

Research Article

Real-Time Trajectory Planning and Control for Constrained UAV Based on Differential Flatness

Dongli Wu ¹, Hao Zhang,² Yunping Liu ², Weihua Fang,³ and Yan Wang²

¹CMA Meteorological Observation Center, Beijing 100081, China

²Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAET), Nanjing University of Information Science & Technology, Nanjing 210044, China

³Nanjing Institute of Hydraulic and Hydrology Automation, Ministry of Water Resources, China

Correspondence should be addressed to Dongli Wu; wudongli666@126.com and Yunping Liu; liuyunping@nuist.edu.cn

Received 7 January 2022; Revised 22 March 2022; Accepted 11 May 2022; Published 20 June 2022

Academic Editor: Hao Wen

Copyright © 2022 Dongli Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The trajectory planning of UAV with nonholonomic constraints is usually taken as differential algebraic equation to solve the optimal control problem of functional extremum under the condition of inequality constraints. However, it can be challenging to meet the requirements of real-time for the high complexity. A differential flat theory based on B-spline trajectory planning can replace the optimal control problem with nonlinear programming and be a good means to achieve the efficient trajectory planning of an UAV under multiple dynamic constraints. This research verifies the feasibility of this theory with actual flight experiments.

1. Introduction

Unmanned aerial vehicles (UAV) [1] have been widely used [2] in challenging operations due to its advantageous features such as strong maneuverability, small size, autonomous flight, and accurate positioning, which turned them into a promising tool in modern military and civilian scenarios [3]. With the continuous development of vision, the combination of the two technologies [4] also has broad prospects, and the closest contact between both fields is the track planning of a UAV [5]. The autonomous flight design of a UAV ultimately depends on its flight path planning [6]. The current planning problems are mainly divided into motion planning and trajectory planning. Among them, the latter refers to the implementation of point-to-point routes in a known environment. Trajectory planning usually combines the dynamic model [7] with the motion planning to solve the overall optimal control issue of the moving body, which leads to other problems, e.g., high complexity, complicated operation, and low computational efficiency, greatly increasing the difficulty of its application [8]. At present, the track planning algorithms are mainly divided into four cate-

gories: (i) the schematic method consisting of the general view method, the Voronoi diagram method, and the contour method [9–11]; (ii) the unit decomposition plan composed of the grid and the quadtree method [12]; (iii) the artificial potential field method consisting of the wave propagation method and the harmonic function method [13, 14]; (iv) the heuristic planning method based on the genetic algorithm and the neural network algorithm [15–17]. The above algorithms all consider the trajectory planning issues with various constraints and verify the feasibility of the methods through simulation. However, they are only suitable for experiments, but their practical feasibility has not been taken into consideration yet. In addition, UAVs also involve related control algorithms. The addition of the above planning algorithms will expand the scale of the problem, significantly increase the time needed, and reduce the planning efficiency, which goes in the opposite direction of the actual application requirements.

The differential flatness method can transform many classical nonlinear systems into flat systems for research, such as robot controlling and planning problems and tracking control problems [18] with nonholonomic

constraints [19]. In reference [20], the theory is applied to the trajectory planning of vehicle-like robots with nonholonomic constraints and compared with the planning method of the opportunity polynomial input. Similarly, reference [21] adopted the theory to design robot controllers in a dynamic environment. Reference [22] proposed the real-time trajectory generation based on differential flatness systems, satisfying state equations, paths, and actuator constraints.

With the development of control algorithms and image processing algorithms, the problem of UAV capturing and trajectory planning has received extensive attention in recent years. Ghadiok et al. built a slam map based on an infrared camera, but it was characterized by a lengthy process [23]. Spica et al. verified the feasibility of an UAV capturing moving objects with the features of the differential flatness theory [24]. Thomas et al. used the motion capture system to achieve visual-based capture with reference to the birds capture behavior, but the price of the motion capture system was too high for practical use [25]. Arputha and Dutta proposed a method for collaborative trajectory planning for multiple UAVs and conducted motion planning on the convex surface by building connections among them [26]. Zhang et al. adopted the ant colony algorithm to accurately estimate the heading angle deviation of a UAV [27]. Penin et al. proposed a minimum time trajectory planning method, using receding horizon control (RHC) to modify the reference trajectory online [28]. Although the above method proposes a path planning method suitable for UAVs, most of them cannot be applied to actual flight control, and many studies lack verification methods based on actual flight experiments. Some of them have achieved the goal of path planning, but with expensive techniques that cannot meet the actual application of promotion.

Aiming at the problems of high planning complexity and poor real-time performance, this study focuses on a UAV equipped with robot arms while taking into consideration the possible constraints of the flight. The first step is to establish the dynamic model of the most suitable system for this UAV, with its position and attitude being used as the flat output. The feasibility of the trajectory planning was proved by verifying the differential flatness theory, and its trajectory was smoothed by the B-spline [29]. We proposed the design of a controller combining the sliding mode and the PID control and verified the feasibility of the abovementioned schemes with proper experiments. Compared with the traditional UAV trajectory planning issue, our method reduces the complexity of planning under the premise of ensuring stability and is more suitable for practical embedded systems. Furthermore, it provides guarantee for high-precision planning actions such as drone capture.

2. Dynamic Model and Controller

2.1. Dynamic Model. Figure 1 illustrates a simplified model image of the UAV. In this paper, the three coordinate systems below are established—earth coordinate system $E -$

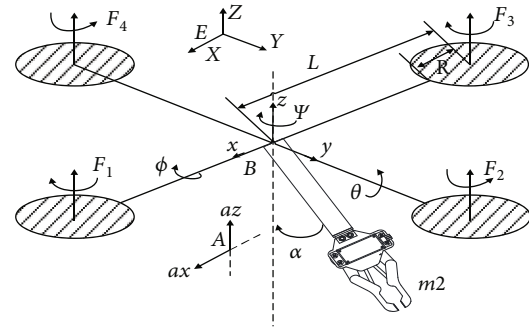


FIGURE 1: System structure diagram.

XYZ , quadrotor carrier coordinate system $B - xyz$, and robot arm coordinate system $A - ax, az$. It is also assumed that the robot arm can only rotate around the center in the plane $ax - az$. $\phi, \theta, \psi, \alpha$ represent the roll angle, pitch angle, attitude angle of the UAV, and the tilt angle of the arm from the axis az , respectively. B represents the centroid position of the quadrotor, and $F_i (i = 1, 2, 3, 4)$ provides the lift for the rotor. The detailed description of the symbols used in this paper is organized in Symbols.

The generalized coordinate vector $q = (\phi, \theta, \psi, x, y, z, \alpha)^T$ and the pseudovelocity vector $p = (\rho, q, r, u, v, w, b)^T$ are defined as outlined above. The pseudovelocity is a multibody system domain that relies on the generalized coordinate to describe the velocity of motion. Here, the velocity vector of the quadrotor in the body coordinate $B - xyz$ system is defined. The spatial position of the system (including the three corners and the displacement along the coordinates) is described by q . p defines the number of velocities of the system (including the three angular velocities and the displacement velocity along the coordinates). The dynamic model of the system is established by the Euler-Ponkale equation as follows:

$$\begin{cases} \dot{q} = V(q)p, \\ M(q)\dot{p} + C(q, p)p + F(p, q, u) = 0. \end{cases} \quad (1)$$

Vector $\eta = (\phi, \theta, \psi)^T$ represents the attitude of the flight system, and vector $X = (x, y, z)^T$ refers to the position of the system centroid in the Earth's coordinate system. $V(q)$ indicates the angular velocity at which the arm rotates, and ω represents the system's pseudovelocity and the generalized velocity attitude transformation matrix. $F(p, q, u)$ represents the sum of aerodynamics, gravity, and control inputs; u is the total system control input; and $M(q)$ is the inertial matrix of the system. $C(p, q)$ describes the gyro matrix of the system and can be defined in a simplified manner as $S_\phi = \sin \phi$, $S_\theta = \sin \theta$, $S_\alpha = \sin \alpha$, $C_\theta = \cos \theta$, $S_\psi = \sin \psi$, $C_\psi = \cos \psi$, and $C_\alpha = \cos \alpha$. The result of the final calculation is

$$F(p, q, u) = \begin{bmatrix} -LU_2 + gm_2RC_\alpha C_\theta S_\phi \\ -LU_3 + gm_2R(C_\theta C_\phi S_\alpha + C_\alpha S_\theta) \\ -LU_4 - gm_2RC_\theta S_\alpha S_\theta \\ -g(m_1 + m_2)S_\theta \\ g(m_1 + m_2)C_\theta S_\phi \\ -U_1 + g(m_1 + m_2)C_\theta C_\phi \\ -T + gm_2R(C_\theta C_\phi S_\alpha + C_\alpha S_\theta) \end{bmatrix}. \quad (2)$$

$$M(q) = \begin{bmatrix} I_x + m_2R^2C_\alpha^2 & 0 & -m_2R^2C_\alpha S_\alpha & 0 & m_2RC_\alpha & 0 & 0 \\ 0 & I_y + J_y + m_2R^2 & 0 & -m_2RC_\alpha & 0 & m_2RS_\alpha & J_y + m_2R^2 \\ -m_2R^2C_\alpha S_\alpha & 0 & I_z + m_2R^2S_\alpha^2 & 0 & m_2RS_\alpha & 0 & 0 \\ 0 & -m_2RC_\alpha & 0 & m_1 + m_2 & 0 & 0 & -m_2RC_\alpha \\ m_2RC_\alpha & 0 & -m_2RS_\alpha & 0 & m_1 + m_2 & 0 & 0 \\ 0 & m_2RS_\alpha & 0 & 0 & 0 & m_1 + m_2 & m_2RS_\alpha \\ 0 & J_y + m_2R^2 & 0 & -m_2RC_\alpha & 0 & m_2RS_\alpha & J_y + m_2R^2 \end{bmatrix}. \quad (3)$$

Among them, the control variables of lifting, rolling, pitching, and yaw are U_1, U_2, U_3 , and U_4 , respectively. T is the control variable of the tilt angle of the arm, and F_i ($i = 1, 2, 3, 4$) is the pulling force of each rotor. The area of the rotor is represented by $A = \pi R^2$, and the coefficient is $K = C_Q/C_T$, then

$$\left\{ \begin{array}{l} U_1 = F_1 + F_2 + F_3 + F_4 \\ U_2 = F_2 - F_4 \\ U_3 = F_1 - F_3 \\ U_4 = K(F_1 - F_2 + F_3 - F_4) \\ F = \frac{1}{2}\rho AC_T R^2 \Omega^2 \end{array} \right\}. \quad (4)$$

The total control input $u = [U_1, U_2, U_3, U_4, T]^T$ is set for the system, and when it experiences low air resistance at low speeds, the force can be ignored. If we assume that ϕ and θ in the flight process are considerably small, and its rate of change is also sufficiently small, it can be substituted into Equation (1) for simplification:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \quad (5)$$

The kinetic model of the system is simplified by Equation (5), and the formulas available are

$$\left\{ \begin{array}{l} \ddot{\phi} = \frac{2LI_z(m_1 + m_2) + m_1m_2R^2(-LU_2C_{2\alpha} + LU_4S_{2\alpha})}{A_5}, \\ \ddot{\theta} = \frac{-T + LU_3}{I_y}, \\ \ddot{\psi} = \frac{L\{2I_x(m_1 + m_2) + m_1m_2R^2\}U_4 + m_1m_2R^2(LR_4C_{2\alpha} + U_2S_{2\alpha})}{A_5}, \\ \ddot{x} = \frac{4m_2(m_1 + m_2)RTC_\alpha - 2m_2R\{2b^2J_y(m_1 + m_2) + 2b^2m_1m_2R^2 + 2m_2RU_1C_\alpha\}S_\alpha + A_4 * gS_\theta}{A_4}, \\ \ddot{y} = \frac{-2I_zLm_2RU_2C_\alpha + 2I_xLm_2RU_4S_\alpha - g[2I_xI_z(m_1 + m_2) + (I_x + I_z)m_1m_2R^2]C_\theta S_\phi + g(I_x - I_z)m_1m_2R^2C_{2\alpha}C_\theta S_\phi}{A_5}, \\ \ddot{z} = \frac{A_1 + A_2 * U_1 - A_3 * T}{A_4}, \\ \ddot{\alpha} = \frac{[(I_y + J_y)(m_1 + m_2) + m_1m_2R^2]T - L\{J_y(m_1 + m_2) + m_1m_2R^2\}U_3 - I_y m_2 R U_1 S_\alpha}{I_y J_y (m_1 + m_2) + I_y m_1 m_2 R^2}. \end{array} \right. \quad (6)$$

The above letters are defined as follows:

$$\begin{cases} A_1 = -4g(m_1 + m_2) [J_y(m_1 + m_2) + m_1 m_2 R^2] C_\theta C_\phi - m_2 R [4\dot{\alpha}^2 J_y(m_1 + m_2) + 4\dot{\alpha}^2 m_1 m_2 R^2] C_\alpha, \\ A_2 = 2 [2J_y(m_1 + m_2) + m_2 R^2 (2m_1 + m_2)] - 2m_2^2 R^2 C_{2\alpha}, \\ A_3 = 4m_2(m_1 + m_2) R S_\alpha, \\ A_4 = 4(m_1 + m_2) [J_y(m_1 + m_2) + m_1 m_2 R^2], \\ A_5 = 2I_x I_y (m_1 + m_2) + 2m_1 m_2 R^2 (I_z C_\alpha^2 + I_x S_\alpha^2). \end{cases} \quad (7)$$

2.2. Controller Design. Our research adopts the differential trajectory planning method based on differential flatness and the sliding mode PID control (combined control of sliding mode control and traditional PID control) [30] to enhance the anti-interference ability and response speed of the system [31]. This methodology was designed based on the control requirements of the UAV. The methods can be applied to the actual control, and the shortest sampling period of the clock is set to 0.02 s, that is, the position of the next desired point of the UAV is determined every 0.02 s. Once the control requirements are implemented, the specific control flow is represented in Figure 2.

The position control of the quadrotor can be divided into horizontal position and height control. Since the system uses four inputs to control six degrees of freedom, it is considered an underactuated and strongly coupled nonlinear system. The system is then decomposed into full a drive module and an underdrive module for easy analysis. We have established the mathematical relationship between the attitude angle, the position, and height control U_1 to achieve the purpose of trajectory tracking by continuously adjusting the attitude angle and the height. Both the horizontal position and the pull-down tilt angle are controlled using PID. The relationship between the desired position $(x_d y_d)$ and the desired acceleration $(\ddot{x}_d \ddot{y}_d)$, the desired tilt angle α_d , the angular acceleration of the robot arm $\ddot{\alpha}$, and the control rates of x, t, T are designed as follows:

$$\begin{cases} \ddot{x}_d = k_{p1}(x_d - x) + k_{i1} \int (x_d - x) + k_{d1}(\dot{x}_d - \dot{x})d_x, \\ \ddot{y}_d = k_{p2}(y_d - y) + k_{i2} \int (y_d - y) + k_{d2}(\dot{y}_d - \dot{y})d_y, \\ \ddot{\alpha}_d = k_{p3}(\alpha_d - \alpha) + k_{i3} \int (\alpha_d - \alpha) + k_{d3}(\dot{\alpha}_d - \dot{\alpha})d_\alpha. \end{cases} \quad (8)$$

The height control and attitude angle control are proposed using the sliding mode PID algorithm. The design of the sliding mode PID controller includes the selection of sliding mode surface and control law design. The surface is generally selected as $s = 0$, and the control objective of the sliding mode PID is designed to make the surface function finally stable, achieving a stable control. Taking the height

as an example of control, Z_d, Z indicates the desired and actual height, and two errors are calculated as

$$e_z = Z_d - Z. \quad (9)$$

Then, the next step is to design the sliding surface function:

$$S_z = \dot{e}_z + d_1 * e_z + c_1 * \int e_z d_t. \quad (10)$$

To ensure that the system can quickly converge to the sliding mode for large values of S . After that, the exponential convergence law rate is chosen to derive U_1 :

$$\dot{S}_z = -\varepsilon \operatorname{sgn}(S_z) - k S_z. \quad (11)$$

In the above formula, ε takes the value 0.01, k takes the value 3, and $\operatorname{sgn}(s)$ denotes the sign function. The Lyapunov function is used to determine the stability of the system, and its positive definite function is designed as follows:

$$V_z = \frac{1}{2} S_z^2. \quad (12)$$

Then, its derivative along the track line is

$$\dot{V}_z = S_z \dot{S}_z = S_z [-\varepsilon \operatorname{sgn}(S_z) - k S_z] = -\varepsilon \|S_z\| - k \|S_z\|^2 \leq 0. \quad (13)$$

Therefore, the kinetic model satisfies the stability criterion of Lyapunov asymptotic stability and is considered an asymptotically stable system, i.e., the error function will slowly converge to 0 over time and can eventually arrive at the sliding mode.

A combination of sliding mode control and PID control is adopted to enhance the anti-jamming and fast response characteristics of the UAV. By combining the control laws of both modes, the sliding mode PID control in this section is obtained, and the stability and robustness of the quadrotor control can be greatly improved by combining the advantages of each control. The position control of the quadrotor can be divided into horizontal position and altitude control.

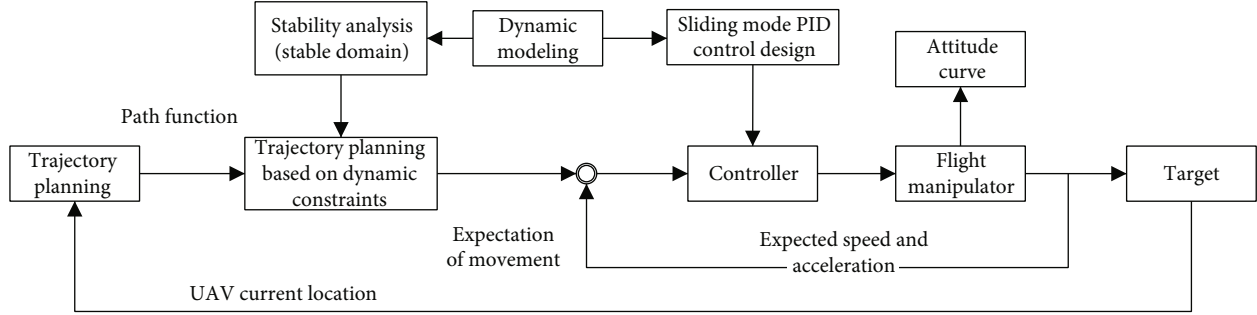


FIGURE 2: Control process.

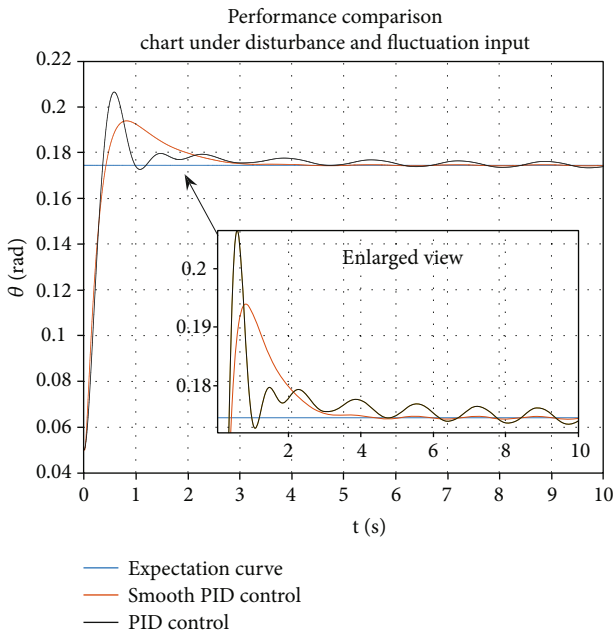


FIGURE 3: Performance comparisons under disturbance and fluctuation control inputs.

Since the system uses four inputs to control six degrees of freedom, it is an under-driven, strongly coupled nonlinear system. It is necessary to establish the mathematical relationship between attitude angle and position and height control quantities to achieve the purpose of trajectory tracking by continuously adjusting both.

Furthermore, to test the performance of the sliding mode PID controller, we first enable the two controls to quickly reach a steady state and stay under a stable input and without disturbance. Then, the control input is converted into a sinusoidal wave input, and complex disturbances are added to it. This process shows that significant control differences exist in traditional PID. It was proved that the sliding mode PID control has strong anti-interference and fast response capability, and Figure 3 is the result of a comparison between the two controllers under disturbance and fluctuation control inputs.

2.3. Robotic Arm Design. The design of the UAVs robot arm contains three servos. Their rotation direction is around the

y -axis forward and backward and around the height direction of the z -axis rotation, and the last servo is responsible for clamping. The control of the robot arm is done through manual grasping, that is, by receiving commands from the remote control to achieve grasping. This means that when the UAV can be moved to the target area by manually controlling the mechanical claw. Figure 4 shows the design of the experiment and the printed product made of ABS (acrylonitrile butadiene styrene plastic). The control of the robot arm is controlled by an MCU that generates PWM waves with a common frequency that ranges between 0.5 ms and 2.5 ms. The time difference between adjacent pulses is 2 ms. The width of the pulse varies by changing its duty cycle to achieve various angular rotations of the servo, and its operating voltage is usually 5 V.

3. Trajectory Planning

UAV trajectory planning with constraints is generally used as an optimal control problem for solving the generalized polar values of differential algebraic equations under inequality constraints. The direct solutions of these equations are normally challenging and inefficient, and their stability is difficult to analyze. To address these issues, this chapter introduces the differential flatness theory based on B-sample trajectory planning to transform the optimal control problem into nonlinear planning. If the solution is successful, we can achieve convenient and real-time planning of the UAV's trajectory under kinetic constraints, e.g., position, velocity, and acceleration.

3.1. Differential Flatness Applied to Trajectory Planning. The definition of differential flatness for nonlinear systems relies on the idea that if the spatial state quantities and control input variables of a system can be expressed using finite order differentials of the system output, then, the nonlinear system with the above characteristics has differential flatness properties.

If we define a nonlinear system $\dot{x} = f(x, u)$, $x \in R^n$, $u \in R^m$ with input u of the system without feedback and define the flat output as $y = h(x, u, \dot{u}, \dots, u^{(r)})$, the trajectory function ($x(t), u(t)$) can be set to be an equation with respect to the flat output y and its differential $x = \phi(y, \dot{y}, \dots, y^{(q)})$, $u = \alpha(y, \dot{y}, \dots, y^{(q)})$. The trajectory planning problem contains a starting

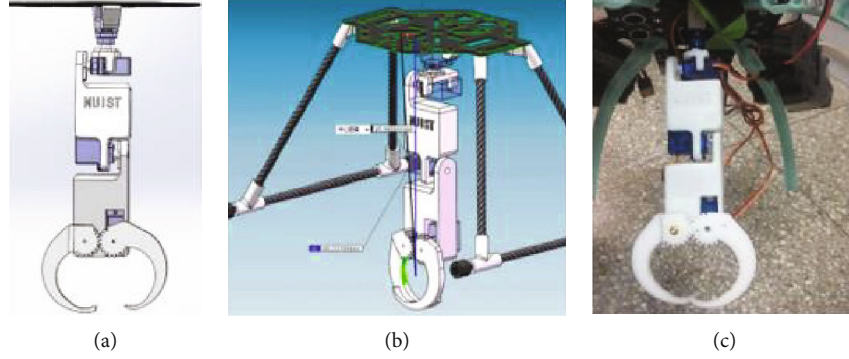


FIGURE 4: UAV robot arm design. (a) Design drawings. (b) Installation schematic. (c) Physical picture.

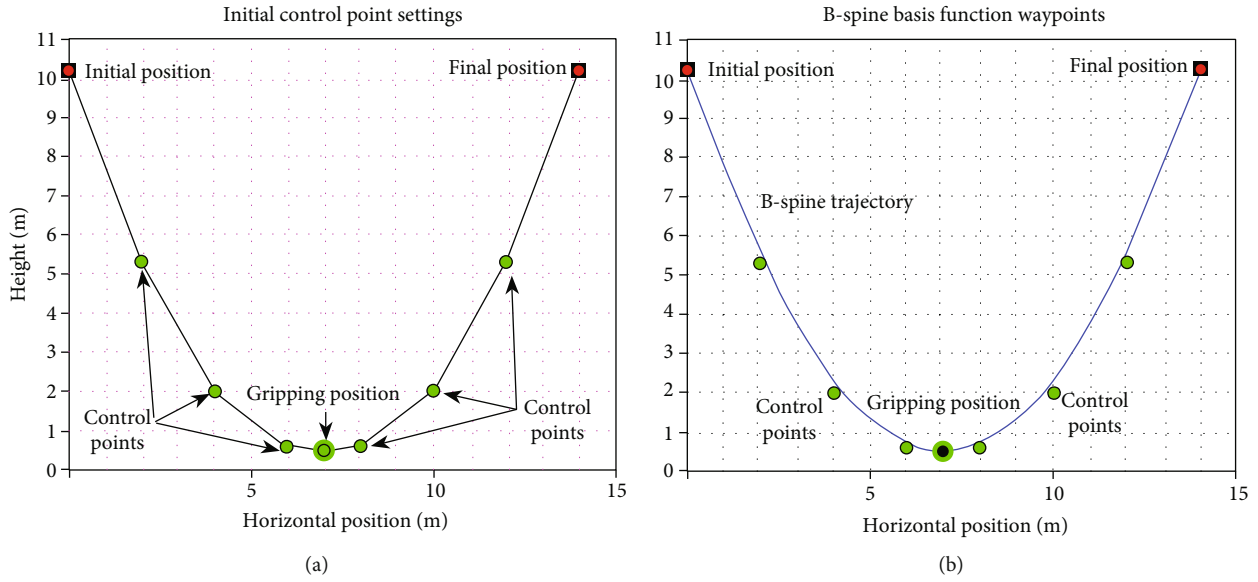


FIGURE 5: Location of the initial control node. (a) Initial control points choice. (b) B-spline basis function waypoints.

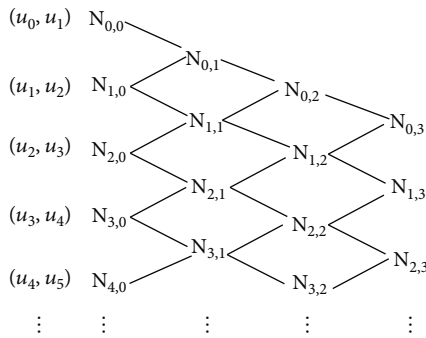


FIGURE 6: B-spline basis function subnode calculation.

where $\lambda_j, j = 1, 2, \dots, N$ is a basis function, and the problem of defining the flat output element y_i is equivalent to finding its spatial prediction in the space composed of the basis functions λ_i . Assuming that the state quantity at the starting moment of the trajectory t_0 is x_0 and the state quantity at the termination moment of t_f is x_f , then, the coefficients A_{ij} should satisfy the following conditions:

$$\begin{aligned}
 y_i(t_0) &= \sum_j A_{ij} \lambda_i(t_0) & y_i(t_f) &= \sum_j A_{ij} \lambda_i(t_f) \\
 \dots & & \dots & \\
 y_i^{(q)}(t_0) &= \sum_j A_{ij} \lambda_j^{(q)}(t_0) & y_i^{(q)}(t_f) &= \sum_j A_{ij} \lambda_j^{(q)}(t_f)
 \end{aligned}
 \tag{15}$$

state and an arrival state, and the flat output of the system is assumed to be a vector containing y_i , so

$$y_i = \sum_j A_{ij} \lambda_j(t), \tag{14}$$

Provided that generality is guaranteed and assuming the dimension $i = 1$ of the planar output vector, $y = [y_1]$ (these

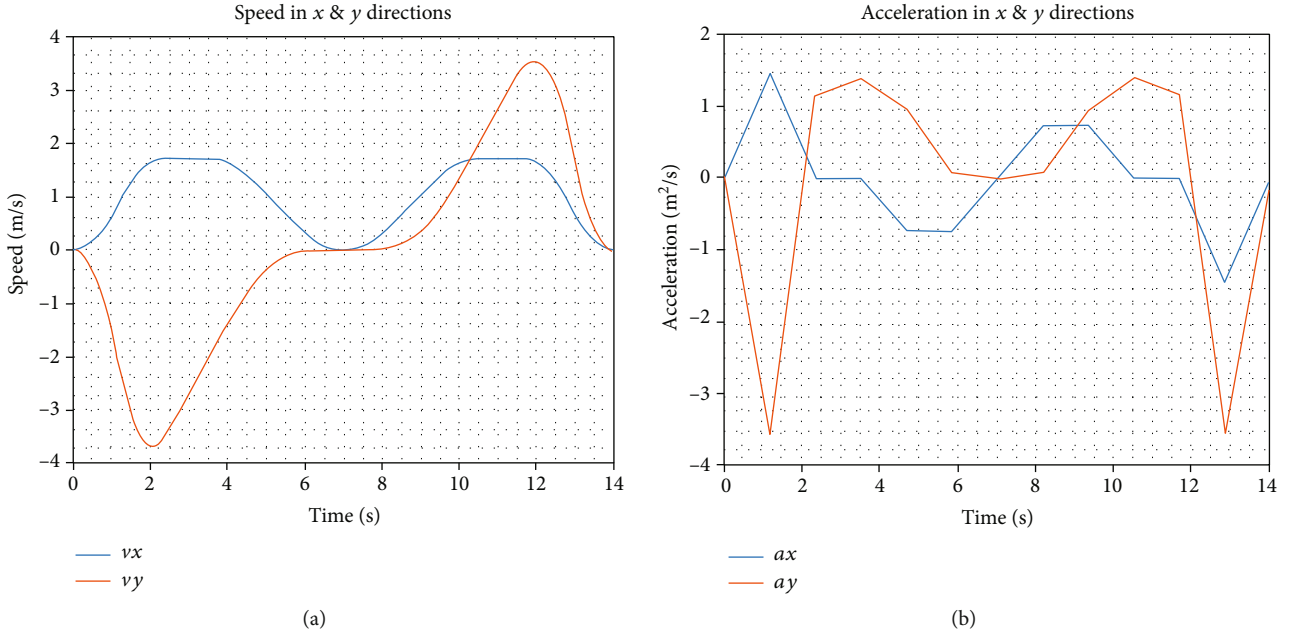


FIGURE 7: B-spline generated basis function path and speed curve. (a) Speed. (b) Acceleration.

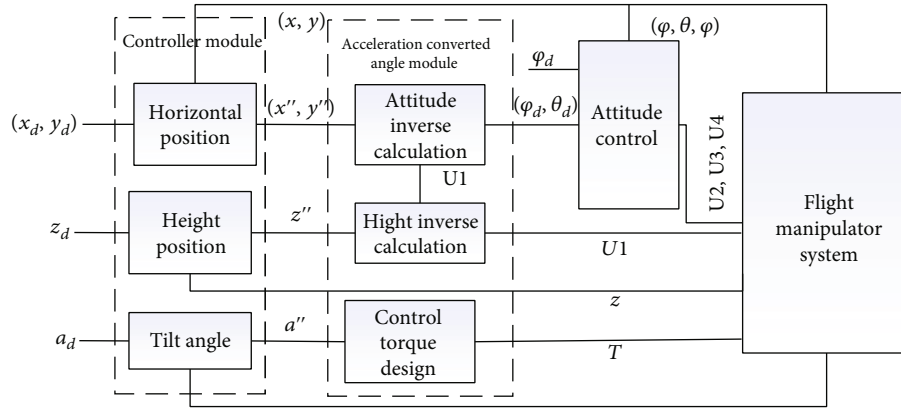


FIGURE 8: Control flow diagram.

results can also be generalized to the multidimensional case), the other variables are defined as follows:

$$\begin{aligned} \tilde{y}_0 &= (y_1(\tau_0), \dots, y_1^{(q)}(\tau_0)), \\ \tilde{y}_f &= (y_1(\tau_f), \dots, y_1^{(q)}(\tau_f)), \\ \tilde{y} &= (\tilde{y}_0, \tilde{y}_f). \end{aligned} \quad (16)$$

Combining equations A and B yields

$$y = \begin{pmatrix} \Lambda(\tau_0) \\ \Lambda(\tau_i) \end{pmatrix} A = \Lambda A, \quad (17)$$

where Λ is a full rank matrix, which means that the space equation λ_j must be an absolute full rank. This shows that the path planning problem for differential flat systems is feasible according to the linear algebraic theory.

3.2. The Cubic B-Spline Curve. The aim of trajectory planning is to achieve gripping. The first step is to set the initial position, gripping position, and final position. Second, a few control points are set for the UAV's obstacle avoidance paths, velocity, and acceleration constraints. Finally, cubic B spline smoothing was carried out on the track.

The cubic B-spline curve generates a path segment for each of the four control points, and then it splices each segment path to obtain a smooth global planning path. A series of smooth nodes can be obtained through the derivation of the finite control points. As shown in Figure 5(b), the plane coordinate system of the UAV's flight is established for convenient analysis, and the initial position of the vehicle is

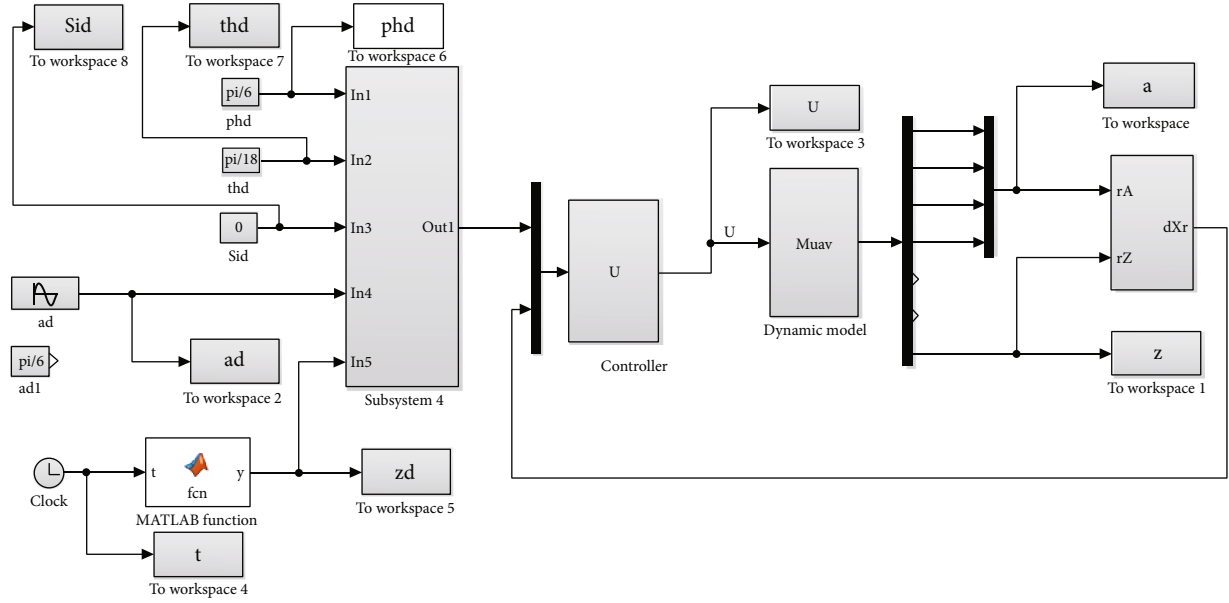


FIGURE 9: Simulation design of UAV with robotic arm.

assumed to be (0, 10), the gripping position (7, 0.5), and the final position (14, 10). According to the data obtained, the maximum descent speed of four-axis rotors is 4 m/s, the maximum rise speed is 10 m/s, and the maximum acceleration is 5 m/s². In addition, we set six control points to constrain the path selection, speed, and acceleration due to the presence of obstacles and other influences during flight. Equation (19) is the constraint condition of the UAV flight, (x_i, y_i) . It indicates the coordinates of the UAV at time i , then sets the shortest path as its penalty function, d_i pointing out the distance from the obstacle. L represents the length of the UAV arm, and v_i and \dot{v}_i represent the speed and acceleration of the vehicle.

$$\begin{cases} \min f(i) = \sum_{i-T}^t \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \\ \text{s.t. } d_i < \sqrt{(x_i - x_o)^2 + (y_i - y_o)^2} + L \\ v_i < 4, \dot{v}_i < 5 \\ i = 0, 1, \dots, t. \end{cases} \quad (18)$$

The coordinates are (2, 5.3), (4, 2), (6, 0.6), (8, 0.6), (10, 2), and (12, 5.3). Figure 5(a) illustrated the initial path of the selected initial control node and its description.

Through the six control points designed above, it is possible to effectively limit the overall speed of the UAV's movement, but due to the large position difference between the points, the vehicle will appear in various processes during point-to-point flight. To enable the UAV to fly stably and smoothly and to consider different deviations, we defined the cubic B-spline basis function to refine the path between the points and the vehicle according to the function. The

task is divided into more subtle phase control objectives, and the basis function of the B-spline is defined as

$$\begin{cases} N_{i,0}(u) = \begin{cases} 0, & u_i \leq u < u_{i+1}, \\ 1, & \text{otherwise,} \end{cases} \\ N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+p}} N_{i+1,p-1}(u), \end{cases} \quad (19)$$

where p represents the degree of basis functions, u_i is the node, and $N_{i,p}(u)$ represents i -th B-spline basis function of degree p . The cubic B-spline function nodes between the step length are set as 0.01, and then the location of the basis functions can be obtained according to the function between adjacent nodes. The specific way is derived as shown in Figure 6, and the basis function location point is the change over time in the process of UAV control at the desired location.

All the desired nodes of UAV movement can be calculated according to the cubic B-spline basis function described above. As shown in Figure 5(b), these are connected to form the desired trajectory of the UAV, and each of the subnode is a subtarget point of the vehicle during the cycle. Figure 7 shows the velocity and acceleration of the planned path and proves that the constraints of speed and acceleration are properly met.

4. Sliding Mode PID Controller Design

The sliding mode PID control presented in this section is obtained by combining the PID control and the sliding mode control laws. The stability and robustness of the

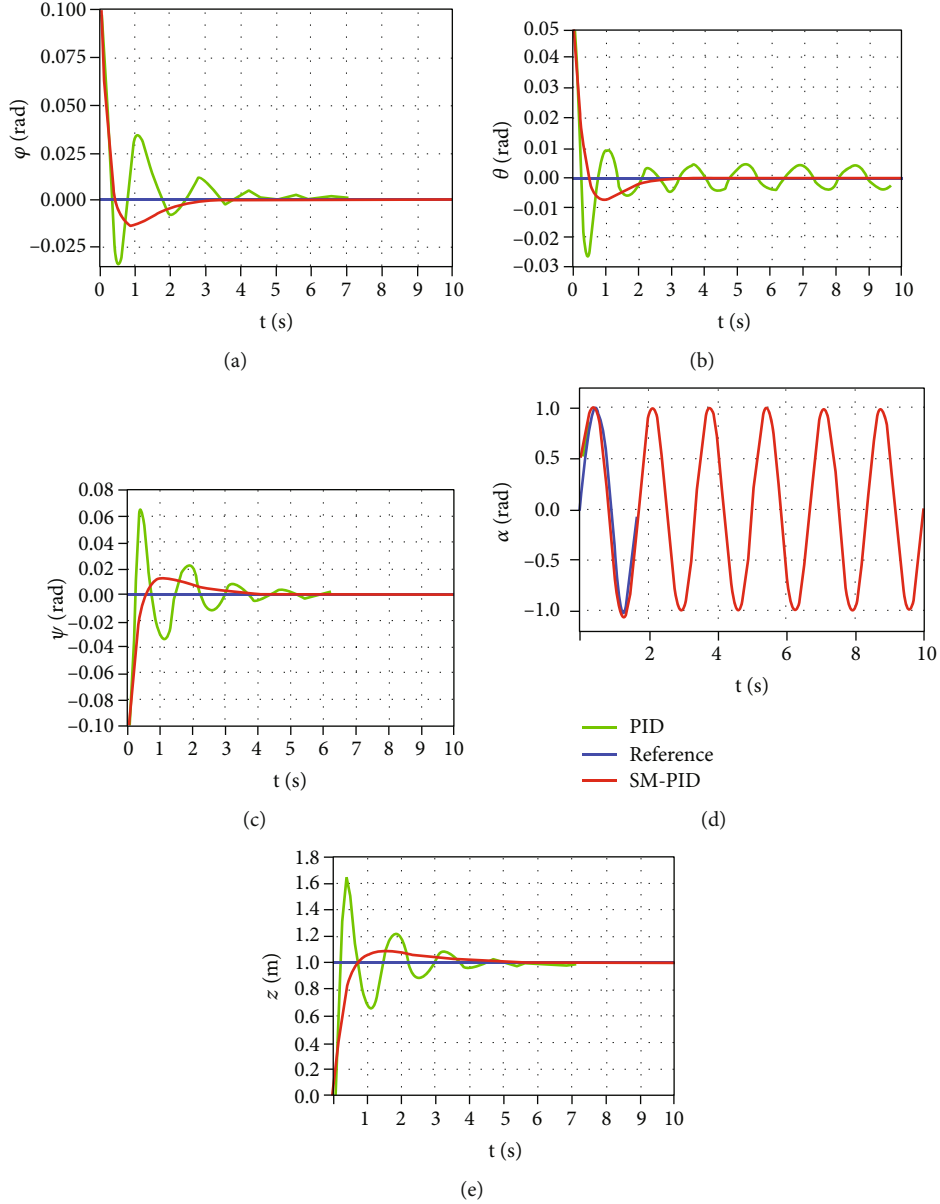


FIGURE 10: SM-PID control and attitude profile under PID control. (a) Rolling angle. (b) Pitch angle. (c) Yaw angle. (d) Pull down tilt angle. (e) Altitude change.

quadrotor control can be greatly improved by combining the advantages of each control. The position control of the quadrotor can be divided into horizontal position and altitude control. Since the system uses four inputs to control six degrees of freedom, it is an under-driven, strongly coupled nonlinear system. We have established the mathematical relationship between attitude angle and position and height control quantities to achieve the purpose of trajectory tracking by continuously adjusting attitude angle and position height. Figure 8 illustrates the control flow diagram of the sliding mode PID control system.

The horizontal position and the pull-down tilt angle are controlled by PID, and the control laws of x, y, α are designed by the relationship between the desired position

(x_d, y_d) and the desired acceleration (\ddot{x}_d, \ddot{y}_d) , the desired tilt angle α_d , and the angular acceleration $\ddot{\alpha}$ of the robot arm:

$$\begin{aligned}
 \ddot{x} &= k_{p_1}(x_d - x) + k_{i_1} \int (x_d - x) dt + k_{d_1}(\dot{x}_d - \dot{x}), \\
 \ddot{y} &= k_{p_2}(y_d - y) + k_{i_2} \int (y_d - y) dt + k_{d_2}(\dot{y}_d - \dot{y}), \\
 \ddot{\alpha} &= k_{p_3}(\alpha_d - \alpha) + k_{i_3} \int (\alpha_d - \alpha) dt + k_{d_3}(\dot{\alpha}_d - \dot{\alpha}).
 \end{aligned} \tag{20}$$

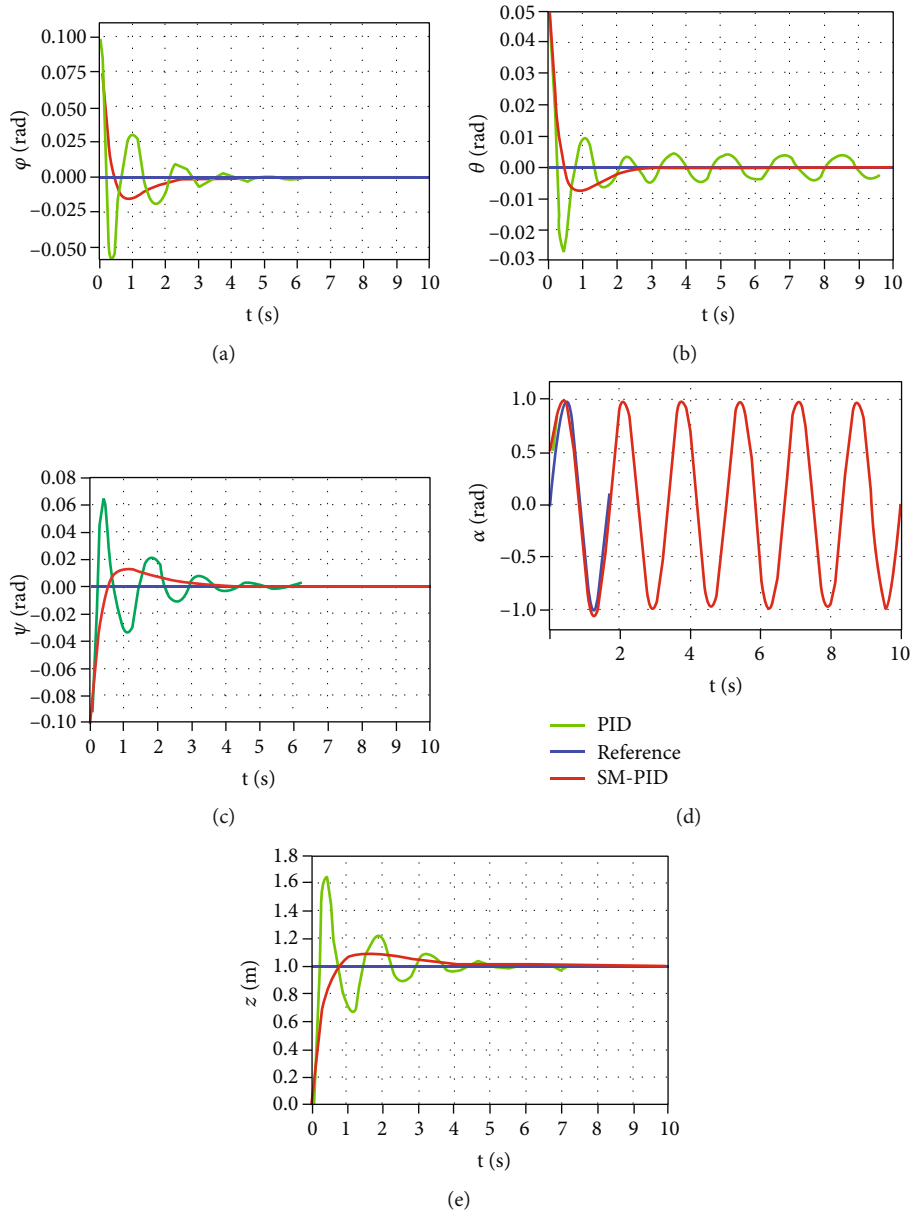


FIGURE 11: Attitude curves under two types of control under sinusoidal motion. (a) Rolling angle. (b) Pitch angle. (c) Yaw angle. (d) Pull down tilt angle. (e) Altitude change.

The design of the sliding mode PID controller includes the selection of the sliding mode surface and the design of the control law. The sliding mode surface is generally selected as $s=0$. Then, the control goal of the sliding mode PID is outlined to make the function of the sliding mode surface finally stable near the sliding mode surface, finally achieving stable control.

We have used the sliding mode PID algorithm to control the height design, and z_d, z indicate the desired and actual height. So, we can define the error between them as

$$e_z = z_d - z. \quad (21)$$

The slip surface function is designed as

$$s_z = \dot{e}_z + d_1 * e_z + c_1 * \int e_z dt. \quad (22)$$

To ensure that the system converges to the sliding mode at high speed for large values of s , the exponential convergence rate is chosen to derive U_1 :

$$\dot{s}_z = -\varepsilon \operatorname{sgn}(s_z) - ks_z, \quad (23)$$

where s takes the value of 0.01, k takes the value of 3, and $\operatorname{sgn}(s)$ denotes the symbolic function. The Lyapunov

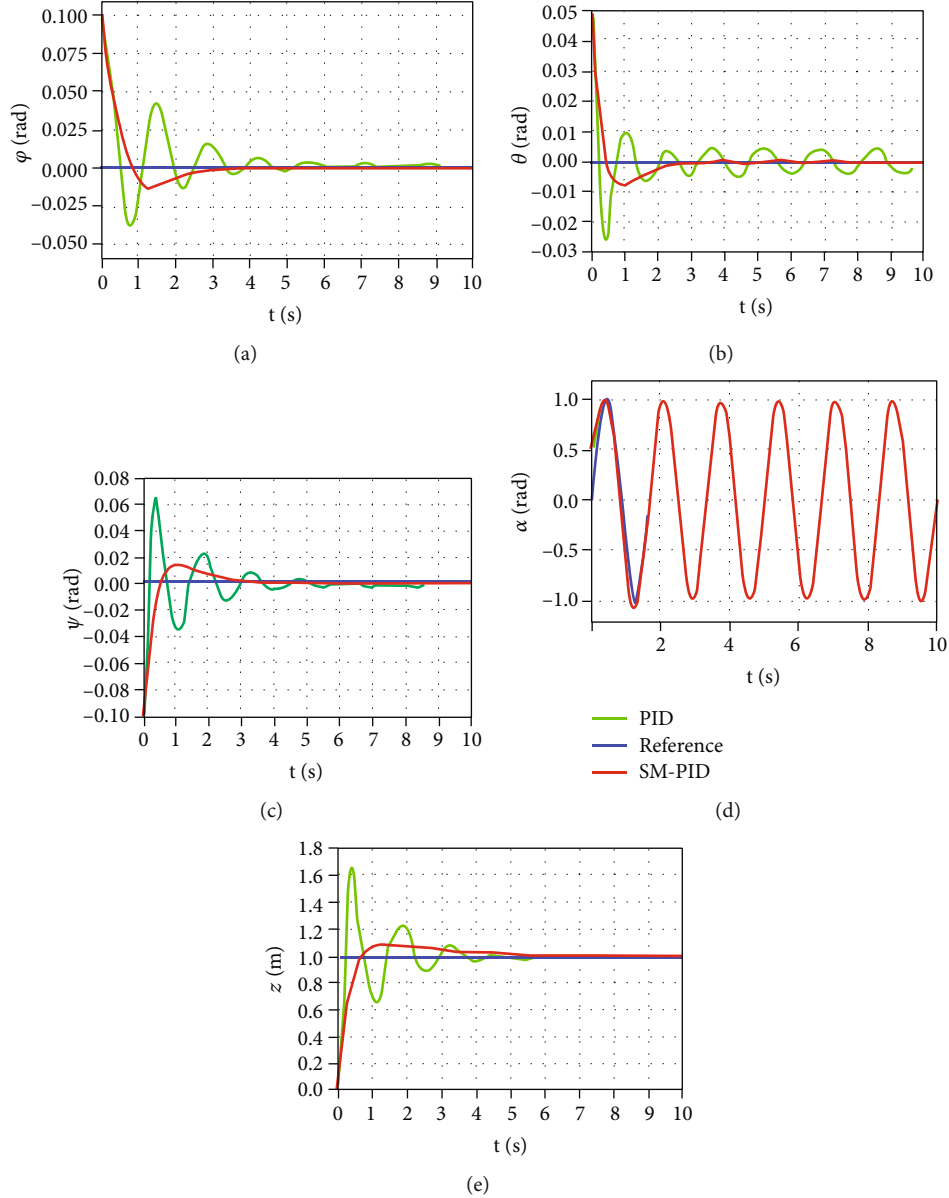


FIGURE 12: Attitude profiles for two controls under sinusoidal motion and perturbation. (a) Rolling angle. (b) Pitch angle. (c) Yaw angle. (d) Pull down tilt angle. (e) Altitude change.

function is used to determine the stability of the system, and its positive definite function is designed:

$$V_z = \frac{1}{2}s_z^2. \quad (24)$$

Then, its derivative along the track line is

$$\dot{V}_z = s_z \dot{s}_z = s_z [-\varepsilon \operatorname{sgn}(s_z) - ks_z] = -\varepsilon \|s_z\| - k \|s_z\|^2 < 0. \quad (25)$$

Therefore, the kinetic model satisfies the stability criterion of Lyapunov asymptotic stability and is an asymptotically stable system. This means that the error function will slowly converge to 0 over time and can eventually arrive at the sliding mode.

5. Experimental Verification

5.1. Theoretical Simulation. The aforementioned steps can be used to obtain the position of the UAV at different time periods. Then, these input points are used as control inputs of the simulation, which is designed according to the control flow in Figure 8. Simulink was used to carry out simulation analysis of the system, and two control modules (controller module U and power module Muav) were built with s-function. The simulation structure diagram is shown in Figure 9. The sequence of points obtained in Section 3 is set as a continuous input that changes with time, which means that the trajectory planning based on the segmentation control is achieved by giving different desired targets to the UAV in different time periods.

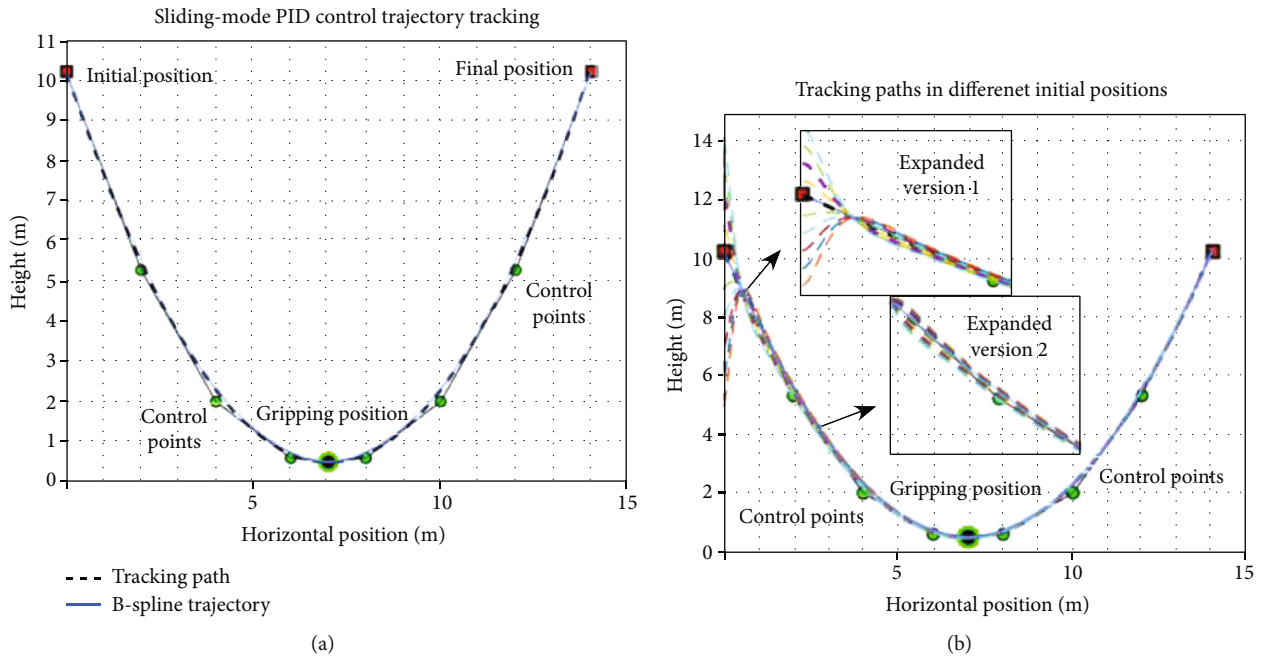


FIGURE 13: Influence of initial position on tracking effect. (a) Trajectory tracking control. (b) Tracking trajectories at different initial positions.

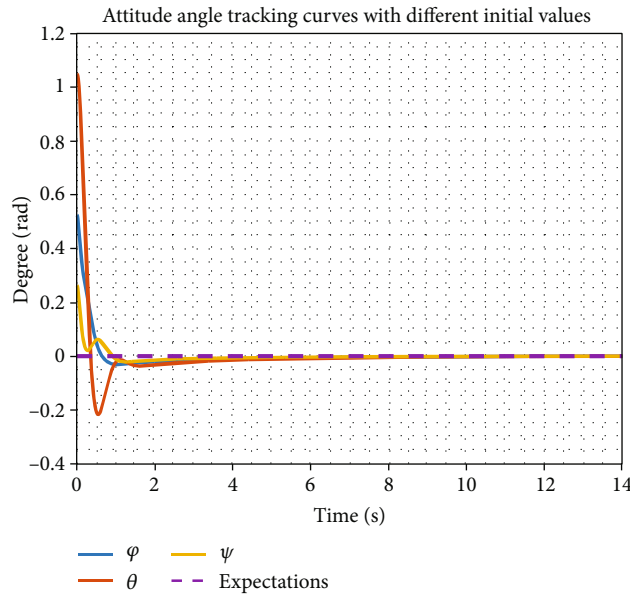


FIGURE 14: Attitude angle curve.

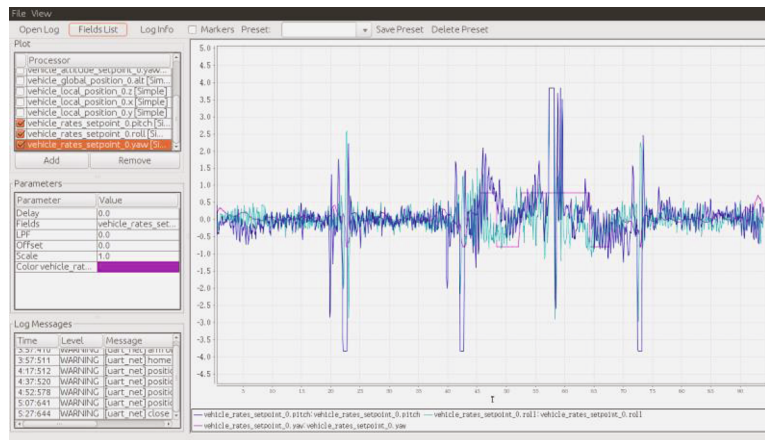
As shown in Figure 9, simulation through Simulink can get the data of each position and attitude angle of the quadrotor. The primary purpose of this research is to study and analyze its stability, so the specific analysis of the attitude angle and the altitude at the following figure is based on the above input values and expected values using sliding mode PID control and PID control. Both are used to obtain the UAV's attitude curve.

As shown in Figure 10, the simulation shows that in the sliding mode PID control, the adjustment time of pitch and roll angle is about 3.2s, and the adjustment time of yaw

angle is 4s, which can converge to the desired value after a reduced adjustment time. The tilt angle of the pull-down robot arm converges to the desired value at 2.5s and remains unchanged, while the height value converges to the desired value at 5.5s. At this stage, the adjustment time in the PID control is 6s, 6s, 6.4s, 7.2s, and 5.5s, respectively. We identified that the adjustment time in each direction tends to increase a lot, while the overshoot of PID is much larger than that of the sliding mode PID. The fluctuation amplitude and flatness are larger, and the sliding mode PID is smoother.



FIGURE 15: UAV flight test.



- Roll
- Yaw
- Pitch

FIGURE 16: UAV flight attitude angle.

Figure 11 shows a periodic sine wave with a desired tilt angle changed to a frequency of 1.2π rad/s and an amplitude of 1. The tracking effect of the two controls is compared when the desired motion of the robotic arm is sinusoidal. The results show that when the desired tilt angle of the robot arm fluctuates sinusoidally, the adjustment time will grow

accordingly, and the sinusoidal motion of the robot arm will affect the change of the pitch angle. Then, the pitch angle in the sliding mode PID control will show smaller fluctuations up and down, while the pitch angle in the PID control will produce larger fluctuations. These fluctuations will always exist and move up and down around the expected value in

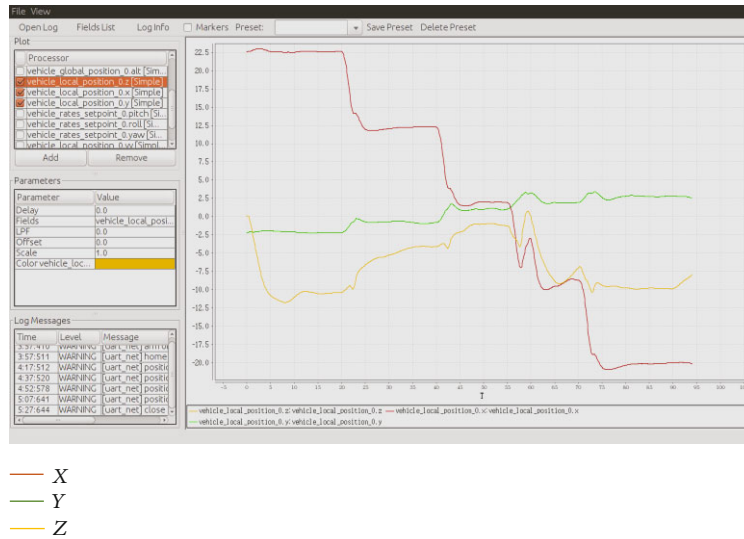


FIGURE 17: Displacement of UAV.



FIGURE 18: Movement speed of the UAV.

an approximate sinusoidal wave, making it difficult to reach the desired state. This is mainly due to the oscillation of the pulled-down object affecting the stability of the whole system in the x -axis direction. Furthermore, the anti-interference ability of PID is not as strong as that of the sliding-mode PID control, so it also triggers inevitable fluctuations in the pitch angle.

Figure 12 shows the attitude waveforms produced by adding appropriate perturbations to both control systems under the condition that the tilt angle expectation is a sine wave. The response time of both controllers increases relatively by a few seconds, but the waveform of the PID controller shows strong vibrations after adding the perturbation. We have also identified small periodic fluctuations in each attitude after a long-time adjustment, but the

same phenomenon was not identified in the sliding mode controller, which also confirms that the sliding mode PID control is more resistant to disturbance than the PID control. Our observations point out that the sliding mode PID control is more robust to external disturbance and parameters, has strong antidisturbance characteristics against external disturbances and parameter uptake, and has strong robustness and self-adaptability.

The simulation model is built by Simulink according to the above control flow and procedures, and the established dynamic model and the control method are input into it. Figure 13(a) shows the path of the UAV on B_x and B_z , and Figure 13(b) shows the influence of different initial heights on the trajectory tracking of the UAV. As shown, as the UAV moves away from the preset initial position

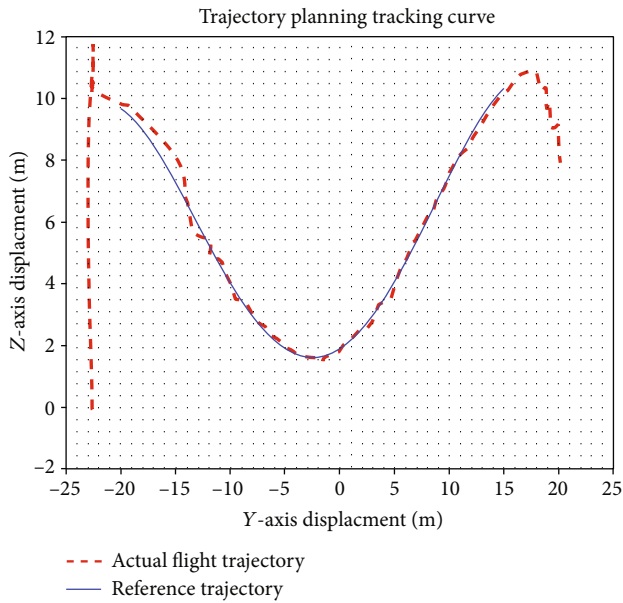


FIGURE 19: Trajectory planning tracking curve.

offset, it needs more real-time tracking. The longer the adjustment time, the greater the deviation and the more time it takes to reach the steady state. Figure 14 shows the performance results of the respective attitude curves in Figure 13 (a) reaching steady state.

5.2. Flight Experiment. The x -axis and z -axis coordinates of the path points obtained by Matlab simulation were imported and then divided into two arrays as the target expectation function of their respective changing with time. The flying robot arm successfully functioned after continuously giving the desired position of the drone through the cycle. Grab the trajectory plan. Figure 15 is a snapshot of several key points of the actual flight. It can be seen that the path point settings are generally in line with the planned Section 4.

Figures 16, 17, and 18 show the attitude angle, relative offset position, and motion speed of the UAV when it is grabbing the track. As shown in Figure 16, the change of the attitude angle of the UAV is small, fluctuating within the range $\pm 1^\circ$, and the maximum deviation of the yaw angle is about $\pm 4^\circ$, which basically meets the stability requirements of the vehicle. According to the above indicators, it can be concluded that the sliding mode PID is effective in the stability control of the drone. Furthermore, in the above flight curve, the movement of the mechanical arm is also controlled by the remote controller to increase the disturbance during the movement through the actual flight. The attitude curve also confirms that the drone can still achieve its stability quickly through the sliding mode PID control with the addition of disturbance.

As shown in Figure 17, in the x and y -axis direction can be found in the UAV in the x -axis direction has been decreasing, which is the direction of flight of the UAV in

the experiment is to fly from north to south, and in the Earth's coordinate system x -axis to north as positive, y -axis to east as positive, so the UAV in the x -axis direction is always decreasing. According to Figure 18, it can be found that the velocity of v_x and v_y is within the range of $\pm 1.5/s$, and the velocity of v_z is within the range of $3.5m/s \sim 6m/s$, which basically conforms to the relevant constraints set in this paper.

Figure 19 is a comparative analysis between the flight trajectory designed using trajectory planning and the actual flight trajectory, it can be found that the flight path of the UAV is flying up and down around the preset trajectory, and the error is basically in line with the control requirements of the UAV from large to small.

6. Conclusion

This research introduced the differential flatness theory as a means to solve the problems in aircraft trajectory planning with nonintegrality constraints. Throughout this research, the steps of trajectory generation based on differential flatness are explained in detail considering the theory's basic concepts and properties. The flat output was determined by the flat property of the dynamics and kinematics of the aircraft, and B-spline was adopted to parameterize the flat output and transform the optimal control problem into a general nonlinear programming problem to reduce the difficulty. In addition, the methodology used in this study optimized the traditional PID control by using the sliding mode control, which enhances the anti-interference and fast response capability of the UAV. Finally, the mathematical model of the UAV contained input into the control model to achieve a good tracking simulation based on the above trajectory. All of these procedures were used to verify the trajectory planning of the B-spline based on the differential flat.

This research proves that the differential flatness theory can better solve the trajectory planning problem of nonintegrality constraint systems. It is easier to find the most suitable setting in the appropriate output space by representing the flat system's input and state by the output and its finite differential term. This methodology can also be used to optimize solutions and then map them to the state space, which greatly reduces the difficulty of solving the control variables.

Symbols

symbol:	Symbol description
m_1 :	Quadrotor mass
m_2 :	Robot arm mass
g :	Local acceleration of gravity
L :	Distance from centroid to motor
Ω :	Rotor speed of revolution
C_T :	Rotor lift factor
C_Q :	Rotor torque factor
R :	Rotor radius
ρ :	Air density
I_x :	The moment of inertia about axis B_x

- I_y : The moment of inertia about axis B_y
 I_z : The moment of inertia about axis B_z .

Data Availability

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

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This project is supported by National Natural Science Foundation of China (Grant no. 51875293).

References

- [1] K. Nonami, "Prospect and recent research & development for civil use autonomous unmanned aircraft as UAV and MAV," *Journal of System Design and Dynamics*, vol. 1, no. 2, pp. 120–128, 2007.
- [2] H. Cai and Q. Geng, "Research on the development process and trend of unmanned aerial vehicle," *Proceedings of 2015 International Industrial Informatics and Computer Engineering Conference*, 2015.
- [3] N. Yin, R. Liu, B. Zeng, and N. Liu, "A review: UAV-based remote sensing," *Proceedings of the 2nd International Symposium on Application of Materials Science and Energy Materials*, vol. 490, 2019.
- [4] D. H. Yeon and J. R. Morrison, "Development of vision tracking and landing for persistent UAV system," *Proceedings of Spring Conference of Korean Society of Industrial Engineering*, pp. 1054–1054, 2018.
- [5] Y. Lu, Z. Xue, G. S. Xia, and L. Zhang, "A survey on vision-based UAV navigation," *Geo-Spatial Information Science*, vol. 21, pp. 1–12, 2018.
- [6] H. Y. Irwanto, "Development of autonomous controller system of high speed UAV from simulation to ready to fly condition," *Journal of Physics Conference Series*, vol. 962, p. 012015, 2018.
- [7] D. M. Vijayakumari, S. Kim, J. Suk, and H. Mo, "Receding-horizon trajectory planning for multiple UAVs using particle swarm optimization," *AIAA Scitech 2019 Forum*, pp. 1165–2019, 2019.
- [8] C. Sampedro, H. Bavle, J. L. Sanchez-Lopez et al., "A flexible and dynamic mission planning architecture for UAV swarm coordination," in *2016 International Conference on Unmanned Aircraft Systems (ICUAS) IEEE*, Arlington, VA, USA, 2016.
- [9] N. B. Hui and D. K. Pratihari, "A comparative study on some navigation schemes of a real robot tackling moving obstacles," *Robotics and Computer-Integrated Manufacturing*, vol. 25, no. 4-5, pp. 810–828, 2009.
- [10] C. Mou, W. Qing-Xian, and J. Chang-Sheng, "A modified ant optimization algorithm for path planning of UCAV," *Applied Soft Computing Journal*, vol. 8, no. 4, pp. 1712–1718, 2008.
- [11] R. Marie, H. B. Said, J. Stéphant, and O. Labbani-Igbida, "Visual servoing on the generalized Voronoi diagram using an omnidirectional camera," *Journal of Intelligent and Robotic Systems*, vol. 94, no. 3-4, pp. 793–804, 2019.
- [12] P. Marin-Plaza, A. Hussein, D. Martin, and A. D. L. Escalera, "Global and local path planning study in a ROS-based research platform for autonomous vehicles," *Journal of Advanced Transportation*, vol. 2018, 10 pages, 2018.
- [13] F. Bounini, D. Gingras, H. Pollart, and D. Gruyer, "Modified artificial potential field method for online path planning applications," *2017 IEEE Intelligent Vehicles Symposium (IV) IEEE*, 2017.
- [14] J. Sun, J. Tang, and S. Lao, "Collision avoidance for cooperative UAVs with optimized artificial potential field algorithm," *IEEE Access*, vol. 5, pp. 18382–18390, 2017.
- [15] J. D. Silva Arantes, M. D. Silva Arantes, C. F. Motta Toledo, O. T. Júnior, and B. C. Williams, "Heuristic and genetic algorithm approaches for UAV path planning under critical situation," *International Journal of Artificial Intelligence Tools*, vol. 26, no. 1, p. 1760008, 2017.
- [16] S. Perez-Carabaza, E. Besada-Portas, J. A. Lopez-Orozco, and J. M. de la Cruz, "Ant colony optimization for multi-UAV minimum time search in uncertain domains," *Applied Soft Computing*, vol. 62, pp. 789–806, 2018.
- [17] V. Ghadiok, J. Goldin, and W. Ren, "Autonomous indoor aerial gripping using a quadrotor," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, CA, USA, 2011.
- [18] Z. Qu, J. Wang, and C. E. Plaisted, "A new analytical solution to mobile robot trajectory generation in the presence of moving obstacles," *IEEE Transactions on Robotics*, vol. 20, no. 6, pp. 978–993, 2004.
- [19] F. Anritter, "On the relations between different flatness based design methods for tracking controllers," in *American Control Conference*, pp. 1740–1745, Seattle, WA, USA, 2008.
- [20] W. Dong and Y. Guo, "New trajectory generation methods for nonholonomic mobile robots," *Proceedings of the 2005 International Symposium on IEEE*, 2005.
- [21] Y. Guo, Y. Long, and W. Sheng, "Global trajectory generation for nonholonomic robots in dynamic environments," *Robotics and Automation*, pp. 1324–1329, 2007.
- [22] N. Faiz, S. Agrawal, and R. Murray, "Trajectory planning of differentially flat systems with dynamics and inequalities," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 2, pp. 219–227, 2001.
- [23] V. Ghadiok, J. Goldin, and W. Ren, "On the design and development of attitude stabilization, vision-based navigation, and aerial gripping for a low-cost quadrotor," *Autonomous Robots*, vol. 33, no. 1-2, pp. 41–68, 2012.
- [24] R. Spica, A. Franchi, G. Oriolo, H. H. Bühlhoff, and P. R. Giordano, "Aerial grasping of a moving target with a quadrotor UAV," *IEEE/RSJ International Conference on Intelligent Robots & Systems IEEE*, 2012.
- [25] J. Thomas, G. Loianno, J. Polin, K. Sreenath, and V. Kumar, "Toward autonomous avian-inspired grasping for micro aerial vehicles," *Bioinspiration & Biomimetics*, vol. 9, no. 2, article 025010, 2014.
- [26] A. A. B. John and R. Dutta, "Cooperative trajectory planning in an intercommunicating group of UAVs for convex plume wrapping," *Sarnoff Symposium*, pp. 1–6, 2017.
- [27] L. Zhang, J. Chen, and F. Deng, "Aircraft trajectory planning for improving vision-based target geolocation performance," *2017 13th IEEE International Conference on Control & Automation (ICCA) IEEE*, 2017.

- [28] B. Penin, R. Spica, P. R. Giordano, and F. Chaumette, "Vision-based minimum-time trajectory generation for a quadrotor UAV," *IEEE/RSJ International Conference on Intelligent Robots & Systems IEEE*, 2017.
- [29] M. B. Milam, K. Mushambi, and R. M. Murray, "A new computational approach to real-time trajectory generation for constrained mechanical systems," *IEEE Conference on Decision & Control*, Vancouver, BC, Canada, 2000.
- [30] O. Mofid and S. Mobayen, "Adaptive sliding mode control for finite-time stability of quad-rotor UAVs with parametric uncertainties," *ISA Transactions*, vol. 72, pp. 1–14, 2018.
- [31] A. K. Singh and L. Dewan, "Adaptive-PID with sliding mode control for longitudinal flight of an UAV," *MR International Journal of Engineering & Technology*, vol. 4, pp. 1–6, 2018.