

Research Article

Ship Track Prediction Based on DLGWO-SVR

Yingyu Chen ¹, Shenhua Yang ^{1,2}, Yongfeng Suo,^{1,2} and Minjie Zheng ^{1,2}

¹Navigation College, Jimei University, Xiamen 361021, China

²Xiamen Key Laboratory of Navigation Simulation and Control, Xiamen 361021, Fujian, China

Correspondence should be addressed to Shenhua Yang; yangshh@163.com

Received 25 July 2021; Revised 12 August 2021; Accepted 28 August 2021; Published 15 September 2021

Academic Editor: Pengwei Wang

Copyright © 2021 Yingyu Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To improve the accuracy of ship track prediction, the improved Grey Wolf Optimizer (GWO) and Support Vector Regression (SVR) models are incorporated for ship track prediction. The hunting strategy of dimensional learning was used to optimize the move search process of GWO and balance exploration and exploitation while maintaining population diversity. Selection and updating procedures keep GWO from being stuck in locally optimal solutions. The optimal parameters obtained by modified GWO were substituted into the SVR model to predict ship trajectory. Dimension Learning Grey Wolf Optimizer and Support Vector Regression (DLGWO-SVR), Grey Wolf Optimized Support Vector Regression (GWO-SVR), and Differential Evolution Grey Wolf Optimized Support Vector Regression (DEGWO-SVR) model trajectory prediction simulations were carried out. A comparison of the results shows that the trajectory prediction model based on DLGWO-SVR has higher prediction accuracy and meets the requirements of ship track prediction. The results of ship track prediction can not only improve the efficiency of marine traffic management but also prevent the occurrence of traffic accidents and maintain marine safety.

1. Introduction

To introduce unmanned vessels into commercial shipping lanes, an effective collision avoidance system must be built to ensure the level of safety required for unmanned vessel autonomy. Ship track prediction is the prediction of a ship's future navigation state based on the target ship's history and current trajectory. The results of ship track prediction can be used to evaluate collision risk, promote active collision avoidance, serve as a reference for ship collision avoidance decisions, and enhance the ability of situational awareness at sea.

Ship track prediction methods can be classified into three categories: Prediction methods based on a statistical model, probability graph model, and machine learning model. The prediction methods based on the statistical model include Gaussian regression [1], Kalman filter [2] and Bayesian network [3], etc., whose characteristics are as follows: In the modeling, the influence of wind, current, and other environments on the ship movement generally need to be considered. The establishment of the ship kinematics equation will increase the complexity of the

model. As the experiment goes on, the error increases. The prediction methods based on the probability graph model include the Markov model [4] and grey model [5]. Their characteristics are as follows: It combines probability theory and graph theory, abstruse knowledge in different fields into probability model, and reduces problems in application to the calculation of probability distribution of probability model variables. Under the condition of a large amount of data, the grey model has low performance and long-running time, but the improved prediction effect is better than that of the least square method. This method improves the prediction accuracy to some extent, but it is still not ideal. Prediction methods based on the machine learning model include recurrent neural network [6], backpropagation neural network [7], long and short time memory network [8], and Support Vector Regression (SVR) [9]. With the following characteristics: As data amount increases, gradient explosion or gradient disappearance may occur, and the convergence rate is slow, which leads to the decrease of prediction accuracy and the low efficiency of sample training. SVR needs to choose parameters independently, which is subjective, and the

prediction time is short, but the prediction accuracy needs to be further improved.

With the rise and widespread application of intelligent optimization algorithms, an increasing number of researchers have introduced optimization algorithms into prediction tasks, such as Particle Swarm Optimization (PSO) [10], Differential Evolution (DE) [11], Artificial Bee Colony (ABC) [12], Gravitational Search Algorithm (GSA) [13], Genetic Algorithm (GA) [14], and Ant Colony Optimization (ACO) [15], and other algorithms are also proposed for combinatorial optimization. These algorithms can explore the search space continually and find the best or relatively close answer in a reasonable time.

Compared to algorithms such as PSO, GSA, DE, ABC, ACO, and the GWO method produces superior outcomes in unknown space. There are several characteristics. First, the GWO algorithm showed superior exploitation. Second, the exploration ability of GWO was confirmed. Third, the GWO showed high local optima avoidance to some extent. Finally, the convergence analysis of GWO confirmed the convergence of this algorithm [16]. GWO requires fewer control parameters and is easy to implement. It has been used in many disciplines to address various optimization issues, such as parameter estimation [17], economic scheduling [18], pattern recognition [19], feature selection [20], and wind speed prediction [21].

In terms of prediction, GWO has not been found to be used in the research of ship track prediction. The GWO search procedure is carried out by α wolf, β wolf, and δ wolf in each iteration, ensuring that the algorithm is convergent. At the same time, it has several limitations, such as lack of population variety, a disparity between exploration and exploitation, local optimization, and premature convergence.

To balance the exploration and exploitation of GWO, Malik et al. [22] proposed Weighted Distance Grey Wolf Optimizer (WDGWO) to calculate the weighted average of the optimal solution. Long et al. [23] proposed Random Opposition-Based Learning Grey Wolf Optimizer (ROL-GWO) by modifying parameter C, which increased the searching ability of the algorithm. N. Singh and S. B. Singh [24] proposed an improved Grey Wolf Optimization algorithm to solve the economic dynamical load scheduling problem, which can both increase the global search and local search simultaneously. This step can coordinate the behavior of grey wolves, global and local random search, and opposition learning. Jayabarathi et al. [18] solved the economic scheduling problem by incorporating HGWO into GWO via variation and crossover operations. It performs well in solving the constrained nonkernel solution problem, although it does not achieve the proper balance of global search and local search. Gaidhane and Nigam [25] proposed GWO-ABC, a hybrid that combines the benefits of GWO and ABC. GWO adopted the information-sharing strategy of ABC to improve the global search capability while retaining the original hunting strategy of grey wolves for local search. Saxena et al. [26] proposed Equipped Grey Wolf Optimizer (E-GWO), which used sinusoidal function bridging, tournament selection, crossover, and mutation

operations in the position update stage. The sinusoidal bridging mechanism and evolutionary operation assist the grey wolf in achieving greater accuracy and avoiding the local optimal solution. It shows good searchability in local scenes, but the lack of local search effect remains the main issue in crossover operations with a single peak problem and unbalanced search.

Many scholars enhanced GWO by modifying the GWO mechanism to avoid it falling into local optimality. Mittal et al. [27] developed Modified GWO (MGWO), which is based on a nonlinear control parameter approach and focuses on the right balance of global and local search, and this method may avoid the difficulties of the local optimal solution and premature convergence. Long et al. [28] proposed Exploration Enhanced GWO (EEGWO) to guide the search of new candidates by applying random individuals in the population for a location update. Nonlinear control parameters were employed to grow linearly during the iteration phase to balance the global and local searches of GWO. However, the primary issues remain falling into a local optimum and early convergence. Nasrabadi et al. [29] improved GWO by combining oppositional learning and parallelization to improve the convergence speed and the accuracy of the final result. However, in some cases, the convergence performance of the algorithm was not good. N. Singh and S. B. Singh [24] proposed Hybrid GWOSCA (HGWOSCA) to update the position of α in GWO by using the position update equation of the Sine Cosine Algorithm (SCA), which improved the global convergence and searchability of GWO when dealing with single-mode problems. But its global searching ability in multimode function is limited, and the combinatorial function balance problem persists. Alomoush et al. [30] proposed Harmony Search with Grey Wolf Optimizer (GWO-HS) Algorithm to solve global optimization problems by using opposing learning strategies. Meng et al. [31] proposed a hybrid Crisscross Search-Based Grey Wolf Optimizer (CS-GWO) algorithm, which used two crossover operators to improve the global search ability of α , β and δ wolves while maintaining the population diversity, but the algorithm convergence occurred too soon.

To solve the problem of poor population diversity and slow convergence rate of GWO, a hybrid Grey Wolf Optimizer based on Elite Opposition (EOGWO) was proposed by introducing the elite opposition-based learning strategy and simplex method into GWO [32]. Inspired by the hierarchical social pyramid of the GWO, Rodriguez et al. [33] proposed to introduce hierarchical operators into GWO to simulate the search process and gives five variations. The cellular automata (CA) concept is embedded into the GWO. A Cellular Grey Wolf Optimizer (CGWO) with a topological structure is proposed [34]. CGWO with a topological structure can also help to improve the diversity of the population. To solve the hybrid flow shop scheduling problem considering noise pollution, Lu et al. [35] proposed a Multiobjective Cellular Grey Wolf Optimizer (MOCGWO) algorithm, which integrates the merits of cellular automata for diversification and variable neighborhood search for intensification.

However, GWO is easy to fall into the local optimal solution, and the balance exploration and exploitation remains to be solved. Meanwhile, the selection parameters of the SVR model are subjective.

Alaa and Abdel [36] explore the advantages of the GWO algorithm in estimating the software reliability growth model's parameters with the objective of minimizing the difference between the estimated and the actual number of failures of the software system. Almazini and Mahamud [37] proposed an enhanced binary grey wolf optimization (EBGWO) algorithm for feature selection in anomaly detection by controlling the balancing parameter. This method focused on obtaining a value for a parameter that controlled the trade-off between exploration and exploitation.

Using the GWO, which was inspired by the previous research, the problem of SVR parameter selection can be solved. Improving the accuracy of ship track prediction for real-time data training and modeling is critical for better collision avoidance guidance at sea. A vessel trajectory prediction model based on Grey Wolf Optimized Support Vector Regression (GWO-SVR), which fully utilizes SVR's advantage of maintaining higher precision in the case of nonlinearity and small samples. However, the model has certain setbacks. The GWO model suffers from the lack of population diversity, the imbalance between exploitation and exploration, and premature convergence. Thus, if the previously trained model is still used, the accuracy of prediction decreases.

The main purpose of this study is to overcome the above deficiency of GWO and improve the effect of ship track prediction. The Dimensional Learning Hunting (DLH) strategy was used to improve the search performance of GWO and track the prediction accuracy of DLGWO-SVR. The DLH search approach is based on individual wolf hunting behavior in nature, and it expands the global search domain by using multineighbors learning. The DLGWO then uses both candidate wolves created by the DLH and GWO search algorithms to move the wolf X_i from its current location to a better one in each iteration. In addition, the DLGWO employs a separate selecting and updating process in each iteration to determine the winner candidate wolf and update the current position for the following iteration. The main work of this paper is as follows:

- (1) Analyze the GWO and SVR models, apply the improved GWO in the training process of the SVR model, and establish the model of ship trajectory prediction.
- (2) The hunting search strategy based on dimensional learning is used to improve the search effect of grey wolves. In each iteration, candidate wolves are generated by the search strategy of DLH and GWO.
- (3) Select and update operations are used in each iteration to select the winning candidate wolf to update the position of the grey wolf in the next iteration.
- (4) The prediction experiment was carried out with AIS data, and the experimental results of GWO-SVR, Differential Evolution Grey Wolf Optimized Support

Vector Regression (DEGWO-SVR), and DLGWO-SVR models were compared and analyzed.

The remainder of this paper can be divided into the following four parts. Section 2 mainly proposes the SVR and GWO related principles. The vessel track prediction based on DLGWO-SVR is then presented in Section 3. Extensive experiments are carried out in Section 4. We end this work by summing up the main contributions in Section 5.

2. The Related Principles

2.1. Principle of SVR. SVR nonlinear mapping of the original data to high-dimensional feature space for linear regression.

Figure 1 shows the structure of the SVR model in the data regression. $K(x_i, y_i)$ represents the kernel function.

In the SVR model, given the data set $\{(x_i, y_i)\}_{i=1}^N \in R$, where x_i is the input vector of the SVR model and y_i is the true output value. N is the total number of data points used. The output of the SVR model is as follows:

$$\hat{y} = f(x) = \sum_{i=1}^N w_i \phi_i(x) + b = w^T \varphi(x) + b. \quad (1)$$

$w = [w_1 w_2, \dots, w_N]^T$, $\varphi = [\phi_1 \phi_2, \dots, \phi_N]^T$, the function $\phi_i(x)$ is called the feature. The parameters w and b are the weight and deviation of the support vector machine, respectively. In the feature space, the input vector x is mapped to the high-dimensional kernel-induced feature space vector $\varphi(x)$, and the nonlinear regression is transformed into linear regression. These parameters are calculated by minimizing the regular risk function:

$$R(w) = \frac{1}{2} w^T w + \lambda \sum_{i=1}^N |y_i - f(x)|_\varepsilon, \quad (2)$$

$$|y_i - f(x)|_\varepsilon = \begin{cases} 0, & \text{if } |y_i - f(x)| < \varepsilon, \\ |y_i - f(x)| - \varepsilon, & \text{otherwise.} \end{cases} \quad (3)$$

In equation (2), $R(w)$ is a regular weight vector representing the complexity of the SVR model. Equation (3) represents the estimated vector. Parameter λ and ε is a user-defined parameter, $|y_i - f(x)|_\varepsilon$ is ε intensive loss function. If the predicted value of $f(x)$ is within the error range ε , the loss is equal to 0. For other predicted points outside the range of the error ε , the loss is equal to the dimension of the difference between the predicted value and the error ε .

The regular parameter λ ensures that a good SVR model is generated. Increasing regularization parameters can penalize larger estimation errors. The estimation error can also be reduced by increasing the weight vector norm of the first expression in equation (2). However, an increase in the weight vector norm does not guarantee the generation of the SVR model.

In classical SVR, the appropriate value of the insensitive parameter ε is difficult to determine in advance. The regular risk function of the minimization equation (2) is equivalent to the minimization constrained risk function:

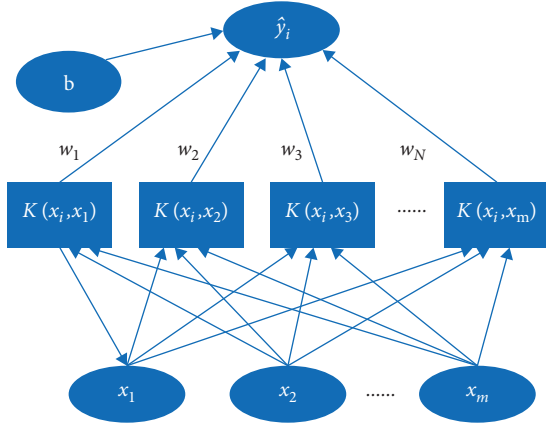


FIGURE 1: Structure of the support vector regression (SVR) model.

$$R(w, \xi, \xi^*) = \frac{1}{2} w^T w + \lambda \sum_{i=1}^N (\xi_i + \xi_i^*). \quad (4)$$

Constraints are as follows:

$$\begin{cases} y_i - w^T \varphi(x) - b \leq \varepsilon + \xi_i, \\ w^T \varphi(x) + b - y_i \leq \varepsilon + \xi_i^*, & i = 1, 2, \dots, N, \\ \xi_i, \xi_i^* \geq 0. \end{cases} \quad (5)$$

Parameter $\xi = [\xi_1, \xi_2, \dots, \xi_N]^T$ and $\xi^* = [\xi_1^*, \xi_2^*, \dots, \xi_N^*]^T$ are the relaxation variable. It represents the maximum and minimum limits on the output of the SVR model, both of which are positive.

The constrained optimization problem of equation (4) can be solved by applying Lagrange multipliers to equations (4) and (5) using standard quadratic programming. The regression function of equation (1) is derived as follows:

$$\begin{aligned} \hat{y} = f(x) &= \sum_{i=1}^N (a_i - a_i^*) \varphi^T(x_i) \varphi(x) \\ &= \sum_{i=1}^N \beta_i K(x, x_i) + b. \end{aligned} \quad (6)$$

Under the Karush-Kuhn-Tucker complementary condition, the parameters w and b are calculated. Parameter K is the kernel function, which is expressed as follows:

$$K(x, x_i) = \varphi^T(x_i) \varphi(x). \quad (7)$$

The Lagrange multiplier a_i and a_i^* are used to express the coefficient β_i . In this study, the SVR model uses the following radial basis kernel functions:

$$K(x, x_i) = \exp\left(-\frac{(x - x_i)^T (x - x_i)}{2\delta^2}\right). \quad (8)$$

A solution is obtained in equation (6) where the coefficient β_i is nonzero only for a subset of the training data.

2.2. Principle of GWO. The grey wolf was inspired by social hierarchy and hunting behavior. Initialize α, β and δ wolves

as the optimal solution, and guide the rest of ω wolves to find the optimal solution. Starting from randomly generating the initial wolves in the search space, the fitting function estimated the position of the wolves and repeated the hunting process of the grey wolves until the predefined number of iterations so as to determine the optimal hunting position, namely the position of α wolves.

There are three main stages in the hunting process: encircling, hunting, and attacking prey.

2.2.1. Encircling Prey. Grey wolves encircle prey during the hunt. Encircling prey is modeled as follows:

$$D = C \cdot Xp(t) - X(t),$$

$$X(t+1) = Xp(t) - A \cdot D, \quad (9)$$

$$A = 2a \cdot r1 - a,$$

$$C = 2 \cdot r2,$$

where t indicates the current iteration, Xp is the prey position, and X is the position vector of a grey wolf. A and C represent coefficient vectors. $r1, r2 \in [0, 1]$. a decreases linearly from 2 to 0.

2.2.2. Hunting. The hunting behavior of grey wolves was modeled on the assumption that α, β and δ wolves knew the location of prey. The following formulas are proposed in this view:

$$D_\alpha = C_1 \cdot X_\alpha - X,$$

$$D_\beta = C_2 \cdot X_\beta - X,$$

$$D_\delta = C_3 \cdot X_\delta - X,$$

$$X_1 = X_\alpha - A_1 \cdot D_\alpha, \quad (10)$$

$$X_2 = X_\beta - A_2 \cdot D_\beta,$$

$$X_3 = X_\delta - A_3 \cdot D_\delta,$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3},$$

where $X_\alpha, X_\beta, X_\delta$ represent the positions of α, β and δ wolves in the t iteration, respectively. X_1, X_2 and X_3 represent the positions of α, β and δ wolves in the $t+1$ iteration, respectively.

2.2.3. Attacking Prey. The prey ceases to move and the hunt ends. The wolves began to attack their prey. The next position of a search agent can be in any position between its current position and the position of the prey. This process is carried out by linear reduction of the iterative curve of local and global search.

Each iteration a updates within the range of $[2, 0]$, half of which is used for global search to achieve a smooth transition effect. The rest is used as a local search, updating the

location of the optimal solution to the location of the prey and the current location of the wolf.

3. The Proposed DLGWO-SVR

3.1. DLGWO Model. In GWO, α, β, δ and ω wolves move toward the search region for the global optimal solution. This behavior could result in a locally optimal solution. Furthermore, the decrease of population diversity makes it easy for grey wolves to fall into local optimum. To address this issue, grey wolf employs dimension learning hunting search strategy to determine candidates and then improves grey wolf algorithm through selection, update, and other operations [38].

Improvements are made in the steps of initialization, movement, and selection and update.

3.1.1. Initialization. In the search space composed of $[l_j, u_j]$, N wolves are randomly distributed.

$$X_{ij} = l_j + \text{rand}_j[0, 1] \times (u_j - l_j), \quad i \in [1, N], j \in [1, D]. \quad (11)$$

The position of the i th Wolf in the t -th iteration is expressed as $X_i(t) = \{x_{i1}, x_{i2}, \dots, x_{iD}\}$. In this case, D represents the dimension of the problem, and the entire population of grey wolves will be stored in a matrix with N rows and D columns. The fitting value of $X_i(t)$ is calculated by the fitting function $f(X_i(t))$.

3.1.2. Movement. Individual hunting of wolves is considered that is learned by its neighbors, the grey wolf in the

neighborhood teaches its neighbor to become another candidate for the new position of $X_i(t)$. The position of X_α, X_β and X_δ determines the position of prey. The first candidate set for the new position of the grey wolf $X_i(t)$ is called $X_i - \text{GWO}(t + 1)$, and this value is calculated by the traditional GWO search strategy.

Each dimension of the new position of the grey wolf $X_i(t)$ is calculated by learning its neighbors and then using equation (14). Another candidate set for the new position of grey wolf $X_i(t)$ forms a DLH search strategy, called $X_i - \text{DLH}(t + 1)$, $R_i(t)$ is solved by calculating the Euclidian distance between the current position $X_i(t)$ and the candidate position $X_i - \text{GWO}(t + 1)$.

$$R_i(t) = \|X_i(t) - X_{i-\text{GWO}}(t + 1)\|, \quad (12)$$

$$N_i(t) = \{X_j(t) \mid D_i(X_i(t), X_j(t)) \leq R_i(t), X_j(t) \in \text{Pop}\}. \quad (13)$$

Through equation (13), considering R_i, D_i is the Euclidian distance between $X_i(t)$ and $X_j(t)$, the neighbors of $X_i(t)$ are denoted as $N_i(t)$. The neighbors of $X_i(t)$ are constructed, and the neighbor $X_{n,d}(t)$ of the D dimension is randomly selected from $N_i(t)$, and the wolf $X_{r,d}(t)$ is randomly selected from the population. Multineighbor learning is performed through equation (14).

3.1.3. Selection and Update. By comparing the fitting values of $X_i - \text{GWO}(t + 1)$ and $X_i - \text{DLH}(t + 1)$, the optimal solution is selected.

$$X_{i-\text{DLH},d}(t + 1) = X_{i,d}(t) + \text{rand} \times (X_{n,d}(t) - X_{r,d}(t)), \quad (14)$$

$$X_i(t + 1) = \begin{cases} X_i - \text{GWO}(t + 1), & \text{if } f(X_i - \text{GWO}) < f(X_i - \text{DLH}), \\ X_i - \text{DLH}(t + 1), & \text{otherwise.} \end{cases} \quad (15)$$

Finally, after all the individuals have completed this step, the iteration counter is incremented by 1, and the next iteration is searched until the predefined number of iterations is reached.

3.2. DLGWO-SVR Model. The improved GWO is used to establish the ship trajectory prediction model based on DLGWO-SVR. Figure 2 shows the flow chart of the DLGWO-SVR prediction model. The detailed experimental steps of DLGWO-SVR are as follows:

Step 1: Initialize the position of N wolves and the relevant parameters of the grey wolf model.

Step 2: Calculate the positions of α, β and δ wolves, input the SVR model for training, and calculate the optimal solution.

Step 3: Update the position of the grey wolf according to the optimal solution.

Step 4: Calculate the distances of α, β and δ wolves to prey, and the positions of α, β and δ wolves, respectively, and calculate the first candidate set $X_i - \text{GWO}(t + 1)$ for the new position of $X_i(t)$.

Step 5: Calculate the Euclidian distance between the candidate position $X_i - \text{GWO}(t + 1)$ and the current position $X_i(t)$, denote as radius $R_i(t)$.

Step 6: Build the neighborhood $N_i(t)$ of $X_i(t)$, and calculate another candidate set $X_{i-\text{DLH},d}(t + 1)$ by using the random neighbors of the d dimension.

Step 7: Select and update operations. By comparing the function values of the candidate set, the position of the next iteration of the grey wolf is determined from $(X_i - \text{GWO}(t + 1), X_i - \text{DLH}(t + 1))$.

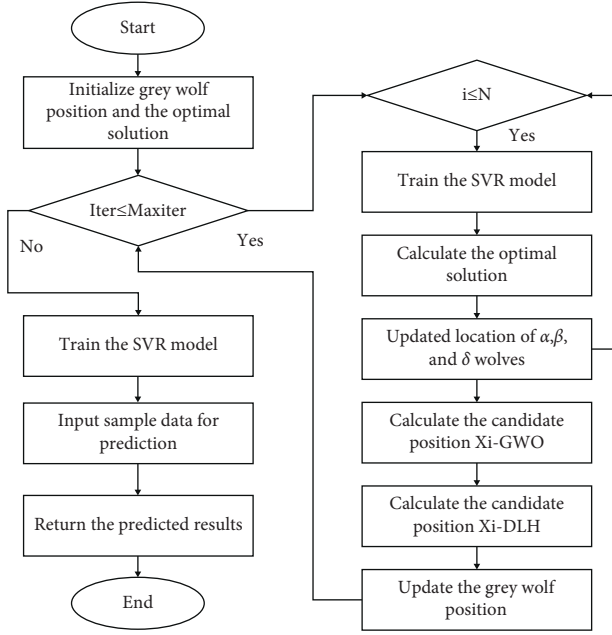


FIGURE 2: Flowchart of the model of DLGWO-SVR.

Step 8: Repeat Steps 2–7 until the preset number of iterations is reached. The experiment is terminated and the predicted results are returned.

4. Results and Discussion

4.1. Datasets. The experiment was modeled and simulated with the help of MATLAB toolbox libsvm-3.24. In this experiment, the AIS data used for prediction are from September to October 2018 of Gulei Port in Zhangzhou, Fujian. The raw data should remove the wrong in the pretreatment and then carry out prediction and verification experiments. There are three types of errors in AIS data.

- (1) The MMSI code of the ship is not 9 digits, or it is an unreasonable record.
- (2) The longitude and latitude of the ship are beyond the reasonable range. For example, the longitude is greater than 180° , the latitude is greater than 90° , or the value of the longitude and latitude is negative.
- (3) The ship's speed and course are beyond the reasonable range, such as the course is greater than 360 or the speed is less than 0 .

Table 1 randomly lists part of the AIS data of September 2018. These data are sample data after preprocessing.

4.2. Predicted Results

4.2.1. Prediction Results for Different MMSI Ships. Input different sample data to the prediction model for the experiment. The predicted experimental results are shown below.

Figure 3 shows the prediction result of MMSI 259681317. Figure 4 shows the prediction result of MMSI 412444715.

Figure 5 shows the prediction result of MMSI 412358490. Figure 6 shows the prediction result of MMSI 412445837. Figure 7 shows the prediction result of MMSI 600003568.

As can be seen from the above figures, the predicted trajectory is basically coincident with the original trajectory, or is parallel to the original trajectory with a very small distance difference near the original trajectory. However, the small distance difference will not affect the actual operation of the staff but is helpful to judge the traffic situation and greatly avoid the occurrence of maritime traffic accidents. Therefore, the above figures show that the new model can accurately predict the ship's trajectory.

4.2.2. Prediction Errors for Different MMSI Ships. To further verify the prediction accuracy of the proposed model, the following is a comparison of the six experimental error curves for trajectory prediction. In the following figure, the num represents the number of track points, and the vertical axis represents the value of the prediction error.

Figure 8 shows the prediction error of MMSI 259681317. Figure 9 shows the prediction error of MMSI 412444715.

Figure 10 shows the prediction error of MMSI 412358490.

Figure 11 shows the prediction error of MMSI 412445837.

Figure 12 shows the prediction error of MMSI 600003568.

As can be seen from Figures 8–12, the prediction error basically fluctuates around the value of 0 . Unlike the others, the error curves in Figure 10 are mostly below 0 and remain within 0.01° . However, they all meet the needs of the nautical practice.

4.3. Model Evaluation. The prediction model is evaluated by means of mean square error, mean absolute error, and mean absolute percentage error. Assume that the true value $y = \{y_1, y_2, \dots, y_n\}$, $y' = \{y'_1, y'_2, \dots, y'_n\}$, then the evaluation index is expressed as follows:

$$E_{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - y'_i|,$$

$$E_{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - y'_i)^2, \quad (16)$$

$$E_{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - y'_i}{y_i} \right|.$$

The E_{MAE} , E_{MSE} , and E_{MAPE} of six trajectories are recorded in Table 2. The prediction error of each trajectory is very small, and the prediction accuracy of the model proposed in this paper can meet the requirements of ship track prediction.

4.4. Error Comparison before and after Model Improvement. To further verify the prediction accuracy of the proposed model, the prediction errors of DLGWO-SVR and GWO-

TABLE 1: Sample data structure.

MMSI	Longitude (°)	Latitude (°)	Course (°)	Speed (kn)	Timestamp (s)
259681317	117.59288	23.687758	293.7	6.8	1535837149
412444715	117.676115	23.611697	305.3	9	1536839207
412358490	117.401963	23.9903	55.9	8.1	1536724254
412445837	117.57915	23.72888	183.4	7.5	1536604590
600003568	117.49264	23.58772	132.1	5.5	1535731218

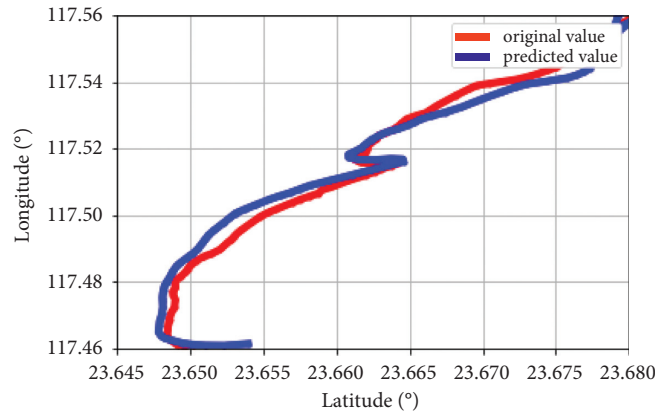


FIGURE 3: Prediction result of MMSI 259681317.

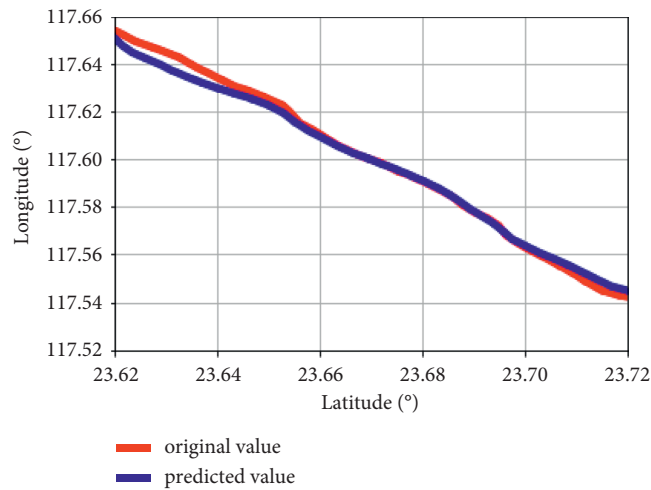


FIGURE 4: Prediction result of MMSI 412444715.

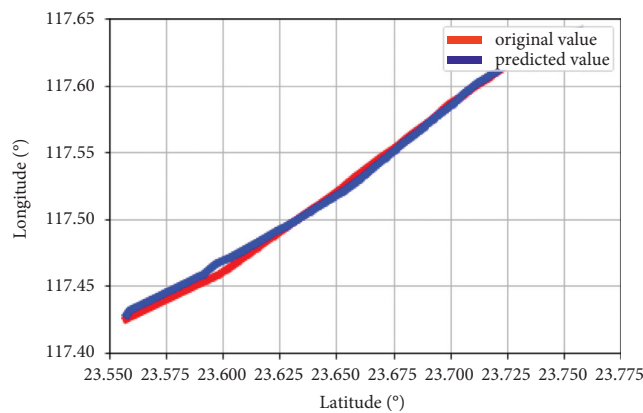


FIGURE 5: Prediction result of MMSI 412358490.

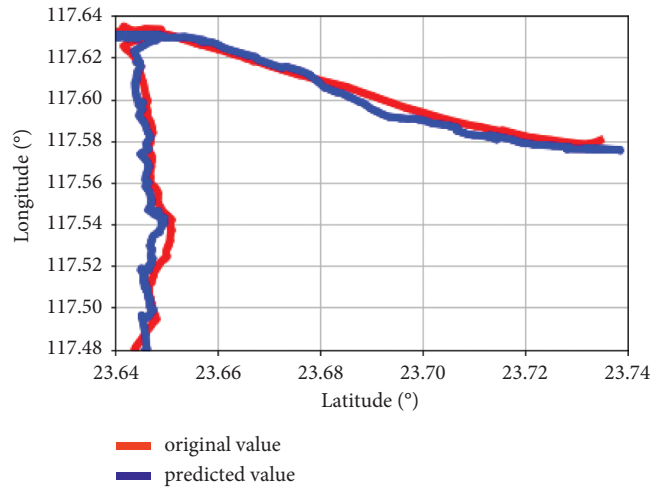


FIGURE 6: Prediction result of MMSI 600003568.

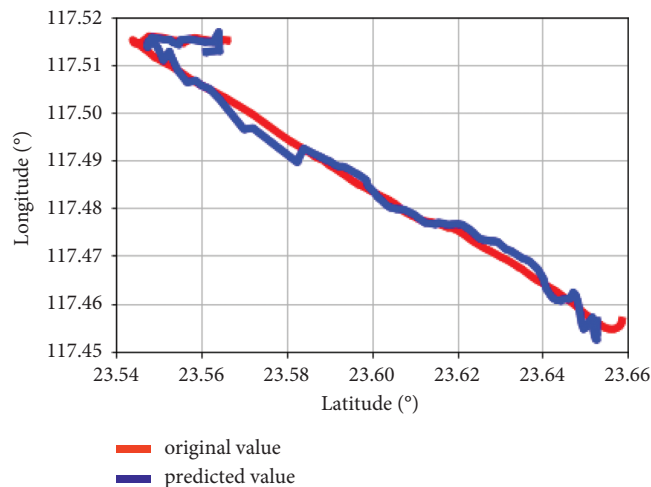


FIGURE 7: Prediction result of MMSI 600003568.

SVR models were compared. In the following figure, num represents the number of track points, and the vertical axis represents the value of the prediction error.

Figure 13 shows the model error comparison of MMSI 259681317.

Figure 14 shows the model error comparison of MMSI 412444715.

Figure 15 shows the model error comparison of MMSI 412358490.

Figure 16 shows the model error comparison of MMSI 412445837.

Figure 17 shows the model error comparison of MMSI 600003568.

Figures 13–17 show that the overall prediction error of DLGWO-SVR is smaller than that of GWO-SVR, and the prediction error of DLGWO-SVR fluctuates less. On the

whole, the performance of the proposed method is better than that of GWO-SVR.

4.5. Error Comparison between the DLGWO-SVR and Other Models.

To further verify the prediction accuracy of the proposed model, it was compared with DEGWO-SVR and GWO-SVR model. As the initial population of GWO is randomly generated, the grey wolf individuals lack diversity in the search space and are prone to fall into the local optimum. DE algorithm maintains population diversity through mutation and selection operations to improve the performance of GWO in local and global search. Figure 18 shows that the prediction error of the proposed prediction model and DEGWO-SVR model. Table 3 shows the mean square error, mean absolute error, and mean absolute

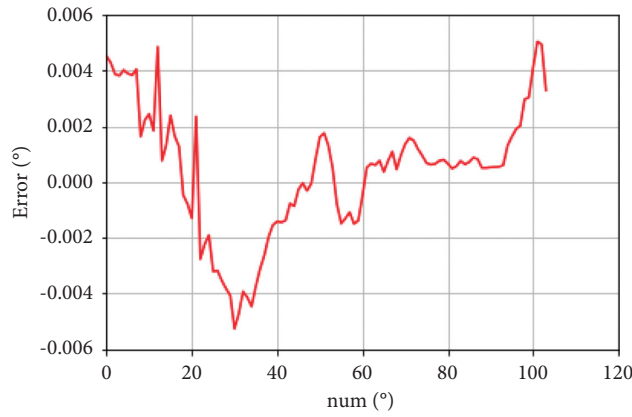


FIGURE 8: The prediction error of MMSI 259681317.

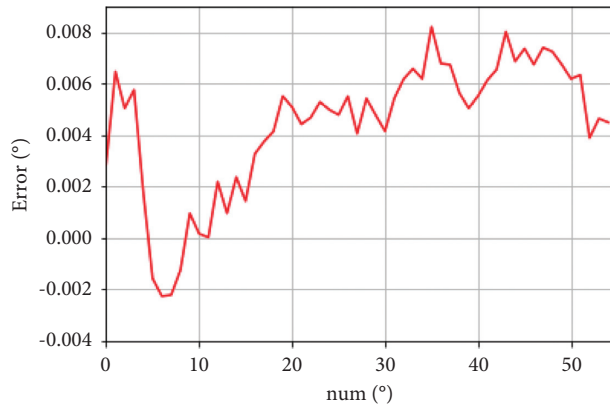


FIGURE 9: The prediction error of MMSI 412444715.

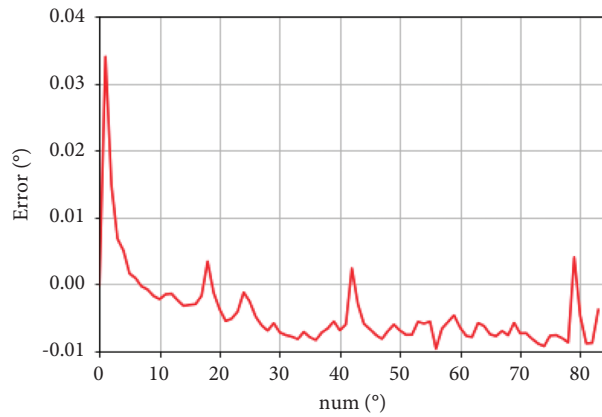


FIGURE 10: The prediction error of MMSI 412358490.

percentage error of the three models. It can be roughly seen that the prediction errors of the DEGWO-SVR model and DLGWO-SVR model are less than GWO-SVR.

Since the proposed model randomly distributes N wolves in the search space during initialization to maintain population diversity and its neighbor learning mechanism

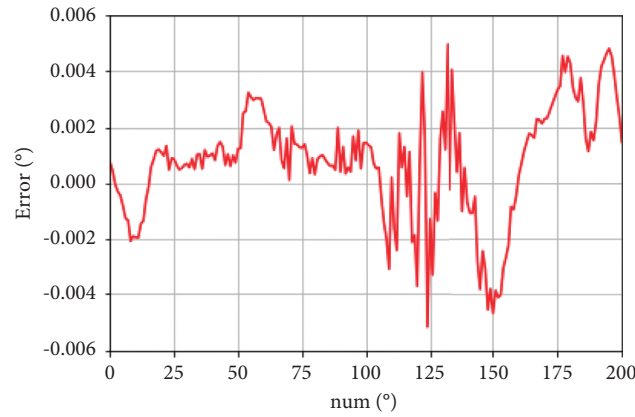


FIGURE 11: The prediction error of MMSI 412445837.

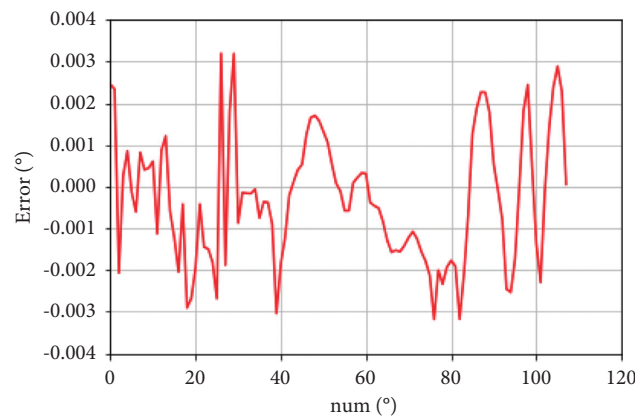


FIGURE 12: The prediction error of MMSI 600003568.

TABLE 2: Evaluation table of DLGWO-SVR prediction error.

MMSI	$E_{MAE} (^\circ)$	$E_{MSE} (^\circ)$	$E_{MAPE} (^\circ)$
259681317	0.001865	0.0000054	0.0000788
412444715	0.005695	0.0002206	0.0002403
412358490	0.006816	0.0000558	0.0000608
412445837	0.001811	0.0000048	0.0000765
600003568	0.001390	0.00000437	0.0000589

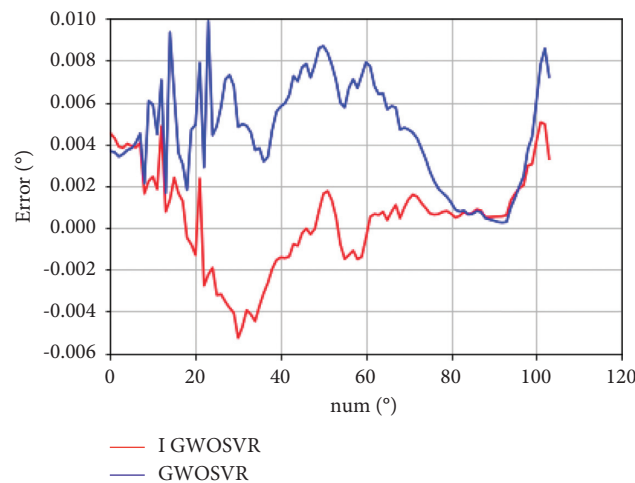


FIGURE 13: The model error comparison of MMSI 259681317.

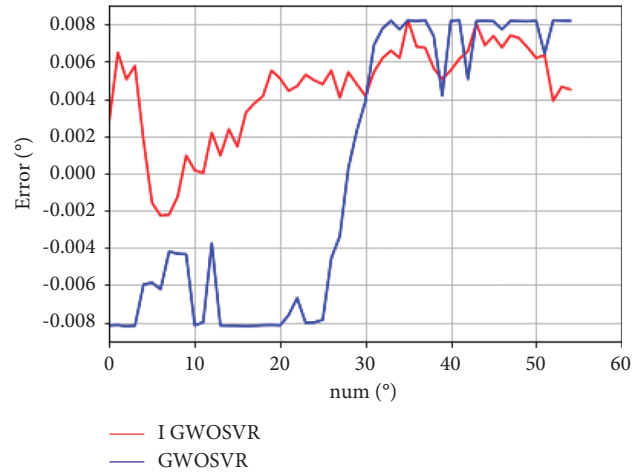


FIGURE 14: The model error comparison of MMSI 412444715.

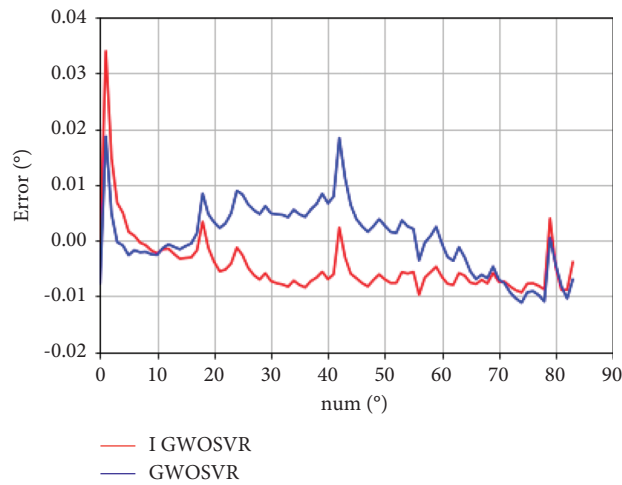


FIGURE 15: The model error comparison of MMSI 412358490.

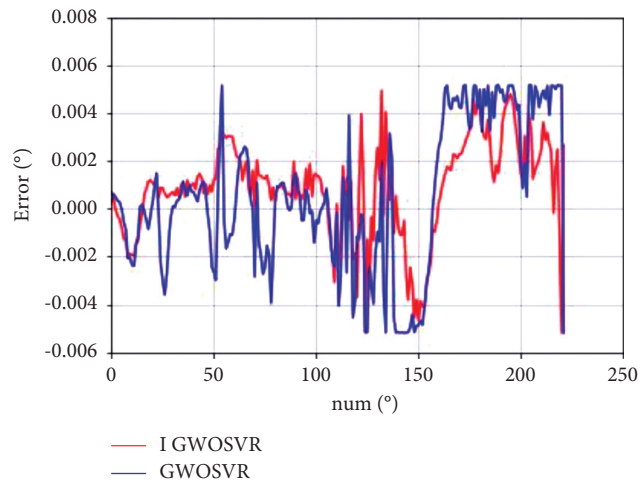


FIGURE 16: The model error comparison of MMSI 412445837.

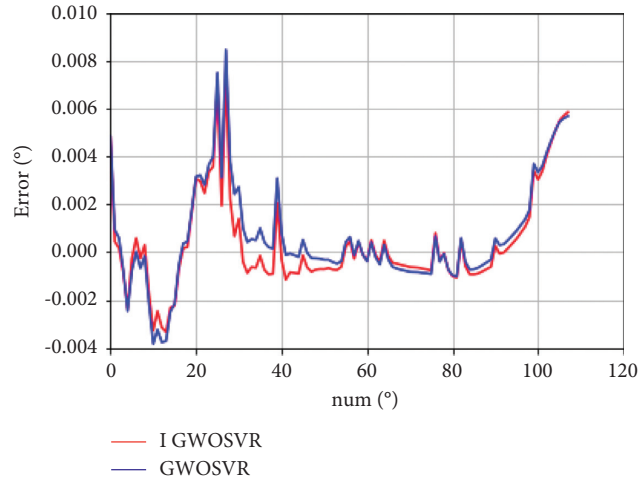


FIGURE 17: The model error comparison of MMSI 600003568.

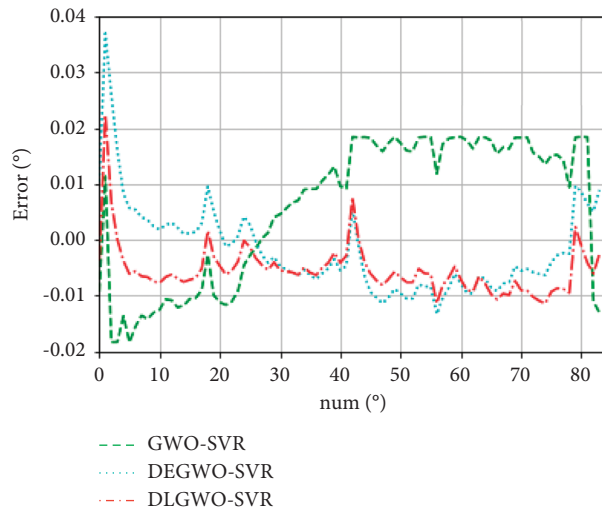


FIGURE 18: Comparison of the prediction errors obtained using the three models.

TABLE 3: Prediction errors of the three models.

Model	$E_{MAE} (^{\circ})$	$E_{MSE} (^{\circ})$	$E_{MAPE} (^{\circ})$
GWO-SVR	0.013199	0.0001983384	0.000112287
DEGWO-SVR	0.0051	0.000069317	0.00021666
DLGWO-SVR	0.0063532	0.00005012	0.00005405

performs better than DE in balancing global and local search, the proposed model performs better than DE and has higher prediction accuracy. As a result, the suggested model in this work meets the prediction requirements and increases the accuracy of the GWO-SVR model greatly.

5. Conclusions

In this paper, we use a DLH strategy to improve GWO for ship track prediction experiments. The improved optimization model can well balance the exploration and

exploitation capabilities of GWO while maintaining the diversity of the population to avoid GWO falling into the local optimal solution, thus improving the accuracy of trajectory prediction.

By conducting prediction experiments on ship trajectories with different MMSI and different lengths, it is not difficult to see that the prediction accuracy of the DLGWO-SVR model meets the requirements of marine ship trajectories prediction and improves the prediction accuracy of ship trajectories. Compared with other methods, DLGWO-SVR can maintain high prediction accuracy in small sample data.

Future research can be combined with the information of surrounding ships and geographical environment to predict the sailing state of ships, and more state information of ships can also be considered for ship track prediction. Furthermore, it is necessary to make a real-time prediction of the trajectory of the target ship sailing in the ocean.

In addition, there are many optimization algorithms not mentioned in this paper, such as Monarch Butterfly Optimization (MBO), earthworm optimization algorithm, Elephant Herding Optimization (EHO), moth search algorithm, slime mould algorithm, and Harris Hawks Optimization (HHO). These optimization algorithms are very effective when dealing with certain problems. MBO and HHO can be used in combination with other machine learning algorithms to deal with issues such as numerical optimization and feature selection. For example, the butterfly optimization algorithm can be combined with ACO and ABC algorithm. EW algorithm can be combined with a GA, PSO, and DE algorithm to deal with specific problems. EHO and SVM sets are better than GA and grid search in classification accuracy. The hybrid optimization algorithm shows better convergence and higher accuracy and lowers computational complexity when solving some suitable problems.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This research was funded by the Natural Science Foundation of Fujian Province (2018J01536, 2018J01484, 2018J01485, and 2020J01660), the Project of Young and Middle-Aged Teacher Education of Fujian Province (JAT170311), Youth Innovation Foundation of Xiamen (3502Z20206019).

References

- [1] M. Smith, S. Reece, S. Roberts, I. Psorakis, and I. Rezek, "Maritime abnormality detection using Gaussian processes," *Knowledge and Information Systems*, vol. 38, no. 3, pp. 717–741, 2014.
- [2] B. Jiang, J. Guan, W. Zhou, and X. Chen, "Vessel trajectory prediction algorithm based on polynomial fitting kalman filtering," *Journal of Signal Processing*, vol. 5, pp. 741–746, 2019.
- [3] Q. J. Zhou, "Prediction of ship navigation safety based on bayesian network," *Ship Science and Technology*, vol. 42, no. 12, pp. 34–36, 2020.
- [4] M. Hanif, F. Sami, M. Hyder, and M. I. Ch, "Hidden Markov model for time series prediction," *Journal of Asian Scientific Research*, vol. 7, no. 5, pp. 196–205, 2017.
- [5] Y. Jiang and Q. Zhang, "Prediction algorithm of dynamic trajectory based on weighted grey model (1, 1)," *Journal of Computer Applications*, vol. 36, no. 5, pp. 1336–1340, 2016.
- [6] L. You, S. Xiao, Q. Peng et al., "ST-Seq2Seq: a s-temporal feature-optimized Seq2Seq model for short-term vessel trajectory prediction," *IEEE Access*, vol. 8, Article ID 218565, 2020.
- [7] R. Zhen, Y. X. Jin, Q. Y. Hu, C. J. Shi, and S. Z. Wang, "Ship navigation behavior prediction based on AIS information and BP neural network," *Navigation of China*, vol. 40, no. 2, pp. 6–10, 2017.
- [8] K. D. Chen, Y. S. Zhu, K. Yan, Y. Q. Cai, Z. J. Ren, and D. W. Gao, "Ship track prediction based on LSTM," *Ship & Ocean Engineering*, vol. 048, no. 6, pp. 121–125, 2019.
- [9] J. Kim, "Vessel target prediction method and dead reckoning position based on SVR seaway model," *International Journal Of Fuzzy Logic And Intelligent Systems*, vol. 17, no. 4, pp. 279–288, 2017.
- [10] G. Chen and Z. Li, "Improved Particle Swarm optimization LSSVM spatial location trajectory data prediction model in health care monitoring system," *Personal and Ubiquitous Computing*, vol. 10, 2019.
- [11] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [12] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee Colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.
- [13] E. Rashedi, H. P. Nezamabadi, and S. Saryazdi, "GSA: a gravitational search algorithm," *Information Sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.
- [14] M. Gen and L. Lin, *Genetic Algorithms*, John Wiley & Sons, NJ, USA, 2008.
- [15] M. Chica, O. Cordon, S. Damas, J. Pereira, and J. Bautista, *Ant Colony Optimization and Swarm Intelligence*, University of Pittsburgh, A, USA, 2006.
- [16] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.
- [17] S. Mirjalili, "How effective is the grey wolf optimizer in training multi-layer perceptrons," *Applied Intelligence*, vol. 43, no. 1, pp. 150–161, 2015.
- [18] T. Jayabarathi, T. Raghunathan, B. R. Adarsh, and P. N. Suganthan, "Economic dispatch using hybrid grey wolf optimizer," *Energy*, vol. 111, pp. 630–641, 2016.
- [19] R. Katarya and O. P. Verma, "Recommender system with grey wolf optimizer and FCM," *Neural Computing & Applications*, vol. 30, no. 5, pp. 1679–1687, 2018.
- [20] Q. Tu, X. Chen, and X. Liu, "Multi-strategy ensemble grey wolf optimizer and its application to feature selection," *Applied Soft Computing*, vol. 76, pp. 16–30, 2019.
- [21] J. Song, J. Wang, and H. Lu, "A novel combined model based on advanced optimization algorithm for short-term wind speed forecasting," *Applied Energy*, vol. 215, pp. 643–658, 2018.
- [22] M. Malik, E. R. Mohideen, and L. Ali, "Weighted distance grey wolf optimizer for global optimization problems," in *Proceedings of the 2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pp. 1–6, Madurai, India, December 2015.

- [23] W. Long, J. J. Jiao, X. M. Liang, S. H. Cai, and M. Xu, "A random opposition-based learning grey wolf optimizer," *IEEE Access*, vol. 99, 2019.
- [24] N. Singh and S. B. Singh, "A novel hybrid GWO-SCA approach for optimization problems," *Engineering Science and Technology, an International Journal*, vol. 20, no. 6, pp. 1586–1601, 2017.
- [25] P. J. Gaidhane and M. J. Nigam, "A hybrid grey wolf optimizer and artificial bee Colony algorithm for enhancing the performance of complex systems," *Journal of Computational Science*, vol. 27, pp. 284–302, 2018.
- [26] A. Saxena, R. Kumar, and S. Mirjalili, "A harmonic estimator design with evolutionary operators equipped grey wolf optimizer," *Expert Systems with Applications*, vol. 145, Article ID 113125, 2020.
- [27] N. Mittal, U. Singh, and B. S. Sohi, "Modified grey wolf optimizer for global engineering optimization," *Applied Computational Intelligence and Soft Computing*, vol. 2016, Article ID 7950348, 16 pages, 2016.
- [28] W. Long, J. Jiao, X. Liang, and M. Tang, "An exploration-enhanced grey wolf optimizer to solve high-dimensional numerical optimization," *Engineering Applications of Artificial Intelligence*, vol. 68, pp. 63–80, 2018.
- [29] M. S. Nasrabadi, Y. Sharafi, and M. Tayari, "A parallel grey wolf optimizer combined with opposition based learning," in *Proceedings of the Swarm Intelligence & Evolutionary Computation; Swarm Intelligence & Evolutionary Computation*, pp. 18–23, Bam, Iran, June 2016.
- [30] A. A. Alomoush, A. Alsewari, H. S. Alamri, K. S. Aloufi, and K. Z. Zamli, "Hybrid Harmony search algorithm with grey wolf optimizer and modified opposition-based learning," *IEEE Access*, vol. 7, 2019.
- [31] A. Meng, C. Zeng, P. Wang et al., "A high-performance Crisscross search based grey wolf optimizer for solving optimal power flow problem," *Energy*, vol. 225, Article ID 120211, 2021.
- [32] S. Zhang, Q. Luo, and Y. Zhou, "Hybrid grey wolf optimizer using elite opposition-based learning strategy and simplex method," *International Journal of Computational Intelligence and Applications*, vol. 16, no. 2, Article ID 1750012, 2017.
- [33] L. Rodriguez, O. Castillo, J. Soria et al., "A fuzzy hierarchical operator in the grey wolf optimizer algorithm," *Applied Soft Computing*, vol. 57, pp. 315–328, 2017.
- [34] C. Lu, L. Gao, and J. Yi, "Grey wolf optimizer with cellular topological structure," *Expert Systems with Applications*, vol. 107, pp. 89–114, 2018.
- [35] C. Lu, L. Gao, Q. Pan, X. Li, and J. Zheng, "A multi-objective cellular grey wolf optimizer for hybrid flowshop scheduling problem considering noise pollution," *Applied Soft Computing*, vol. 75, pp. 728–749, 2019.
- [36] F. Alaa and A. R. Abdel, "Estimating the parameters of software reliability growth models using the grey wolf optimizer algorithm," *International Journal of Advanced Computer Science and Applications*, vol. 7, 2016.
- [37] H. Almazini and K. R. K. Mahamud, "Grey wolf optimization parameter control for feature selection in anomaly detection," *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 2, 2021.
- [38] M. H. N. Shahraki, S. Taghian, and S. Mirjalili, "An improved grey wolf optimizer for solving engineering problems," *Expert Systems with Applications*, vol. 166, Article ID 113917, 2021.