65,162,160,158,155,130,125,122,120,118,115,110,105,101,100,100,98,96,95,90,88,82,80,77,75, 73,72,70,69,66,65,63,60,58,56,50,30,20,15,10,8,5,3,1] |
| Goods weight | $V$ = [80,82,85,70,72,72,66,50,55,25,50,55,40,48,50,32,22,60,30,32,40,38,35,32,25,28,30,22,50,30,45,30,60,50,20,65,20,25, 30,10,20,25,15,10,10,10,4,4,2,1] |

**Table 2.** Result Comparison Table of 0 – 1 Knapsack Problem

| Algorithm | Max | Min | Mean | Standard deviation |
|---|---|---|---|---|
| WOA | 3,176 | 3,158 | 3,162 | 2.57E-01 |
| MWOA | 3,275 | 3,257 | 3,257 | 0.00E-00 |

### 4.2 Standard Function Tests

In Table 3, Bukin function and Bohachevsky function are single peak functions, which are mainly used to test the convergence performance of the improved algorithm in the operation process. Shubert function and Drop-wave are complex multi-peak function, which are easy to make the algorithm fall into the local optimal value, so as to get the real optimal value. The Holder-table and Rastrigin function have complex spatial property, which are used to test the computational accuracy, convergence stability and time complexity of the improved algorithm. The function images are shown in Fig. 4.

For the fairness of comparison, the improved algorithm is compared with Genetic Algorithm (GA) (Guo et al., 2017) and other improved whale optimization algorithms (Guo et al., 2017; He et al., 2019). According to the results in Table 4, the parameters of Genetic Algorithm and improved whale algorithm are still selected from the references (Guo et al., 2017; He et al., 2019).

In order to more intuitively reflect the advantages of MWOA performance, Fig. 5 shows the running results under the above six standard functions.

Convergence analysis: In the optimization process of six test functions, the convergence speed and convergence precision of MWOA are better than the other four comparison algorithms when the number of iterations is the same in the early iteration. At the late stage of optimization, by analyzing the mean solution corresponding to Table 2, MWOA found the optimal value, which was no less than the other four algorithms.

Robustness analysis: The robustness of the algorithm can be reflected from the size of standard deviation. The smaller the deviation is, the better the robustness is. On the contrary, the robustness of the algorithm is worse. In the column of standard deviation in Table 2, in the optimization process of $f_1$, $f_2$ and $f_4$, the standard deviation of MWOA is 0, showing good robustness. In the process of $f_3$ optimization, the standard deviation is smaller than GA and WOA and slightly larger than MS-WOA. In the optimization process of $f_5$ and $f_6$, the standard deviation of MWOA is smaller than that of WOAWC and slightly larger than that of MS-WOA.

To sum up, MWOA overall optimization ability is better than the other four algorithms.

## 5. Optimization of Truss Structure by MWOA

### 5.1 Optimization Model of Truss Structure

When WOA is used for truss structure optimization, all members of the truss structure can be considered as each goods in the 0-1

**Table 3.** Test Functions

| Function name | Expression | Dimensions | Scope | Global minimum |
|---|---|---|---|---|
| Bukin | $f_1(x) = 100\sqrt{\lvert x_2 - 0.01x_1^2\rvert} + 0.01\lvert x_1 + 10\rvert$ | 2 | [−10,1] | 0 |
| Bohachevsky | $f_2(x) = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1) - 0.4\cos(4\pi x_2) + 0.7$ | 2 | [−100,100] | 0 |
| Drop-wave | $f_3(x) = -\dfrac{1+\cos(12\sqrt{x_1^2+x_2^2})}{0.5(x_1^2+x_2^2)+2}$ | 2 | [0,0] | -1 |
| Rastrigin | $f_4(x) = 10d + \sum_{i=1}^{d}[x_i^2 - 10\cos(2\pi x_i)]$ | $d$ | [−5.12,5.12] | 0 |
| Shubert | $f_5(x) = \left(\sum_{i=1}^{5} i\cos((i+1)x_1+i)\right)\left(\sum_{i=1}^{5} i\cos((i+1)x_2+i)\right)$ | 2 | [−10,10] | -186.7309 |
| Holder-table | $f_6(x) = -\left\lvert \sin(x_1)\cos(x_2)\exp\left(\left\lvert 1 - \dfrac{\sqrt{x_1^2+x_2^2}}{\pi}\right\rvert\right)\right\rvert$ | 2 | [−10,10] | -19.2085 |

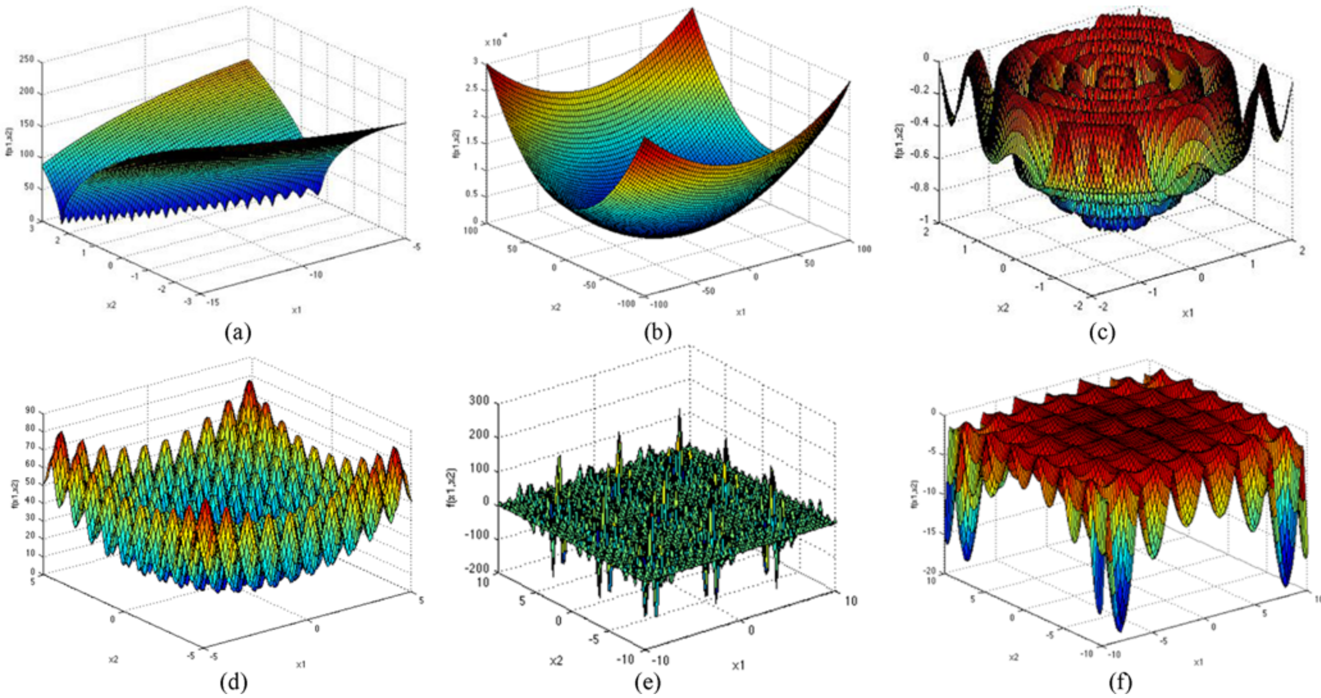

**Fig. 4.** Standard Function Graph: (a) Bukin Function Graph, (b) Bohachevsky Function Graph, (c) Drop-Wave Function Graph, (d) Rastrigin Function Graph, (e) Shubert Function Graph, (f) Holder-Table Function Graph

**Table 4.** Comparison of Optimization Performance for Benchmark Functions

| Function | Indicator | GA (Guo et al., 2017) | WOA (He et al., 2019) | MS-WOA (He et al., 2019) | WOAWC (Guo et al., 2017) | MWOA |
|---|---|---|---|---|---|---|
| $f_1$ | Mean | 10.18E-01 | 31.71E-01 | 18.02E-03 | 8.32E-04 | 5.36E-04 |
| | Standard deviation | 3.51E-02 | 2.35E-05 | 0.00E+00 | 1.06E-01 | 0.00E+00 |
| | Running time (s) | 5.3519 | 3.6213 | 2.9483 | 3.5826 | 2.3461 |
| $f_2$ | Mean | 3.93E-01 | 2.80E-01 | 0.13E+01 | 2.83E-02 | 1.59E-05 |
| | Standard deviation | 3.86E-01 | 4.22E-01 | 3.64E-01 | 2.59E-03 | 0.00E+00 |
| | Running time (s) | 4.5629 | 4.5061 | 3.7054 | 2.5785 | 2.0016 |
| $f_3$ | Mean | 31.46E-01 | 12.21E-01 | 8.88E-01 | 6.24E-01 | -9.59E-01 |
| | Standard deviation | 0.52E-01 | 2.62E-01 | 0.00E+00 | 5.49E-02 | 1.38E-03 |
| | Running time (s) | 4.8623 | 5.8611 | 3.2864 | 2.9187 | 2.5462 |
| $f_4$ | Mean | 6.18E-01 | 0.00E+00 | 0.00E+00 | 25.97E-02 | 0.00E+00 |
| | Standard deviation | 2.94E-01 | 0.00E+00 | 0.00E+00 | 3.47E-03 | 0.00E+00 |
| | Running time (s) | 3.5491 | 2.8462 | 2.1826 | 1.8364 | 2.3061 |
| $f_5$ | Mean | -1.65E+02 | -0.43E+02 | 1.15E-02 | -1.68E+02 | -1.71E+02 |
| | Standard deviation | 3.46E-01 | 1.47E-02 | 3.43E-03 | 4.58E-01 | 4.30E-03 |
| | Running time (s) | 6.513 | 5.426 | 3.592 | 3.018 | 2.105 |
| $f_6$ | Mean | -1.17E+01 | -1.54E+01 | -1.20E+01 | -1.63E+01 | -1.71E+01 |
| | Standard deviation | 3.59E-02 | 2.63E-02 | 1.13E-03 | 3.52E-01 | 1.82E-03 |
| | Running time (s) | 6.5926 | 3.9214 | 2.8436 | 2.1824 | 1.7621 |

Knapsack Problem. Therefore, the cross-sectional area of each member that constitutes the truss structure is the weight of each goods in the 0 – 1 Backpack Problem, and the optimization result is the objective function which is the backpack quality searched by WOA.

The optimization model problem of truss with sectional area as the design variable is described as follows:

$$minF = W(x), \tag{16}$$

$$s.t.\ \ g_i(x) \le 0,\ i = 1, 2, \cdots, m, \tag{17}$$

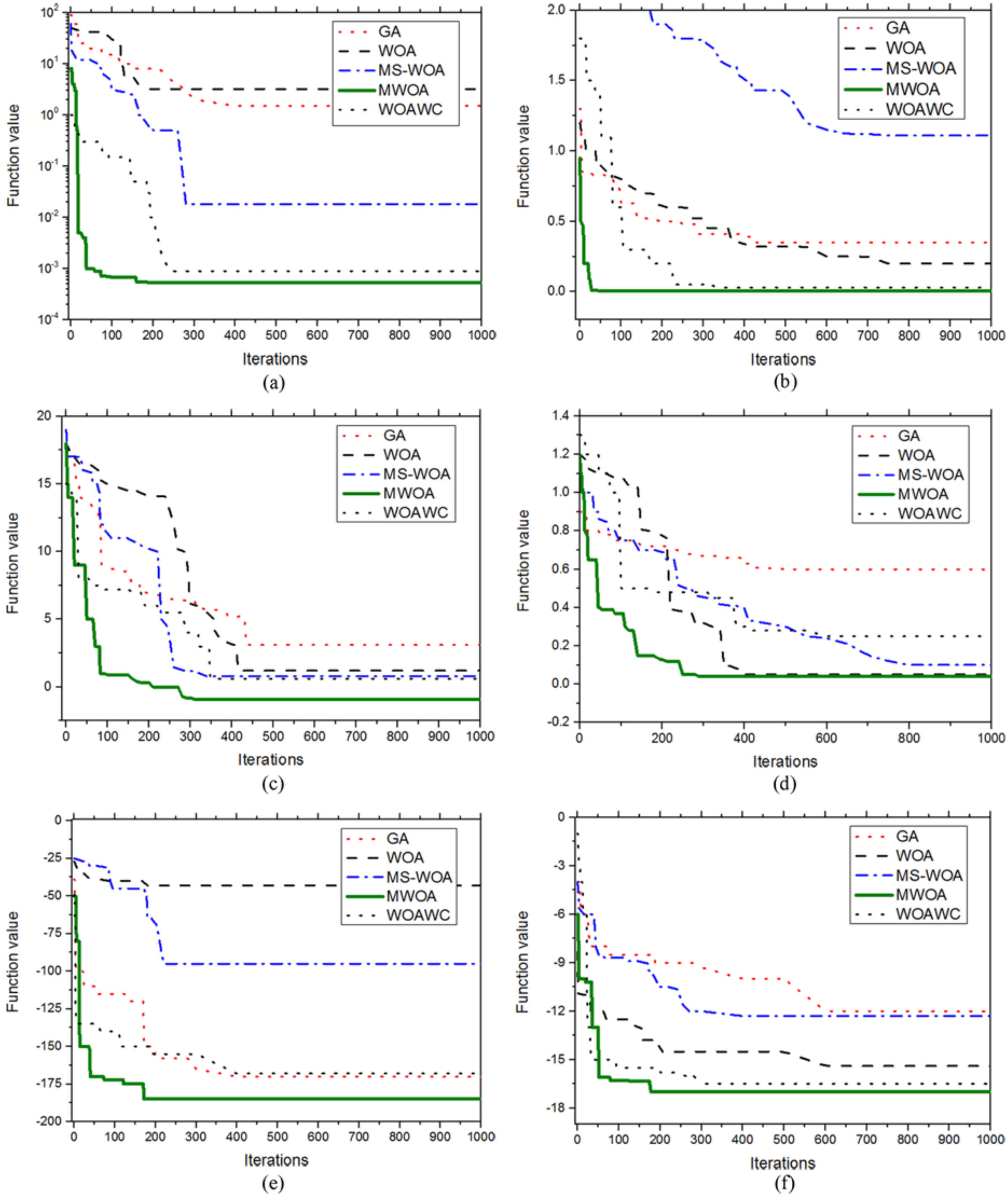


**Fig. 5.** The Iterative Curve of the Algorithm under Four Functions: (a) Bukin Function, (b) Bohachevsky Function, (c) Drop-Wave Function, (d) Rastrigin Function, (e) Shubert Function, (f) Holder-Table Function

where $g_i(x)$ is the constraint function; $m$ is the number of constraints.

## 5.2 Optimization of 15-Bar Truss Structure

The 15-bar truss structure model (Li and Yan, 2019) as shown in Fig. 6, Tables 5 and 6.

The optimization results of truss structure are shown in Table 7. Under the same constraint conditions, MWOA was used to optimize the 25-bar truss structure, and the total mass of the optimized structure was 95.67 kg. Compared to ACO-SA, quality decreased (129.71 – 95.67)/95.67 = 35.58%. Similarly, compared with WOA, PSO and SWOA, MWOA's quality decreased by 10.52%, 48.54% and 7.58%, respectively. In conclusion, it is feasible to optimize the truss structure with the MWOA to obtain the best quality and the least iteration times.

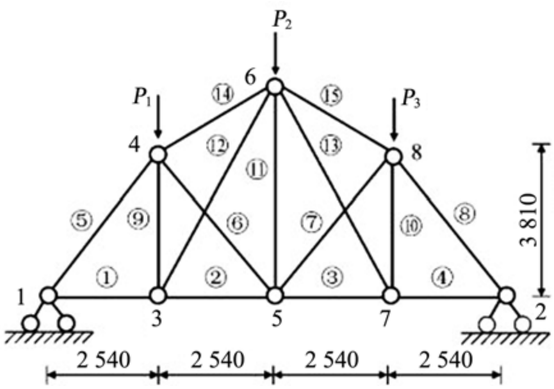


Fig. 6. The Spatial Structure of 15-Bar

Table 5. Parameters of 15-Bar Truss Structures

| Parameter | Value |
|---|---|
| Material density | $\rho = 7.8 \times 10^3$ kg/m$^3$ |
| Elastic modulus | $E = 200$ GPa |
| Allowable stress range | [-120 MPa, 120 MPa] |
| Node displacement range | [-10 mm, 10 mm] |
| Cross section | D = [113.2,143.2,145.9,174.9,185.9,235.9, 265.9,297.1,308.6,334.3,497.8,507.6,736.7, 791.2,1063.7] (unit: m$^2$) |

Table 6. Load Conditions of 15-Bar Truss Structure

| Load condition (unit: *KN*) | $P_1$ | $P_2$ | $P_3$ |
|---|---|---|---|
| 1 | 35 | 35 | 35 |
| 2 | 35 | 0 | 35 |
| 3 | 35 | 0 | 0 |

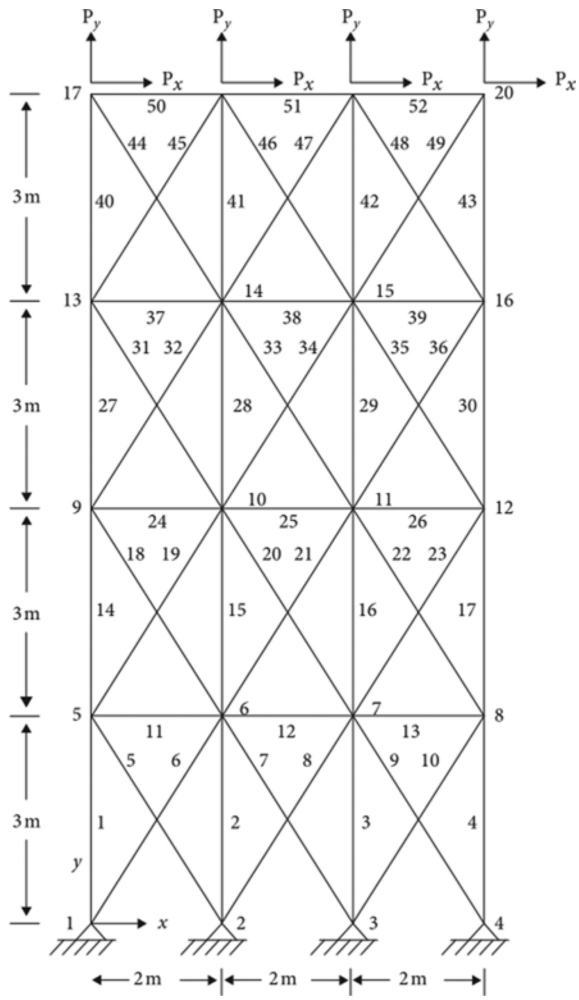


Fig. 7. The Spatial Structure of 52-Bar

Table 7. Comparison of Optimization Results of 15-Bar Structure

| Bar number | Bar section area (unit: m$^2$) | | | | |
|---|---|---|---|---|---|
| | ACO-SA (Zhou and Han, 2013) | WOA (Liu et al., 2020) | PSO (Liu et al., 2020) | SWOA (Liu et al., 2020) | MWOA |
| $A_1$ | 165.318 | 113.241 | 187.103 | 113.149 | 110.613 |
| $A_2$ | 151.337 | 113.241 | 113.145 | 113.145 | 113.145 |
| $A_3$ | 151.337 | 113.241 | 113.145 | 113.145 | 113.145 |
| $A_4$ | 165.318 | 113.241 | 187.103 | 113.149 | 110.613 |
| $A_5$ | 736.307 | 736.738 | 736.942 | 736.636 | 736.107 |
| $A_6$ | 143.516 | 113.241 | 143.136 | 113.149 | 110.128 |
| $A_7$ | 113.427 | 113.241 | 143.436 | 113.149 | 110.128 |
| $A_8$ | 736.913 | 736.728 | 736.419 | 735.713 | 736.107 |
| $A_9$ | 121.642 | 113.145 | 113.336 | 113.145 | 110.128 |
| $A_{10}$ | 121.642 | 113.145 | 113.413 | 113.145 | 110.128 |
| $A_{11}$ | 113.617 | 113.145 | 113.413 | 113.107 | 113.101 |
| $A_{12}$ | 113.208 | 113.145 | 113.153 | 113.107 | 103.113 |
| $A_{13}$ | 113.208 | 113.145 | 113.153 | 100.143 | 103.113 |
| $A_{14}$ | 334.331 | 334.331 | 334.331 | 334.331 | 334.331 |
| $A_{15}$ | 334.331 | 334.331 | 334.331 | 334.331 | 334.331 |
| Total weight (unit: kg) | 129.71 | 105.73 | 142.11 | 102.92 | 95.67 |

**Table 8.** Parameters of 52-Bar Truss Structures

| Parameter | Value |
|---|---|
| Material density | $\rho = 7.86 \times 10^3$ kg/m$^3$ |
| Elastic modulus | $E = 207$ GPa |
| Allowable stress range | [-180 MPa, 180 MPa] |
| Stress condition | $P_x = 100$ KN, $P_y = 200$ KN |

**Table 9.** Grouping of 52-Bar Structures

| Group number | Bar number | Group number | Bar number |
|---|---|---|---|
| $A_1$ | 1,2,3,4 | $A_7$ | 27,28,29,30 |
| $A_2$ | 5,6,7,8,9,10 | $A_8$ | 31,32,33,34,35,36 |
| $A_3$ | 11,12,13 | $A_9$ | 37,38,39 |
| $A_5$ | 14,15,16,17 | $A_{10}$ | 40,41,42,43 |
| $A_5$ | 18,19,20,21,22,23 | $A_{11}$ | 44,45,46,47,48,49 |
| $A_6$ | 24,25,26 | $A_{12}$ | 50,51,52 |

## 5.3 Optimization of 52-Bar Truss Structure

The 52-bar truss structure model as shown in Fig. 7 and Table 8 (Assimi et al., 2018). The grouping condition shows in Table 9.

The optimization results of truss structure are shown in Table 10. Under the same constraint conditions, MWOA was used to optimize the 52-bar truss structure, and the total mass of the optimized structure was 1897.91 kg. Compared to ACO-SA, MWOA's quality decreased (1936.12 – 1897.91)/1897.91 = 2.01%. Similarly, compared with WOA, PSO and SWOA, MWOA's quality decreased by 1.48%, 1.90% and 1.99%, respectively. In conclusion, it is feasible to optimize the truss structure with the MWOA to obtain the best quality and the least iteration times.

**Table 10.** Comparison of Optimization Results of 52-Bar Structure

| Bar group number | Bar section area (unit: m$^2$) | | | | |
|---|---|---|---|---|---|
| | ACO-SA (Zhou and Han, 2013) | WOA (Liu et al., 2020) | PSO (Liu et al., 2020) | SWOA (Liu et al., 2020) | MWOA |
| $A_1$ | 46.575 | 46.575 | 46.575 | 46.575 | 46.575 |
| $A_2$ | 11.632 | 11.632 | 11.632 | 11.632 | 11.632 |
| $A_3$ | 4.985 | 4.985 | 5.064 | 4.985 | 4.962 |
| $A_4$ | 33.028 | 33.028 | 33.028 | 33.028 | 33.028 |
| $A_5$ | 9.402 | 9.402 | 9.402 | 9.402 | 9.402 |
| $A_6$ | 4.945 | 4.945 | 4.945 | 4.945 | 4.945 |
| $A_7$ | 22.905 | 22.867 | 22.873 | 22.868 | 22.855 |
| $A_8$ | 10.078 | 10.078 | 10.078 | 10.078 | 10.078 |
| $A_9$ | 5.125 | 4.962 | 5.163 | 4.965 | 4.955 |
| $A_{10}$ | 14.921 | 14.599 | 15.356 | 14.591 | 13.602 |
| $A_{11}$ | 11.072 | 10.886 | 10.958 | 10.986 | 10.765 |
| $A_{12}$ | 5.127 | 5.065 | 5.062 | 5.067 | 5.138 |
| Weight (unit: kg) | 1936.12 | 1925.92 | 1933.89 | 1935.61 | 1897.91 |

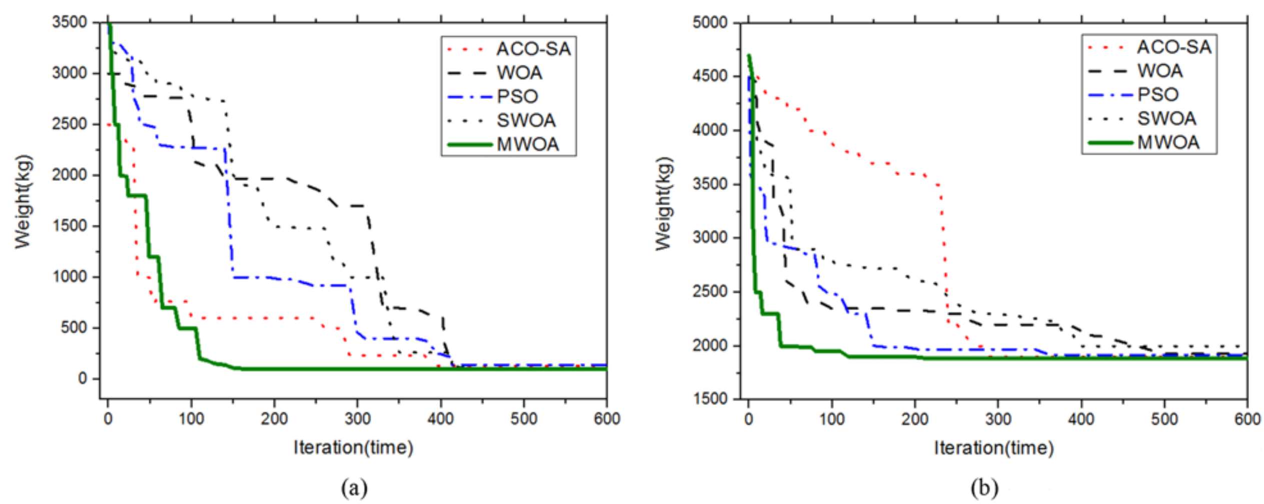


**Fig. 8.** Optimization Iteration Curve of Two Truss Structures: (a) Iteration Curve of the 15-Bar Truss Structure, (b) Iteration Curve of the 52-Bar Truss Structure

### 5.4 Algorithm Iterative Curve Comparison

According to the optimization results in Tables 7 and 10, it can be seen that the new algorithm MWOA performs better in solving speed and optimization degree, and can achieve the expected results. In order to make the comparison results clearer, Fig. 8 is the iterative curve of optimal design of truss structures.

## 6. Conclusions

This paper proposes a modified whale optimization algorithm based on Tent chaotic mapping and tournament selection mechanism to solve structural optimization problem. The Tent map expands the diversity of the population. Adaptive chaotic search is used to find the local optimal solution. The championship selection strategy can effectively identify the local and global optimal solutions, thus greatly improving the accuracy and performance of the algorithm. The functions test results show that the operation result of the MWOA is closer to the extreme value, with the lowest standard deviation and the shortest running time. When solving the 0 – 1 Knapsack Problem, the improved algorithm achieves satisfactory results in the test. Finally, the MWOA is applied to the optimal design of truss structure, and the objective function value is obtained with fewer iterations, and the minimized weight. In conclusion, the improved algorithm proposed in this paper is feasible and provides new ideas for the improvement and application of intelligent algorithm. In the future research work, the convergence of MWOA will be analyzed from the perspective of theoretical research and combined with other engineering applications.

## Acknowledgments

The work was supported by the Science and Technology Research Project of Institutions of Higher Education of Hebei Province (ZD2019114), and the Hebei Graduate Innovation Funding Project (CXZZSS2020079).

## ORCID

Yancang Li https://orcid.org/0000-0002-0229-7982
Muxuan Han https://orcid.org/0000-0002-3642-0782

## References

Agrawal RK, Kaur B, Sharma S (2020) Quantum based whale optimization algorithm for wrapper feature selection. *Applied Soft Computing* 89:1-14, DOI: 10.1016/j.asoc.2020.106092

Assimi H, Jamali A, Nariman-Zadeh N (2018) Multi-objective sizing and topology optimization of truss structures using genetic programming based on a new adaptive mutant operator. *Neural Computing and Applications* 23(24):1-21, DOI: 10.1007/s00521-018-3401-9

Batista CAS, Viana RL (2020) Chaotic maps with nonlocal coupling: Lyapunov exponents, synchronization of chaos, and characterization of chimeras. *Chaos, Solitons and Fractals* 58:3-18, DOI: 10.1016/j.chaos.2019.109501

Bozorgi SM, Yazdani S (2019) IWOA: An improved whale optimization algorithm for optimization problems. *Journal of Computational Design and Engineering* 6(3):243-259, DOI: 10.1016/j.jcde.2019.02.002

Chang XG, Zhao YJ (2018) Weak structure balance analysis of signed network based on WSB-EA evolutionary algorithm. *Journal of Intelligent Systems* 13(8):783-790, DOI: 10.11992/tis.201706054 (in Chinese)

Chu TH, Nguyen QU, Michael O (2018) Semantic tournament selection for genetic programming based on statistical analysis of error vectors. *Information Sciences* 436:352-366, DOI: 10.1016/j.ins.2018.01.030

Elaziz MA, Mirjalili S (2019) A hyper-heuristic for improving the initial population of whale optimization algorithm. *Knowledge-Based Systems* 172(15):42-63, DOI: 10.1016/j.knosys.2019.02.010

Fu SP, Qiao JF, Han HG (2013) Improved differential evolutionary algorithm based on mutation strategy of tournament selection for function optimization. *Computer Science* 40(6A):15-18 (in Chinese)

Gharehchopogh FS, Gholizadeh H (2019) A comprehensive survey: Whale optimization algorithm and its applications. *Swarm and Evolutionary Computation* 48:1-24, DOI: 10.1016/j.swevo.2019.03.004

Guo ZZ, Wang P, Ma YF, Wang Q, Gong CQ (2017) Whale optimization algorithm based on adaptive weight and cauchy mutation. *Microelectronics and Computers* 34(9):20-25, DOI: 10.19304/j.cnki.issn1000-7180.2017.09.005

Hao XH, Song JX, Zhou Q, Ma M (2020) Optimization algorithm of whale with hybrid strategy improvement. *Computer Application Research* 37(12):1-7, DOI: 10.19734/j.issn.1001-3695.2019.09.0528 (in Chinese)

He Q, Wei KY, Xu QS (2019) Improved whale optimization algorithm based on hybridstrategy. *Computer Application Research* 36(12):3647-3651, DOI: 10.19734/j.issn.1001-3695.2018.07.0382 (in Chinese)

Jain L, Katarya R, Sachdeva S (2020) Opinion leader detection using whale optimization algorithm in online social network. *Expert Systems with Applications* 142(15):1-22, DOI: 10.1016/j.eswa.2019.113016

Jiang RY, Ming Y, Wang SY, Tao C (2020) An improved whale optimization algorithm with armed force program and strategic adjustment. *Applied Mathematical Modelling* 81:603-623, DOI: 10.1016/j.apm.2020.01.002

Kalemci EN, Ikizler SB, Dede T, Angın Z (2020) Design of reinforced concrete cantilever retaining wall using grey wolf optimization algorithm. *Structures* 23:245-253, DOI: 10.1016/j.istruc.2019.09.013

Kaur G, Arora S (2018) Chaotic whale optimization algorithm. *Journal of Computational Design and Engineering* 5(3):275-284, DOI: 10.1016/j.jcde.2017.12.006

Kılıç H, Yüzgeç U (2019a) Tournament selection based antlion optimization algorithm for solving quadratic assignment problem. *Engineering Science and Technology* 22:673-691, DOI: 10.1016/j.jestch.2018.11.013

Kılıç H, Yüzgeç U (2019b) Improved antlion optimization algorithm via tournament selection and its application to parallel machine scheduling. *Computers & Industrial Engineering* 132:166-186, DOI: 10.1016/j.cie.2019.04.029

Kuang FJ, Xu WH, Jin Z (2014) Artificial colony algorithm for adaptive tent chaos search. *Control Theory and Application* 31(11):1502-1509, DOI: 10.7641/CTA.2014.31114 (in Chinese)

Li YC, Wang X (2019) Improved dolphin swarm algorithm based on information entropy and its truss optimization. *Journal of Chongqing University* 42(5):76-85, DOI: 10.11835/j.issn.1000-582X.2019.05.009 (in Chinese)

Li YC, Yan Z (2019a) Application of improved bat algorithm in truss optimization. *KSCE Journal of Civil Engineering* 23(6):2636-2643, DOI: 10.1007/s12205-019-2119-2

Liu L, He Q (2019b) An improved whale optimization algorithm for solving function optimization problem. *Computer Application Research* 37(4):1-8, DOI: 10.19734/j.issn.1001-3695.2018.11.0726 (in Chinese)

Liu LB, Zhao TT, Li YC, Xu MD, Wang ZY, Wang B (2020) Application of improved whale algorithm in truss structure optimization. *Fly Ash Comprehensive Utilization* 34(1):19-25 (in Chinese)

Manjit K, Dilbag S, Kehui S, Umashankar R (2020) Color image encryption using non-dominated sorting genetic algorithm with local chaotic search based 5D chaotic map. *Future Generation Computer Systems* 107:333-350, DOI: 10.1016/j.future.2020.02.029

Mansouri A, Wang XY (2020) A novel one-dimensional sine powered chaotic map and its application in a new image encryption scheme. *Information Sciences* 520:46-62, DOI: 10.1016/j.ins.2020.02.008

Memarzadeh R, Zadeh HG, Dehghani M, Hossien RM, Mortazavi SM (2020) A novel equation for longitudinal dispersion coefficient prediction based on the hybrid of SSMD and whale optimization algorithm. *Science of the Total Environment* 716(10):1-12, DOI: 10.1016/j.scitotenv.2020.137007

Mirjalili S, Lewis A (2016) The whale optimization algorithm. *Advances in Engineering Software* 95(5):51-67, DOI: 10.1016/j.advengsoft.2016.01.008

Naderi A, Sohrabi MR, Ghasemi, MR, Dizangian B (2020) Total and partial updating technique: A swift approach for cross-section and geometry optimization of truss structures. *KSCE Journal of Civil Engineering* 24:1219-1227, DOI: 10.1007/s12205-020-2096-5

Pelusi D, Mascella R, Tallini L, Nayak J, Naik B, Deng Y (2020) An improved moth-flame optimization algorithm with hybrid search phase. *Knowledge-Based Systems* 191:1-14, DOI: 10.1016/j.knosys.2019.105277

Rajeshkumar J, Kousalya K (2017) Diabetes data classification using whale optimization algorithm and backpropagation neural network. *International Research Journal of Pharmacy* 8(11):219-222, DOI: 10.7897/2230-8407.0811242

Samar MI, Lobna AS, Ahmed GR, Ahmed HM, Mohamed FA (2020) A novel image encryption system merging fractional-order edge detection and generalized chaotic maps. *Signal Processing* 167:1-21, DOI: 10.1016/j.asoc.2020.106162

Sayed GI, Darwish A, Hassanien AE (2018) A new chaotic whale optimization algorithm for features selection. *Journal of Classification* 35:1-45, DOI: 10.1007/s00357-018-9261-2

Shahna KU, Mohamed A (2020) A novel image encryption scheme using both pixel level and bit level permutation with chaotic map. *Applied Soft Computing Journal* 90:1-17, DOI: 10.1016/j.asoc.2020.106162

Wu TB, Zhu HQ, Long W, Li YG, Liu YL (2020) Improved whale optimization algorithm and its application in sintering ingredients. *Journal of Central South University (Natural Science Edition)* 51(1):103-111, DOI: 10.11817/j.issn.1672-7207.2020.01.013 (in Chinese)

Zheng WD, Li ZG, Jia HZ, Gao C (2019) Prediction module of steelmaking end point based on improved whale optimization algorithm and least squares support vector machine. *Acta Electrica Sinica* 47(3):700-706, DOI: 10.3969/j.issn.0372-2112.2019.03.026 (in Chinese)

Zhou SJ, Han X (2013) Improved ant colony algorithm and its application in truss structure optimization. *Steel Structure* 28(3):1-5, DOI: 10.3969/j.issn.1007-9963.2013.03.001 (in Chinese)