Hindawi
Advances in Multimedia
Volume 2022, Article ID 6169331, 10 pages
https://doi.org/10.1155/2022/6169331

Research Article

Optimization of Aircraft Flight Trajectory Combined with Thinking Navigation Algorithm

Yifan Yang

Chinese Flight Test Establishment, Xi'an, Shaanxi 710089, China

Correspondence should be addressed to Yifan Yang; yyfyy616648@163.com

Received 10 June 2022; Revised 29 June 2022; Accepted 11 July 2022; Published 3 August 2022

Academic Editor: Qiangyi Li

In order to improve the optimization effect of the flight trajectory of the aircraft, this paper combines the thinking navigation algorithm to optimize the flight trajectory of the aircraft and analyzes the flight trajectory of the aircraft through the intelligent model. By processing the original satellite clock error data by the first-order difference method, the modeling data can be more suitable for nonlinear characteristics. Moreover, this paper chooses a simple network structure and uses the MEA to select the optimal initial parameters of the model for the BP neural network, which can avoid the local optimization of the BP neural network results. In addition, this paper conducts experimental analysis on the MEA-BP model through fitting data of different lengths. The simulation test results show that the thinking navigation algorithm proposed in this paper has a very obvious effect on the optimization of the flight trajectory of the aircraft.

1. Introduction

The trajectory optimization of the aircraft refers to the solution of parameters such as speed and altitude during the entire flight process under the condition of given flight performance indicators (the shortest flight time, and the least fuel consumption in the whole process). Under the action of the engine thrust, aerodynamic force, and the gravity of the aircraft, we need to study the law of the movement of the center of mass of the aircraft, that is, to optimize the takeoff and landing performance of the aircraft, as well as the climb, cruise, and descent performance. Therefore, when studying the problem of aircraft flight performance, it is necessary to establish the equation of motion of the center of mass of the aircraft. The establishment of the equation of motion for the center of mass of the aircraft is closely related to the flight environment. In addition to this, flight performance is affected by the interaction between the aircraft and the air, including the aerodynamic characteristics of the aircraft, the total weight of the aircraft, and the relationship of atmospheric and environmental conditions to the aircraft. Therefore, accurate calculation of flight performance must be based on reliable atmospheric data, aerodynamic data, motorized data, and aircraft data.

The essence of civil aircraft flight management system is trajectory optimization and tracking. Modern aircraft trajectory optimization is based on aircraft kinematics equations, dynamic equations, and multiple constraints under the premise of designing control variables according to flight plan requirements, flight technical indicators, etc., and constructing target functionals. Therefore, the optimal control variables are solved to find an optimal flight trajectory that satisfies a certain performance index. The trajectory optimization design is an important part of the overall design of modern aircraft. In the overall design of the aircraft, trajectory optimization and tracking play an important guiding role in other subsystems such as overall layout analysis, overall parameter determination, aerodynamic design, material and structural strength, control system, and power plant system. It is one of the overall goals pursued by the design of civil aviation aircraft to combine with the dynamic parameter design to make all the subsystems of the aircraft cooperate with each other to achieve the optimal performance indicators. Only by considering multiple aspects can the designed aircraft be more practical and valuable. In the three-stage

(climbing, cruising, and descending) trajectory optimization, in the trajectory optimization tasks of each stage, early people focused on local point optimization and segment optimization, and the performance indicators that were concerned were the shortest time, the most fuel-efficient, and the shortest distance climb. In index [1], the methods used mainly include retrieval method, iterative method, dynamic variation method, energy state management method, and singular perturbation method. The advantage of the retrieval method [2] is that it is convenient and concise. The disadvantage is that it only considers the local optimum and requires a large storage space. When the flight task and flight environment change, the accuracy, and accuracy and search efficiency of this method are low. The iterative method in the static optimization method expresses the quantity to be optimized, such as the cost of flight time and fuel cost, as an objective function composed of certain parameters, and then substitutes the objective function into the parameter estimated value to continuously iterate to finally obtain the optimal solution of the objective function. The method is simple and direct, but the disadvantage is that the whole trajectory cannot be optimized. The variational method transforms the trajectory optimization problem into a state equation based on the optimal control theory and uses the minimum value principle or the variational principle to obtain the global optimal solution [3].

At present, the research on aircraft trajectory planning mainly focuses on UAVs and military aircraft, such as terrain avoidance, terrain tracking, and UAV formation flight trajectory planning in low-altitude penetration of military aircraft. The main applied planning algorithms are as follows: grid method, artificial potential field method, A* search algorithm, sparse A* search algorithm, D* search algorithm, genetic algorithm, ant colony algorithm, particle swarm algorithm, etc. Scholars like Amin JN discussed several problems of UAV track planning based on genetic algorithm [4]. The literature [5] applied the navigation algorithm to online track optimization, but this method is complicated and difficult to understand. The literature [6] proposed that the distance propagation method is used, which can effectively solve some dynamic trajectory optimization problems, but with the increase of the number of grids, the optimization time increases significantly. The literature [7] uses the threat cost weighting method to construct a Voronoi diagram and analyzes it according to the characteristics of real-time trajectory planning. Two (strategic and tactical) trajectory planning ideas are proposed. The dual-population particle swarm planning algorithm proposed by the literature [8] uses a parallel search strategy to meet the requirements of real-time planning. The literature [9] uses dynamic path algorithm and terrain tracking. The algorithm replans the helicopter's online track and improves the mission survivability of the aircraft. The literature [10] uses the differential evolution algorithm (DE) to randomly search the space and points out that this algorithm has the advantages of fast search and good robustness. Regarding the study of aircraft avoiding restricted areas such as flight restricted areas and threats, the literature [11] uses genetic algorithm to solve the flight trajectory with constraints, and the literature [12] uses MILP method to transform the trajectory planning problem into a linear programming problem under constraints. In order to obtain the optimal obstacle avoidance trajectory, the literature [13] discusses the trajectory optimization problem of UAV in the presence of flight restricted areas. The goal of trajectory planning is to detect and analyze real-time environmental data in a timely and effective manner when the aircraft reaches the destination, so as to change the preplanned flight path in real time and reduce the probability of the aircraft being threatened by maneuvering. The literature [14] applied the adaptive genetic algorithm and D* search algorithm to the offline and online trajectory planning of the aircraft cruise segment on the basis of referring to relevant theories and algorithms, and achieved good planning results.

With the complexity of the mathematical model of the aircraft, the multicoupling and aerodynamic nonlinearity phenomena in the flight process make this method almost no longer applicable [15]. With the development of science and technology and the study of nonlinear control theory, many nonlinear control methods have appeared. Judging from the current research work on aircraft track tracking, most scholars focus on neural network control, fuzzy control, LQR control, sliding mode control, and adaptive control. In the literature [16], a method for decoupling the strongly coupled transformation of the aircraft is given. When designing the sliding mode control law, the motion model needs to be linearized. The literature [17] designed a sliding mode control law to stably track the three-dimensional flight path of the aircraft. The literature [18] combined the predictive function control command with the fuzzy inference compensation to achieve the optimal track tracking control of the aircraft problem, and the simulation shows the feasibility of this method. The literature [19] proposed an improved robust adaptive control, which reduces the complexity of the control law by introducing a dynamic surface, and uses the RBF neural network approximation method to control the uncertain parameters inside the model, and the overall realization of the pitch angle, attack angle, and level tail is achieved. *Deflected instruction tracking*. The literature [20] designed a robust adaptive reverse thrust sliding mode trajectory tracking method, which realized trajectory tracking with state variables and angle of attack as commands, and realized observation compensation for atmospheric disturbance and modeling error by designing an observer. And we introduce the tangent function and attenuation factor to reduce the signal chattering problem.

This paper combines the thinking navigation algorithm to optimize the flight trajectory of the aircraft, analyzes the flight trajectory of the aircraft through the intelligent model, and combines the intelligent algorithm to improve the planning effect of the flight trajectory of the aircraft.

2. Thinking Navigation Algorithm

2.1. The Basic Theory of Satellite Clock Error Forecasting. Satellite clock error data, also known as phase data, are the time offset data obtained by standard time and atomic clock output time, in nanoseconds or seconds. In satellite

navigation and positioning, phase data represent satellite clock error data and receiver clock error data. The raw phase data can be directly used for navigation positioning and satellite clock error prediction. However, the magnitude of the phase data is large, and it is impossible to judge and identify the location of outliers such as gross errors and clock jumps. Therefore, the data need to be preprocessed before the actual clock error forecasting. By making corresponding mathematical changes to the phase data, the corresponding frequency data can be obtained, and by processing the frequency data, the gross errors and clock jumps existing in the clock error can be effectively detected.

The relative deviation of the standard output frequencies of different frequencies can obtain frequency data, which is defined as:

$$D = \frac{f_A - f_B}{f_0}. \tag{1}$$

In the formula, f_A is the output frequency of the frequency standard A, f_B is the output frequency of the frequency standard B, f_0 is the nominal frequency, and D is the obtained frequency data.

Over time, the actual output frequency of the atomic clock will be affected by the internal components and the external environment, and it usually presents a linear trend of increasing or decreasing. When time changes, the frequency of satellite clock also changes, and the rate of change is called frequency drift rate or aging rate. The calculation formula is as follows:

$$z_i = \frac{y_{i+1} - y_i}{\tau_0}. \tag{2}$$

In the formula, y_i is the frequency value of epoch i, y_{i+1} is the frequency value of epoch $i+1$, and τ_0 is the sampling interval of adjacent epochs.

The frequency of atomic clocks is often not a specific value, but fluctuates within a certain range. It obtains correct information on whether the atomic clock frequency is accurate or not by comparing the actual frequency obtained by measuring the 1BP neural network algorithm or calculation with the nominal frequency. The calculation formula is as follows:

$$A = \frac{f - f_0}{f_0}. \tag{3}$$

In the formula, f is the actual frequency measured or calculated, f_0 is the real frequency, that is, the nominal frequency, and A is the calculated frequency accuracy. Since the frequency of atomic clocks fluctuates within a certain range, the average of multiple measurement results is usually taken as the final result.

2.2. Preprocessing of Satellite Clock Error Data. The frequency data are obtained from the phase data by mathematical transformation, and the conversion formula is as follows:

$$y_i = \frac{x_{i+1} - x_i}{\tau_0}. \tag{4}$$

In the formula, x_i is the value of the satellite clock error at epoch i, x_{i+1} is the value of the satellite clock error at epoch $i+1$, τ_0 is the sampling interval of the clock error data in adjacent epochs, and y_i is the obtained clock difference frequency data.

After the detection of gross errors and clock jumps is completed for the frequency data, the processed frequency data can be recovered by the following formula to obtain new phase data, that is, the clock error data.

$$x_i = \int_0^t y(\lambda)\mathrm{d}\lambda. \tag{5}$$

In order to check the characteristics of phase data and frequency data, in this paper, IGS postevent satellite clock error is used for experimental analysis, and one day's GPS satellite clock error data are randomly selected with a sampling rate of 30 s. Taking the PRN10 satellite as an example, its satellite clock error data and corresponding frequency data are given, as shown in Figure 1 and Figure 2.

It can be seen from Figures 1 and 2 that the clock error data (phase data) of the No. 10 satellite are in a linear trend as a whole, and its order of magnitude is 10–4. It is difficult to see with the naked eye whether there are outliers such as gross errors in the data. However, the frequency data obtained through mathematical changes are far less than the order of magnitude of the satellite clock error. Since the gross error value is small, it is easier to find outliers such as gross error using frequency data.

Most gross error detection methods are processed based on satellite frequency data. Currently, the robust estimation method based on median (MAD) is used more, and the median representation is as follows:

$$\mathrm{MAD} = \mathrm{Median}\left\{\frac{|y_i - m|}{0.6745}\right\}. \tag{6}$$

In the formula, y_i is the satellite frequency data, and $m = \mathrm{Median}\{y_i\}$ is the middle number of the frequency data sequence.

After the algorithm obtains the median (MAD), the middle number m of the frequency data is added to the n times of the obtained median, and the sum of the above two items is compared with the magnitude of the absolute value $|y_i|$ of each frequency. According to its relationship to judge gross error, the formula is as follows:

$$|y_i| > (m + n \cdot \mathrm{MAD}). \tag{7}$$

Among them, the integer n is selected according to the actual situation, and the other parameters are the same as those in formula (6).

When the satellite frequency data satisfy the above formula, the data are judged as gross error and marked. After the gross error detection is carried out on all the frequency data, according to the number of marked epochs, the frequency data of the corresponding epoch are set to 0, and the frequency data are filled by means of mathematical calculation. The method is simple and easy to implement, and has weak sensitivity to gross error size, strong error resistance,

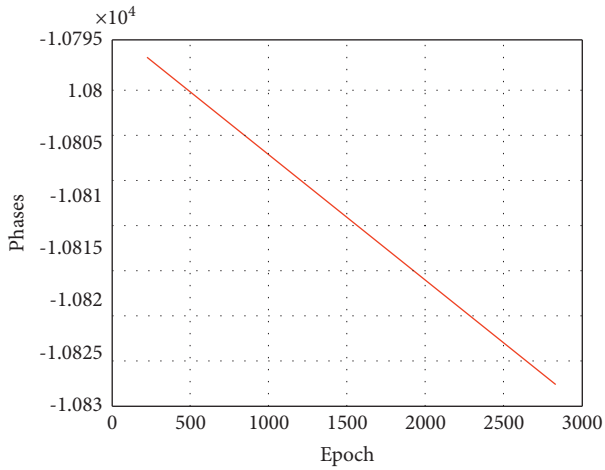

Figure 1: PRN10 satellite clock error data.

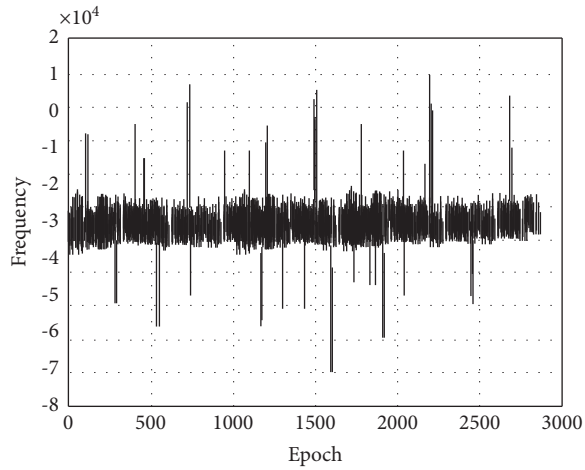

Figure 2: PRN10 satellite frequency data.

and good applicability to frequency data. Using this method, the satellite frequency sequence in Figure 2 is processed for gross error detection, and the processed satellite frequency data are shown in Figure 3.

By comparing Figures 2 and 3, it can be seen that the frequency data quality after using the median method to remove gross errors has been significantly improved, and the overall fluctuation is relatively stable. The clock error data with better quality are obtained by restoring the processed satellite frequency data. Using these data for modeling prediction will greatly improve the accuracy of the clock error prediction value.

In order to judge whether there is a clock jump in the clock difference data, this paper adopts the following methods to detect: the algorithm first processes the satellite frequency data using the median method mentioned above. After the processing is completed, the epochs marked as gross errors are stored, and sometimes adjacent epochs are marked at the same time. There is a high probability of a clock jump in this case. The algorithm sums the adjacent two items of the satellite frequency data to form a new data sequence $\overline{y_i} = y_i + y_{i+1}$ and marks the epoch according to

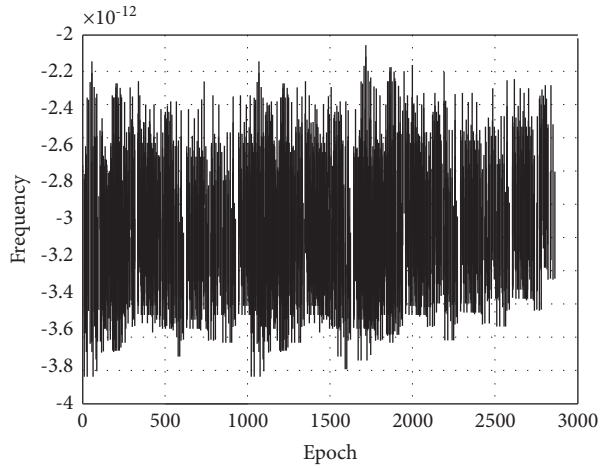

Figure 3: Frequency data of PRN10 satellite after removing gross errors.

the above stored gross errors. If $\overline{y_i}$ only has a peak value in the marked epoch, the epoch is considered to be a gross error, and if both the marked epoch and its next epoch have peak values, it can be judged that the abnormal value is a clock jump. The specific judgment is shown in Figure 4.

2.3. Commonly Used Clock Error Prediction Models. The quadratic polynomial model of satellite clock error forecast is as follows:

$$x_i = a_0 + a_1(t_i - t_0) + a_2(t_i - t_0)^2 + \varepsilon. \tag{8}$$

In the formula, $i = 1, 2, \ldots, n$; x_i is the value of the satellite clock offset at time t_i; t_0 is the satellite clock reference time; a_0, a_1, a_2 are the satellite phase, frequency offset, and frequency drift at the reference time; and ε is the random error of the model.

The linearization formula of this model is as follows:

$$L = Ba + v. \tag{9}$$

In the formula, there are

$$B = \begin{bmatrix} 1 & t_1 - t_0 & (t_1 - t_0)^2 \\ 1 & t_2 - t_0 & (t_2 - t_0)^2 \\ \vdots & \vdots & \vdots \\ 1 & t_i - t_0 & (t_i - t_0)^2 \end{bmatrix},$$

$$L = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \end{bmatrix}, \tag{10}$$

$$v = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_i \end{bmatrix}.$$

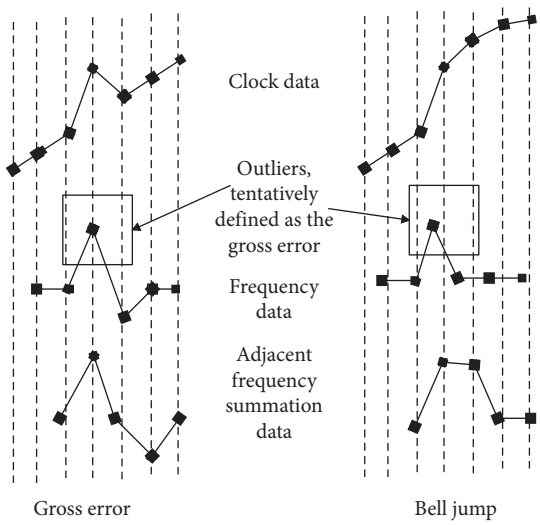

Figure 4: Schematic diagram of the difference between gross errors and clock jumps in satellite clock errors.

When the number of satellite clock errors exceeds three, based on the principle of least squares, the estimation of the parameters in the fitting function can be obtained as follows:

$$\hat{a} = \begin{bmatrix} \hat{a}_0 & \hat{a}_1 & \hat{a}_2 \end{bmatrix}^T = \left(B^T B\right)^{-1} B^T L. \tag{11}$$

The fitting of the clock error sequence can be completed by substituting the parameter estimates into formula (8). After the model is determined, the time extrapolation can be used to complete the prediction of the satellite clock error at the next moment

$$\begin{bmatrix} \hat{x}_{i+1} \\ \hat{x}_{i+2} \\ \vdots \\ \hat{x}_{i+n} \end{bmatrix} = \begin{bmatrix} 1 & t_{i+1} - t_0 & \left(t_{i+1} - t_0\right)^2 \\ 1 & t_{i+2} - t_0 & \left(t_{i+2} - t_0\right)^2 \\ \vdots & \vdots & \vdots \\ 1 & t_{i+n} - t_0 & \left(t_{i+n} - t_0\right)^2 \end{bmatrix} \begin{bmatrix} \hat{a}_0 \\ \hat{a}_1 \\ \hat{a}_2 \end{bmatrix}. \tag{12}$$

In the formula, n is the number of forecast clock errors.

In the actual satellite clock error prediction, the appropriate polynomial order is generally selected according to the actual satellite clock characteristics. When the frequency drift of the satellite clock cannot be ignored, the quadratic polynomial fitting model is often used in the actual clock error prediction; otherwise, the linear model is used.

The clock error prediction model with additional periodic term is as follows:

$$\begin{aligned} x_i &= a_0 + a_1\left(t_i - t_0\right) + a_2\left(t_i - t_0\right)^2 \\ &\quad + \sum_{k=1}^{p} A_k \sin\left[2\pi f_k\left(t_i - t_0\right) + \varphi_k\right] + \varepsilon. \end{aligned} \tag{13}$$

In the formula, the first half is the quadratic polynomial model, and the latter is the periodic term. Among them, A_k, f_k, $an\ d\ \varphi_k$ are related to the period term, which are amplitude, frequency, and phase, respectively; p is the number of main periods; and the rest parameters are the same as formula (8).

After the parameters p and f_k are determined, the unknown parameters in the model are solved by using the clock difference data after substituting into formula (13). For the convenience of calculation, we set as follows:

$$\begin{aligned} b_k &= A_k \cos\varphi_k, \\ c_k &= A_k \sin\varphi_k. \end{aligned} \tag{14}$$

Then, the linearized form of formula (13) is as follows:

$$\begin{aligned} x_i &= a_0 + a_1\left(t_i - t_0\right) + a_2\left(t_i - t_0\right)^2 \\ &\quad + \sum_{k=1}^{p} \{b_k \sin\left[2\pi f_k\left(t_i - t_0\right)\right] \\ &\quad + c_k \cos\left[2\pi f_k\left(t_i - t_0\right)\right]\} + \varepsilon. \end{aligned} \tag{15}$$

Its matrix form is as follows:

$$L = Aa + v. \tag{16}$$

In the formula, L is the n-dimensional satellite clock error data, $a = \begin{bmatrix} a_0 & a_1 & a_2 & b_1 & c_1 & \cdots & \cdots & b_p & c_p \end{bmatrix}^T$ is the parameter to be determined, A is the coefficient matrix, and the dimension is $n \times (2p + 3)$. The specific expression is as follows:

$$A = \begin{bmatrix} A_1 & A_2 \end{bmatrix}, \tag{17}$$

Among them, there are

$$\begin{aligned} A_1 &= \begin{bmatrix} 1 & \delta t_1 & \delta t_1^2 \\ 1 & \delta t_2 & \delta t_2^2 \\ \vdots & \vdots & \vdots \\ 1 & \delta t_n & \delta t_n^2 \end{bmatrix}, \\ A_2 &= \begin{bmatrix} \sin\left(2\pi f_1 \delta t_1\right) & \cos\left(2\pi f_1 \delta t_1\right) & \cdots & \sin\left(2\pi f_p \delta t_1\right) & \cos\left(2\pi f_p \delta t_1\right) \\ \sin\left(2\pi f_1 \delta t_2\right) & \cos\left(2\pi f_1 \delta t_2\right) & \cdots & \sin\left(2\pi f_p \delta t_2\right) & \cos\left(2\pi f_p \delta t_2\right) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \sin\left(2\pi f_1 \delta t_n\right) & \cos\left(2\pi f_1 \delta t_n\right) & \cdots & \sin\left(2\pi f_p \delta t_n\right) & \cos\left(2\pi f_p \delta t_n\right) \end{bmatrix}. \end{aligned} \tag{18}$$

In the formula, $\delta t_n = t_n - t_0$.

The same polynomial model is solved, and the least squares principle is used as the criterion to obtain the estimated value of the unknown parameter $\hat{a}$:

$$\hat{a} = \begin{bmatrix} \hat{a}_0 & \hat{a}_1 & \hat{a}_2 & \hat{b}_1 & \hat{c}_1 & \cdots & \cdots & \hat{b}_p & \hat{c}_p \end{bmatrix}^T = \left(A^T A\right)^{-1} A^T L. \tag{19}$$

After obtaining the parameter estimates in the model, we substitute them into formula (13) to complete the model fitting and then perform the clock error prediction at the next moment as follows:

$$L' = A'\hat{a}. \tag{20}$$

In the formula, $A' = [\, A_1' \;\; A_2' \,]$. Among them, there are

$$\begin{aligned} A_1' &= \left[\, 1 \;\; \delta t_{n+i} \;\; \delta t_{n+i}^2 \,\right], \\ A_2' &= \lfloor \sin\left(2\pi f_1 \delta t_{n+i}\right) \cos\left(2\pi f_1 \delta t_{n+i}\right) \\ &\quad \ldots \;\; \sin\left(2\pi f_p \delta t_{n+i}\right) \cos\left(2\pi f_p \delta t_{n+i}\right) \rfloor. \end{aligned} \tag{21}$$

The basic contents of the grey model suitable for satellite clock error forecasting are as follows.

We assume that there is a sequence of original clock offsets, as follows:

$$X^0 = \left\{x^0(1), x^0(2), \ldots, x^0(n)\right\}. \tag{22}$$

The new clock error sequence obtained by accumulation is as follows:

$$X^1 = \left\{x^1(1), x^1(2), \ldots, x^1(n)\right\}, \tag{23}$$

Among them, $x^1(k) = \sum_{i=1}^{k} x^0(i), k = 1, 2, \ldots, n$.

The differential formula of the grey model is as follows:

$$\frac{\mathrm{d}x}{\mathrm{d}t} + bx = u. \tag{24}$$

Substitute the new data sequence generated by accumulating satellite clock error data into the above formula, we get

$$\frac{\mathrm{d}X^1}{\mathrm{d}t} + bX^1 = u. \tag{25}$$

In the formula, $t \in [\, 0, \;\; +\infty \,]$.

Integrating formula (25) in the interval [k, k+1], we have

$$X^1(k+1) - X^1(k) + b\int_k^{k+1} X^1(t)\mathrm{d}t = u. \tag{26}$$

In the formula, $k = 1, 2, \ldots, n-1$.

Taking into account $X^0(k+1) = X^1(k+1) - X^1(k)$, and expressing the integral in the form of mean, the above formula can be written as follows:

$$X^0(k+1) = -bZ^1(k+1) + u. \tag{27}$$

In the formula, $Z^1(k+1) = (X^1(k) + X^1(k+1)/2)$ is the arithmetic mean of points $X^1(k)$、$X^1(k+1)$.

The matrix form of formula (27) is as follows:

$$L = Cb + v. \tag{28}$$

In the formula, there are

$$\begin{aligned} L &= \begin{bmatrix} X^0(2) \\ X^0(3) \\ \vdots \\ X^0(n) \end{bmatrix}, \\ C &= \begin{bmatrix} -Z^{1)}(2) & 1 \\ -Z^{1)}(3) & 1 \\ \vdots & \vdots \\ -Z^{1)}(n) & 1 \end{bmatrix}, \\ b &= \begin{bmatrix} a \\ u \end{bmatrix}. \end{aligned} \tag{29}$$

Based on the principle of least squares, the parameter $\hat{b} = \begin{bmatrix} \hat{a} \\ \hat{u} \end{bmatrix} = (C^T C)^{-1} C^T L$ of the grey model is obtained.

Substituting the obtained parameters into the differential formula, taking into account the initial parameter $X^1(1) = X^0(1)$, we get

$$X^1(k+1) = \left[X^0(1) - \frac{u}{a}\right]e^{-ak} + \frac{u}{a}. \tag{30}$$

The above formula can be used for prediction, and the obtained prediction value is the new data after one accumulation, and the accumulation and subtraction operation is performed, and then, the prediction value of the original clock error data before accumulation is as follows:

$$\begin{aligned} X^0(k+1) &= X^1(k+1) - X^1(k) \\ &= \left(1 - e^a\right)\left[X^0(1) - \frac{u}{a}\right]e^{-ak}. \end{aligned} \tag{31}$$

Formula (31) is the grey model suitable for satellite clock error forecasting. The model can perform modeling prediction with more than 4 data volumes, and its effect is better in the long-term prediction of clock errors.

The autoregressive moving average model ARMA (p, q) is expressed in the following form:

$$\begin{aligned} x_t &= \varphi_1 x_{t-1} + \varphi_2 x_{t-2} + \cdots + \varphi_p x_{t-p} \\ &\quad + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \cdots - \theta_q a_{t-q}. \end{aligned} \tag{32}$$

In the formula, $\{x_t\}$ is the data sequence calculated by the substitution model; φ_i and θ_i are the unknown autoregressive (AR) coefficient and moving average (MA) coefficient, respectively; p and q refer to the order of the autoregressive model and the order of the moving average model, respectively; and $\{a_t\}$ is white noise, which conforms to the normal distribution $N\left(0, \;\; \sigma^2\right)$.

2.4. BP Neural Network Algorithm. Neural networks are effective in fitting and predicting nonlinear time series, in which BP neural network is the most widely used. Through iteration, the output value is gradually approached to the expected value of the output.

The BP neural network structure can set up multiple hidden layers. However, too many hidden layers will increase the computational workload. In order to shorten the calculation amount and speed up the calculation processing, this paper sets the hidden layer as one layer. During the training process, the input layer of each layer node is calculated as follows:

$$\mathrm{In}_{n,j} = \sum_{j=1}^{n} \omega_{ij} x_i + \theta_j. \tag{33}$$

In the formula, n is the number of network layers, ω_{ij} is the weight of the n-th layer of the network, θ_j is the threshold of the n-th layer, x_i is the input data, which corresponds to the input layer in Figure 1, and $\mathrm{In}_{n,j} = \sum_{j=1}^{n} \omega_{ij} x_i + \theta_j$ in the formula (33) is the intermediate parameter of the model.

The activation function often uses the sigmoid function

$$f(x) = \frac{1}{1 + e^{-x}}. \tag{34}$$

The function is continuously differentiable, and it can project data from $[-\infty, +\infty]$ into the interval [0, 1].

After the input layer is calculated, the corresponding output layer is calculated as follows:

$$\mathrm{Out}_{n,j} = f\left(\mathrm{In}_{n,j}\right) = \frac{1}{1 + e^{-\left(\sum_{j=1}^{n} \omega_{ij} x_i + \theta_j\right)}}. \tag{35}$$

In the formula, $\mathrm{Out}_{n,j}$ is the output value obtained by the BP neural network.

According to the gradient descent algorithm, the estimated value $\mathrm{Out}_{n,j}$ of the network output is compared with the real value of the product released by IGS, and the error value is obtained. Through the error back propagation, the BP neural network completes the update of the weight and threshold parameters, and the update formula is as follows:

$$\begin{aligned} \omega(t+1) &= \omega(t) + \eta\varepsilon(t) y(t), \\ \theta(t+1) &= \theta(t) + \eta\varepsilon(t) y(t). \end{aligned} \tag{36}$$

In the formula, t is the number of iterations; $\varepsilon(t)$ is the error term, that is, the difference between the expected output value and the actual output result; η is the learning efficiency of the neural network; and y(t) is the output value of the neuron.

2.5. Thinking Evolution Algorithm. Mind evolutionary algorithm (MEA) deals with optimization problems in the way of evolution of human thinking. The algorithm first uses the convergence operation to optimize the individuals in the subgroup, and then, the mature subgroups compete globally through the alienation operation, which greatly improves the optimization efficiency. Figure 5 is a schematic diagram of the structure of the MEA.

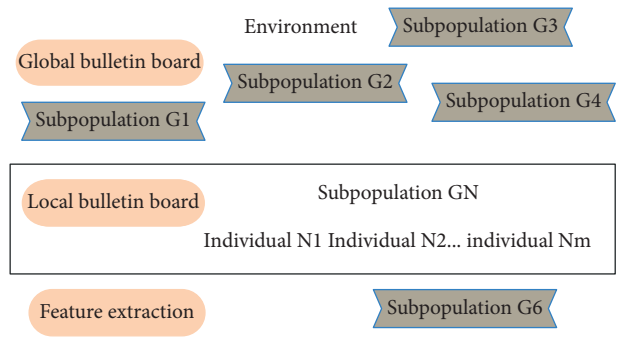

Figure 5: :Schematic diagram of the structure of the MEA.

We assume that there is a set of satellite clock offset data $\{x_1, x_2, \ldots, x_n\}$, and use the data at these n times to model and predict the clock offset value at future times. Usually, the number of output layers is the same as the number of output data types, and the output value in this paper is the clock difference. Therefore, the output node is set to 1, and the neural network is used to establish the mapping relationship between $\{x_1, x_2, \ldots, x_N\}$ and $\{x_{n+1}\}$ (N is the number of input nodes), and the idea of sliding window is adopted. On the premise of keeping the number of samples unchanged, the new prediction data are constantly used to replace the previous known data, and the multi-epoch satellite clock error prediction is realized.

According to formula (37) and formula (38), the algorithm first calculates the mean square error E and then takes the reciprocal of it to obtain the score function score of various groups and individuals.

$$E = \sqrt{\frac{1}{m} \sum_{i=1}^{m} (\hat{y}_i - y_i)^2}, \tag{37}$$

$$\mathrm{score} = \frac{1}{E}. \tag{38}$$

In the formula, $\hat{y}_i$ is the output value after iteration, y_i is the sample value, and m is the total number of samples.

The algorithm flow is shown in Figure 6.

After completing the neural network training using n-dimensional modeling data, according to the neural network structure, the number of input nodes is N, and the number of output nodes is 1. The algorithm adopts the idea of sliding window to ensure that the number of samples remains unchanged and so on to achieve multistep forecasting.

We assume that $X = \{x(1), x(2), \ldots, x(n)\}$ is a set of n-dimensional satellite clock error sequences, where $x(i), i = 1, 2, \ldots, n$ is the clock error data of different epochs. The algorithm divides the clock difference data into two groups and makes the difference in turn to obtain a new first-order difference data sequence

$$\Delta X = \{\Delta x(1), \Delta x(2), \ldots, \Delta x(n-1)\}. \tag{39}$$

Among them, $\Delta x(i) = x(i+1) - x(i)$.

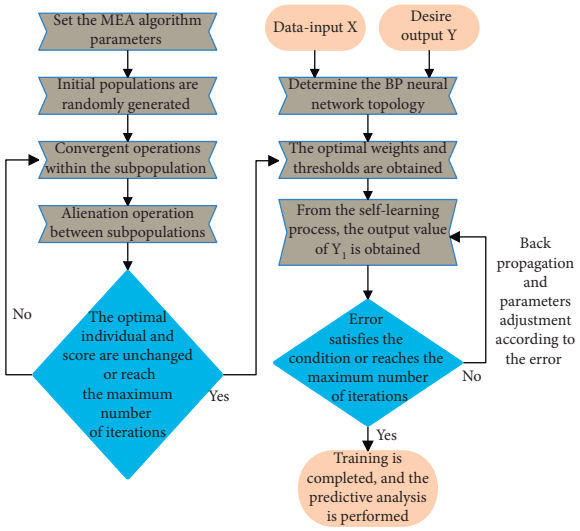

Figure 6: MEA to optimize BP neural network process.

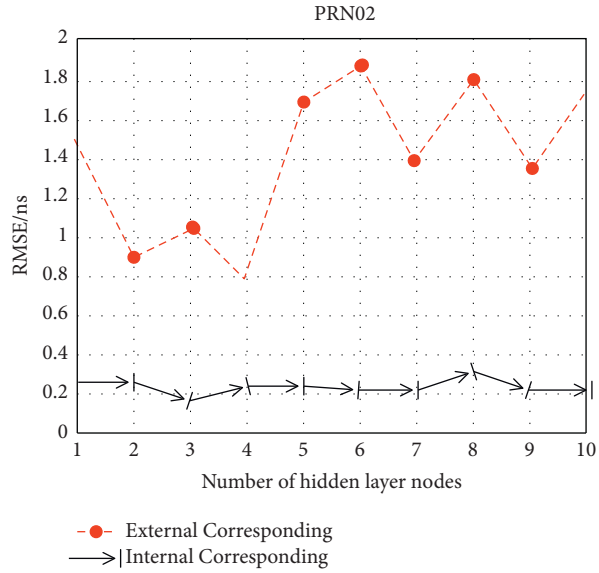

Figure 7: The relationship between the number of hidden layer nodes and the model accuracy.

After the original clock error is processed by one time difference, different from the conventional operation, the data used to solve the parameters in the modeling process are the data sequence after one time difference. When determining the model, a neural network is used to model the formed data sequence ΔX, and the satellite clock error of the next m epochs is predicted: the algorithm models the n-1-dimensional data, and after the modeling process is completed, the m-dimensional data sequence is predicted, $\Delta X' = \{\Delta x(n), \Delta x(n+1), \ldots, \Delta x(m+n-1)\}$. The algorithm recovers the clock error of the epoch to be determined by using the predicted first-order difference and the modeling last clock error data x(n), that is,

$$x(k) = x(n) + \sum_{n}^{k-1} \Delta x(k). \tag{40}$$

3. Comparative Test Analysis

In order to verify the optimization effect of the thinking navigation algorithm on the flight trajectory of the aircraft, the effect of the method in this paper is studied by means of comparative experiments.

The modeling uses 12 hours of data, the satellite clock error data sampling interval is 30 s, and there are 1440 data in total. In this paper, the input layer is set to 1, and the number of hidden layer nodes is in the range of [1 ,10]. This paper takes the G02 satellite as an example and uses its clock error data to conduct experiments. Different mean square errors (RMSE) can be obtained by selecting different hidden layer nodes. The relationship between the two is shown in Figure 7.

It can be seen from the analysis in Figure 7 that the overall fitting accuracy of the model is relatively stable, fluctuating up and down at 0.3 ns. When the number of hidden layer nodes is 5, the prediction accuracy suddenly deteriorates, and the curve fluctuates greatly. The reason is that when there are too many hidden layer nodes, the BP neural network algorithm is prone to "overfitting." That is, although the model has a high fitting accuracy at this time, the prediction accuracy suddenly deteriorates. If the number of hidden layer nodes is not selected properly, it will seriously affect the prediction accuracy of the model. According to the experimental results and fully considering the fitting and prediction accuracy, this paper selects 3 as the number of nodes in the hidden layer of the BP neural network.

In summary, in this paper, the input layer node is set to 1, the output layer node is set to 1, and the hidden layer node is calculated to be 3.

The random selection of initial parameters in BP neural network will increase the convergence time and workload in the training process, which may cause abnormal results in the model solving process and affect the stability of the model. Moreover, the results of the algorithm are easy to fall into the local minimum, which reduces the performance of the model to obtain the global optimum. In this paper, the MEA is used to select the initial parameters required by the BP neural network. In the MEA, the population size is set to 1440, and 5 winning subpopulations and 5 temporary subpopulations are selected. In order to make the algorithm fully find the optimal individual, the number of iterations is set to 200.

The prediction performance of the BP model and the MEA-BP model for satellite clock errors is compared. In order to fully compare the forecasting effects, the clock offset data of the first 12 hours are used to independently predict the clock offset values of the next 3 hours, 6 hours, and 12 hours for 10 times. Figure 8 shows the changes of the RMS values predicted by the two models.

In this experiment, the MEA-BP model and the traditional BP model are used to forecast the No. 2 satellite in different time periods. It can be seen from Figure 8 that under the same network structure, the multiple prediction accuracy of the MEA-BP model and the BP model does not fluctuate significantly, which proves the feasibility of the network structure. When the forecast time increases, the MEA-BP model obtains higher accuracy and has better

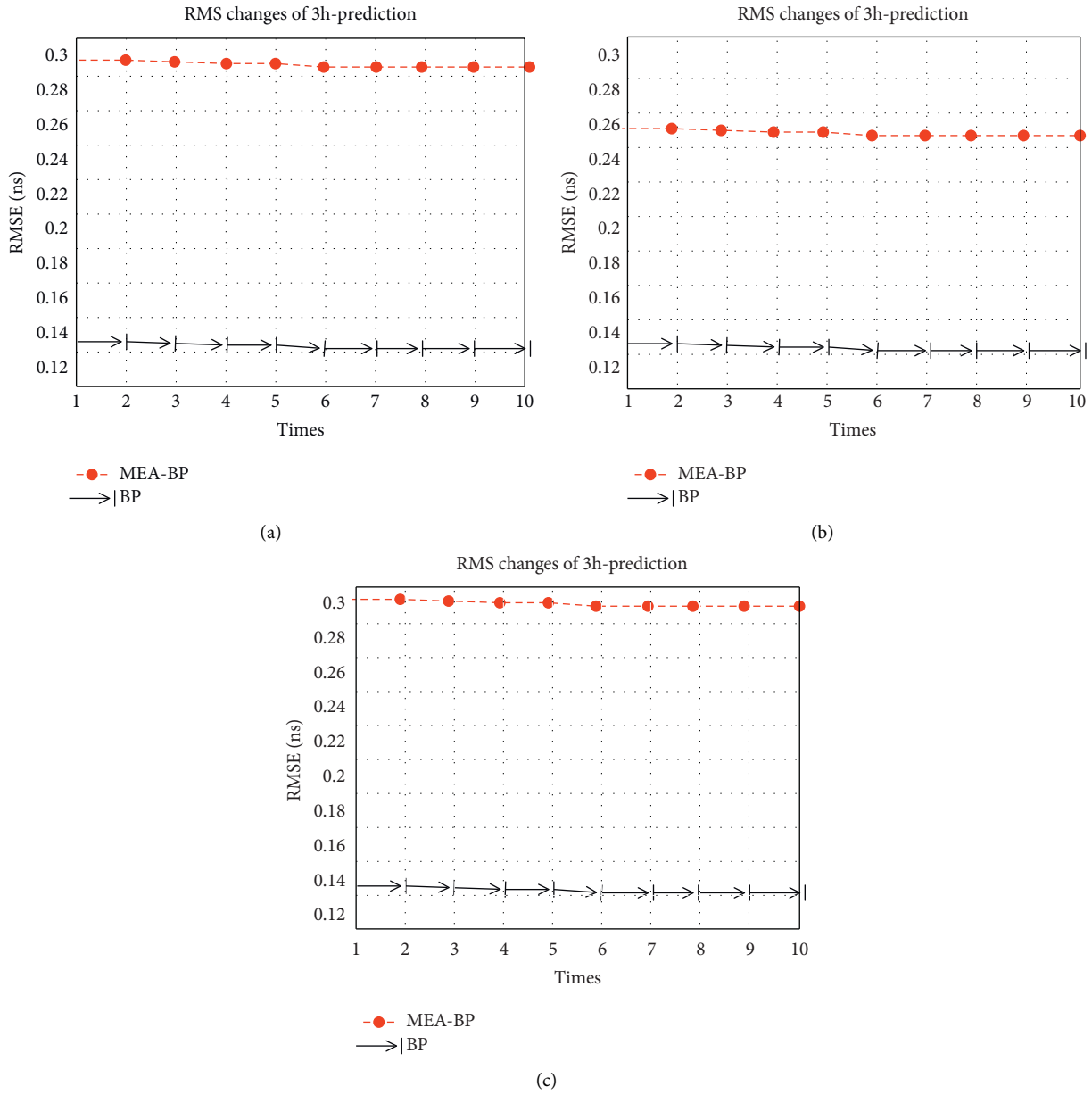

Figure 8: Variation of 10 times RMS of MEA-BP and BP forecast (a) (b) (c).

forecast performance than the traditional BP model. In the clock error prediction of the three time periods, the prediction accuracy of the MEA-BP model is better than that of the BP model. It shows that the initial parameters (weights and thresholds) required by the BP neural network are better selected by MEA, so as to avoid the local optimal results obtained by the BP model, and effectively improve the accuracy of the prediction results. Moreover, it shows the feasibility of the MEA-BP model for clock error prediction, so it is a relatively stable aircraft flight trajectory optimization algorithm.

On the basis of the above research, the role of the thinking navigation proposed in this paper in the optimization of the flight trajectory of the aircraft is analyzed through multiple sets of simulation experiments, and the simulation evaluation results shown in Table 1 are obtained.

Table 1: Evaluation of aircraft flight trajectory optimization.

Num	Trajectory optimization	Num	Trajectory optimization
1	88.48	10	91.48
2	88.49	11	90.77
3	88.05	12	91.57
4	89.39	13	89.74
5	90.29	14	90.46
6	91.14	15	91.02
7	89.40	16	91.06
8	89.07	17	91.11
9	88.04	18	89.94

From the above experimental research, it can be seen that the thinking navigation algorithm proposed in this paper has a very obvious effect on the optimization of the flight trajectory of the aircraft.

4. Conclusion

Track tracking is based on the precompleted flight track, and it performs flight track planning in real time according to the external interference, environment, and other internal and external factors in the actual flight. When completing a certain task or operation, it calculates and tracks the required optimal flight path, and generates navigation and control instructions. Correspondingly, it controls the force and attitude angle of the aircraft (including engine thrust, attack angle, roll angle, and heading angle) to make the aircraft fly along the planned flight path. This paper combines the thinking navigation algorithm to optimize the flight trajectory of the aircraft and analyzes the flight trajectory of the aircraft through the intelligent model. The simulation test study shows that the thinking navigation algorithm proposed in this paper has a very obvious effect on the optimization of the flight trajectory of the aircraft.

Data Availability

The labeled datasets used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interests.

Acknowledgments

This study was sponsored by the Chinese Flight Test Establishment.

References

[1] W. Yi, Y. Liu, E. Bodanese, A. Nallanathan, and G. K. Karagiannidis, "A unified spatial framework for UAV-aided MmWave networks," *IEEE Transactions on Communications*, vol. 67, no. 12, pp. 8801–8817, 2019.

[2] W. Feng, J. Wang, Y. Chen, X. Wang, N. Ge, and J. Lu, "UAV-Aided MIMO communications for 5G internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1731–1740, 2018.

[3] H. He, S. Zhang, Y. Zeng, and R. Zhang, "Joint altitude and beamwidth optimization for UAV-enabled multiuser communications," *IEEE Communications Letters*, vol. 22, no. 2, pp. 344–347, 2018.

[4] B. Li, S. Zhang, X. Zhang, B. Xi, and Y. Tian, "Application and research of scheduling mechanism for UAV cluster launching control system," *Xibei Gongye Daxue Xuebao/Journal of Northwestern Polytechnical University*, vol. 36, no. 2, pp. 353–358, 2018.

[5] W. Mei, Q. Wu, and R. Zhang, "Cellular-connected UAV: uplink association, power control and interference coordination," *IEEE Transactions on Wireless Communications*, vol. 18, no. 11, pp. 5380–5393, 2019.

[6] Q. Zhu, K. Jiang, X. Chen, W. Zhong, and Y. Yang, "A novel 3D non-stationary UAV-MIMO channel model and its statistical properties," *China Communications*, vol. 15, no. 12, pp. 147–158, 2018.

[7] X. Wang and M. C. Gursoy, "Coverage analysis for energy-harvesting UAV-assisted mmWave cellular networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 12, pp. 2832–2850, 2019.

[8] M. Y. Arafat and S. Moh, "Localization and clustering based on swarm intelligence in UAV networks for emergency communications," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8958–8976, 2019.

[9] D. Ebrahimi, S. Sharafeddine, P. H. Ho, and C. Assi, "UAV-aided projection-based compressive data gathering in wireless sensor networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1893–1905, 2019.

[10] S. Fu, Y. Tang, N. Zhang, L. Zhao, S. Wu, and X. Jian, "Joint unmanned aerial vehicle (UAV) deployment and power control for internet of things networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4367–4378, 2020.

[11] D. Liu, Y. Xu, J. Wang et al., "Opportunistic UAV utilization in wireless networks: motivations, applications, and challenges," *IEEE Communications Magazine*, vol. 58, no. 5, pp. 62–68, 2020.

[12] X. Liu, Y. Liu, and Y. Chen, "Reinforcement learning in multiple-UAV networks: deployment and movement design," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 8036–8049, 2019.

[13] T. Yu, X. Wang, and A. Shami, "UAV-enabled spatial data sampling in large-scale IoT systems using denoising autoencoder neural network," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1856–1865, 2019.

[14] Y. S. Wang, Y. W. P. Hong, and W. T. Chen, "Trajectory learning, clustering, and user association for dynamically connectable UAV base stations," *IEEE Transactions on Green Communications and Networking*, vol. 4, no. 4, pp. 1091–1105, 2020.

[15] M. Khabbaz, J. Antoun, and C. Assi, "Modeling and performance analysis of UAV-assisted vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 9, pp. 8384–8396, 2019.

[16] D. Liu, J. Wang, K. Xu et al., "Task-driven relay assignment in distributed UAV communication networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 11003–11017, 2019.

[17] W. Rahmaniar and A. E. Rakhmania, "Online digital image stabilization for an unmanned aerial vehicle (UAV)," *Journal of Robotics and Control (JRC)*, vol. 2, no. 4, pp. 234–239, 2021.

[18] A. Thibbotuwawa, G. Bocewicz, P. Nielsen, and B. Zbigniew, "Planning deliveries with UAV routing under weather forecast and energy consumption constraints," *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 820–825, 2019.

[19] N. H. Motlagh, M. Bagaa, and T. Taleb, "UAV-based IoT platform: a crowd surveillance use case," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 128–134, 2017.

[20] X. Yu, X. Dong, X. Yang et al., "Air-ground integrated deployment for UAV-enabled mobile edge computing: a hierarchical game approach," *IET Communications*, vol. 14, no. 15, pp. 2491–2499, 2020.