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 categories: (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
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 $E$, 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 $q$. $\rho$ 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:

$$
\begin{align*}
\dot{q} &= V(q)p, \\
M(q)\ddot{p} + C(q, p)p + F(p, q, u) &= 0.
\end{align*}
$$

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 $\omega$ represents the system’s pseudovelocities 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 $S_\phi = \sin \phi$, $S_\theta = \sin \theta$, $S_\alpha = \sin \alpha$, $C_\phi = \cos \theta$, $S_\psi = \sin \psi$, $C_\psi = \cos \psi$, and $C_\alpha = \cos \alpha$. The result of the final calculation is
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 $\frac{A}{C}$, and the coefficient is $K = \frac{C_Q}{C_T}$, then

\begin{equation}
F(p, q, u) = \begin{bmatrix}
-I_x + m_2 R^2 C_a & 0 & -m_2 R^2 C_a & 0 & m_2 R C_a & 0 & 0 \\
0 & I_y + f_y + m_2 R^2 & 0 & -m_2 R C_a & 0 & m_2 R S_a & f_y + m_2 R^2 \\
-m_2 R^2 C_a & 0 & I_z + m_2 R^2 S_a^2 & 0 & m_2 R S_a & 0 & 0 \\
0 & -m_2 R C_a & 0 & m_1 + m_2 & 0 & 0 & -m_1 R C_a \\
m_2 R C_a & 0 & -m_2 R S_a & 0 & m_1 + m_2 & 0 & 0 \\
0 & m_2 R S_a & 0 & 0 & 0 & m_1 + m_2 & m_2 R S_a \\
0 & f_y + m_2 R^2 & 0 & -m_2 R C_a & 0 & m_2 R S_a & f_y + m_2 R^2
\end{bmatrix}.
\end{equation}

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 $\phi$ and $\theta$ in the flight process are considerably small, it can be substituted into Equation (1) for simplification:

\begin{equation}
\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}
\phi \\
\theta \\
\psi
\end{bmatrix} + \begin{bmatrix}
\rho & v \ U_1 + m_2 R C_a S_a \\
\rho & v & 0 \\
\rho & 0 & 0
\end{bmatrix} \begin{bmatrix}
\rho \\
v \\
\gamma
\end{bmatrix}.
\end{equation}

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

\begin{equation}
\begin{cases}
\dot{\phi} = \frac{2L I_z (m_1 + m_2) + m_1 m_2 R^2 (-LU_2 C_{2a} + LU_4 S_{2a})}{A_3}, \\
\dot{\theta} = \frac{-T + LU_4}{I_y}, \\
\dot{\psi} = \frac{L [2L I_z (m_1 + m_2) + m_1 m_2 R^2] U_1 + m_1 m_2 R^2 (LR C_{2a} + U_2 S_{2a})}{A_3}, \\
\dot{x} = \frac{4m_1 (m_1 + m_2) R T C_a - 2m_2 R (2L^2 I_z (m_1 + m_2) + 2L^2 m_1 m_2 R^2 + 2m_1 R U_1 C_a) S_a + A_4 \ast g S_a}{A_4}, \\
\dot{y} = \frac{-2L I_z m_2 R U_2 C_a + 2L I_z m_2 R U_4 S_a - g [2L I_z (m_1 + m_2) + (I_y + I_z) m_1 m_2 R^2] C_Q S_a + g I_z - I_y] m_1 m_2 R^2 C_{2a} C_Q S_a}{A_3}, \\
\dot{z} = \frac{A_1 + A_2 \ast U_1 - A_3 \ast T}{A_1}, \\
\dot{a} = \frac{[I_y + I_z] (m_1 + m_2) + m_1 m_2 R^2] T - L [I_y (m_1 + m_2) + m_1 m_2 R^2] U_1 - I_y m_2 R U_4 S_a}{I_y (m_1 + m_2) + I_y m_2 R^2}.
\end{cases}
\end{equation}
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 altitude 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 $U_i$ 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 $(\dot{x}_d, \dot{y}_d)$, the desired tilt angle $\alpha$, the angular acceleration of the robot arm $\ddot{\alpha}$, and the control rates of $x, t, T$ are designed as follows:

$$\begin{align*}
\dot{x}_d &= k_p1(x_d - x) + k_i1 \int (x_d - x) + k_d1(\dot{x}_d - \dot{x})dx, \\
\dot{y}_d &= k_p2(y_d - y) + k_i2 \int (y_d - y) + k_d2(\dot{y}_d - \dot{y})dy, \\
\ddot{\alpha} &= k_p3(\alpha_d - \alpha) + k_i3 \int (\alpha_d - \alpha) + k_d3(\dot{\alpha}_d - \dot{\alpha})d\alpha.
\end{align*}$$

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

$$e_z = Z_d - Z.$$  

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

$$S_z = \dot{e}_z + d_1 \ast e_z + c_1 \ast \int e_z dt.$$  

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 $U_1$:

$$\dot{S}_z = -\epsilon \text{ sgn}\,(S_z) - kS_z.$$  

In the above formula, $\epsilon$ takes the value 0.01, $k$ takes the value 3, and $\text{ sgn}\,(s)$ 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:

$$V_z = \frac{1}{2}S_z^2.$$  

Then, its derivative along the track line is

$$\dot{V}_z = \dot{S}_z \dot{S}_z = S_z[-\epsilon \text{ sgn}\,(S_z) - kS_z] = -\epsilon\|S_z\| - k\|S_z\|^2 \leq 0.$$  

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.

The above letters are defined as follows:

$$\begin{align*}
A_1 &= -4g(m_1 + m_2)[J_p(m_1 + m_2) + m_1m_2R^2]C_\phi C_p - m_2R[4\ddot{x}^2J_p(m_1 + m_2) + 4\ddot{y}^2m_1m_2R^2]C_a, \\
A_2 &= 2[2J_p(m_1 + m_2) + m_2R^2(2m_1 + m_2)] - 2m_1^2R^2C_{2a}, \\
A_3 &= 4m_2(m_1 + m_2)RS_{a}, \\
A_4 &= 4(m_1 + m_2)\{J_p(m_1 + m_2) + m_1m_2R^2\}, \\
A_5 &= 2I_zJ_p(m_1 + m_2) + 2m_1m_2R^2(I_zC_a^2 + I_sS_a^2).
\end{align*}$$
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 $\dot{x} = f(x, u), x \in \mathbb{R}^n, u \in \mathbb{R}^m$ with input $u$ of the system without feedback and define the flat output as $y = h(x, u, u_1, \ldots, u^{(i)})$, the trajectory function $(x(t), u(t))$ can be set to be an equation with respect to the flat output $y$ and its differential $x = \phi(y, y, \cdots, y^{(k)}), u = a(y, y, \cdots, y^{(k)})$. The trajectory planning problem contains a starting
state and an arrival state, and the flat output of the system is assumed to be a vector containing $y_i$, so

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

where $\lambda_j, j = 1, 2, \cdots, N$ 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 $\lambda_j$. 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:

$$y_i(t_0) = \sum_j A_{ij} \lambda_j(t_0) \quad y_i(t_f) = \sum_j A_{ij} \lambda_j(t_f)$$

$$y_i^{(q)}(t_0) = \sum_j A_{ij} \lambda_j^{(q)}(t_0) \quad y_i^{(q)}(t_f) = \sum_j A_{ij} \lambda_j^{(q)}(t_f)$$

(15)

Provided that generality is guaranteed and assuming the dimension $i = 1$ of the planar output vector, $y = [y_1]$ (these
results can also be generalized to the multidimensional case), the other variables are defined as follows:

\[
\begin{align*}
\tilde{y}_0 & = \left( y_1(\tau_0), \ldots, y_1^{(q)}(\tau_0) \right), \\
\tilde{y}_f & = \left( y_1(\tau_f), \ldots, y_1^{(q)}(\tau_f) \right), \\
\tilde{y} & = \left( \tilde{y}_0, \tilde{y}_f \right).
\end{align*}
\]  

Combining equations A and B yields

\[
y = \begin{pmatrix} \Lambda(\tau_0) \\ \Lambda(\tau_i) \end{pmatrix} A = \Lambda A,
\] 

where \( \Lambda \) is a full rank matrix, which means that the space equation \( \lambda_j \) 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...
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 \(v_r\) represent the speed and acceleration of the vehicle.

\[
\begin{align*}
\min f(i) &= \sum_{i=1}^{t} \sqrt{(x_{i} - x_{i-1})^2 + (y_{i} - y_{i-1})^2} \\
\text{s.t.} &
\begin{cases}
  d_i < \sqrt{(x_i - x_o)^2 + (y_i - y_o)^2} + L \\
  v_i < 4, v_r < 5 \\
  i = 0, 1, \ldots, t.
\end{cases}
\end{align*}
\]  

(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

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

(19)

where \(p\) represents the degree of basis functions, \(u_i\) is the node, and \(N_{lp}(u)\) 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
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 $x, y, \alpha$ 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 $\alpha_d$, and the angular acceleration $\ddot{\alpha}$ of the robot arm:

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

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.
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

$$e_z = z_d - z.$$  \hspace{1cm} (21)

The slip surface function is designed as

$$s_z = \dot{e}_z + d_1 \cdot e_z + c_1 \cdot \int e_z \, dt.$$  \hspace{1cm} (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 $U_1$:

$$\dot{s}_z = -c \cdot \text{sgn} \left(s_z \right) - ks_z,$$  \hspace{1cm} (23)

where $c$ takes the value of $0.01$, $k$ takes the value of $3$, and $\text{sgn} \left(s \right)$ denotes the symbolic function. The Lyapunov
function is used to determine the stability of the system, and its positive definite function is designed:

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

Then, its derivative along the track line is

$$\dot{V}_z = s_z\dot{s}_z = s_z[-\varepsilon \text{ sgn}(s_z) - k_s||s_z|| - k_\parallel||s_z||^2 < 0.$$  \hspace{1cm} (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.
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 3.2 s, and the adjustment time of yaw angle is 4 s, 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 2.5 s and remains unchanged, while the height value converges to the desired value at 5.5 s. At this stage, the adjustment time in the PID control is 6 s, 6 s, 6.4 s, 7.2 s, and 5.5 s, 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 11 shows a periodic sine wave with a desired tilt angle changed to a frequency of $1.2\pi$ 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
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 ant-disturbance 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 Bx and Bz, 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
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 of $\pm 1^\circ$, 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 $v_y$ is within the range of 3.5m/s–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 nonintegrity 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 nonintegrity 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
$\Omega$: Rotor speed of revolution
$C_T$: Rotor lift factor
$C_Q$: Rotor torque factor
$R$: Rotor radius
$\rho$: Air density
$I_x$: The moment of inertia about axis $B_x$
$I_z$: The moment of inertia about axis $By$

$I_y$: The moment of inertia about axis $Bz$.

**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**


