RESEARCH ON PATH PLANNING OF LOGISTICS INTELLIGENT UNMANNED AERIAL VEHICLE

Hai-Wu Lee,* Shoaib Ahmed,** and Chi-Shiuan Lee***

Abstract

As unmanned aerial vehicle (UAV) starts to be frequently used in industry, agriculture, reconnaissance, and logistics, flight research involving UAV also starts to cover a wider range. In the civil field, UAVs are generally used as an auxiliary tool to deal with urban problems, but buildings are the main factors hindering the flight of UAVs. Therefore, it is necessary to find out the optimal flight path of UAVs under certain constraints. The intelligent logistics UAV proposed in this paper is used to replace the courier to deliver small goods. It is a quadcopter integrated with a webcam, ultrasound ground proximity warning system (GPWS), and is controllable through a mobile APP. Linear quadratic regulator (LQR) and improved proportion-integral-derivative (PID) controllers are applied in its flight control system. In the path planning, compared with the traditional A* algorithm and artificial potential field algorithm, the ant colony algorithm and dynamic path planning used in this paper can quickly solve the optimal path of UAV in complex terrain. UAV can detect the working state of components during flight and the surrounding obstacles. The operator's phone will receive the feedback information of the UAV immediately and carries out the process manually or automatically to ensure flight safety.

Key Words

Logistics intelligent UAV, improve PID, linear quadratic regulator, dynamic path planning, emergency treatment system

1. Introduction

In recent years due to their applications in prospect, logistics, photography, and communications [1]. UAVs are usually easy to control, even though WIFI smartphones and tablets [2] based on the development of a mobile communication network, UAV further break through the limitation of information transmission distance. At present, the application demand for UAVs in cities is increasing [3]. UAVs are commonly used to perform various tasks in dangerous and complex environments [4]. UAVs have the advantages of strong flexibility, quick response, avoiding casualties [5] in inclement weather, toxic, explosion, and other conditions.

In recent years, the path planning of unmanned aerial vehicles (UAV) in a three-dimensional environment has become a highly focused scholars' research topic. UAV path planning finds the optimal flight path from the starting point to the target point under certain constraints [6]. There are many ways to realise UAV path planning. The author summarises the following feasible methods by consulting materials and studying the UAV itself. There are similarities between UAV path planning and common transportation vehicle path planning methods in life and the main algorithms are the Dijkstra algorithm, Floyd algorithm [7], A* algorithm, and ant colony algorithm [8]. Among them, the multi-objective optimisation Dijkstra algorithm based on dimension reduction mapping has been proven effective [9]. The method of applying A* algorithm to path planning modelling and simulation [10], Chen *et al.* [11] introduced an accurate algorithm based on mixed-integer linear programming (MILP) to optimise the A* algorithm based on UAV regional model so as to realise path planning. Xavier *et al.* [12] proposed a new path tracking algorithm (nlgl+) on the original algorithm by comparing the carrot chase method, pure tracking line of sight algorithm, and nonlinear guidance law algorithm in the model. Yi *et al.* [13] proposed a two-dimensional obstacle avoidance method based on pH curve and proved the algorithm's effectiveness in autonomous path planning of UAV in complex dynamic threat environment. Che *et al.* [14] proposed an improved ant colony optimisation algorithm based on particle swarm optimisation algorithm for path planning of autonomous underwater vehicle.

We finally decided to adopt an ant colony algorithm and dynamic path planning in a bionic optimisation algorithm by comparing different methods and combining

* School of Big Data, Fuzhou University of International Studies and Trade, No.28, Yuhuan Road, Shouzhan New District, Changle District, Fuzhou 350202, Fujian Province, PR. China; e-mail: 2439141397@qq.com

** Graduate School of Science and Engineering, University of the Ryukyus, Okinawa, Japan 903-0213; e-mail: eshoa1996@outlook.com

*** Public Affairs, Fo Guang University, Taiwan ROC 262307; e-mail: 0988632906@gmail.com

Corresponding author: Hai-Wu Lee

Recommended by Howard Li
(DOI: 10.2316/J.2024.206-0842)

them with the actual situation. By simulating the potential mechanism of self-organising behaviour in nature, the bionic optimisation algorithm has been proved to be a promising technology for solving complex optimisation problems.

By simulating the underlying mechanisms of self-organising behaviour in nature, bionic optimisation algorithms have proven to be promising techniques for solving complex optimisation problems [15]. Unlike traditional mathematical methods, bionic optimisation algorithms are developed and implemented by mimicking biological behaviours in nature [16]. A boundary control strategy is employed to govern a two-link rigid-flexible wing. Its design is based on the bionics approach to improve aircraft mobility and flexibility [17]. The ant colony algorithm is a meta-heuristic algorithm proposed by Colorni, Dorigo, and Maniezzo. Since then, several variants of the ant colony algorithm have been proposed, such as ant colony systems, antrank, elite ant colony systems [18], max–min ant colony systems, and ant colony system algorithms. The ant colony cloning algorithm is one of the best variants of the ant colony optimisation algorithm, developed by Dorigo and Gambardella [19]. However, there are still some shortcomings in-database processing. Zheng *et al.* [20] proposed the parallel ant colony optimisation algorithm (PACO) to solve the problems of local convergence and multi-joint query optimisation of the ant colony algorithm. On this basis, he proposed an ant colony optimisation algorithm and then used the proposed ant colony optimisation algorithm to select the optimal flight path from the candidate path according to the proposed utility function [21].

Fixed-wing aircraft have nonlinear dynamics and require a robust control system to achieve stable flight under most outdoor conditions [22]. Although the control algorithm has made significant progress in model predictive control, adaptive control, sliding mode control, these algorithms are computationally intensive and unsuitable for small, embedded processors in UAV avionics [23]. Proportional integral derivative (PID) is one of the most used controllers in autopilots, which is simple to implement and has low computational intensities. However, the disadvantage of this linear controller is that it may not meet the design requirements for a wide range of operating conditions. To solve this problem, the fixed-wing aircraft adopts a gain scheduling method. The main advantage of the gain scheduling PID controller is its ability to remain stable under different operating conditions [24]. The University of Alberta has conducted experimental tests using an internal and external circulation control structure. Due to the physical constraints of actuators, such as the rotor speed of drones, many researchers use input saturation to study motion control [25]. In addition, restrictions on the country should also be considered in practical issues. For example, limiting the attitude angle to a safe range is necessary to avoid unnecessary drone configuration [26]. Attitude constraints are also conducive to visual servo providing airborne cameras to obtain feedback information of motion control [27]. The trajectory tracking problem of flapping-wing micro aerial vehicles

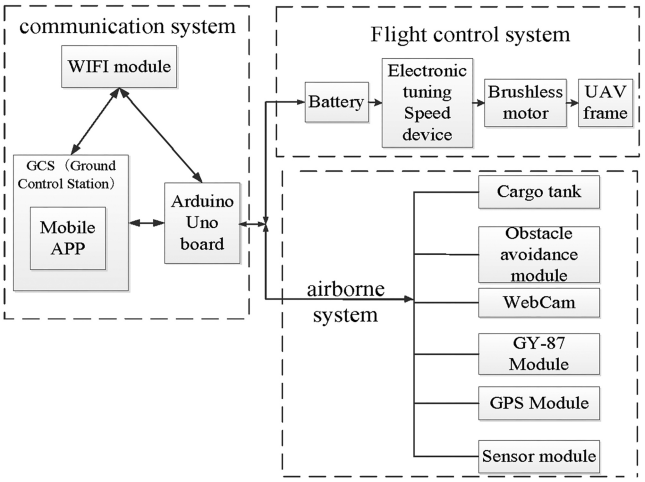


Figure 1. System architecture diagram.

(FWMAVs) in the longitudinal plane is investigated under the hierarchical structure, and an adaptive control strategy is proposed to achieve autonomous tracking [28]. Che and Yu [29] processed a neural-network estimators based fault-tolerant tracking control for AUV via ADP with rudders faults and ocean current disturbance. Che [30] the author processed again single critic network based fault-tolerant tracking control for under actuated AUV with actuator fault.

Therefore, we have studied a logistics UAV, which is an unmanned low altitude aircraft operated by radio remote control equipment to carry packages, combined with the ant colony algorithm and dynamic path planning to calculate the accurate flight path of the UAV and finally achieve safe and accurate delivery. The structure of this paper is as follows. In the second part, the system architecture of the logistics drone is proposed. The third part introduces the related methodology, including improved PID-based flight control theory, ant colony algorithm, and emergency treatment system. The fourth part describes the principles and flowcharts of UAV communication control and path planning. The fifth and sixth parts compare and analyse different methodologies to give experimental and simulation results. Finally, the conclusion is drawn.

2. System Architecture

Nowadays, the research on intelligent control has begun to turn from the ground to the air. Subsequently, UAV technology has become an important research direction of automatic control. However, since the birth of quadrotor aircraft, it has reached a bottleneck state due to poor control, low safety performance, and other problems. This paper studies a four-rotor aircraft equipped with an ultrasonic obstacle avoidance system. The aircraft architecture is shown in Fig. 1, which mainly includes the following aspects:

1) Ground Control Station (GCS): This android-based mobile APP is designed based on MIT APP Inventor 2 development environment. The operator connects the

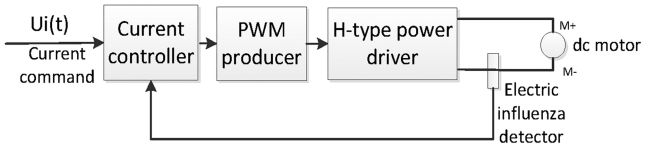


Figure 2. Current control mode diagram of dc motor driver.

mobile device with the UAV through the WIFI module to control the take-off, landing, and flight direction of the UAV.

2) Arduino Uno Board: The flight control board of this system, which can achieve the bidirectional transmission of instructions between the ground control terminal, the flight control system, and the airborne system.
3) WIFI Module (ESP8266): It is used for two-way communication between mobile phone and Arduino to realise the function of sending and receiving information between mobile phone and UAV [31].
4) Obstacle Avoidance Module: It is mainly composed of HC-SR04 ultrasonic wave, SG90 steering gear, and tri-colour light. The steering gear combined with the ultrasonic module can detect obstacles in the range of 180 degrees ahead; the tricolour light displays different colours to reflect the distance between the quad and the obstacle [32].
5) Webcam: Image taken during a drone flight.
6) GY-87 Module: The 10-dof module, which is composed of a three-axis gyro, an accelerometer, a magnetic field, and a pneumatic pressure, is used to calculate the current attitude of the four-axis aircraft to control its state.
7) GPS Module: The NEO-7N UBLOX satellite locator is used to obtain the position of the UAV.
8) Motor Module: Drive the brushless motor by Arduino output PWM to the electronic governor.
9) Sensing Module: Mainly consists of an alarm buzzer, photosensitive resistor, and LED lamp, among which the 5539-type photosensitive resistor determines the current light intensity. When the light is dim, the LED light on the wing will be lit.

3. Methodology

This system proposes three methods: flight control system, improved ant colony algorithm, and emergency treatment system.

3.1 Flight Control System

The motor driver is a dc motor driver with the current return. The UAV drive system in this paper uses the current control mode of the motor driver. The schematic diagram of the system is shown in Fig. 2. The input voltage $U_i(t)$ represents the command value of the current.

The control system of the UAV uses the kinetic energy equation of the motor control system and the closed-loop PID controller to control the balance. Its control input is the dc motor current driver input voltage $U_i(t)$ and output

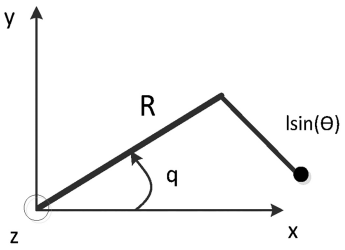


Figure 3. X-Y plane projection of the motor axis.

is the angle of reversion $\theta(t)$ of the central motor axis. We will first use the Lagrange energy method to obtain the dynamic equation of the motor and then obtain the linear state equation and its input and output transfer function under the vertical equilibrium state of the motor. Finally, the PID controller is designed according to the transfer function.

Let the origin of the coordinate system of the UAV motor be located at the centre of the motor rotation axis, the x-y axis is a plane relatively parallel to the ground, and the z-axis is vertical and upward to the x-y plane, as shown in Fig. 3. The motor axis is tilted at an angle θ(t) around the X-axis and has a vertical position of zero degree, and the x-y plane angle q(t) is rotated around the z-axis. Let (x, y, z) be the position of the centre of gravity of the motor and can be obtained from the geometric relationship:

$$x = R\cos q + l\sin\theta\sin q \tag{1}$$
$$y = R\sin q - l\sin\theta\cos q \tag{2}$$
$$z = l\cos\theta \tag{3}$$

and

$$\dot{x} = (-R\sin q + l\sin\theta\cos q)\,\dot{q} + l\cos\theta\sin q\dot{\theta} \tag{4}$$
$$\dot{y} = (R\cos q + l\sin\theta\sin q)\,\dot{q} - l\cos\theta\cos q\dot{\theta} \tag{5}$$
$$\dot{z} = -l\sin\theta\dot{\theta} \tag{6}$$

Therefore, the total kinetic energy of the motor can be expressed as (7), where m is the mass of the motor itself, and I is the equivalent moment of inertia.

$$\begin{aligned} K &= \frac{1}{2}I\dot{q}^2 + \frac{1}{2}m\left(\dot{x}^2 + \dot{y}^2 + \dot{z}^2\right) \\ &= \frac{1}{2}\left(I + mR^2 + ml^2\sin^2\theta\right)\dot{q}^2 \\ &\quad + \frac{1}{2}ml^2\theta^2 - mRl\cos\theta\dot{q}\dot{\theta} \end{aligned} \tag{7}$$

The total potential energy:

$$P = mgl\cos\theta \tag{8}$$

The dynamic equation of UAV motor can be deduced from Lagrangian function $L = K$ - P and Lagrange dynamic equation as follows:

$$ml^2\ddot{\theta} - mRl\cos\theta\ddot{q} - ml^2\sin\theta\cos\theta\dot{q}^2 - mgl\sin\theta = 0 \tag{9}$$

The characteristic equation of the motor linearisation system is:

$$\Delta(s) = |sI - A| = s^4 + as^3 - \frac{g}{l}s^2 - \frac{g}{l}as = 0 \quad (10)$$

If the state variable is $x(t)$, the input variable is $u(t)$, and the output variable is $y(t)$, then the PID algorithm of the surrounding aircraft is shown in the equation of state (11), where A and B are matrices, and t is the time coefficient.

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (11)$$

$$\mathrm{y}(t) = Cx(t) + Du(t) \quad (12)$$

Then, performance indicators were obtained using the optimal control under the linear quadratic regulator (LQR). The object is the linear system given in the form of state space in the present control theory, and the objective function is the quadratic function of object state and control input. LQR optimal design refers to the state feedback controller K designed to minimise the quadratic objective function J, and the weight matrix Q and R only determine k, so the choice of Q and R is particularly important. Let the state equation of the linear-time-varying system be:

$$u(t) = -kx(t) \quad (13)$$

$$\dot{x}(t) = (A - Bk)x(t) \quad (14)$$

$$y(t) = (c - dk)x(t) \quad (15)$$

Performance index:

$$J = \frac{1}{2}\int_0^\infty \left[x^T(t)\,\mathrm{Qx}(t) + u^T(t)\,Ru(t)\right]\mathrm{dt} \quad (16)$$

In the formula, the vector $x(\mathrm{t}) \in Rn$, $u(\mathrm{t}) \in Rm$, the matrix $A(\mathrm{t})$ is $n \times n$-dimensional time-varying system matrix, $B(\mathrm{t})$ is $n \times m$ dimensional gain matrix, $Q(\mathrm{t})$ is the semi-positive definite matrix, which is the weighted state, $R(\mathrm{t})$ is the symmetric positive definite matrix, which is the weighted control input $u(\mathrm{t})$.

For the time-varying state regulator problem in infinite time, if the matrix pairs $\{A(\mathrm{t}),\ B(\mathrm{t})\}$ are completely controllable, there exists A unique optimal control.

$$u(t) = -kx(t) = -R^{-1}b^T Px(t) \quad (17)$$

The matrix P is the solution of the Riccati algebraic (18), the matrix Q and the scalar R are the state variables and the proportion of the control input, respectively.

$$\mathrm{Pb}R^{-1}b^T P - Q - PA - A^T P = 0 \quad (18)$$

The optimal performance index is:

$$J^* = \frac{1}{2}x^T(t)\,P(t)\,x(t) \quad (19)$$

3.2 Improved Path Planning of Ant Colony Algorithm

This system proposes two algorithms: the ant colony algorithm and adaptive strategy.

3.2.1 Ant Colony Algorithm and Mathematical Model

Ants leave through the path of a volatile secretion (hereafter called pheromones), pheromone disappears with time will gradually evaporate. Ants in the foraging process can perceive the existence of the pheromone and concentration, to guide its direction of movement, and tend to move in the direction of the pheromone concentration is high, the path of the pheromone concentration, the higher, the better, by means of this kind of positive feedback can get the best path.

Let k $(k = 1, 2 \ldots, m)$ in the process of motion. The information concentration determines the direction of motion transfer on each path. We record all nodes currently passed by the K ant. This collection of storage nodes dynamically adjusts to the movement of the ants. During the search process of the algorithm, the ant will intelligently choose the path to go next.

Suppose m represents the total number of ants, and d_{ij} $(i, j = 0, 1 \ldots, m-1)$ represents the distance between node i and node j, and the pheromone concentration on the connection between i and j at time t. At the initial moment, m ants will be randomly placed, and the initial pheromone concentration in each path is the same. At time t, the state transition probability of ant k from node i to node j is:

$$\rho_{\mathrm{ij}}^k = \begin{cases} \frac{[\tau_{\mathrm{ij}}(t)]^\alpha \times [\eta_{\mathrm{ij}}(t)]^\beta}{\sum_{k \in \mathrm{allowed}_k} [\tau_{\mathrm{ik}}(t)]^\alpha \times [\eta_{\mathrm{ik}}(t)]^\beta}, & j \in \mathrm{allowed}_k \\ 0, & \mathrm{other} \end{cases} \quad (20)$$

In (18), allowed k represents all nodes that can be selected in the next step, and c is the set of all nodes; As an information heuristic factor, α represents the relative importance of trajectory in the algorithm, reflecting the influence of path selection by ants on the amount of information in the path. The larger this value is, the stronger the collaboration between ants will be. β can be called an expected heuristic factor and represents the relative importance of visibility in an algorithm. η_{ij} is the inspiration function. In the algorithm, it represents the expected degree of transfer from node i to node j normally, $\eta_{\mathrm{ij}} = 1/d_{\mathrm{ij}}$. When the algorithm is running, each ant searches forward according to (18).

In the process of ant movement, to avoid leaving too many pheromones on the road, the enlightening information is flooded. Residual information should be updated and processed after each ant traversal is completed. Thus, at time $t + n$, the information on the path (i, j) is adjusted as (19) and (20).

$$\tau_{\mathrm{ij}}(t+n) = (1-p) \times \tau_{\mathrm{ij}}(t) + \Delta\tau_{\mathrm{ij}}(t) \quad (21)$$

$$\Delta\tau_{\mathrm{ij}}(t) = \sum_{k=1}^{m} \Delta\tau_{\mathrm{ij}}^k(t) \quad (22)$$

On the type of constant $\rho\epsilon(0, 1)$ represents pheromone volatilisation factor to express the degree of loss of path information. ρ is related to the size of the algorithm's global search ability and convergence speed, usable 1 - ρ on behalf of the residual pheromone factor. $\Delta\tau_{\mathrm{ij}}^k(t)$ is a search

path pheromone increment (i, j). At the initial moment as $\Delta\tau_{\mathrm{ij}}(0) = 0$, $\Delta\tau_{\mathrm{ij}}^{k}(t)$ said the first ant k at the end of the traversal path pheromone (i, j).

As a result of the pheromone update strategy is different, the scholar Dorigo. M study found that the three different kinds of basic ant colony algorithm model, respectively, as the "ant system," the "ant quantity system," and "ant close system," three models of as $\Delta\tau_{\mathrm{ij}}^{k}(t)$ are different.

Ant system:

$$\Delta\tau_{\mathrm{ij}}^{k} = \begin{cases} \frac{Q}{L_k} & \text{ant K walks past ij} \\ 0, & \text{other} \end{cases} \tag{23}$$

Ant quantity system:

$$\Delta\tau_{\mathrm{ij}}^{k} = \begin{cases} \frac{Q}{d_{\mathrm{ij}}}, & \text{Ant k walks ij between t and t} + 1 \\ 0, & \text{other} \end{cases} \tag{24}$$

Ant close system:

$$\Delta\tau_{\mathrm{ij}}^{k} = \begin{cases} Q, & \text{Ant k walks ij between t and t} + 1 \\ 0, & \text{other} \end{cases} \tag{25}$$

Pheromones in "ant quantity system" and "ant density system" are updated after the ant completes one step, local information is adopted. In the "ant-week system", pheromones in the path are updated after the ant completes a cycle, that is, the application of the overall information. Experiments on a series of standard test problems show that the algorithm is more efficient.

3.2.2 Adaptive Strategy

The value of the pheromone volatility factor ρ in the ant colony algorithm directly affects the global search ability and its convergence speed.

This paper proposes an improved strategy of the ant colony algorithm based on adaptive pheromone volatiles. Set the volatile pheromone factor ρ initial value $\rho(t_0) = 1$ when the algorithm on many times calculation but without obvious change, will ρ based on (24) type adjustment:

$$\rho(t) = \begin{cases} 0.95\rho(t-1), & \text{if } 0.95\rho(\mathrm{t}-1) \geq \rho_{\min} \\ \rho_{\min}, & \text{otherwise} \end{cases} \tag{26}$$

In the above formula, $\rho_{\min}$ for the value of the minimum value, the existence of $\rho_{\min}$ can prevent reducing the speed of the convergence algorithm when the ρ is too small. To improve the ability of the optimal global solution and on the end of each cycle search, retain the optimal solution, let ρ adaptive adjustment. The volatile pheromone factor ρ redefine also needs to make corresponding adjustments to the pheromone update strategy. Update the pheromone distribution according to the distribution of pheromones, readjust the pheromone distribution on the path so that it is not easy to be too concentrated or too dispersed, avoiding the premature mechanism of the algorithm. The local update of pheromone can adopt the strategy as shown in (25).

$$\tau_{\mathrm{ij}}(t1) \begin{cases} \tau_{\mathrm{ij}}(t) - \frac{5}{d_{\mathrm{ij}}}, & \text{If } \frac{m}{2} \text{or } \frac{m}{4} \text{ ants choosing} \\ & \text{the same path} \\ \tau_{\mathrm{ij}}(t) + \frac{1}{d_{\mathrm{ij}}}, & \text{otherwise} \end{cases} \tag{27}$$

Since ants often choose the path with high pheromone content, when many ants choose the same path, the increase of pheromone is so large that more ants tend to gather in this path, so $1/d_{\mathrm{ij}}$ is taken as the increased pheromone. If the number of ants choosing this path reaches m/2, or if the ants with m/4 choose this path, the traversal will be terminated and the $5/d_{\mathrm{ij}}$ pheromone will be reduced to the average pheromones on each path to reduce the possibility of getting into local optimal. Global pheromones are updated according to (26) and (27).

$$\tau_{\mathrm{ij}}^{k}(t+1) = (1-\rho)\tau_{\mathrm{ij}}^{k} + \varphi_k * \Delta\tau_{\mathrm{ij}}^{k}(t) \tag{28}$$

$$\tau_{\mathrm{ij}}(t+1) = (1-\rho)\tau_{\mathrm{ij}} + \sum_{k=1}^{m} \varphi_k * \Delta\tau_{\mathrm{ij}}^{k}(t) \tag{29}$$

In the above formulas, φ_k represent the ant k of the path (i, j) on the influence degree of the pheromone update, the calculation method is as follows: set the route for a total of s (i, j) of ants in the current iteration to traverse the path length from small to a large order, will get the serial number in the rank [ ], the rank $[k]$ said the ant k corresponding serial number, taking

$$\varphi_k = \frac{s}{2} - \mathrm{rank}[k] + 1 \tag{30}$$

In the (28), if the value of rank $[k]$ is bigger, the φ_{k} is negative, the global update the ants to reduce the amount of the corresponding path pheromone. The greater the value of rank $[k]$, then the path pheromone reduces. The greater the degree of pheromone on the path to making the worse retain less, it is beneficial to keep better path information.

The initial path is determined according to the path information, and the decision problem of the optimal path is solved by dynamic programming.

With ultrasound on a small scale search obstacles and do real-time planning to avoid obstacles, algorithms running in circle method selects the influence of obstacles, painted garden, the operation is simple and easy, and the characteristics of the circle facilitate a tangent, by the tangent to a small scale can be got the most change path, add such a simple dynamic algorithm is better than the original ant colony more flexible.

3.3 Emergency Treatment System

The system consists of a signal emergency plan, electricity emergency plan, and power and signal emergency plan.

3.3.1 Signal Emergency Plan

The UAV is allowed to fly only when the transmitter and receiver are successfully connected *via* WiFi. When

the UAV rotor is started, the system records the starting point position through GPS positioning and stores it in the database. In the process of UAV flight, if the UAV signal is insufficient or lost, the control system will make the UAV automatically switch from flight to hover state and try to connect again. The connection time is two minutes. If the connection timeout, the UAV will read the take-off position and plan the path to return to the landing.

3.3.2 Electricity Emergency Plan

The UAV is allowed to fly when the power is higher than 40%. When the UAV rotor starts, the system records the starting point position through GPS positioning and stores it in the database. When the UAV battery power is less than 20%, the system will remind the operator to operate the aircraft immediately through the human–computer interaction interface. When the UAV power is less than 10%, no one has the opportunity to read the position of the starting point and plan the path to force the return to prevent the crash caused by the insufficient power supply. When the power of the UAV is less than 5% and the UAV has not returned to the initial position, the UAV will make a forced landing nearby according to the ground area given by the GPS map, so as to avoid the forced landing in the dangerous areas, such as rivers and roads. After the forced landing, the UAV will send the position to the operator and enter the sleep state so that the operator can find the UAV according to the positioning.

3.3.3 Power and Signal Emergency Plan

If the battery power of the UAV is less than 5% during the process of reconnection or during the process of returning after the connection timeout, the UAV will make a forced landing near the safe range of the map and send positioning by GPS.

4. System Software and Hardware Design

The system is divided into three parts: communication control, ultrasonic obstacle avoidance, and path planning.

4.1 Communication Control System

The communication control system of the UAV is mainly to connect to the Internet through WIFI, and then control the UAV to execute commands, such as forward, back, left and right steering, take-off and landing through the APP control interface of the ground control terminal [33]. Figure 4 is the flowchart of the flight control system.

The system first initialises the port and the sensor, then detects and reads the remote-control command data. The PID is adaptively adjusted through command data and attitude data. The adjusted PID parameters are input to the PID controller to calculate the speed of the corresponding motor. Finally, the PWM signal is adjusted to drive the motor.

During the flight of the UAV, the presence of obstacles is detected by ultrasound. If the ultrasonic detects

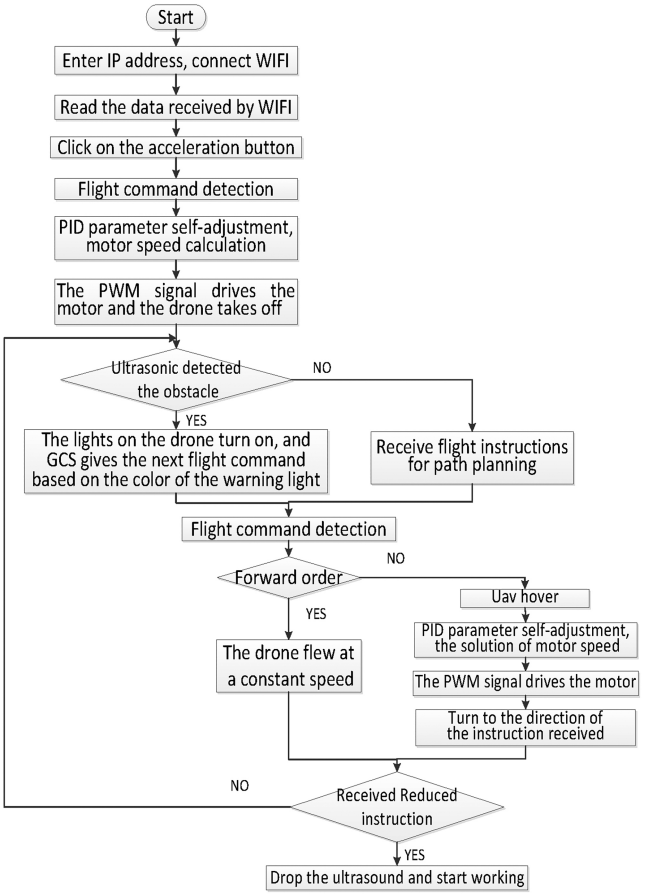


Figure 4. Flowchart of flight control system.

obstacles, the warning light will be on and wait for the next command at the ground end. Otherwise, if the obstacles are not detected, the path planning flight command at the ground end will be received. By detecting the flight command, if the forward command is received, the UAV will fly forward at a uniform speed, otherwise the UAV will hover, and the PID will be adaptively adjusted through the command data and attitude data. Input the adjusted PID parameters to the PID controller to calculate the speed of the corresponding motor, and finally drive the motor by adjusting the PWM signal.

In the process of controlling the UAV flight, ultrasonic detects the surrounding obstacles by rotating the steering gear. If detected, the tri-colour LED light on the UAV will be on, and the GCS will judge the safe direction of the next flight according to the LED light colour. If no obstacle is detected, the UAV will execute the next action according to the flight instruction of path planning, and the planning process is shown in Fig. 6. Until the landing command is received, the below ultrasonic starts to work, and the landing process is shown in Fig. 5.

4.2 UAV Landing Obstacle Avoidance System

When the UAV receives the descending command, it is assumed that point P is the pre-landing point on the ground, and point O is the position of the UAV hovering in the air, and OP is perpendicular to the landing surface.

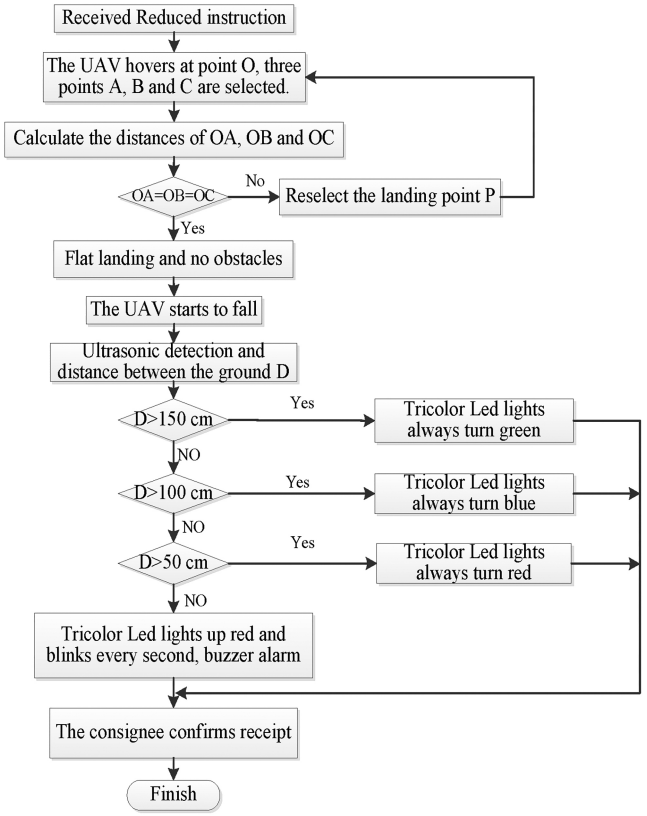


Figure 5. Flowchart of descending ultrasonic obstacle avoidance system.

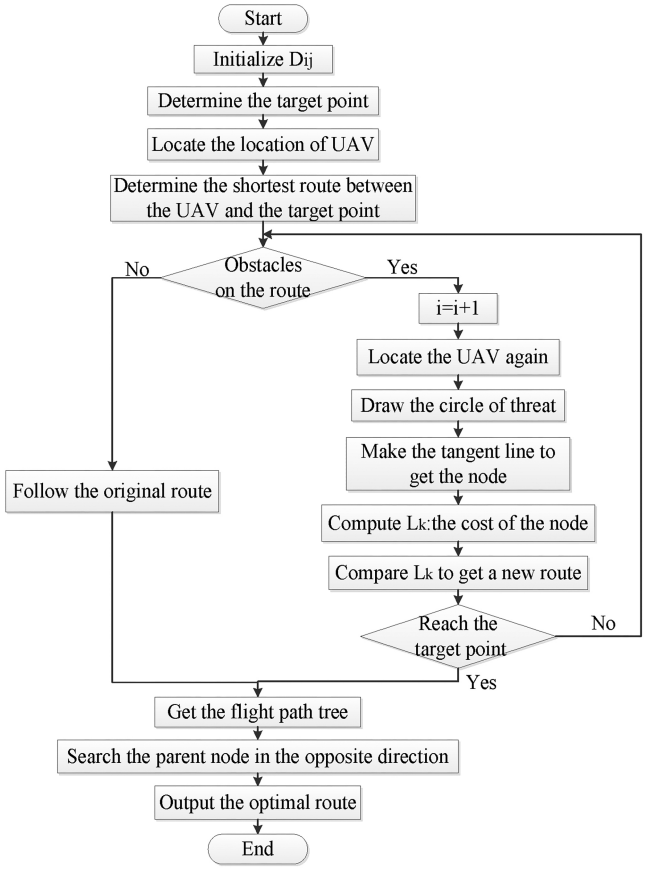


Figure 6. Dynamic programming flowchart.

Choose A, B, and C within the radius of 1 m with P as the centre of the circle. The height between these three points and the UAV can be determined by calculation. If it is the same, it means that the landing surface is flat, then it begins to fall. At the same time, ultrasonic measure the distance D from the ground to remind the operator to pay attention and ensure a safe landing. If not, then re-select the landing point. After the UAV landed, the staff took out the goods for acceptance, and the UAV distribution task was complete.

4.3 The Path Planning of UAV

The following are the main description of Fig. 6.

1) D_{ij}: Track transfer point; the counter with i as the number of obstacles ($i = 1,\ 2,\ 3\ldots$),
2) j: The order of the point cuts ($j = 1, 2, 3, 4$).
3) L_k: Two tangents before and after the node and the total length of the arc.
   - Step 1: Determine the position of starting point A and target point B.
   - Step 2: Select the initial optimal route.
   - Step 3: Deal with the threat area passed by line AB during the flight.
   - Step 4: If an obstacle is found, redefine the starting point.
     Instead, continue on the same course [34].
   - Step 5: Take the obstacle as the centre of the circle, confirm a safe distance as the radius, and construct a threat circle.
   - Step 6: Make tangents between point A and point B and the threatening circle, respectively, and the tangents are nodes at all levels.
   - Step 7: Calculate the sum of the tangent and tangent segments and the length of the arc and calculate the node cost.
   - Step 8: Compare node costs and plan a new optimal path.
   - Step 9: Make repeated obstacle judgements before reaching the target point.
   - Step 10: If there are obstacles, repeat the above process to get the new node and calculate the node cost.
   - Step 11: After evading all obstacles, obtain the track tree [35]
   - Step 12: Reverse search the parent node and trace all the way back to the initial node.
   - Step 13: Get the complete optimal track.

5. Simulation Comparative and Analysis

The simulation, comparison, and analysis are divided into two parts: improved PID and dynamic path planning.

5.1 Improved PID Algorithm

Symbol description:

1) $y(k)$ – discrete value of system response output.
2) $u(k)$ – discrete value of digital PID control output.

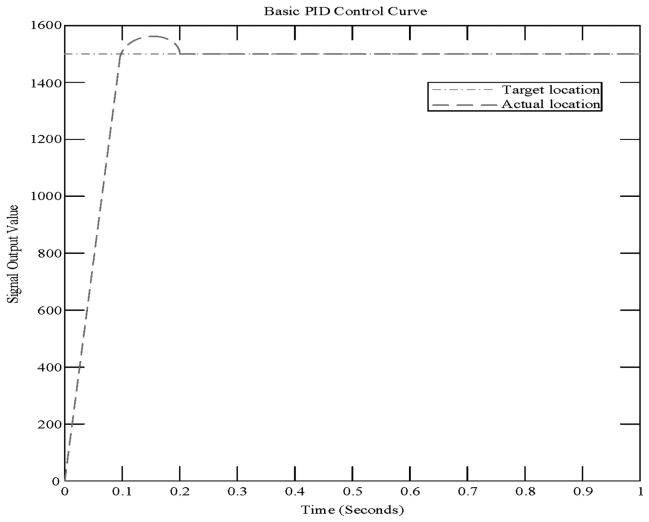


Figure 7. Basic PID control curve.

3) $r(k)$ – the discrete value of the expected output (known), which in this case is a constant (*i.e.*, step input).
4) $e(k) - e(k) = r(k) - y(k)$, is expected value - actual value, is the error comparison signal with unit negative feedback.

5.1.1 Basic PID Control Principle

When the continuous PID control is converted to digital, the differential link is replaced by difference, the integral link is replaced by summation, and the proportional link remains unchanged. The difference implementation only needs to be equivalent to $e(k)$-e_1. The implementation of the integral accumulates the original error after each operation, *i.e.*, $E_e = E_e = E_e + e_1$. PID controller output:

$$U(k) = K_p \cdot e(k) + K_d \cdot (e(k) - e_1) + K_i \cdot E_e \quad (31)$$

The discrete response points can be solved by the output of (29). The output PID control curve is shown in Fig. 7.

5.1.2 Compare and Analyse the Influence of PID Output Parameters

In this paper, to study the influence of three PID parameters K_p, K_i, and K_d, we set up an array of PID, K_p, K_i, and K_d. Take a value of the array each time, and then set a loop function for simulation. Then, all PID effects are output to a graph for comparison, and the output subgraph matrix is shown in Fig. 8(a)–(f).

According to Fig. 8(a) and 8(b), modification of K_p will shorten the rise time but may also bring about a large overshoot. Overshoot is not absolute. A small K_p may cause a large overshoot, while a large K_p will cause a small overshoot [for example, in Fig. 8, the comparison between Fig. 8(a) and 8(b)]. However, the introduction of integral is also necessary; otherwise, it will take a long time to reduce the error e(k) [such as the Fig. 8(e) diagram]. The introduction of the differential is equivalent to a leading correction, which will reduce overshoot, but the transitional differential is likely to cause tail oscillation, and the system will gradually become unstable. So there's a balance between the differential and the integral, and when that balance is met, the system has almost no oscillations, and the response is fast. It can be known from Fig. 8 that Fig. 8(c) overshoot is caused by excessive integration in Fig. 8(d), and Fig. 8(f) is ideal.

5.1.3 Improve PID Algorithm

When $U(k) > U_{\max}$, if $e(k) > 0$ means that the output value has not reached the specified value, or if $U(k) < 0$, if $e(k) < 0$ means that the output value exceeds the specified value. The integration will bring a lag, no more integration. $U_{\max} = r(k)$ is assumed in this simulation, and the comparison curve of the output of the improved PID algorithm is shown in Fig. 9.

According to Fig. 9, the overshoot of the system is significantly reduced, and the adjustment time is also shortened. This paper proposes an improved PID method, which eliminates the steady-state error and reduces the lag effect brought by the integral link.

Based on the above, the PID regulation can be summarised into the following two points:

1) When K_p is small, the system is sensitive to the introduction of differential and integral links. Integral will cause overshoot, so it should not be too large; Differentials can cause oscillations, and overshoot increases when oscillations are severe.
2) When K_p Increases, the overshoot of integral due to lag gradually decreases, but it should not be too small. Otherwise, the adjustment time will be too long. In this case, if you want to continue to reduce the overshoot, you can introduce the differential appropriately. The system may be unstable if K_p is increased continuously. Therefore, when K_p is increased and K_d is introduced to reduce overshoot, good steady-state characteristics and dynamic performance can be achieved even when K_p is not very large.

5.2 Analysis and Explanation of Path Planning

5.2.1 The Algorithm Description

Assume A as the starting point and B as the ending point and set an obstacle on the way to draw a circle with it as the centre to limit the influence range of the obstacle. The UAV flies on the outside and above the circle through real-time track planning, so two optimal paths can be obtained by making a tangent. According to the geometrical properties, the tangent lines of two points outside the circle are the same length. Therefore, the optimal path can be obtained only by comparing the arc length between two tangent points and selecting the shorter one, as shown in Fig. 10:

If there are multiple obstacles, set them as $D1$, $D2$, $D3$, and $D4$ successively. Repeat the above process to avoid all obstacles that influence the original route, and do not consider the obstacles that do not influence the original route. In the end, the optimal path can be obtained as shown in Fig. 11:

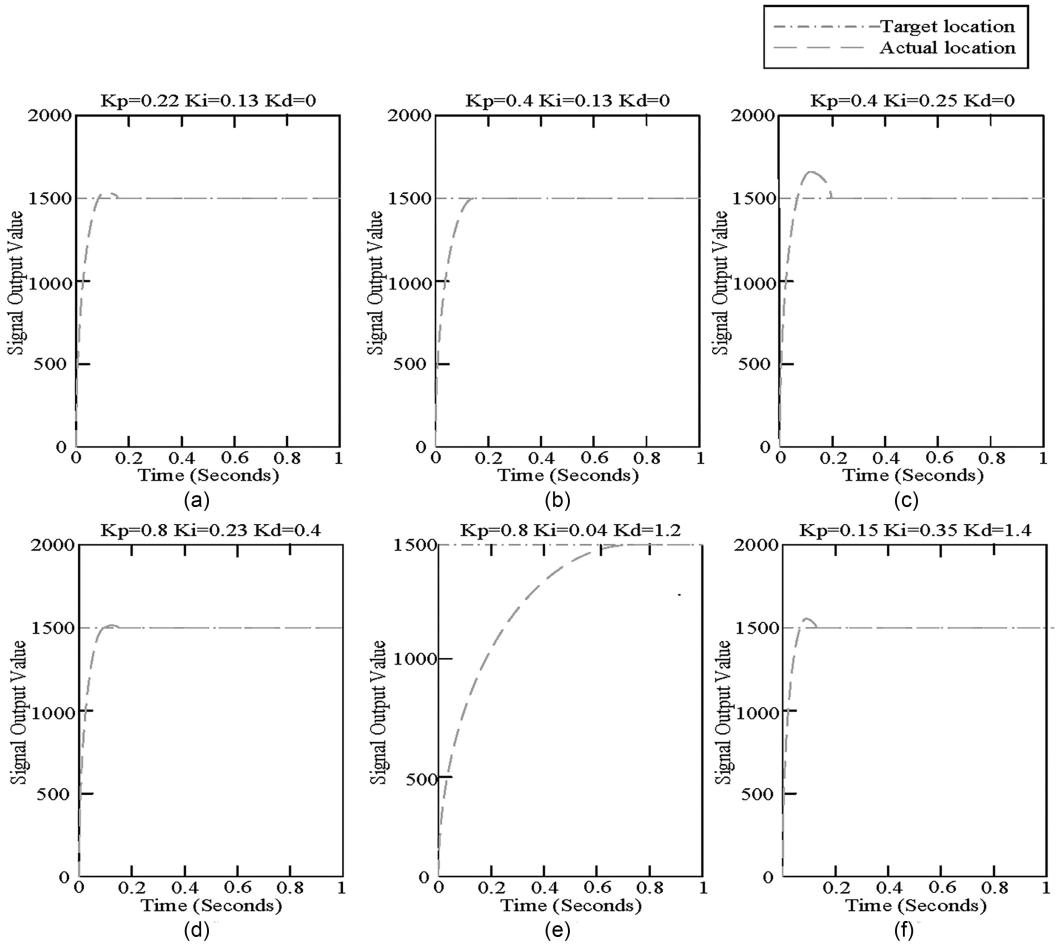


Figure 8. Comparison diagram of PID parameter influence effect.

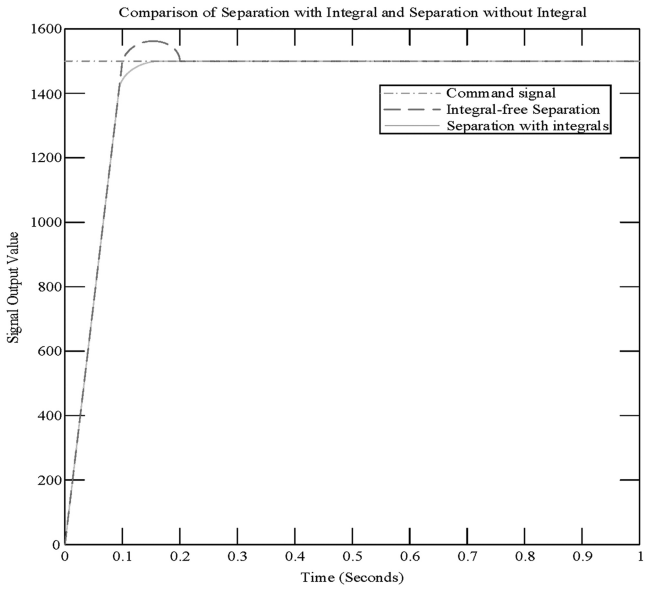


Figure 9. Improved PID algorithm curve.

By following the above approach, the problem can be transformed into a multi-stage decision-making problem. Finally, a track tree can be obtained as shown in Fig. 12:

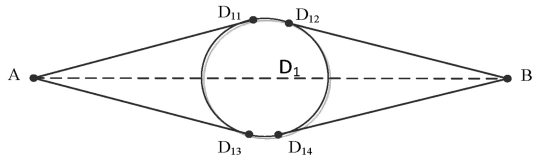


Figure 10. Dynamic programming diagram of the original path from A to B.

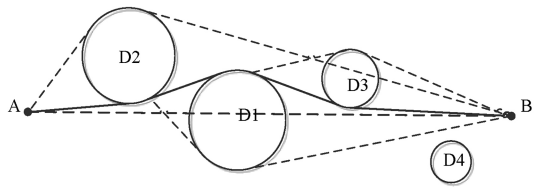


Figure 11. Dynamic planning diagram of multiple obstacles from A to B.

Conclusion: the dynamic programming method is faster than the traditional A* and artificial potential field algorithms. In this method, the UAV only needs to search and optimise in a small range and does not need to search in a large range. In this way, other uncertainties can be avoided.

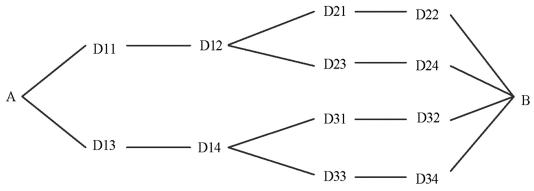


Figure 12. Real-time track tree diagram.

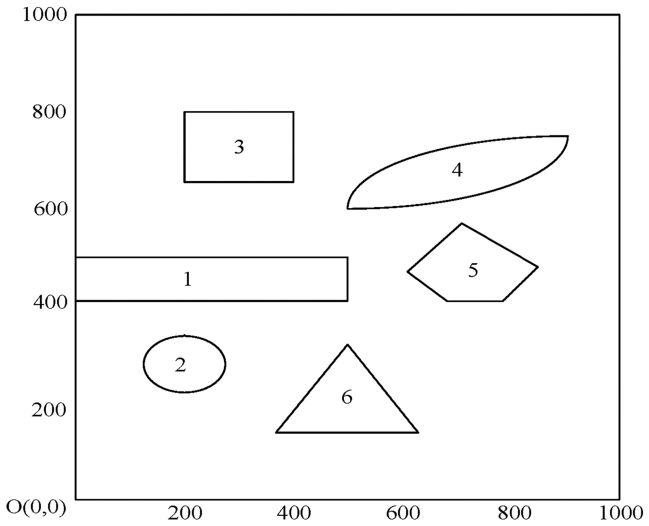


Figure 13. Schematic diagram of multiple obstacles.

5.2.2 Simulation and Analysis of Experimental Results

First, in a 1000 × 1000 simulation scene, suppose there is a UAV at the origin $O(0, 0)$, and there are six obstacles (including regular and irregular graphics) in the scene. As shown in Fig. 13:

Second, to meet the requirements of planning within a small range, the scene area coordinates are finely differentiated, an appropriate length unit is selected, and the large area is divided into several small areas, as shown in Fig. 14:

Finally, suppose the drone starts at zero. The path is planned according to the dynamic programming method described in Fig. 6, and the two cases, *o-m* and *o-n*, are simulated. The shortest path obtained is shown in Fig. 15:

For the purpose of experimental simulation, the following ideal hypothesis is made:

1. Suppose the UAV is a particle.
2. Assume that the UAV is flying at a constant speed all the time.
3. Don't take into account the change of the position of obstacles and other factors.
4. For the moment, the flying birds and other living creatures enter the flying area suddenly in the UAV flight.

5.2.3 Data Display Simulation Result

To simplify the analysis of the system, there are ten nodes in the statistics Fig. 16 is the MATLAB path simulation

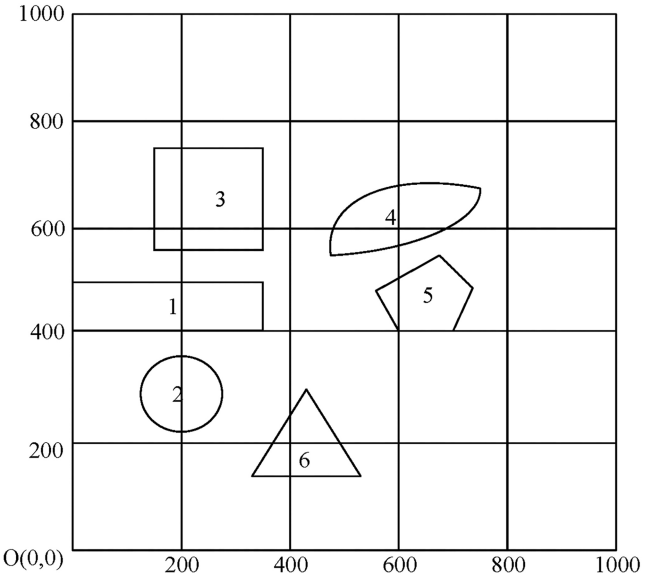


Figure 14. Schematic diagram of a small area.

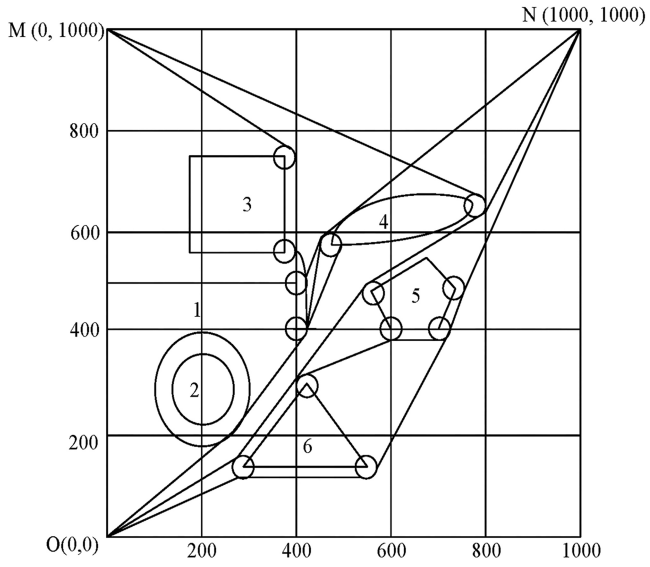


Figure 15. Simulation path planning diagram.

where the red indicating arrow is the best path. The connection node data between two points is shown in Table 1, and the value of the shortest distance data is shown in Table 2.

The results verified by simulation show the distance between nodes and are compared with different paths to calculate the shortest path. Under the condition that the number of nodes conforms to the hypothesis, the time of completing the whole dynamic programming process is about 0.008 s. The simulation results show that the proposed algorithm is fast and feasible.

6. Experimental Result

The experimental results are divided into two parts: UAV flight test and path planning.

Table 1
Connection Nodes Between Two Points

Nodes#	1	2	3	4	5	6	7	8	9	10
1	[1], [1]	[1], [2]	[1], [2], [3]	[1], [10], [4]	[1], [5]	[1], [6]	[1], [10], [7]	[1], [10], [4], [8]	[1], [9]	[1], [10]
2	[2], [3], [1]	[2], [2]	[2], [3]	[2], [4]	[2], [5]	[2], [4], [6]	[2], [7]	[2], [8]	[12,4,6,9]	[2], [10]
3	[3], [1]	[3], [1], [2]	[3], [3]	[3,1,10,41]	[3], [1], [5]	[3], [1], [6]	[3], [1], [10], [7]	[3], [1], [10], [4], [8]	[3], [1], [9]	[3], [1], [10]
4	[4], [6], [1]	[4], [6], [2]	[4], [6], [2], [3]	[4], [4]	[4], [6], [5]	[4], [6]	[4], [8], [7]	[4], [8]	[4], [6], [9]	[4], [6], [10]
5	[5], [6], [1]	[5], [6], [2]	[5,6,2,31]	[5], [4]	[5], [5]	[5], [6]	[15,4,8,7]	[5], [4], [8]	[5], [6], [9]	[5], [6], [10]
6	[6], [1]	[6], [2]	[6], [2], [3]	[6], [10], [4]	[6], [5]	[6], [6]	[6], [10], [7]	[6], [8]	[6], [9]	[6], [10]
7	[7], [1]	[7], [2]	[7], [2], [3]	[7], [1], [10], [4]	[7], [5]	[7], [8], [6]	[7], [7]	[7], [8]	[7], [1], [9]	[7], [1], [10]
8	[8], [7], [1]	[8], [6], [2]	[8], [3]	[8], [4]	[8], [7], [5]	[8], [6]	[8], [7]	[8], [8]	[8], [6], [9]	[8], [10]
9	[9], [10], [3], [1]	[9], [10], [4], [6], [2]	[9,10,31	[9], [10], [4]	[9], [7], [5]	[9], [10], [4], [6]	[9], [7]	[9], [10], [4], [8]	[9], [9]	[9], [10]
10	[10], [3], [1]	[10], [4], [6], [2]	[10,31]	[10], [4]	[10], [4], [6], [5]	[10], [4], [6]	[10], [7]	[10], [4], [8]	[10], [4], [6], [9]	[10], [10]

Table 2
The Shortest Distance

Nodes#	1	2	3	4	5	6	7	8	9	10
1	0	8	33	35	22	14	38	42	18	4
2	45	0	2	2	19	34	13	13	42	3
3	6	48	0	14	38	33	25	41	29	27
4	46	24	47	0	40	8	35	12	27	39
5	32	40	34	5	0	6	45	46	46	47
6	5	7	38	41	24	0	48	17	14	6
7	14	21	37	35	22	48	0	10	38	28
8	27	46	20	16	32	17	7	0	38	23
9	48	40	33	48	35	29	7	31	0	1
10	48	48	9	2	38	11	13	24	28	0

6.1 UAV Flight Test Result

We flew the drone in a relatively empty playground for safety reasons. Figure 17(a)–(g) refers to the whole process of logistics UAV from delivery to picking up by operators. Figure 17(a) shows the operator putting a small cargo into the UAV cargo box. Then, we connect the phone App to the drone successfully, click the "Faster" button, and the drone takes off. Figure 17(b) shows the UAV in the process of taking off. During the flight, the front steering engine continuously rotates ultrasonic waves to detect obstacles. If obstacles are not detected, the route will be planned according to part B. If obstacles are detected, as shown in Fig. 17(c), the UAV will hover in the air. Meanwhile, the mobile APP of the ground control terminal displayed "Warning: Suggest to turn left", as shown in Fig. 17(d). After receiving the obstacle alarm, the operator will send corresponding steering instructions according to the prompt, and the UAV will continue to fly, as shown in Fig. 17(e). The UAV flies over the express station and stops on the flat ground, as shown in Fig. 17(f). The worker opens the hatch, takes out the cargo, scans and registers the barcode on the cargo, and the drone completes the distribution task, as shown in Fig. 17(g). If the drone waits for the pick-up time is too long, as shown in Fig. 17(h), the drone will send abnormal pick-up information to the logistics system and take off and return to the delivery point, as shown in Fig. 17(i).

6.2 Path Planning

To verify the actual flying effect of the UAV, the site was selected for test fight in a scene with trees and houses. See if the drone can avoid trees and tall buildings. The flight process is shown in Fig. 18(a)–(h). the direction of arrow

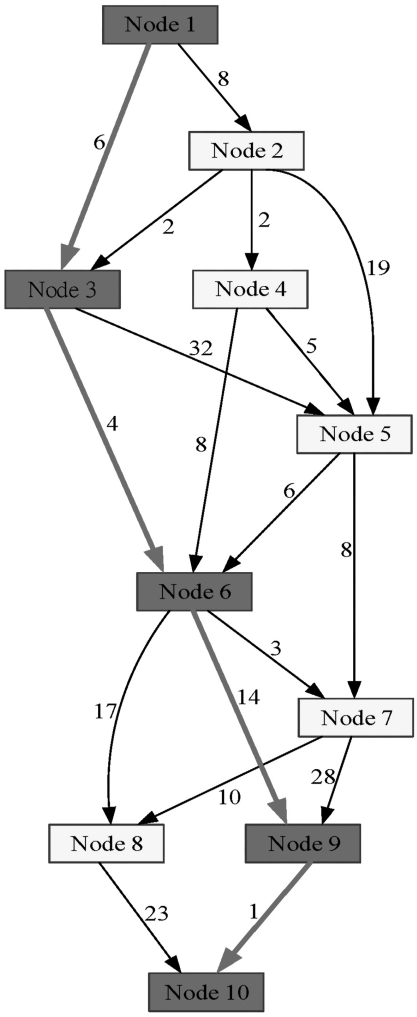


Figure 16. Path diagram of simulation experiment.

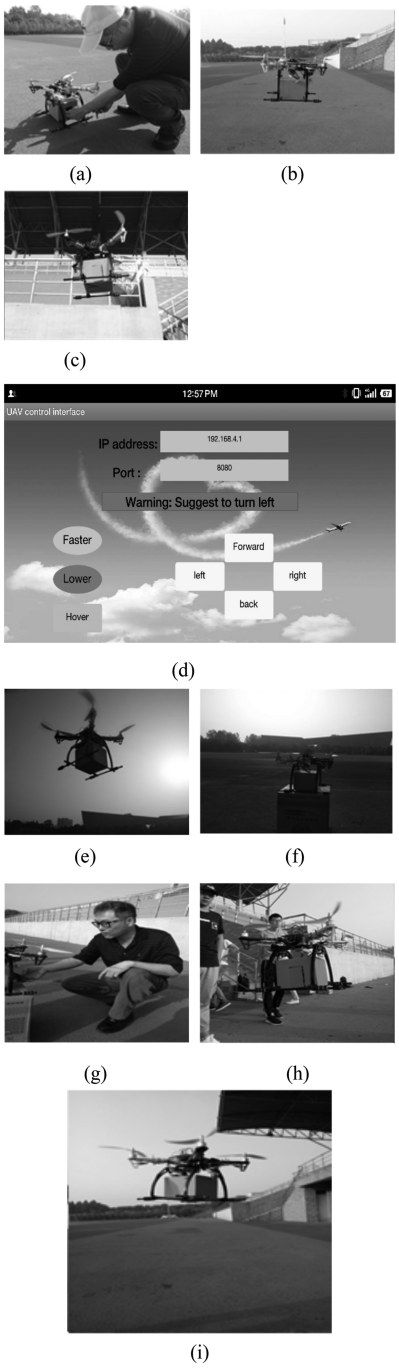


Figure 17. The process of logistics UAV delivering and pick-up.

shows the path of the UAV, and the position of the drone is encircled. Starting from Fig. 18(a), the UAV takes off from the ground, Fig. 18(b) is the UAV after lift-off. At this moment, there is a tree on the way. AS shown in Fig. 18(c), the drone avoided trees and made a successful detour. In Fig. 18(d), the UAV determines the optimal path to cross the road according to the obstacle density. The UAV then successfully avoids trees again, as shown in Fig. 18(e). After flying across the river, the UAV encountered buildings as it flew northwest as shown in Fig. 18(f). Ultrasonic waves detect obstacles and rise vertically over the roof, as shown in Fig. 18(g). Finally, the UAV avoids the building complex and enters the high altitude to avoid all obstacles as shown in Fig. 18(h), the fast flight is successful.

To improve the reliability of experimental results, we conducted a second experiment, as shown in Fig. 19. Similarly, the UAV takes off from the ground. In Fig. 19(a), the UAV does not affect pedestrians when flying. In Fig. 19(b) and 19(c), the UAV avoids trees twice through track planning, and in Fig. 19(d), the UAV encounters tall buildings and then takes off vertically, as shown in Fig. 19(e). In Fig. 19(f), the UAV overrides a tall building and the experiment is successful again.

7. Conclusion

According to the environmental adaptability of quadrotor logistics UAV, an improved flight control system of intelligent UAV is proposed, and the hardware design, attitude calculation, software simulation, and flight test of the system are carried out. The system uses a GPS positioning system and dynamic path planning based on an ant colony algorithm to transport the goods from the main express station to the sub-express stations in the community, which are received by the staff, temporarily store the goods and notify the consignee to pick up the goods at the express station. Therefore, the advantages

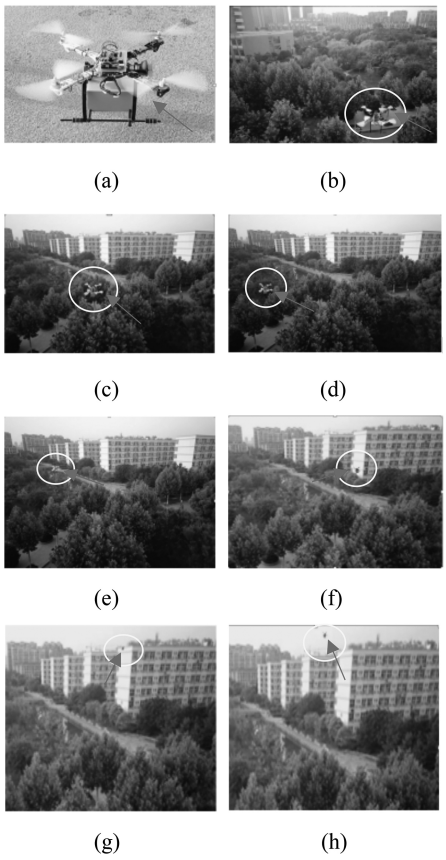


Figure 18. Reliability of the first flight path.

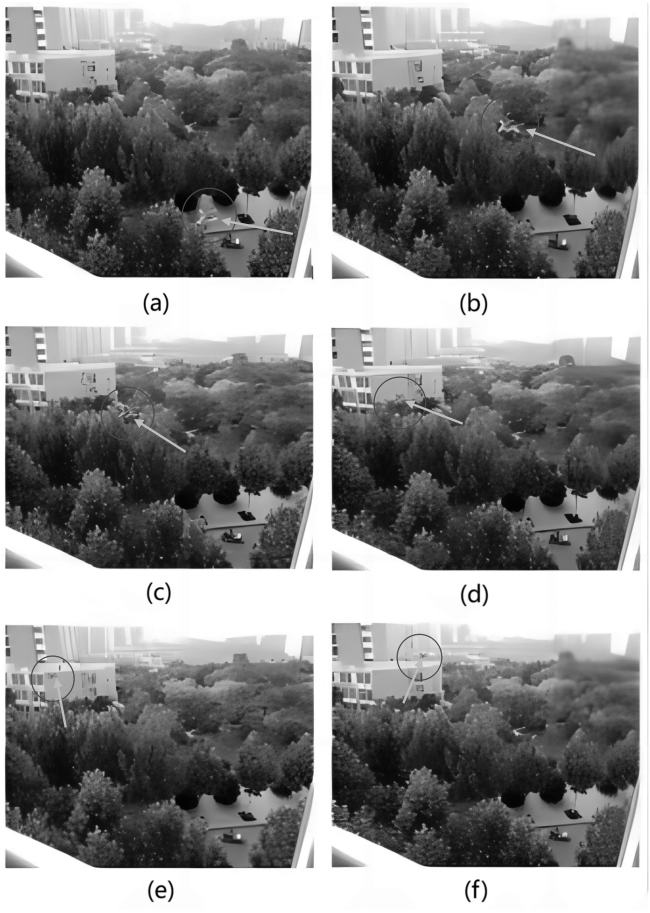


Figure 19. Reliability of the second flight path.

of this kind of logistics UAV mainly lie in solving the distribution problem of small packages, improving work efficiency, and reducing labour costs. In addition, the system also has UAV tracking function. When the battery power is too low, it is feed back to the control end through the communication system to track the position of the UAV and return in time, so as to improve the safety of the UAV. Experimental results show that the system is feasible.

Intelligent logistics UAV is used to replace the courier and plays an important role in logistics transportation. With the progress of science and technology, its most important result can reduce human costs, improve work efficiency, promote economic development, and can be widely used in the logistics industry in the near future.

References

[1] N.H. Motlagh, T. Taleb, and O. Arouk, Low-altitude unmanned aerial vehicles-based internet of things services: Comprehensive survey and future perspectives, *IEEE Internet Things J., 3*(6), 2016, 899–922.

[2] I. Bisio, C. Garibotto, and F. Lavagetto, A. Sciarrone, and S. Zappatore, Blind detection: Advanced techniques for WiFi-based drone surveillance, *IEEE Transactions on Vehicular Technology, 68*(1), 2019, 938–946.

[3] L. Zhu, D. Yin, L. Shen, X. Xiang, and G. Bai, Research on urban application-oriented route planning of UAV based on mobile communication network, *Proc. 2015 4th International Conf. on Computer Science and Network Technology (ICCSNT), Harbin*, 2015, 1562–1570

[4] X. Hu, H. Ma, Q. Ye, and H. Luo, Hierarchical method of task assignment for multiple cooperating UAV teams, *Journal of Systems Engineering and Electronics, 26*(5), 2015, 1000–1009.

[5] F. Zheng, F. Wang, J. Wu, and X. Zheng, A methodology of UAV route planning for fast image mosaicking, *Proc. 2015 23rd International Conf. on Geoinformatics*, Wuhan, 2015, 1–5.

[6] Z. He and L. Zhao, The comparison of four UAV path planning algorithms based on geometry search algorithm, *Proc. 2017 9th International Conf. on Intelligent Human-Machine Systems and Cybernetics (IHMSC),* Hangzhou, 2017, 33–36.

[7] J. Wang, Y. Sun, Z. Liu, P. Yang, and T. Lin, Route planning based on FLOYD algorithm for intelligence transportation system, *Proc. 2007 IEEE International Conf. on Integration Technology*, Shenzhen, 2007, 544–546.

[8] L. Yang, D. Li, and R. Tan, Research on the shortest path solution method of interval valued neutrosophic graphs based on the ant colony algorithm, *IEEE Access, 8*, 2020, 88717–88728.

[9] L. Wang and Y. Li, A multi-objective optimization method based on dimensionality reduction mapping for path planning of a HALE UAV, *Proc. 2019 Chinese Automation Congress (CAC),* Hangzhou, China, 2019, 3189–3194.

[10] Z. Lv, L. Yang, Y. He, Z. Liu, and Z. Han, 3D environment modeling with height dimension reduction and path planning for UAV, *Proc. 2017 9th International Conf. on Modelling, Identification and Control (ICMIC)*, Kunming, 2017, 734–739.

[11] J. Chen, M. Li, Z. Yuan, and Q. Gu, An improved A* algorithm for UAV path planning problems, *Proc. 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conf. (ITNEC),* Chongqing, China, 2020, 958–962.

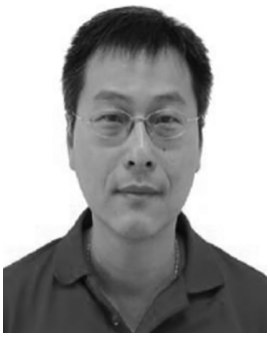
[12] D.M. Xavier, N.B.F. Silva, and K.R.L.J.C. Branco, Path-following algorithms comparison using Software-in-the-Loop simulations for UAVs, *Proc. 2019 IEEE Symposium on Computers and Communications (ISCC*), Barcelona, Spain, 2019, 1216–1221.

[13] Z. Yi, Y. Xiuxia, and Z. Weiwei, Flyable path planning for multiple UAVs in complicated threat environment, *Proc. 2014 International Conf. on Multisensor Fusion and Information Integration for Intelligent Systems (MFI),* Beijing, 2014, 1–5.

[14] G. Che, L. Liu, and Z. Yu, An improved ant colony optimization algorithm based on particle swarm optimization algorithm

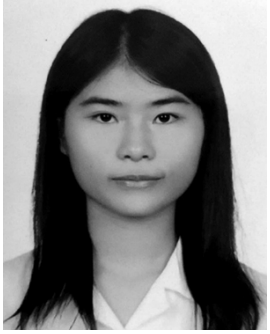
for path planning of autonomous underwater vehicle, *Journal of Ambient Intelligence and Humanized Computing, 11*,2020, 3349–3354.
[15] H. Duan and P. Li, *Bio-inspired computation in unmanned aerial vehicles* (Berlin, Germany: Springer-Verlag, 2014).
[16] H. Duan, P. Li, Y. Shi, X. Zhang, and C. Sun, Interactive learning environment for bio-inspired optimization algorithms for UAV path planning, *IEEE Transactions on Education, 58*(4), 2015, 276–281.
[17] W. He, T. Wang, X. He, L.-J. Yang, and O. Kaynak, Dynamical modeling and boundary vibration control of a rigid-flexible wing system, *IEEE/ASME Transactions on Mechatronics, 25*(6), 2020, 2711–2721.
[18] D. Zhang, X. You, S. Liu, and K. Yang, Multi-colony ant colony optimization based on generalized jaccard similarity recommendation strategy, *IEEE Access, 7*, 2019, 157303–157317.
[19] M.M. Alobaedy, A.A. Khalaf, and I.D. Muraina, Analysis of the number of ants in ant colony system algorithm, *Proc. 2017 5th International Conf. on Information and Communication Technology (ICoIC7),* Malacca City, 2017, 1–5.
[20] W. Zheng, X. Jin, F. Deng, S. Mao, Y. Qu, Y. Yang, X. Li, S. Long, C. Zheng, and Z. Xie, Database query optimization based on parallel ant colony algorithm, *Proc. 2018 IEEE 3rd International Conf. on Image, Vision and Computing (ICIVC),* Chongqing, 2018, 653–656.
[21] Q. Yang and S. Yoo, Optimal UAV path planning: sensing data acquisition over IoT sensor networks using multi-objective bio-inspired algorithms, *IEEE Access, 6*, 2018, 13671–13684.
[22] A. Mohamed, S. Watkins, R. Clothier, M. Abdulrahim, K. Massey, and R. Sabatini, Fixed-wing MAV attitude stability in atmospheric turbulence—Part 2: Investigating biologically-inspired sensors, *Progress in Aerospace Sciences, 71*, 2014, 1–13.
[23] M. Euston, P. Coote, R. Mahony, J. Kim, and T. Hamel, A complementary filter for attitude estimation of a fixed-wing UAV, *Proc. IEEE/RSJ International Conf. on Intelligent Robots and Systems (IROS),* Nice, 2008, 340–345.
[24] P. Poksawat, L. Wang, and A. Mohamed, Gain scheduled attitude control of fixed-wing UAV with automatic controller tuning, *IEEE Transactions on Control Systems Technology, 26*(4), 2018, 1192–1203.
[25] G. Fink, H. Xie, A.F. Lynch, and M. Jagersand, Nonlinear dynamic image-based visual servoing of a quadrotor, *Journal of Unmanned Vehicle Systems, 3*(1), 2015, 1–21.
[26] M. Burger and M. Guay, A backstepping approach to multivariable robust constraint satisfaction with application to a VTOL helicopter, *Proc. 48th IEEE. Decision Control*, Shanghai, China, 2009, 5239–5244.
[27] N. Cao and A.F. Lynch, Inner–outer loop control for quadrotor UAVs with input and state constraints, *IEEE Transactions on Control Systems Technology, 24*(5), 2016, 1797–1804.
[28] W. He, X. Mu, L. Zhang, and Y. Zou, Modeling and trajectory tracking control for flapping-wing micro aerial vehicles, *IEEE/CAA Journal of Automatica Sinica, 8*(1), 2021, 148–156.
[29] G.F. Che and Z. Yu, Neural-network estimators based fault-tolerant tracking control for AUV via ADP with rudders faults and ocean current disturbance, *Neurocomputing*, 411, 2020, 442–454.
[30] G.F. Che, Single critic network based fault-tolerant tracking control for underactuated AUV with actuator fault, *Ocean Engineering*, 254, 2022, 111380.
[31] Z. Chen, D. Yin, D. Chen, M. Pan, and J. Lai, Wifi-based UAV communication and monitoring system in regional inspection, *Proc. 2018 International Computers, Signals and Systems Conf. (ICOMSSC),* Dalian, China, 2018, 386–392.
[32] M. Itani, A. Haroun, and W. Fahs, "Obstacle avoidance for ultrasonic unmanned aerial vehicle monitoring using android application, *Proc. 2018 International Arab Conf. on Information Technology (ACIT),* Werdanye, Lebanon, 2018, 1–4.
[33] H.-W. Lee, Y. Zhu, X. Shi, F.-F. Peng, and W.-J. Jin, Research of four-axis aircraft using WIFI and rotary anti-collision system, *Proc. 2018 IEEE International Conf. on Applied System Invention (ICASI)*, Chiba, 2018, 665–668.
[34] H. Jing-Lin, S. Xiu-Xia, L. Ri, D. Xiong-Feng, and L. Mao-Long, UAV real-time route planning based on multi-optimized RRT algorithm, *Proc. 2017 29th Chinese Control and Decision Conf. (CCDC),* Chongqing, 2017, 837–842.
[35] H. Jun and Z. Qingbao, Multi-objective mobile robot path planning based on improved genetic algorithm, Proc. *2010 International Conf. on Intelligent Computation Technology and Automation*, Changsha, 2010, 752–756.

Biographies



Hai-Wu Lee (Member, IEEE) received the degree from the Department of Electronic Engineering, Kun Shan University, in 2000, the master's degree from the Institute of Computers, Communications, and Control, National Taipei University of Technology, in 2003, and the Ph.D. degree from the Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, in 2014. He is currently an AI Academic Leader and a Professor with the Department of School of Science and Engineering, Xiangsihu College of Guangxi University for Nationalities. His research interests include the design and application of optimal control systems for biped walking robots, image processing, and intelligence RFID. He is a Reviewer of journals, such as IEEE Transactions on Education and IEEE Access *etc.*

Shoaib Ahmed received the B.Sc. degree in electrical engineering and technology from the University of Engineering and Technology Lahore, Pakistan, in 2019, and the master's degree in electrical engineering from the School of Electronics, Electrical Engineering & Physics, Fujian University of Technology, China, in 2022. He is currently pursuing the Ph.D. degree with the Graduate School of Science and Engineering, University of the Ryukyus, 903-0213, Okinawa, Japan. His research interest is in image processing, machine learning, cyber-physical power systems (CPPS), and blockchain-based smart grid auction.



Chi-Shiuan Lee studying in the Department of Public Affairs at Fo Guang University, is currently an exchange student at university of the west in the United States. Her current research focuses on the interdisciplinary field of combining long term care services with artificial intelligence.