Hindawi
International Journal of Aerospace Engineering
Volume 2022, Article ID 4373879, 16 pages
https://doi.org/10.1155/2022/4373879

Research Article

Position Deployment Optimization of Maneuvering Conventional Missile Based on Improved Whale Optimization Algorithm

Yaxiong Li, Xinwei Zhang, Jiufen Zhao, Xinzhi Yang, and Mingyuan Xi

Xi'an Research Institute of High Technology, Xi'an, Shaanxi 710025, China

Correspondence should be addressed to Xinwei Zhang; kins265@163.com

Received 11 April 2022; Revised 12 July 2022; Accepted 2 August 2022; Published 25 August 2022

Academic Editor: Jinchao Chen

To reasonably allocate mobile conventional missile battle positions and improve the survivability and combat effectiveness of the missile weapon system, a missile position deployment optimization design model based on Mixed-Integer Nonlinear Programming (MINLP) is established. First, for the multilevel position network, an optimization model with the goal of maximizing the viability is established, and a two-stage solution method is proposed. Second, the whale optimization algorithm (WOA) is improved, and the convergence factor which changes nonlinearly with the number of evolution is introduced to coordinate the local development and global search ability of the algorithm. The diversity mutation operation is carried out on the optimal individual to reduce the probability of premature convergence of the algorithm, and the improved whale algorithm is used to solve the optimization model. Finally, the performance of the improved algorithm is verified by example analysis and simulation experiments, which provides a reference for the deployment mode of mobile missile positions.

1. Introduction

Missile position is an important basis for the missile weapons to carry out combat missions. The reasonable deployment and scientific configuration of positions are not only conducive to the full play of the operational effectiveness of missile weapons but also can effectively improve the command ability and survivability of the weapon system, which is of the significance to the firepower utilization of missile weapons. Therefore, it is necessary to further optimize the position deployment, explore the rational allocation scheme, and promote the combat effectiveness of missile forces.

To solve the problem of position configuration optimization of mobile missile, Wang and Gao [1] first proposed a polygon position configuration mode and solved the model with an enumeration method and linear programming method. At the same time, Wang and Wei [2] aimed at improving the launch stability of missiles and discussing the influence of mutual interference between guidance systems on the position configuration mode when multiple missiles were fired at the same time. On this basis, Hao [3] applied the polygonal position configuration mode to land-based conventional missiles and achieved considerable results. However, the model constructed in the above research is relatively simple, and the position configuration is also weakened in the model. In response to this problem, Xie et al. [4] conducted an in-depth study on the position configuration of mobile missiles, proposed a fan-shaped position configuration model, and constructed a model solving algorithm based on the enumeration method and Monte Carlo method, which provided a reference for the decision-making of conventional missile position configuration. Zeng et al. [5] applied graph theory and neural network theory to the deployment and selection of air defense missile positions and put forward the indicators affecting position selection. On the basis of Zeng, Wang et al. [6] established the configuration model of air defense missile position from the perspective of improving operational efficiency and solved it by genetic algorithm, but the established model is relatively simple. Liu et al. [7] discussed configuration of the naval air defense position, which provided ideas for the establishment of the naval air defense position model. Subsequently, Gao et al. [8] took into account the number of missile shots into the problem of position deployment and established the deployment model of naval air defense weapon position based on the number of shots. However, due to the complexity of the position deployment problem, the optimization model has quite a few constraints, and it is difficult to solve. In the following period of time, the research on the position deployment of maneuvering conventional missiles has entered a stagnation period.

In recent years, with the continuous development of the bionic swarm intelligent optimization algorithm, which simulates the behavior of animals in nature, it provides a new and powerful tool for solving complex optimization problems [9]. Some traditional swarm intelligence optimization algorithms include ant colony optimization (ACO) [10], fruit fly optimization algorithm (FOA) [11], and grey wolf optimization (GWO) [12]. With the continuous development of intelligent bionic algorithm, more and more new algorithms are proposed, mainly including monarch butterfly optimization (MBO) [13], moth swarm algorithm (MSA) [14], and slime mold algorithm (SMA) [15]. In 2016, Mirjalili and Lewis [16] proposed a new intelligent optimization algorithm for bionic humpback whale behavior: whale optimization algorithm (WOA). Compared with other intelligent algorithms, this algorithm has simple principle, less parameter settings, and strong optimization performance. At present, it has been successfully applied to complex optimization problems [17]. However, at the same time, the algorithm also has some shortcomings, such as local development and global search ability is difficult to coordinate and prone to premature convergence in the late iteration [18]. Based on the shortcomings of WOA, many scholars have also improved it in recent years. For example, Ning and Cao [19] improved WOA from three aspects: initial population, convergence factor, and mutation operation, and verified the performance of the improved algorithm. Li et al. [20] proposed a mathematical model considering linear and nonlinear failure criteria for locating the critical sliding surface of soil slope and improved the WOA. The experimental results show that the improved algorithm has better performance. Donyaii et al. [21] improved WOA for water resource management and reservoir operation and analyzed water resource management problems using WOA. Bozorgi and Yazdani [22] improved WOA by combining WOA with differential evolution (DE) to solve the premature convergence problem of WOA and verified the performance of the improved algorithm through experiments. However, in the context of military applications, there is no relevant research on using this algorithm to solve the optimization problem of missile position deployment.

Therefore, this paper attempts to apply WOA to the position deployment optimization problem of conventional mobile missiles, improve the shortcomings of the algorithm, and overcome the disadvantages that the position deployment optimization model is difficult to solve. Aiming at this goal, this paper optimizes the position deployment with the goal of maximizing the survivability of the weapon system and proposes a two-stage solution method combined with the actual problem. The WOA is improved: the convergence factor is introduced to coordinate the global search and local development ability of the algorithm, and diversity mutation is performed on the optimal individual to reduce the probability of premature convergence of the algorithm. It provides a new solution method and configuration mode reference for the position deployment optimization problem and verifies the superiority and effectiveness of the improved WOA through examples.

The main contributions and innovations of this paper are as follows:

(1) WOA is applied to the position deployment optimization of mobile conventional missiles. It overcomes the disadvantages of many constraints and difficult to solve the position deployment optimization problem and provides a new method for solving the problem

(2) A three-level sector position deployment model and a position optimization model based on MINLP [23] are established. The deployment mode is more complex, and the constraint conditions of the model are more complete, which is more in line with the actual combat of conventional missile weapons

(3) A two-stage method for solving the model is proposed. By this method, the MINLP problem is transformed into a nonlinear programming problem, which effectively reduces the difficulty of solving the original problem

(4) An improved WOA is proposed. The convergence factor of nonlinear variation is introduced to balance the global search and local development ability of the algorithm. The diversity mutation of the optimal whale individuals reduces the probability of premature convergence of the algorithm

(5) The optimization method in this paper is applied to an example, which can provide a reference for the configuration and deployment of missile positions and is beneficial to the performance of missile weapons

The organizational structure of this paper is as follows: Section 2 describes the operational scenario of maneuvering conventional missiles, puts forward the position deployment mode, and gives the background of the position optimization problem. Section 3 establishes the optimization model of the position deployment problem. Section 4 proposes a two-stage solution method for the optimization model. Section 5 introduces the WOA, improves the algorithm, and proposes an Improved WOA. Section 6 verifies the model and method of this paper through examples and analyzes the advantages of the improved WOA by comparing it with other algorithms. Finally, the conclusions of this paper and the direction for future efforts are expounded.

2. Operational Scenario of Conventional Mobile Missile

When maneuvering conventional missile weapons perform operational tasks, their ability to move quickly and damage targets mainly depends on the missile weapon system, operators, and missile position system. Therefore, this paper defines the conventional mobile missile weapons system as a weapon-personnel-position system.

2.1. Position Deployment Pattern Scenario. Deployment and configuration of positions refer to the layout optimization of positions at all levels to meet the operational requirements of mobile conventional missile weapons, namely, the configuration and layout of the main position, forward positions, and launching positions in the combat zone, and other supporting facilities. Without considering the topography of the positions at all levels and the road traffic conditions among the positions, the position deployment mode of the mobile conventional missile system is proposed as follows: with the main position as the radiation center, forward positions and launching positions in the combat zone are fan-shaped radiation distribution, as shown in Figure 1.

In the figure, x is the deployment number of forward positions in the operation area, and y is the deployment number of launch positions in the operation area. θ is the angle of sector deployment of forward positions and launch positions, and its size can be determined by the number of forward positions and launch positions. To facilitate the rapid movement of missile weapons, there is a road between the main position, forward positions, and launch positions. Assuming that the road lengths between the main position and forward positions are equal, both L_1. Road lengths between forward positions and launch positions are, also equal, both L_2. At the same time, Figure 1 shows that the number of mobile roads between positions depends on the number of positions, that is, the number of roads between the main position and forward positions is x, and the number of roads between forward positions and launch positions is y.

2.2. Missile Weapon Maneuver Mode Scenario. It is assumed that the missile weapon system adopts the scheme of formation maneuver and batch maneuver. Missile weapon systems are maneuvered from the main missile position to the forward position in the area of operations to wait for mission assignment or directly to launch positions. In addition, the number of formations of mobile missile weapons mainly depends on the combat task level. The number of formations of missile weapons is also different with different task levels.

2.3. Enemy Attack Mode Scenario. Combat tasks are inevitably accompanied by confrontation between the two sides. While one side carries out combat operations, the other side must take corresponding measures. In the future war, it is assumed that the enemy's attack mode is as follows: first, the first round of accurate strikes is carried out on missile positions at all levels, trying to destroy the missile positions. After the first round of attack, the remaining missile weapon maneuver from the main position and forward positions to launch positions. During the maneuver, the enemy used reconnaissance satellites and high precision weapons to strike the maneuvering missile weapons twice, making the opponent completely incapacitated.

3. Establishment of Position Deployment Optimization Model

Based on the position deployment pattern shown in Figure 1, a planning model is established to optimize the position deployment and configuration.

3.1. Determination of Decision Variables. According to the scenario of position deployment mode, four decision variables are determined, which are

x: number of forward positions in the operational area.

y: number of launch positions in the operational area.

L_1: road distance from main position to forward position.

L_2: road distance from forward position to launch position.

3.2. Construction of Objective Function. Survivability is the premise that missile weapons can perform combat tasks and cause damage to the target [24]. The deployment optimization of missile positions is aimed at improving the viability of missile weapon systems. Therefore, this paper takes the survival probability of maneuvering conventional missile system as the objective function to model.

There are two main aspects to measure the survivability of the missile weapon systems [25]. First, the missile weapon system still has the ability to fight after being attacked by the enemy. The second is the protective ability of the weapon system to prevent from being attacked by the enemy. According to the meaning of survivability and the scheme of Section 2, several random events that may occur in the process of the missile weapons performing combat tasks are as follows:

Random event A: the enemy carried out the first round of attacks to strike the main missile position accurately. Assuming that the probability of the main position being discovered by the enemy is P_{Mf}, and the probability of being destroyed after discovery is P_{Md}, the survival probability of the missile weapon system under this random event is

$$P(A) = 1 - P_{Mf} \cdot P_{Md}. \tag{1}$$

Random event B: the enemy carried out the first round of attacks to accurately strike the forward positions in the operational area. Assuming that the probability of any forward position being found by the enemy in the operational area is P_{Ff}, and the probability of being destroyed after discovery is P_{Fd}, the survival probability of the missile weapon system under this random event is

$$P(B) = 1 - {P_{Ff}}^{x} \cdot {P_{Fd}}^{x}. \tag{2}$$

Random event C: the enemy carried out the first round of attacks to accurately strike the launch positions in our operational area. Assuming that the probability of any launch position being found by the enemy in the combat zone is P_{Lf}, and the probability of being destroyed after discovery is P_{Ld}, the survival probability of the missile weapon

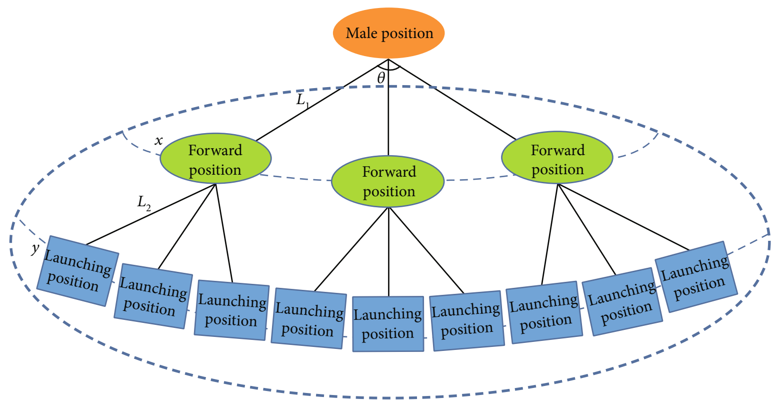

Figure 1: Position deployment pattern scenario.

system under this random event is

$$P(C) = 1 - P_{Lf}{}^{y} \cdot P_{Ld}{}^{y}. \tag{3}$$

Random event D: after the first round of attack, missile weapon moved from the main position to the forward position, and the enemy attacked our mobile missile weapons. Assuming that the maneuvering speed of the missile weapon is a constant v, and the maneuvering time is t, the probability of being found in the maneuvering process is a proportional function of time t, and the proportional coefficient is λ. If the probability of the missile weapon being found by the enemy during the maneuver is P_{Zf1}, then:

$$P_{Zf1} = \lambda \cdot t = \lambda \cdot \frac{L_1}{v}. \tag{4}$$

Assuming that the number of mobile missile weapons at this time is m and the probability of destruction after being found by the enemy during the maneuver is P_{Zd}, the survival probability of the missile weapon system under this random event is

$$P(D) = \begin{cases} 1 - \left(\lambda \dfrac{L_1}{v}\right)^m \cdot P_{Zd}{}^m & , m > 0, \\ 1 & , m = 0. \end{cases} \tag{5}$$

Random event E: after the first round of attack, our missile weapons moved from the forward position to the launch position, and the enemy attacked our mobile missile weapons. If the probability of the missile weapon being found by the enemy during the maneuver is P_{Zf2}, then:

$$P_{Zf2} = \lambda \cdot t = \lambda \cdot \frac{L_2}{v}. \tag{6}$$

Assuming that the number of mobile missile weapons at this time is n, the survival probability of the missile weapon system under this random event is

$$P(E) = \begin{cases} 1 - \left(\lambda \dfrac{L_2}{v}\right)^n \cdot P_{Zd}{}^n & , n > 0, \\ 1 & , n = 0. \end{cases} \tag{7}$$

The above five random events are independent of each other. If the survival probability of the maneuvering conventional missile weapon system is P_S, then:

$$P_S = P(A) \cdot P(B) \cdot P(C) \cdot P(D) \cdot P(E). \tag{8}$$

Corresponding to different mission levels, the values of m and n of the number of mobile missile weapons are different. Assuming that the probability coefficient [26] of maneuvering missile weapon system to perform tasks at all levels is $\alpha_1, \alpha_2, \cdots$, respectively, and $\sum \alpha_i = 1$, then, the objective function is

$$\max P_S = \max \prod P_{Si}{}^{\alpha_i}. \tag{9}$$

3.3. Selection of Constraint Conditions. Constraint 1: the maneuvering time t of the missile weapon cannot be greater than a certain value T. If the maneuver time is greater than T, the probability that the missile weapon is destroyed by the enemy during the maneuver is 1. Then, constraint 1 is

$$\frac{L_1}{v} \le T, \frac{L_2}{v} \le T. \tag{10}$$

Thus, the proportional coefficient is

$$\lambda = \frac{1}{T}. \tag{11}$$

Constraint 2: the construction cost of the position system cannot exceed the budget C. Assuming the construction cost of the main position is C_M, the construction cost of the forward position is C_F, the construction cost of the launch position is C_L, and the road construction cost per unit length is C_R. Then, constraint 2 is

$$C_M + xC_F + yC_L + (xL_1 + yL_2)C_R \le C. \tag{12}$$

Constraint 3: to meet the communication requirements between positions, the mobile road distance between the main position and the forward position and between the forward position and the launch position cannot exceed a certain value L_C. Then, constraint 2 is

$$L_1 \le L_C, L_2 \le L_C. \tag{13}$$

Constraint 4: to ensure that one enemy missile cannot destroy two forward positions at the same time, the distance between each forward position cannot be less than a certain value L_D. The distance between adjacent forward positions can be calculated from Figure 1:

$$l = 2L_1 \cdot \sin \frac{\theta}{2(x-1)}. \tag{14}$$

Then, constraint 4 is

$$2L_1 \cdot \sin \frac{\theta}{2(x-1)} \ge L_D. \tag{15}$$

Constraint 5: similarly, the distance between launch positions cannot be less than a certain value L_D. Assuming that the distance between each launching position and the main position is $(L_1 + L_2)$, then, constraint 5 is

$$2(L_1 + L_2) \cdot \sin \frac{\theta}{2(y-1)} \ge L_D. \tag{16}$$

Constraint 6: to meet the requirements of the combat area and damage capacity, the distance between the main position and the launch position should not be less than a certain value L_A, the number of forward positions and launch positions should not be less than M and N, and the number is integer. Then constraint 6 is

$$\begin{cases} L_1 + L_2 > L_A, \\ x \ge M, y \ge N, x, y \in Z. \end{cases} \tag{17}$$

4. Model Solving Algorithm Construction

In summary, the position deployment optimization model has been established, and the constraint conditions are sorted out to obtain the model expression as follows:

$$\max P_S = \max \prod {P_{Si}}^{\alpha_i} \begin{cases} L_1 \le \min(vT, L_C), L_2 \le \min(vT, L_C), \\ C_M + xC_F + yC_L + (xL_1 + yL_2)C_R \le C, \\ 2L_1 \cdot \sin \dfrac{\theta}{2(x-1)} \ge L_D, \\ 2(L_1 + L_2) \cdot \sin \dfrac{\theta}{2(y-1)} \ge L_D, \\ L_1 + L_2 \ge L_A, \\ x \ge M, y \ge N, x, y \in Z. \end{cases} \tag{18}$$

Clearly, the above model is a Mixed-Integer Nonlinear Programming (MINLP) model. The objective function of the model is complex, the constraint condition is nonlinear and contains integer constraints, so it is difficult to solve [27]. To facilitate the solution of the model, according to the practical problem of position deployment optimization, a two-stage solution method is proposed in this paper. The basic idea is since the decision variables x and y are integers, the value range of x and y can be determined through the constraint conditions of the model, and all possible value sequences can be obtained. Then, the obtained value sequence is substituted into the optimization model, and the MINLP model is transformed into the nonlinear programming model of variables L_1 and L_2. Finally, the improved WOA is used to solve the obtained nonlinear programming model. This method avoids integer constraints and reduces the original four decision variables to two, which effectively reduces the difficulty of solving the original model.

4.1. Model Stage One Solution. According to the constraint conditions of the model, the value sequence of the forward position and launch position is solved first.

By transforming constraints condition 2, condition 4, and condition 5, we can get

$$xL_1 + yL_2 \le \frac{C - C_M + xC_F + yC_L}{C_R}, \tag{19}$$

$$L_1 \ge \frac{L_D}{2} \csc \frac{\theta}{2(x-1)}, \tag{20}$$

$$L_1 + L_2 \ge \frac{L_D}{2} \csc \frac{\theta}{2(y-1)}. \tag{21}$$

Set the deployment angle θ as the maximum, that is:

$$\theta = 2\pi. \tag{22}$$

The plane rectangular coordinate system is established with L_1 as abscissa axis, L_2 as ordinate axis, and O point as origin, as shown in Figure 2. Line l_1 represents constraint condition 2, l_2 represents constraint condition 4, and l_3 represents constraint condition 5.

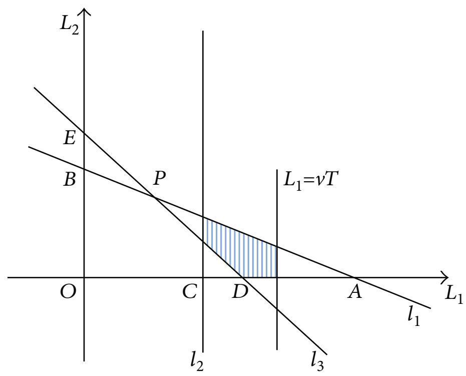

Figure 2: Coordinate diagram.

In the coordinate system, the shadow part is the feasible region of the model. Point A and point B are the intersection points of l_1 and coordinate axis, the coordinates are $(C - C_M + xC_F + yC_L/xC_R, 0)$ and $(0, C - C_M + xC_F + yC_L/yC_R)$, respectively. Point C is the intersection of l_2 and the transverse axis, and the coordinate is $(L_D/2 \csc \pi/x - 1, 0)$. Point D and point E are the intersection of l_3 and coordinate axis, and the coordinates are $(L_D/2 \csc \pi/y - 1, 0)$ and $(0, L_D/2 \csc \pi/y - 1)$, respectively. Point P is the intersection of l_1 and l_3, and the abscissa is $y/y - x(L_D/2 \csc \pi/y - 1 - C - C_M - xC_F - yC_L/yC_R)$.

Therefore, to exist a feasible domain, the following conditions are required.

Condition 1: when point D is to the left of point A, and point E is above point B, that is

$$\begin{cases} \dfrac{C - C_M - xC_F - yC_L}{yC_R} < \dfrac{L_D}{2} \csc \dfrac{\pi}{(y-1)}, \\ \dfrac{C - C_M - xC_F - yC_L}{xC_R} > \dfrac{L_D}{2} \csc \dfrac{\pi}{(y-1)}. \end{cases} \tag{23}$$

At this time, if the feasible region exists, it needs to satisfy

$$\frac{y}{y-x}\left(\frac{L_D}{2} \csc \frac{\pi}{y-1} - \frac{C - C_M - xC_F - yC_L}{yC_R}\right) < vT. \tag{24}$$

Condition 2: when point D is on the right side of point A , and point E is above point B, the feasible region does not exist. Therefore, needs to be met:

$$\frac{L_D}{2} \csc \frac{\pi}{y-1} < \frac{C - C_M - xC_F - yC_L}{xC_R}. \tag{25}$$

Condition 3: when point D is on the left side of point A, point E is below point B, namely,

$$\begin{cases} \dfrac{L_D}{2} \csc \dfrac{\pi}{(y-1)} < \dfrac{C - C_M - xC_F - yC_L}{yC_R}, \\ \dfrac{L_D}{2} \csc \dfrac{\pi}{(y-1)} < \dfrac{C - C_M - xC_F - yC_L}{xC_R}. \end{cases} \tag{26}$$

At this time, if the feasible region exists, it needs to satisfy

$$\frac{L_D}{2} \csc \frac{\pi}{y-1} < vT. \tag{27}$$

Condition 4: to make the feasible region exist, line l_2 needs to be on the left side of point A and line $L_1 = vT$, which satisfies

$$\begin{cases} \dfrac{L_D}{2} \csc \dfrac{\pi}{x-1} < vT, \\ \dfrac{L_D}{2} \csc \dfrac{\pi}{x-1} < \dfrac{C - C_M - xC_F - yC_L}{xC_R}. \end{cases} \tag{28}$$

By combining Equations (24), (25), (27), and (28), all value sequences of the forward position and launch position can be obtained.

4.2. Model Stage Two Solution. By substituting all value sequences of x and y into the position deployment optimization model, the original model can be transformed into the nonlinear programming model of variables L_1 and L_2, and then, the improved WOA is used to solve the optimal value.

5. Improved Whale Optimization Algorithm

5.1. Original Whale Optimization Algorithm. Whale predation method is relatively special, that is, bubble net predation [28], as shown in Figure 3. A detailed description of bubble-net predatory behavior can be seen in reference [28]. Based on the characteristics of bubble-net predatory behavior, Mirjalili and Lewis abstracted a new bionic intelligent optimization algorithm, namely, whale optimization algorithm. The mathematical model was used to simulate the process of whale encirclement, bubble-net attack, and random search for prey.

5.1.1. Surrounding Prey. Assuming that the number of whales is N, each whale is an individual, the search space is d-dimensional, the position of each individual in d -dimensional space is a solution, and the position of prey is the optimal solution of the problem. Whales identify and surround prey positions by echo. The update formula of whale position is as follows:

$$X(k+1) = X_p(k) - A \cdot \left|C \cdot X_p(k) - X(k)\right|, \tag{29}$$

where k is the current iteration number, and $X(k)$ is the current whale position vector.

$X_p = (X_p^1, X_p^2, \cdots X_p^D)$ is the position of prey, namely, the current optimal position vector, and D is the vector dimension. $A \cdot |C \cdot X_p(k) - X(k)|$ is the surrounding step, and A and C are defined as follows:

$$\begin{cases} A = 2\delta \cdot \text{rand}_1 - \delta, \\ C = 2 \cdot \text{rand}_2, \end{cases} \tag{30}$$

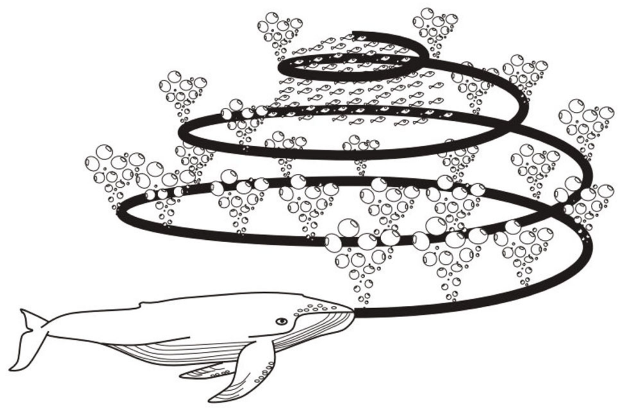

Figure 3: Bubble-net predation behavior.

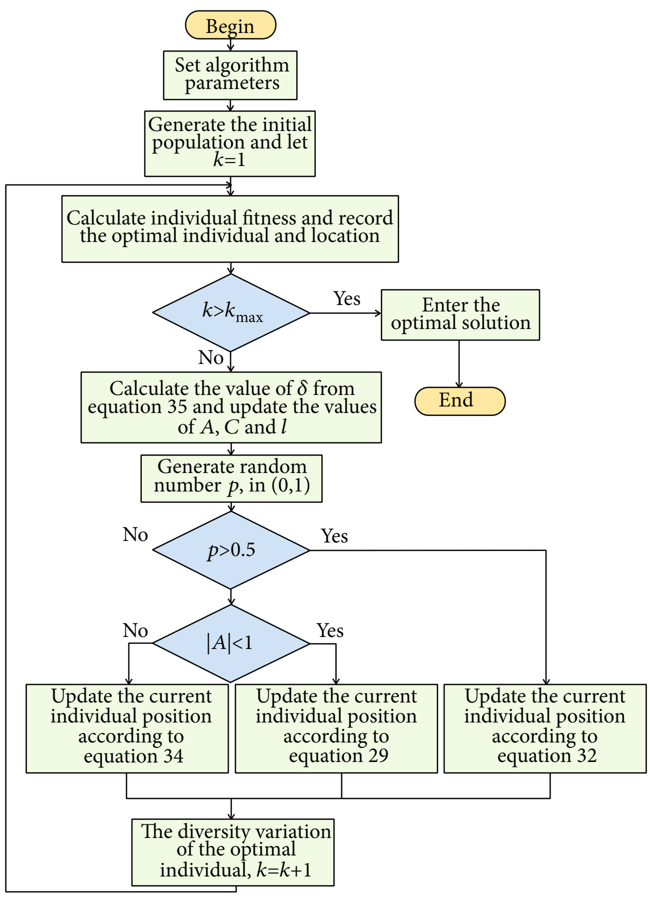

Figure 4: Improved WOA flow chart.

where rand_1 and rand_2 represent random numbers in the range of [0,1], and δ is the convergence factor, which decreases linearly from 2 to 0 with the increase of iteration number k, that is:

$$\delta = 2 - \frac{2k}{k_{\max}}. \tag{31}$$

$k_{\max}$ is the maximum number of iterations.

5.1.2. Bubble-Net Attacks. WOA designs two methods, namely, spiral update position and contraction surrounding mechanism. In the spiral update position method, the whale spiral motion approaches the prey, and the mathematical model is

$$X(k+1) = X_p(k) + D^k \cdot e^{bl} \cdot \cos(2\pi l). \tag{32}$$

In the formula, $D^k = |X_p(k) - X(k)|$ denotes the distance between the whale and prey, b denotes the constant of the limiting logarithmic spiral shape, and l is the random number in the range of [-1,1].

The contraction bounding mechanism is realized with the decrease of convergence factor. When the convergence factor decreases linearly from 2 to 0, the fluctuation range of A is $[-\delta, \delta]$. When A is a random value in the range of [-1,1], $X(k+1)$ can be any position between $X(k)$ and $X_p(k)$.

In the process of optimization, it is considered that the probability of choosing spiral position update and contraction bounding mechanism is equal to 0.5 [16]. Therefore, the mathematical model is

$$X(k+1) = \begin{cases} X_p(k) - A \cdot |C \cdot X_p(k) - X(k)|, p < 0.5, \\ X_p(k) + D^k \cdot e^{bl} \cdot \cos(2\pi l), p \geq 0.5. \end{cases} \tag{33}$$

5.1.3. Random Search for Prey. When $|A| > 1$, the whale swims outside the prey contraction circle. At this time, the whale individuals search for prey randomly according to their positions. The mathematical model is

$$X(k+1) = X_{\text{rand}}(k) - A \cdot |C \cdot X_{\text{rand}}(k) - X(k)|, \tag{34}$$

where X_{rand} is the randomly selected whale individual position vector.

5.2. Nonlinear Variation Strategy of Convergence Factor. As a swarm intelligence optimization algorithm, the coordination of global search and local development ability of WOA is essential [29]. The analysis of Section 5.1 shows that the main parameters of WOA are A and C. When $|A| > 1$, the search area of whale population is more extensive, which is reflected in the global search ability of the algorithm. When $|A| \leq 1$, the whale population performs a local fine search, which is reflected in the local development ability of the algorithm. The value of A depends largely on the convergence factor δ. When δ is large, the algorithm has better global search ability and avoids falling into the local optimum. When δ is small, the algorithm has strong local development ability and accelerates the convergence speed. It can be seen from Section 5.1.1 that in the original WOA, the convergence factor δ decreases linearly from 2 to 0 with the increase of iteration number k. However, under the linear decreasing strategy, the global search ability of the algorithm is strong in the early stage, but the convergence speed is slow. The convergence speed is accelerated in the later

Table 1: Model parameter values.

Parameter	α_1	α_2	α_3	C	C_M	C_F	C_L	C_R	P_{Mf}	P_{Md}	P_{Ff}
Value	0.5	0.3	0.2	3000	500	180	60	1	0.65	0.55	0.60
Parameter	P_{Fd}	P_{Lf}	P_{Ld}	P_{Zd}	L_C	L_D	L_A	v	T	M	N
Value	0.50	0.75	0.60	0.55	45	2	20	28	2	5	20

Table 2: Values of m and n under different task levels.

Task level	m	n
Level 1	5	0
Level 2	5	5
Level 3	0	10

Table 3: Value sequence of x and y.

x	5	5	5	5	5
y	20	21	22	23	24
x	5	5	6	6	6
y	25	26	20	21	22

Table 4: Model simulation results.

x	y	L_1^*	L_2^*	P_S^*
5	20	9.31076	10.6892	0.6409329
5	21	8.88211	11.1179	0.6409331
5	22	9.55878	10.4412	0.6409328
5	23	13.6141	6.38594	0.6409166
5	24	16.9736	3.02639	0.6408727
5	25	17.0312	2.96875	0.6408716
5	26	17.2723	2.72770	0.6408667
6	**20**	**13.1647**	**6.83533**	**0.6420127**
6	21	17.4328	2.56721	0.6419560
6	22	17.4937	2.50627	0.6419547

stage, but it is easy to fall into the local optimal solution. Therefore, the strategy of linear decreasing convergence factor cannot fully reflect the actual optimization search process [30].

In fact, we expect the algorithm to have strong global search ability and fast convergence speed in the early stage. In the later stage, the convergence speed is fast while avoiding falling into local optimization. Based on the above considerations, the convergence factor should first increase slightly with the increase in the number of iterations and then decrease rapidly when it increases to a certain extent, and finally increase slowly, showing a nonlinear change with the increase in the number of iterations. Therefore, the convergence factor update formula is as follows:

$$\delta = \frac{1 - k/k_{\max}}{1 - \mu \cdot k/k_{\max}} + (\delta_{\text{initial}} - \delta_{\text{final}}), \tag{35}$$

where μ is the nonlinear regulation coefficient, $\mu > 0$. δ_{initial} is the initial value of the convergence factor, and δ_{final} is the final value of the convergence factor.

5.3. Optimal Individual Diversity Variation Strategy. In the late iteration of WOA, all whale individuals in the population are close to the current optimal individuals, resulting in insufficient diversity of the population [31]. If the current optimal individual is not the global optimal solution, but the local optimal solution, the algorithm will converge prematurely and fall into the local optimal solution, that is, the algorithm has premature convergence.

To reduce the probability of this phenomenon in the algorithm, the diversity mutation of the current optimal individual is carried out. Assuming that an element $x_j (j = 1, 2, \cdots, d)$ is randomly selected from the individual $X_i = (x_{i1}, x_{i2}, \cdots x_{id})$ with the probability of $1/d$, and then a real number is randomly generated in $[l_i, u_i]$ to replace an element x_j in X_i, thus, a new individual $X_i' = (x_{i1}', x_{i2}', \cdots, x_{id}')$ is generated. The mathematical formula is

$$X_i' = \begin{cases} \varepsilon \cdot (u_i - l_i) + l_i, i = j, \\ X_i, i \neq j. \end{cases} \tag{36}$$

ε is the random number in [0,1], u_i is the upper bound of the variable x_i, and l_i is the lower bound.

5.4. Steps to Improve WOA. In summary, the steps of improved WOA are shown in Figure 4.

5.5. Algorithm Time Complexity Analysis. Time complexity can measure the computing speed and execution efficiency of the algorithm, which is one of the key indicators to measure the performance of the algorithm. Suppose that the population number of the algorithm is n, the maximum number of iterations is T, the search latitude is d, the average time of a single inspection beyond the boundary is t_1, and the average time of a single update of the location is t_2.

The WOA uses a linear convergence factor, assuming that the calculation time is t_3. Then, the time complexity of the algorithm is $O(T * n * d * t_2 + T * n * t_1 + T * t_3)$, and the simplified complexity is $O(T * n * d)$. The nonlinear convergence factor is used in the IWOA, and the calculation time is assumed to be t_4. The diversity variation is added, and the calculation time is assumed to be t_5. Then, the time complexity of the algorithm is $O(T * n * d * t_2 + T * n * t_1 + T * (t_4 + t_5))$, and the simplified complexity is $O(T * n * d)$.

It can be seen that the time complexity of IWOA and WOA remains the same order of magnitude, and the time complexity of the improved algorithm does not increase.

Table 5: Comparison of test results.

Tests	x	y	IWOA L_1^*	IWOA L_2^*	IWOA P_S^*	WOA L_1^*	WOA L_2^*	WOA P_S^*	GA L_1^*	GA L_2^*	GA P_S^*
1	5	20	9.31076	10.6892	**0.6409329**	8.10199	11.8980	0.6409328	12.1908	15.5498	0.6409101
2	5	21	8.88211	11.1179	**0.6409331**	8.30287	11.6971	0.6409329	12.3602	13.2476	0.6409180
3	5	22	9.55878	10.4412	**0.6409328**	9.81265	10.1873	0.6409325	13.5191	9.65475	0.6409161
4	5	23	13.6141	6.38594	**0.6409166**	13.7714	6.22863	0.6409153	15.6792	6.15669	0.6408942
5	5	24	16.9736	3.02639	**0.6408727**	17.1397	2.86029	0.6408694	18.2783	2.85869	0.6408431
6	5	25	17.0312	2.96875	**0.640871**6	17.1265	2.87352	0.6408696	17.9551	2.92603	0.6408513
7	5	26	17.0650	2.93495	**0.6408709**	18.0227	1.97730	0.6408496	18.3023	2.85368	0.6708425
8	6	20	13.1647	6.83533	**0.6420127**	13.3492	6.65084	0.6420114	15.6296	6.31112	0.6419876
9	6	21	17.4328	2.56721	**0.641956**0	17.6574	2.34260	0.6419510	18.5669	2.31422	0.6419280
10	6	22	17.4937	2.50627	**0.6419547**	17.8038	2.19622	0.6419476	18.7671	2.25703	0.6419223

6. Results and Discussion

6.1. Case Analysis of Position Optimization. Assuming that the task level is three, the possibility coefficients under each task level and other parameters in the optimization model are taken as shown in Table 1.

In addition, the values of m and n under different task levels are shown in Table 2.

Using the constructed two-stage model solving algorithm, Equations (24), (25), (27), and (28) are solved to obtain the value sequence of x and y, as shown in Table 3.

Then, the improved WOA is used to solve the problem. The parameters of the algorithm are as follows: population number is 200, maximum iteration number is 300, constant $b = 1$, initial convergence factor $\delta_{\text{initial}} = 2$, final value $\delta_{\text{final}} = 1$, and nonlinear adjustment coefficient $\mu = 25$. The simulation results are shown in Table 4.

It can be seen from Table 4 that when $x = 6$, $y = 20$, $L_1 = 13.1647$, and $L_2 = 6.83533$, the survival probability of missile weapon system is the largest and the probability is 0.6420127. At the same time, other data obtained by solving the model can also be used for reference. When considering the actual factors such as topography, road traffic, and so on, decision makers can choose a more realistic position configuration scheme according to multiple sets of data.

6.2. Performance Analysis of Improved Algorithm

6.2.1. Comparison of Model Test Problems. The 10 value sequences of x and y in the deployment optimization model are used as test problems, as shown in Table 3. The improved WOA, WOA, and genetic algorithm (GA) [32] are used to solve each test problem, and Table 5 is obtained. In order to ensure the effectiveness of the experiment, the population number of the three algorithms is set to 200, and the maximum number of iterations is 300. The other parameters of GA are crossover ratio is 0.8, and mutation probability is 0.2.

Compared with the three groups of data in Table 5, when using the improved WOA and WOA to solve the survival probability, the numerical value is greater than that of the genetic algorithm, indicating that the improved WOA and WOA model are better. In the 10 groups of test problems, the survival probability obtained by the improved WOA is slightly better than those obtained by the WOA, but the amplitude is not large. To better reflect the advantages of the improved WOA, this paper selects 10 classical test functions from reference [17] and combines them with 10 basic test functions in CEC2017 test set to further verify the performance of the improved WOA.

6.2.2. Comparison of Classical Test Functions. The function names, expressions, and other features of the 10 test functions are shown in Table 6.

In this paper, the improved WOA and WOA are used to solve the test function, respectively. The dimensions of the test function are set to 50 dimensions, 200 dimensions, and 1000 dimensions, respectively. Through the solution results of the two algorithms, the performance of the improved algorithm is analyzed and compared.

In this paper, two indexes, accuracy (AC) and optimization success rate (SR), are selected to evaluate the performance of the improved algorithm [33]. The closeness between the result obtained by the algorithm and the global optimal solution is called the accuracy. Assuming that the global optimal solution of the problem is X^* and the optimal solution obtained after $k_{\max}$ iterations of the algorithm is P_{best}, the expression of AC is

$$AC = |f(X^*) - f(P_{\text{best}})|. \tag{37}$$

After many experiments, the proportion of the algorithm converging to the optimal solution of the problem is called the success rate of optimization. If the total number of experiments is z and the number of times the algorithm converges to the optimal solution of the problem is z_e, the expression of SR is

$$\text{SR} = 100\% \times \frac{z_e}{z}. \tag{38}$$

In each experiment, if the AC of the algorithm is less than the convergence accuracy of the test function, it is

Table 6: Classical test function.

Name	Function	Range	$f_{\min}$	Convergence accuracy
Sphere	$f_1(x) = \sum_{i=1}^{d} x_i^2$	[-100, 100]	0	1×10^{-08}
Schwefel 2.22	$f_2(x) = \sum_{i=1}^{d} \lvert x_i \rvert + \prod_{i=1}^{d} \lvert x_i \rvert$	[-10, 10]	0	1×10^{-08}
Sumpower	$f_3(x) = \sum_{i=1}^{d} \lvert x_i \rvert^{(i+1)}$	[-1, 1]	0	1×10^{-08}
Rosenbrock	$f_4(x) = \sum_{i=1}^{d} \left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right]$	[-5, 10]	0	1×10^{0}
Quartic	$f_5(x) = \sum_{i=1}^{d} i x_i^4 + \text{random}(0, 1)$	[-1.28, 1.28]	0	1×10^{-04}
Rastrigin	$f_6(x) = \sum_{i=1}^{d} \left[x_i^2 - 10 \cos(2\pi x_i) + 10\right]$	[-5.12, 5.12]	0	1×10^{-08}
Ackley	$f_7(x) = -20 \exp\left(-0.2\sqrt{1/d \cdot \sum_{i=1}^{d} x_i^2}\right) - \exp\left(1/d \cdot \sum_{i=1}^{d} \cos(2\pi x_i)\right) + 20 + e$	[-32, 32]	0	1×10^{-08}
Griewank	$f_8(x) = 1/4000 \cdot \sum_{i=1}^{d} x_i^2 - \prod_{i=1}^{d} \cos\left(x_i/\sqrt{i}\right) + 1$	[-600, 600]	0	1×10^{-08}
Inverted cosine mixture	$f_9(x) = 0.1n - \left(0.1\sum_{i}^{n} \cos(5\pi x_i) - \sum_{i=1}^{n} x_i^2\right)$	[-1, 1]	0	1×10^{-08}
Levy Montalo	$f_{10}(x) = 0.1 \cdot \left\{\sin^2(3\pi x_1) + \sum_{i=1}^{d} (x_i - 1)^2 \left[1 + \sin^2(3\pi x_{i+1})\right] + (x_d - 1)^2 \cdot \left[1 + \sin^2(2\pi x_d)\right]\right\} + \sum_{i=1}^{d} u(x_i, 5, 100, 4)$	[-5, 5]	0	1×10^{-08}

Table 7: Optimization comparison of two algorithms (classical test function).

F	Algorithm	$D = 50$			$D = 200$			$D = 1000$		
		AC_{ave}	SD	SR	AC_{ave}	SD	SR	AC_{ave}	SD	SR
$f_1(x)$	WOA	2.14E − 083	1.49E − 082	100%	3.90E − 081	2.01E − 080	100%	3.24E − 080	1.39E − 079	100%
	IWOA	**2.19E − 086**	**1.51E − 085**	100%	**4.05E − 089**	**2.60E − 088**	100%	**1.69E − 087**	**7.04E − 087**	100%
$f_2(x)$	WOA	9.16E − 035	5.60E − 034	100%	9.67E − 052	6.07E − 051	100%	1.94E − 049	1.24E − 048	100%
	IWOA	**1.37E − 050**	**6.21E − 050**	100%	**1.21E − 053**	**5.76E − 053**	100%	**9.06E − 052**	**6.34E − 051**	100%
$f_3(x)$	WOA	3.39E − 125	1.90E − 124	100%	4.07E − 125	2.81E − 124	100%	1.01E − 123	5.72E − 123	100%
	IWOA	**1.00E − 179**	**0.00E + 00**	100%	**4.63E − 194**	**0.00E + 00**	100%	**1.09E − 195**	**0.00E + 00**	100%
$f_4(x)$	WOA	4.77E + 001	4.00E − 001	0%	1.97E + 002	2.15E − 001	0%	9.92E + 002	9.00E − 001	0%
	IWOA	**4.76E + 001**	**2.60E − 001**	0%	1.97E + 002	**1.74E − 001**	0%	**9.91E + 002**	**5.66E − 001**	0%
$f_5(x)$	WOA	2.30E − 003	2.39E − 003	4%	2.93E − 003	3.40E − 003	2%	2.89E − 003	3.00E − 003	4%
	IWOA	**5.23E − 003**	**6.01E − 003**	**6%**	**1.36E − 003**	**1.62E − 003**	**4%**	**1.53E − 003**	**1.56E − 003**	**10%**
$f_6(x)$	WOA	3.34E − 015	1.76E − 014	100%	0.00E + 00	0.00E + 00	100%	0.00E + 00	0.00E + 00	100%
	IWOA	**2.27E − 015**	**1.59E − 014**	100%	0.00E + 00	0.00E + 00	100%	0.00E + 00	0.00E + 00	100%
$f_7(x)$	WOA	5.08E − 015	2.32E − 015	100%	4.87E − 015	2.42E − 015	100%	4.44E − 015	2.46E − 015	100%
	IWOA	**3.94E − 015**	**2.25E − 015**	100%	**3.38E − 015**	**2.38E − 015**	100%	**4.30E − 015**	**2.35E − 015**	100%
$f_8(x)$	WOA	6.50E − 003	3.20E − 002	96%	0.00E + 00	0.00E + 00	100%	0.00E + 00	0.00E + 00	100%
	IWOA	**2.22E − 018**	**1.55E − 017**	**100%**	0.00E + 00	0.00E + 00	100%	0.00E + 00	0.00E + 00	100%
$f_9(x)$	WOA	0.00E + 00	0.00E + 00	100%	0.00E + 00	0.00E + 00	100%	0.00E + 00	0.00E + 00	100%
	IWOA	0.00E + 00	0.00E + 00	100%	0.00E + 00	0.00E + 00	100%	0.00E + 00	0.00E + 00	100%
$f_{10}(x)$	WOA	1.43E − 001	9.52E − 002	0%	6.17E − 001	**2.81E − 001**	0%	6.66E + 00	3.50E + 00	0%
	IWOA	**4.64E − 002**	**2.54E − 002**	0%	**5.71E − 001**	4.39E − 001	0%	**4.80E + 00**	**1.48E + 00**	0%

Table 8: Comparison of convergence algebra of two algorithms.

F	WOA						IWOA					
	$D = 50$		$D = 200$		$D = 1000$		$D = 50$		$D = 200$		$D = 1000$	
	ca_{ave}	ca_{min}	ca_{ave}	ca_{min}	ca_{ave}	ca_{min}	ca_{ave}	ca_{min}	ca_{ave}	ca_{min}	ca_{ave}	ca_{min}
$f_1(x)$	142	119	151	124	157	135	**81**	**73**	**84**	**72**	**86**	**80**
$f_2(x)$	170	153	179	159	185	167	**99**	**90**	**103**	**92**	**101**	**90**
$f_3(x)$	**45**	**27**	**47**	**32**	**48**	**28**	56	48	56	43	56	47
$f_4(x)$	0	0	0	0	0	0	0	0	0	0	0	0
$f_5(x)$	346	271	351	351	229	224	**318**	**177**	**410**	**387**	**244**	**144**
$f_6(x)$	141	119	147	113	155	130	**111**	**51**	**92**	**78**	**92**	**77**
$f_7(x)$	181	168	187	171	187	163	**105**	**93**	**105**	**97**	**103**	**15**
$f_8(x)$	142	120	145	125	152	125	**85**	**72**	**83**	**72**	**82**	**76**
$f_9(x)$	110	89	123	100	132	105	**72**	**63**	**73**	**65**	**74**	**65**
$f_{10}(x)$	0	0	0	0	0	0	0	0	0	0	0	0

considered that this experiment has converged to the global optimal solution.

In the comparative experiment, the two algorithms adopt the same parameter settings, the population size $N = 30$, and the maximum number of iterations $k_{max} = 500$. Other parameters of the Improved WOA are constant $b = 1$, initial value of convergence factor $\delta_{initial} = 2$, final value $\delta_{final} = 1$, and nonlinear adjustment coefficient $\mu = 25$. Each test function is run 30 times with the two algorithms, respectively, and the optimal precision value, the worst precision value, the average accuracy (AC_{ave}), the precision standard deviation (SD), and the SR obtained from the experiment are recorded. The experimental results are shown in Table 7. Among them, to more clearly reflect the

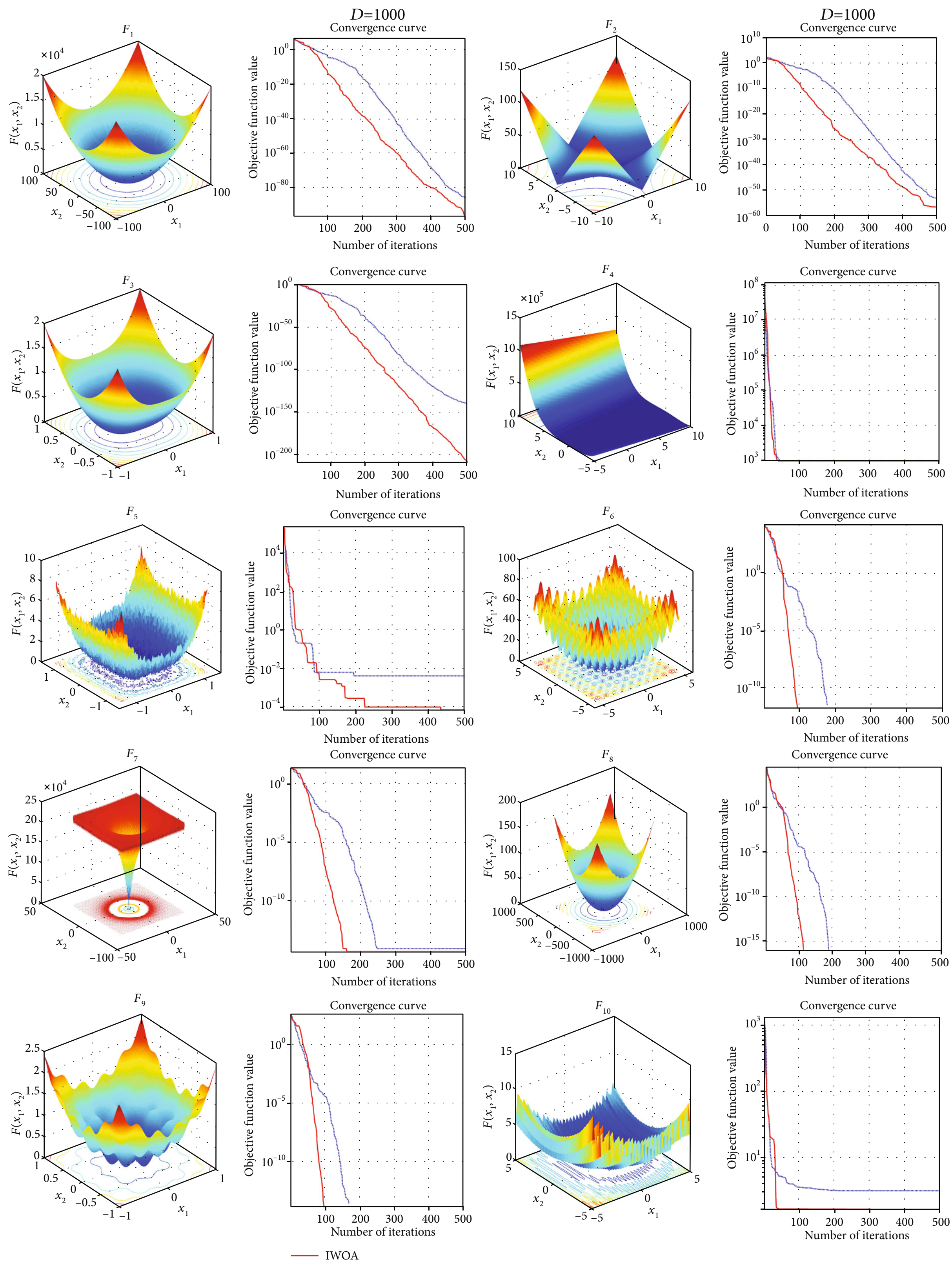

FIGURE 5: Comparison of convergence curves between improved WOA and WOA.

Table 9: Rank sum test value under classical test function.

F	p_1	p_2	p_3
$f_1(x)$	$1.23E-06$ +	$2.23E-09$ +	$5.06E-09$ +
$f_2(x)$	$7.55E-10$ +	$2.12E-05$ +	$1.20E-08$ +
$f_3(x)$	$7.56E-10$ +	$7.56E-10$ +	$7.56E-10$ +
$f_4(x)$	$1.16E-01$ –	$2.50E-01$ –	$8.15E-08$ +
$f_5(x)$	$5.04E-03$ +	$3.00E-03$ +	$9.00E-03$ +
$f_6(x)$	NA=	NA=	NA=
$f_7(x)$	$3.94E-02$ +	$6.10E-03$ +	$9.42E-03$ +
$f_8(x)$	$5.00E-01$ =	NA=	NA=
$f_9(x)$	NA=	NA=	NA=
$f_{10}(x)$	$1.42E-08$ +	$1.99E-01$ –	$2.48E-03$ +

experimental comparison, better results have been marked with black bold.

It can be seen from Table 7 that when the function dimension is 50, the improved WOA can converge to the global optimal solution when solving the other eight functions except $f_4(x)$ and $f_{10}(x)$ and can converge to the theoretically optimal value 0 when solving $f_9(x)$. When solving $f_1(x)$, $f_2(x)$, and $f_3(x)$, the solution obtained by the improved algorithm is very close to the theoretically optimal value 0 ($E-086$, $E-054$, and $E-179$, respectively). Compared with WOA, the improved WOA obtains better optimization results when solving $f_1(x)$, $f_2(x)$, $f_3(x)$, $f_8(x)$, and $f_{10}(x)$. When solving $f_4(x)$, $f_5(x)$, $f_6(x)$, and $f_7(x)$, the result of the improved WOA is slightly better than that of the WOA. For the success rate of optimization, when solving $f_1(x)$, $f_2(x)$, $f_3(x)$, $f_6(x)$, $f_7(x)$, and $f_9(x)$, the SR of both algorithms is 100%. When solving $f_5(x)$ and $f_8(x)$, the SR of the improved WOA is slightly higher than that of the WOA. When solving $f_4(x)$, the success rate of both algorithms is 0.

When the function dimension is 200, for $f_6(x)$, $f_8(x)$, and $f_9(x)$, both algorithms can converge to the theoretically optimal solution 0. In contrast, the improved WOA can get better results when solving most functions.

When the function dimension is 1000, the optimization performance of the two algorithms has little change. This also fully demonstrates that the robustness of the two algorithms is strong when solving large-scale optimization problems.

Table 8 shows the comparison of the average convergence algebra (ca_{ave}) and minimum convergence algebra (ca_{min}) of the two algorithms when solving test functions with different dimensions.

It can be seen from Table 8 that compared with the two algorithms, when solving the nine functions except $f_3(x)$, the average convergence algebra and the minimum convergence algebra of the improved WOA are smaller than those of the WOA, indicating that the improved WOA has faster solving speed. At the same time, it can be seen from Tables 7 and 8 that the improved WOA can effectively deal with large-scale optimization problems.

To show the performance of the improved WOA more intuitively, Figure 5 shows the image of the test function and the convergence curves of the two algorithms when solving the test function (dimension is 1000). It can be clearly seen that the convergence speed and convergence accuracy of the improved WOA are better than those of the WOA.

The above mainly compares the performance of the improved algorithm from the optimal solution and the minimum number of iterations obtained by the algorithm. From the comparison results, the performance of the improved algorithm is better than the original algorithm. Next, the performance of the improved algorithm is evaluated from the perspective of statistics. Wilcoxon test [34] is used to test and count the data that the algorithm runs 50 times independently when solving classical test functions with different dimensions. The significance level is 0.05. If the inspection value $p < 0.05$, it indicates that the performance difference is significant, otherwise, the difference is not obvious. Use p_1, p_2, and p_3 to represent the inspection value when the dimension is 50, 200, and 1000, respectively. The test results are shown in Table 9, in which, "+" indicates that the performance ratio result of the improved algorithm is "excellent", "-" indicates that the comparison result is "inferior", "=" indicates that the performance of the two algorithms is equivalent, and "NA" indicates that the comparison is invalid, that is, the performance of the two algorithms is equivalent.

According to the Wilcoxon rank sum test results, most of the test values compared with the improved algorithm are less than 0.05. The test value of functions $f_6(x)$ and $f_9(x)$ is "NA," that is, when solving $f_6(x)$ and $f_9(x)$, the performance of the improved algorithm is equivalent to that of the original algorithm. When the dimension of function $f_4(x)$ is low, the performance of the improved algorithm is inferior to that of the original algorithm. In addition, there are significant differences in solving other functions. Therefore, according to the Wilcoxon rank sum test results, from a statistical point of view, the improved algorithm is effective.

6.2.3. Comparison of CEC2017 Test Functions. In order to further verify the performance of the improved algorithm, this paper selects 10 basic test functions of different types in CEC2017 test set [35] to compare and test the algorithm. Test function information is shown in Table 10.

Similarly, the improved WOA and WOA are used to solve the CEC2017 test function, and the dimensions are set to 10, 50, and 100 dimensions, respectively. The parameter setting of the algorithm is the same as the previous section. Each test function was run 50 times with two algorithms, and the optimal precision value, the worst precision value, the average precision value, the standard difference of precision, and the success rate were recorded. The experimental results are shown in Table 11. In order to more clearly reflect the experimental comparison, better results have been marked with black bold.

Table 11 shows that when the dimension is 10, the improved algorithm can converge to the global optimal solution when solving 10 basic test functions and can converge

Table 10: CEC2017 test function.

	F	Name	Range	$f_{\min}$
Unimodal functions	CEC01	Bent cigar function	[-100, 100]	0
	CEC02	Sum of different power function	[-100, 100]	0
	CEC03	Zakharov function	[-100, 100]	0
Simple multimodal functions	CEC04	Rosenbrock's function	[-100, 100]	0
	CEC05	Rastrigin's function	[-100, 100]	0
	CEC08	Noncontinuous rotated Rastrigin's function	[-100, 100]	0
	CEC09	Levy function	[-100, 100]	0
Hybrid functions	CEC11	High conditioned elliptic function	[-100, 100]	0
	CEC12	Discus function	[-100, 100]	0
	CEC13	Ackley's function	[-100, 100]	0

Table 11: Optimization comparison of two algorithms (CEC2017).

F	Algorithm	$D = 50$			$D = 200$			$D = 1000$		
		AC_{ave}	SD	SR	AC_{ave}	SD	SR	AC_{ave}	SD	SR
CEC01	WOA	$2.25E-50$	$1.49E-49$	100%	$6.59E-50$	$2.95E-49$	100%	$2.41E-49$	$1.14E-48$	100%
	IWOA	$\mathbf{6.76E-72}$	$\mathbf{3.41E-71}$	100%	$\mathbf{1.53E-64}$	$\mathbf{9.75E-64}$	100%	$\mathbf{1.14E-64}$	$\mathbf{5.83E-64}$	100%
CEC02	WOA	$0.00E+00$	$0.00E+00$	100%	$0.00E+00$	$0.00E+00$	100%	$0.00E+00$	$0.00E+00$	100%
	IWOA	$0.00E+00$	$0.00E+00$	100%	$0.00E+00$	$0.00E+00$	100%	$0.00E+00$	$0.00E+00$	100%
CEC03	WOA	$1.64E+01$	$2.04E+01$	16%	$8.92E+02$	$1.59E+02$	0%	$1.80E+03$	$3.03E+02$	0%
	IWOA	$\mathbf{1.93E+00}$	$\mathbf{2.48E+00}$	**62%**	$\mathbf{8.12E+02}$	$\mathbf{1.36E+02}$	0%	$\mathbf{1.69E+03}$	$\mathbf{1.64E+02}$	0%
CEC04	WOA	$1.08E+01$	$2.83E+01$	0%	$6.87E+00$	$4.09E-01$	0%	$6.87E+00$	$4.24E-01$	0%
	IWOA	$\mathbf{6.35E+00}$	$\mathbf{1.28E+00}$	**4%**	$\mathbf{6.11E+00}$	$\mathbf{1.75E+00}$	**8%**	$\mathbf{6.48E+00}$	$\mathbf{9.28E-01}$	**2%**
CEC05	WOA	$0.00E+00$	$0.00E+00$	100%	$0.00E+00$	$0.00E+00$	100%	$0.00E+00$	$0.00E+00$	100%
	IWOA	$0.00E+00$	$0.00E+00$	100%	$0.00E+00$	$0.00E+00$	100%	$0.00E+00$	$0.00E+00$	100%
CEC08	WOA	$3.58E+00$	$8.22E+00$	68%	$7.11E-17$	$3.48E-16$	100%	$3.55E-17$	$2.49E-16$	100%
	IWOA	$\mathbf{0.00E+00}$	$\mathbf{0.00E+00}$	**100%**	$\mathbf{0.00E+00}$	$\mathbf{0.00E+00}$	**100%**	$\mathbf{0.00E+00}$	$\mathbf{0.00E+00}$	**100%**
CEC09	WOA	$1.13E-01$	$1.17E-01$	100%	$1.88E+00$	$6.52E-01$	100%	$4.35E+00$	$6.82E-01$	100%
	IWOA	$\mathbf{4.95E-04}$	$\mathbf{7.59E-04}$	100%	$\mathbf{7.21E-01}$	$\mathbf{3.23E-01}$	100%	$\mathbf{3.49E+00}$	$\mathbf{6.53E-01}$	100%
CEC11	WOA	$1.67E-53$	$8.70E-53$	100%	$4.64E-50$	$3.25E-49$	100%	$4.62E-53$	$2.34E-52$	100%
	IWOA	$\mathbf{9.59E-71}$	$\mathbf{5.49E-70}$	100%	$\mathbf{4.51E-68}$	$\mathbf{2.82E-67}$	100%	$\mathbf{1.45E-67}$	$\mathbf{6.72E-67}$	100%
CEC12	WOA	$2.70E-56$	$1.18E-55$	100%	$1.62E-58$	$6.20E-58$	100%	$1.20E-55$	$5.23E-55$	100%
	IWOA	$\mathbf{3.53E-76}$	$\mathbf{1.67E-75}$	100%	$\mathbf{5.52E-71}$	$\mathbf{3.75E-70}$	100%	$\mathbf{6.53E-69}$	$\mathbf{4.54E-68}$	100%
CEC13	WOA	$4.80E-15$	$\mathbf{2.27E-15}$	100%	$4.65E-15$	$\mathbf{2.06E-15}$	100%	$4.87E-15$	$\mathbf{2.53E-15}$	100%
	IWOA	$\mathbf{3.87E-15}$	$2.60E-15$	100%	$\mathbf{3.94E-15}$	$2.25E-15$	100%	$\mathbf{4.16E-15}$	$2.64E-15$	100%

to the theoretically optimal value 0 when solving functions CEC02, CEC05, nd CEC08. Compared with the original algorithm, the improved algorithm obtains better optimization results except functions CEC02 and CEC05. When solving functions CEC03, CEC04, and CEC08, the optimization success rate of the improved algorithm is better than that of the original algorithm.

When the dimension is 50, the improved algorithm can converge to the theoretically optimal value 0 when solving CEC02, CEC05, and CEC08. Compared with the original algorithm, the improved algorithm also obtains better optimization results.

When the dimension is 100, the optimization performance of the two algorithms has little change. From the data in Table 11, the performance of the improved algorithm is better than the original algorithm.

Similarly, Wilcoxon rank sum test is used to test and count the data independently running 50 times when the algorithm solves the basic test function of CEC2017 in different dimensions. p_1, p_2, and p_3 are used to represent the test values of dimensions 10, 50, and 100, respectively. The test results are shown in Table 12, where "+" denotes that the performance ratio of the improved algorithm is "excellent," "-" denotes that the comparison result is "inferior,"

Table 12: Rank sum test value under CEC2017.

F	p_1	p_2	p_3
CEC01	$7.56E-10$ +	$7.56E-10$ +	$8.03E-10$ +
CEC02	NA=	NA=	NA=
CEC03	$5.58E-07$ +	$9.54E-03$ +	$4.31E-02$ +
CEC04	$3.50E-03$ +	$1.73E-04$ +	$5.27E-04$ +
CEC05	NA=	NA=	NA=
CEC08	$3.91E-03$ +	NA=	NA=
CEC09	$8.03E-10$ +	$1.47E-09$ +	$7.52E-07$ +
CEC11	$8.53E-10$ +	$8.03E-10$ +	$7.56E-10$ +
CEC12	$7.56E-10$ +	$7.56E-10$ +	$1.30E-09$ +
CEC13	$4.84E-02$ +	$8.09E-02$ −	$1.92E-01$ −

"=" indicates that the performance of the two algorithms is equivalent, and "NA" indicates that the comparison is invalid, that is, the performance of the two algorithms is equivalent.

According to the Wilcoxon rank sum test results, most of the test values obtained by comparison with the improved algorithm are less than 0.05. The test values of functions CEC02, CEC05, and CEC08 are "NA," indicating that when solving these three test functions, the performance of the improved algorithm is equivalent to that of the original algorithm. Except function CEC13, there are significant differences in solving other functions. Therefore, it can be considered that the improved algorithm is effective.

7. Conclusions

In this paper, an optimal deployment model based on MINLP is established for the position deployment optimization of mobile conventional missiles, and a two-stage solution method of the model is proposed. The whale optimization algorithm is improved from two aspects of convergence factor and optimal individual variation, which provides a new method for solving the position deployment optimization problem. The example shows that the model established in this paper is easy to solve and is helpful to the rational allocation of missile positions. And the calculation result of the model is no longer a single data, but a group of data with reference significance, which is more conducive to the deployment and construction of missile positions. At the same time, the experimental result of test functions shows that the improved WOA proposed in this paper has fast convergence speed and high convergence accuracy and can effectively solve large-scale optimization problems.

At the same time, the models and methods proposed in this paper also have certain limitations. On the one hand, the deployment optimization of positions is a complex optimization problem, involving many factors, which needs to be further improved in the assumption of operation scheme and position configuration mode. On the other hand, although the improved WOA proposed in this paper has a good optimization effect, compared with the WOA, the convergence speed of the improved algorithm in the early stage of search is not much improved, and it also needs to be further improved.

Next, the research will be improved from two aspects: the first is to explore the more complex location configuration mode for location optimization problems. The second is for the improved WOA, and the generation method of the initial population of the algorithm will be improved to accelerate the convergence speed of the improved algorithm in the initial search.

Data Availability

The simulation data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there is no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was financially supported by the National Natural Science Foundation of China (no. 11701564) and the Natural Science Foundation of Shaanxi Province (no. 2020JQ-485). The author would like to thank Dr. You for his useful guidance and Dr. Ma for his help in the experiment.

References

[1] M. L. Wang and X. G. Gao, "Optimal design of strategic missile maneuvering position deployment," *System Engineering Theory and Practice*, vol. 10, pp. 94–98, 1999.

[2] G. J. Wang and K. P. Wei, "Study on position allocation and fire distribution of certain missile," *Tactical Missile Technology*, vol. 4, pp. 21–25, 2003.

[3] J. Z. Hao, "Optimal design of land-based conventional missile position deployment," *Military Operations and Systems Engineering*, vol. 2, pp. 70–73, 2004.

[4] T. Xie, M. L. Wang, J. Q. Zhong, and Q. Fang, "Research on position configuration optimization of conventional missile," *Journal of System Simulation*, vol. 21, no. 6, pp. 1535–1537, 2009.

[5] H. Zeng, Y. L. Wang, and F. X. Liu, "Graph theory and application of neural network in ground to air missile force position," *Fire Control and Command Control*, vol. 2, pp. 104–106, 2007.

[6] K. Wang, X. B. Yuan, and M. L. Lu, "Study on optimization of ground-air missile position configuration," *Ship Electronic Engineering*, vol. 31, no. 6, pp. 37–41, 2011.

[7] R. Liu, X. M. Li, and L. J. Liu, "Analysis on optimal configuration of naval air defense positions," *Ship Electronic Engineering*, vol. 32, no. 8, 2012.

[8] Z. H. Gao, X. Y. Fan, Y. Sun, and M. Y. Li, "Research on position deployment of air defense weapon system based on shooting number model," *Computer and Digital Engineering*, vol. 41, no. 4, pp. 549–555, 2013.

[9] S. Cheng, R. Wang, G. H. Wu, Y. N. Guo, L. B. Ma, and Y. H. Shi, "Swarm intelligence optimization algorithm," *Journal of Zhengzhou University*, vol. 39, no. 6, pp. 1-2, 2018.

[10] M. Dorigo, V. Maniezzo, and A. Colorni, “Ant system: optimization by a colony of cooperating agents,” *IEEE Trans on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 26, no. 1, pp. 29–41, 1996.

[11] W. T. Pan, “A new fruit fly optimization algorithm: taking the financial distress model as an example,” *Knowledge-Based Systems*, vol. 26, pp. 69–74, 2012.

[12] S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey wolf optimizer,” *Advances in Engineering Software*, vol. 69, no. 7, pp. 46–61, 2014.

[13] G. G. Wang, S. Deb, and Z. H. Cui, “Monarch butterfly optimization,” *Neural Computing and Applications*, vol. 31, no. 7, pp. 1995–2014, 2019.

[14] A. A. Mohamed, S. M. Yahia, A. M. Ahmed, and A. M. Hemeida, “Optimal power flow using moth swarm algorithm,” *Electric Power Systems Research*, vol. 142, pp. 190–206, 2017.

[15] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, “Slime mould algorithm: a new method for stochastic optimization,” *Future Generation Computer Systems*, vol. 111, pp. 300–323, 2020.

[16] S. Mirjalili and A. Lewis, “The whale optimization algorithm,” *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.

[17] W. Long, S. H. Cai, J. J. Jiao, M. Tang, and T. Wu, “Improved whale optimization algorithm for large scale optimization problems,” *System Engineering Theory and Practice*, vol. 37, no. 11, pp. 2983–2994, 2017.

[18] J. H. Wang, L. Zhang, C. Shi, F. Che, G. Ding, and j. Zhou, “Whale optimization algorithm based on chaos search strategy,” *Control and Decision*, vol. 34, no. 9, pp. 1893–1900, 2019.

[19] G. Y. Ning and D. Q. Cao, “Improved whale optimization algorithm for solving constrained optimization problems,” *Discrete Dynamics in Nature and Society*, vol. 2021, Article ID 8832251, 13 pages, 2021.

[20] S. H. Li, X. H. Luo, and L. Z. Wu, “An improved whale optimization algorithm for locating critical slip surface of slopes,” *Advances in Engineering Software*, vol. 157, article 103009, 2021.

[21] A. Donyaii, A. Sarraf, and H. Ahmadi, “A novel approach to supply the water reservoir demand based on a hybrid whale optimization algorithm,” *Shock and Vibration*, vol. 2020, Article ID 8833866, 15 pages, 2020.

[22] S. M. Bozorgi and S. Yazdani, “IWOA: an improved whale optimization algorithm for optimization problems,” *Journal of Computational Design and Engineering*, vol. 6, no. 3, pp. 243–259, 2019.

[23] Z. C. Ye, X. Y. Mo, and L. Zhao, “MINLP model for operational optimization of LNG terminals,” *Processes*, vol. 9, no. 4, p. 599, 2021.

[24] M. L. Wang and X. D. Yang, “Analysis on survivable threat environment of maneuvering missile system,” *Modern Defense Technology*, vol. 42, no. 1, 2014.

[25] M. L. Wang and S. P. Peng, *Analysis method of missile weapon system survivability*, National Defense Industry Press, Beijing, 2015.

[26] L. Y. Liu, J. S. Zhu, J. G. Wu, and F. Sun, “Application of expert scoring method in L / HIRF evaluation factors,” *Aviation Maintenance & Engineering*, vol. 7, pp. 41–43, 2017.

[27] Y. F. Lin, D. Jiang, and W. L. Du, “A hybrid optimization method for mixed integer nonlinear programming problems,” *Computers and Applied Chemistry*, vol. 31, no. 12, pp. 1447–1451, 2014.

[28] W. A. Watkins and W. E. Schevill, “Aerial observation of feeding behavior in four baleen whales: Eubalaena glacialis, Balaenoptera borealis, Megaptera novaeangliae, and Balaenoptera physalus,” *Journal of Mammalogy*, vol. 60, no. 1, pp. 155–163, 1979.

[29] Z. Z. Wu and F. Song, “Whale optimization algorithm based on improved spiral update position model,” *System Engineering Theory and Practice*, vol. 39, no. 11, pp. 2928–2944, 2019.

[30] Z. L. Wei, H. Zhao, M. D. Li, and Y. Wang, “A grey wolf optimization algorithm based on nonlinear adjustment strategy of control parameter,” *Journal of Air Force Engineering University*, vol. 17, no. 3, pp. 68–72, 2016.

[31] A. D. Tang, T. Hang, D. W. Xu, and L. Xie, “Chaos multi - elite whale optimization algorithm,” *Computer Engineering and Science*, vol. 47, no. 7, pp. 1481–1494, 2021.

[32] H. T. Mei, J. X. Hua, and Y. Wang, “Improved intuitionistic fuzzy genetic algorithm for solving nonlinear programming problems,” *Journal of Beijing University of Aeronautics and Astronautics*, vol. 47, no. 7, pp. 1481–1494, 2021.

[33] Z. L. Geng, M. Li, S. R. Cao, and C. X. Liu, “Whale optimization algorithm based on hybrid reverse learning strategy,” *Computer Engineering and Science*, vol. 44, no. 2, pp. 355–363, 2022.

[34] F. Wilcoxon, “Individual comparisons by ranking methods,” *Biometrics Bulletin (S0099-4987)*, vol. 1, no. 6, pp. 80–83, 1945.

[35] N. H. Awad, M. Z. Ali, J. J. Liang, B. Y. Qu, and P. N. Suganthan, *Problem Definitions and Evaluation Criteria for the CEC 2017 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization*, Nanyang Technological University, Singapore, 2016.