

杭州电子科技大学

本科毕业设计（论文）

（2023届）

题 目 Research on Control Algorithm of
Quadcopter

学 院 圣光机联合学院

专 业 自动化

班 级 19320211

学 号 19322103

学生姓名 徐森

指导教师 尹克 Sergei V. Shavetov

完成日期 2023 年 6 月

诚信承诺

我谨在此承诺：本人所写的毕业设计（论文）《Research on Control Algorithm of Quadcopter》均系本人独立完成，没有抄袭行为，凡涉及其他作者的观点和材料，均作了注释，若有不实，后果由本人承担。

承诺人（签名）徐水水

2023年6月15日

ABSTRACT

The quadcopter has become widely used in various military and civilian fields due to its high flight safety, low production cost, and flight flexibility, making it highly valuable. The quadcopter's control system is key to its ability to perform preset tasks with flexibility and accuracy. Given the complexity of quadcopter flight, including nonlinearity, strong coupling, underactuation, and unknown uncertainties, this paper conducted a series of work on the design, simulation, testing, and hardware-in-the-loop simulation of the quadcopter's control system, after investigating the research status and mainstream mathematical models and control methods of the quadcopter.

Firstly, this paper analyzes the force situation and internal power structure of the quadcopter to establish a mathematical nonlinear model. The model is simplified to a linear model suitable for controller design based on the quadcopter's flight characteristics. Additionally, a nonlinear mathematical model of the quadcopter for simulation is built in MATLAB/Simulink.

Secondly, based on the analysis of the principles of two control methods, PID and linear active disturbance rejection control (LADRC), corresponding cascaded PID and cascaded LADRC controllers are designed according to the coupling relationship and characteristics of the quadcopter's mathematical model. The effectiveness of the two controllers in point hovering, anti-interference ability, and robustness is evaluated and compared through MATLAB/Simulink simulation, proving that both controllers are effective in the absence of interference, and the cascaded LADRC controller has better robustness and anti-interference ability than the cascaded PID controller.

Finally, to verify the effectiveness of the above control algorithms in actual flight, a quadcopter experimental platform is built by combining the open-source Pixhawk flight controller with the Airsim real-time motion simulation platform. Through the combination of hardware and software, the effectiveness of the designed cascaded PID and cascaded LADRC controllers is verified through hardware-in-the-loop simulation.

Key words: Quadcopter; PID; active disturbance rejection control(ADRC); linear active disturbance rejection control(LADRC)

CONTENTS

1 INTRODUCTION	1
1.1 Research Background and Significance	1
1.2 Literature Review	2
1.2.1 Mathematical model of quadcopter	3
1.2.2 Control methods for quadcopter	4
1.3 Thesis Outline	6
2 MATHEMATICAL MODELING OF QUADCOPTER	8
2.1 The structure of quadcopter	8
2.2 Working principle of quadcopter	9
2.3 Mathematical modeling of quadcopter	11
2.3.1 Coordinate system establishment	11
2.3.2 Attitude representation	12
2.3.3 Nonlinear mathematical model for quadcopter control .	13
2.3.4 Simplification of mathematical model of quadcopter .	21
2.3.5 Quadcopter MATLAB simulation model	23
2.4 Chapter Summary	24
3 PID-BASED QUADCOPTER CONTROLLER DESIGN	25
3.1 PID control theory	25
3.2 Cascade PID controller design	27
3.3 Simulation experiment of cascade PID controller for quadcopter	27
3.3.1 Fixed-point hovering experiment	28
3.3.2 Anti-interference ability and robustness experiments .	29
3.4 Chapter Summary	34
4 LADRC-BASED QUADCOPTER CONTROLLER DESIGN	35
4.1 Overview of Active Disturbance Rejection Control (ADRC) technology	35
4.2 Active Disturbance Rejection controller (ADRC) structure . . .	37
4.3 Linear Active Disturbance Rejection Controller (LADRC) structure	41
4.4 Cascade LADRC controller design	44
4.4.1 Design of LADRC controller for height z channel	45

4.4.2	Design of LADRC controller for yaw angle ψ channel	46
4.4.3	Design of ADRC controller for horizontal position $x, y - \phi, \theta$ channel.	47
4.4.4	Analysis of control system stability	49
4.5	Simulation experiment of cascade LADRC controller for quadcopter	51
4.5.1	Fixed-point hovering experiment	51
4.5.2	Anti-interference ability and robustness experiments	52
4.6	Chapter Summary	57
5	QUADCOPTER HARDWARE-IN-THE-LOOP(HIL) SIMULATION EXPERIMENTS	58
5.1	Introduction of hardware-in-the-loop simulation platform for quadcopters	58
5.1.1	Hardware system: Autopilot system (flight control system)	58
5.1.2	Software system: Real-time motion simulation platform Airsim	59
5.1.3	Hardware-in-the-loop simulation experiment process	60
5.2	Fixed-point hovering experiment	60
5.2.1	Experiment on HIL Simulation with Cascaded PID Controller	61
5.2.2	Experiment on HIL Simulation with Cascaded LADRC Controller	62
5.3	Chapter Summary	63
CONCLUSION AND FUTURE WORK	64	
5.4	Conclusion	64
5.5	Future Work	65
ACKNOWLEDGEMENTS	67	
REFERENCES	67	

1 INTRODUCTION

1.1 Research Background and Significance

Unmanned aerial vehicles(UAV), or "drones", are aircraft that do not require a pilot. They can be flown automatically through onboard computers, or remotely controlled by ground personnel or pilots on other aircraft. Compared to manned aircraft, drones can autonomously execute pre-programmed tasks using onboard sensors and flight control systems, or receive instructions through wireless remote control devices [1]. Drones are characterized by low operational requirements, zero risk of human casualties, flexible flight capabilities, and high task efficiency. As a result, they have been widely used in military and civilian applications such as crop inspection, reconnaissance and strike, damage assessment, express transportation, aerial photography, and formation flying [2, 3].

One of the most popular types of drones is the rotorcraft, which can be classified based on the number and configuration of its rotors[4]. Among this classification, quadcopters have recently become widely used due to their high flight safety, low space environment restrictions, simple controls, good maneuverability, large payload capacity, ability to carry various mission equipment to achieve different functions, low production costs, ease of assembly and disassembly, and ease of maintenance[5].

However, the control field of quadcopters still faces many challenges. Firstly, from the perspective of control systems, quadcopters have highly non-linear, strongly coupled[6], underactuated, and dynamically complex models, as well as measurement noise, making it difficult to design effective high-performance flight controllers[7]. Secondly, due to their inherent structural errors, small size, and light weight, quadcopters are susceptible to interference from internal and external factors, such as motor vibrations and changes in air-flow during high-speed rotations[5], which can alter their original aerodynamic characteristics and make it difficult to establish accurate models. Therefore, control systems need to have strong robustness and weak model correlation. Additionally, quadcopters are easily affected by unknown uncertainties dur-

ing autonomous flight, making it necessary for control systems to have strong adaptability and anti-interference capabilities.

The autonomous flight control algorithm is a key technology that ensures quadcopters have excellent anti-interference ability and robustness, enabling them to successfully complete preset tasks. Therefore, researching and designing control algorithms that enable quadcopter systems to rapidly converge while possessing good anti-interference ability and robustness has always been an important hotspot in the development of quadcopter industry[8–12]. Additionally, quadcopter research provides an experimental platform for the integration of many fields such as automatic control, advanced sensing technology, and electronic engineering. In the fields of intelligent robot control, 3D path planning, and multi-aircraft cooperation, quadcopter autonomous flight technology has high research value.

Thus, designing high-performance autonomous control algorithms and applying them to quadrotor systems can enable them to safely and efficiently complete a variety of tasks and be widely used in civilian and military fields. It can also provide a safer and more efficient experimental platform for the integration of multiple fields, with extremely high research value.

1.2 Literature Review

research on multi-rotors has increasingly focused on autonomy and group behavior. In June 2013, Raffaello D 'Andrea, a professor at ETH Zurich, gave a TED Global talk on competitive robotic sports (where machines perform to the full limits of their abilities, including catching the ball, balancing, and co-decision making) And showed off a quadrotor controlled by the depth camera Kinect[13]. In June 2015, the machine intelligence column of the journal Nature published an article entitled "Science, Technology, and Future of Small Autonomous Drones"[14]. This paper summarized the challenges of designing, manufacturing, perceiving, and controlling small UAVs and analyzed the future research trends of small UAVs.

At the end of 2012, DJI released the Phantom quadcopter, which could take off and fly with ease. At the time, action cameras were extremely popular for capturing extreme sports, and so the Phantom, equipped with an action

camera, quickly became a hit in the aerial photography market. Since then, DJI has released a series of consumer drones including the Spark, Mavic, and Inspire. Okada et al.[15] demonstrated a four-rotor helicopter with a rotating shell structure, which helps the aircraft to hover steadily in the air and avoid obstacles. This can be applied to surveillance and inspection. Shen et al.[16] designed an autonomous quadcopter equipped with a magnetometer, GPS sensor, altimeter, tiltable camera, and sensors for indoor and outdoor hovering. This can be applied to surveillance, mission planning, search, and rescue. Latscha et al. [17] developed a hybrid quadcopter that combines two snake-shaped mobile robots for monitoring disaster scenes. Darivianakis et al.[18] designed and developed a quadcopter for inspecting infrastructure and engaging in physical interaction.

The research on quadcopter involves many fields, including the establishment of precise mathematical models of dynamics, the design of controllers for the aircraft, trajectory optimization and tracking, the acquisition of sensor and visual information, and the establishment of ground station human-machine friendly interfaces[19]. The controller designed for quadcopter is the key and core for the quadcopter to achieve autonomous flight, complete various tasks and flight actions. The design of the controller is currently a hot topic and challenge that many universities and companies at home and abroad are focusing on.

To design high-performance controllers, it is necessary to first establish as accurate mathematical models as possible, and then based on the mathematical models and the characteristics of the quadcopter itself, research and design controllers that are strong in anti-interference, robust and easy to adjust parameters, to ensure that the quadcopter can fly stably under various circumstances.

1.2.1 Mathematical model of quadcopter

In order to achieve effective control of quadcopter, it is necessary to accurately establish its mathematical model under various flight states. However, due to the structural characteristics of quadcopter, including structural errors, and the various physical effects it experiences during flight (aerodynamics, grav-

ity, gyroscopic effects, and rotor inertia moments), it is also susceptible to deformation caused by external environmental factors, such as airflow, making it difficult to establish an effective and reliable dynamic model[5]. Therefore, it is important to comprehensively analyze the key forces and moments acting on the quadcopter, and based on the corresponding physical laws, establish the most accurate quadcopter dynamic equations possible. At the same time, the control system should have strong robustness and weak model correlation.

1.2.2 Control methods for quadcopter

Currently, there are several main control methods for quadcopter systems, which are characterized by their small size, light weight, susceptibility to external environmental interference, and difficulty in identifying parameters for their nonlinear, strongly coupled[7], underactuated, and dynamic models. These control methods include:

1. Proportional Integral Derivative, PID Control

The Proportional Integral Derivative (PID) control method is the earliest developed and applied control method. Its algorithm is simple, and parameter adjustment is relatively easy, making it convenient for implementation. It has been widely used in engineering practice for unmanned aerial vehicle flight control. However, classical PID control overlooks nonlinear factors in the system. As a result, its decoupling ability and anti-interference performance are flawed[5]. Therefore, for quadcopter, which are difficult to model accurately and have strong coupling, conventional PID control cannot meet design requirements. Various improved PID control methods need to be adopted.

2. Sliding Mode Control(SMC)

Sliding Mode Control (SMC) is a variable structure nonlinear control method with a simple structure, strong robustness, and superior control performance. Its control idea is to change the internal feedback control structure of the system based on the current state of the system, so that the system state slides on the sliding mode surface and eventually reaches the equilibrium point. SMC does not require high mathemati-

cal model accuracy for the objects it controls and can adapt to internal perturbations and external environmental interference, making it highly anti-interference. Additionally, its algorithm design is simple and easy to implement, making it widely used in various industrial controls. However, SMC has a significant disadvantage of accompanying chattering[5].

3. Back-Stepping Control

Back-Stepping control is a commonly used method for nonlinear system control. Its design concept is to decompose complex nonlinear systems into multiple subsystems until the system's complete controllability can be achieved. Then, starting from the lower order, a function and intermediate virtual variables that satisfy the Lyapunov law are designed for each subsystem, moving forward until the entire control system is designed[20]. This method is suitable for systems with strict feedback control structures and has good overshoot-free tracking performance. However, because backstepping control relies on prior information about the system model, model errors have a significant impact on control accuracy[5].

4. Active Disturbance Rejection Control(ADRC)

Active Disturbance Rejection Control (ADRC) is a control technology proposed by control theory expert Mr. Han Jingqing, which combines modern control theory and PID control technology. The controller consists of three parts: Tracking Differentiator (TD), Nonlinear Extended State Observer (NESO), and Nonlinear State Error Feedback (NSEF)[21]. The ADRC control scheme is a "observation + compensation" control method that inherits the core spirit of classical PID control theory and integrates the advantages of modern control theory. It can effectively solve the nonlinear and uncertainty problems of the controlled system without relying on the mathematical model of the controlled object. It has superior control performance and provides a new solution for the control problems of nonlinear and strongly coupled objects. Currently, many scholars have applied ADRC control technology to quadcopter control systems. For example, the literature [22]designed a cascaded ADRC controller, in which the inner loop uses nonlinear ADRC for attitude control and the outer loop uses linear ADRC for position control, achieving sta-

ble hovering and fast tracking control of the quadcopter. The literature [23]designed an attitude controller based on ADRC and a back-stepping sliding mode controller combined with Lyapunov stability analysis, to stabilize the attitude control of the quadcopter in the presence of model uncertainty and external disturbances. The control effect is better than that of the classical ADRC controller in terms of stability and dynamic performance.

1.3 Thesis Outline

This article mainly focuses on three aspects: first, establishing a mathematical model for quadcopter flight as accurately as possible and linearizing it; second, designing two control systems, namely cascaded PID and cascaded LADRC controllers, and verifying their effectiveness through MATLAB/Simulink simulations, while comparing their performance; third, setting up a hardware-in-the-loop simulation platform to verify the effectiveness of the control system.

Chapter 1 mainly introduces the application prospects of quadcopter and its main research and development progress, and presents some mainstream control schemes adopted for the quadcopter control system.

Chapter 2 is based on the analysis of the structure and flight principle of the quadcopter, and establishes its nonlinear mathematical model through mechanical theory. To simplify the nonlinear model for controller design, reasonable assumptions are made to obtain the simplified linear mathematical model. Finally, the nonlinear mathematical model of the quadcopter is built in MATLAB/Simulink for simulation experiments.

Chapter 3 first introduces the PID control theory and the role of each element, then designs the cascaded PID controller for the quadcopter. Finally, the effectiveness of this controller is verified by simulation experiments in MATLAB/Simulink, and the robustness and anti-interference capability of this controller are tested experimentally.

Chapter 4 first introduces the theoretical knowledge and composition structure of Active Disturbance Rejection Control (ADRC), and then studies in detail the improved control algorithm of ADRC - Linear Active Disturbance

Rejection Control (LADRC) and its parameter tuning method, due to ADRC's complex parameter tuning. Then, a cascaded LADRC controller is designed based on the LADRC control principle, and its effectiveness is verified through MATLAB/Simulink simulation experiments, testing its robustness and anti-interference ability. At the same time, to better illustrate the superiority of the LADRC algorithm, its simulation experiment results are compared and analyzed with the series PID controller.

Chapter 5 first introduces the open-source cross-platform physics and visual simulation software Airsim based on game engines and the open-source drone flight control hardware Pixhawk. Then, the designed controllers are applied to the flight control hardware and simulation experiment platform for hardware-in-the-loop(HIL) simulation experiments, and the effectiveness and rationality of the designed controllers are verified through actual experimental results, achieving the integration of theory and practice.

2 MATHEMATICAL MODELING OF QUADCOPTER

For a controlled object, establishing an accurate mathematical model is the foundation for system analysis and control. It can simplify complex practical problems into mathematical problems, replace physical experiments, and improve the efficiency and accuracy of problem-solving. This is a critical step in designing the control system for quadcopter.

This chapter first introduces the structure and flight principles of quadcopters, followed by two coordinate systems and three attitude representation methods required for quadcopter modeling. Then, by considering the forces and internal structure of the quadcopter, a nonlinear control model with six degrees of freedom is established under two attitude representation methods. To facilitate subsequent controller design, the nonlinear model is appropriately simplified and linearized to obtain a simplified linear model. Finally, a nonlinear control model for quadcopters is built in MATLAB/Simulink for subsequent simulation experiments.

2.1 The structure of quadcopter

A quadcopter with a conventional structure utilizes four rotors as direct power sources, with two rigid rods intersecting at the center and symmetrically distributing the rotors at the ends of the four arms. The space in the middle of the body is typically used for placing the autopilot and other external devices.

As shown in Figure 1, quadcopters can be classified into a cross-shaped and an X-shaped structure based on the angle between the nose and the frame. The flight principles of quadcopters with a cross-shaped and an X-shaped layout are the same. However, the X-shaped layout is more popular due to its higher maneuverability (more rotors involved in pitch and roll attitude control) and the reduced obstruction of the gimbal's field of view caused by the arms and rotors. Therefore, this paper chooses the X-shaped quadcopter for modeling and analysis.

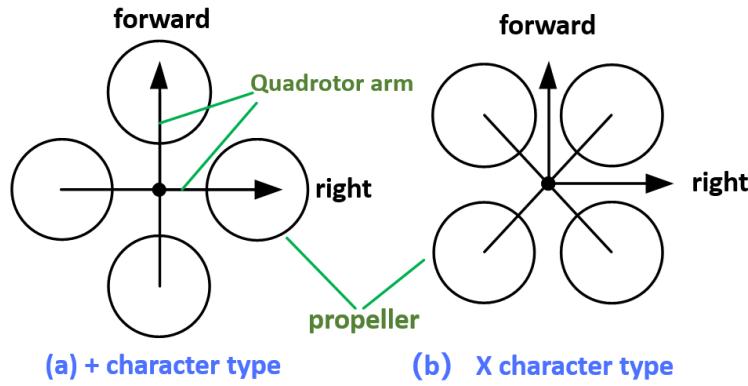


Figure 1 — Quadcopter conventional rack layout

2.2 Working principle of quadcopter

Taking the X-shaped quadcopter as an example, we establish the quadcopter body coordinate system as shown in Figure 2 and introduce the flight principles of the quadcopter. In this configuration, rotors #1 and #3 rotate counterclockwise, while rotors #2 and #4 rotate clockwise, with the x_b axis serving as the quadcopter's nose.

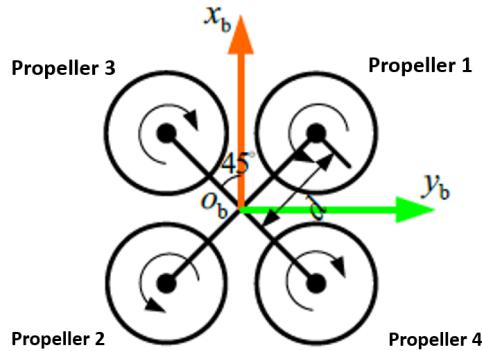


Figure 2 — X-shaped quadcopter frame layout

Vertical Movement: As shown in Figure 10a, if we increase the speed of all four rotors simultaneously and equally, the total thrust generated by the propellers will increase, but the total torque will remain at 0. If the quadcopter is placed on a horizontal surface, it will ascend when the thrust exceeds the weight, and descend when the thrust is less than the weight.

Pitch Movement: As shown in Figure 10b, if we simultaneously and equally decrease the speed of propellers #1 and #4 while increasing the speed of propellers #2 and #3, the quadcopter will pitch forward. This will generate a forward component of thrust, while simultaneously reducing the vertical component of thrust, which will no longer be equal to the weight of the quadcopter.

To compensate for this, we need to increase the speed of all four propellers accordingly, enabling the quadcopter to achieve horizontal forward flight. Similarly, horizontal backward flight can be achieved using the same principle.

Rolling Movement: As shown in Figure 10c, if we simultaneously and equally decrease the speed of propellers #1 and #2 while increasing the speed of propellers #3 and #4, the quadcopter will roll to the right. This will generate a rightward component of thrust, while simultaneously reducing the vertical component of thrust, which will no longer be equal to the weight of the quadcopter. To compensate for this, we need to increase the speed of all four propellers accordingly, enabling the quadcopter to achieve horizontal rightward flight. Similarly, horizontal leftward flight can be achieved using the same principle.

Yaw Movement: As shown in Figure 10d, decreasing the speed of propellers #2 and #4 while increasing the speed of propellers #1 and #3 by the same amount will result in zero torque for both forward/backward and left/right flight. By Newton's Third Law, every action has an equal and opposite reaction. As propellers #1 and #3 rotate counterclockwise and propellers #2 and #4 rotate clockwise, the clockwise yaw torque increases while the counterclockwise yaw torque decreases, resulting in a net clockwise yaw torque. This causes the quadcopter to turn clockwise and change its yaw direction. Similarly, counterclockwise turning can be achieved.

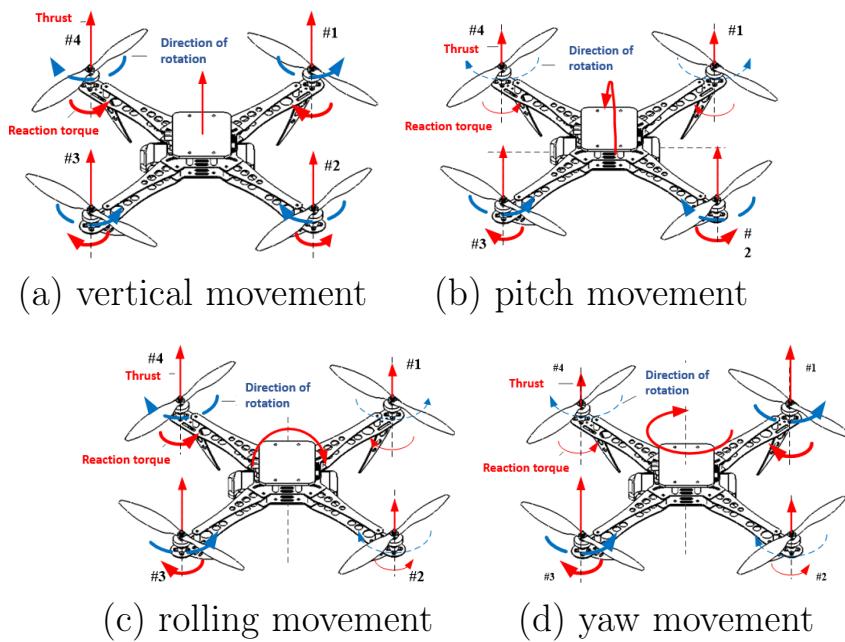


Figure 3 — Flight principle of quadcopter

2.3 Mathematical modeling of quadcopter

2.3.1 Coordinate system establishment

In order to establish a mathematical model with six degrees of freedom for a quadcopter to accurately describe its attitude and position, it is necessary to establish the body-fixed coordinate system and the earth-fixed coordinate system as shown in Figures 4a and 4b, respectively.

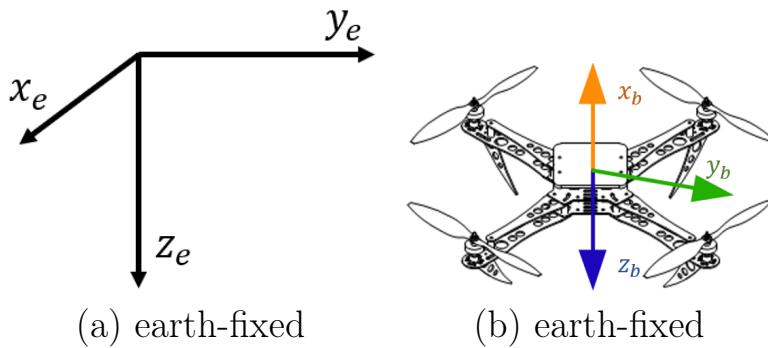


Figure 4 – earth-fixed coordinate system(a)& body-fixed coordinate system(b)

- **earth-fixed coordinate system**

As shown in Figure 4a, the Earth-fixed coordinate system $o_e x_e y_e z_e$ is used to study the motion of the quadcopter relative to the ground and determine its three-dimensional spatial position. Assuming the Earth's surface is flat and neglecting its curvature, the origin o_e is set at the quadcopter's takeoff position, with the $o_e x_e$ axis pointing towards the quadcopter's nose direction and the $o_e z_e$ axis pointing down towards the ground. The $o_e y_e$ axis is determined by the right-hand rule.

- **body-fixed coordinate system**

As shown in Figure 4b, the body-fixed coordinate system $o_b x_b y_b z_b$ is fixed to the quadcopter's body, with the quadcopter's center of gravity serving as the origin o_b . The $o_b x_b$ axis points towards the quadcopter's nose direction, while the $o_b z_b$ axis points downwards perpendicular to the aircraft's plane. The $o_b y_b$ axis is determined by the right-hand rule.

2.3.2 Attitude representation

Currently, there are multiple ways to represent the attitude of a quadcopter during flight, each with its own advantages and disadvantages. This section will introduce three methods for representing attitude: Euler angles, rotation matrices, and quaternions.

- **Euler angle**

According to Euler's theorem, the rotation of a rigid body about a fixed point can be seen as the composition of several finite rotations around that point[24]. By performing three basic rotations, the Earth-fixed coordinate system can be transformed into the body coordinate system. In each of these rotations, the rotation axis corresponds to a coordinate axis of the system being rotated, and the rotation angle is the Euler angle. Intuitively, if the Earth-fixed coordinate system is identical to the body coordinate system, then the yaw angle ψ , pitch angle θ , and roll angle ϕ are as shown in Figure 5, with the direction of the angles determined by the right-hand rule.

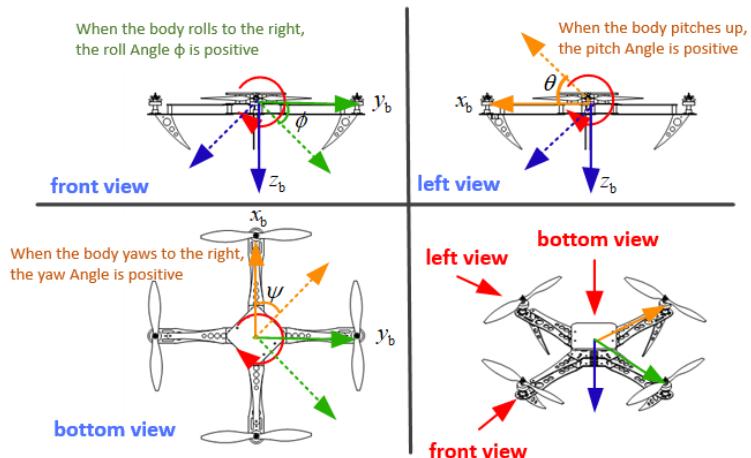


Figure 5 — Visual representation of Euler angles

- **rotation matrix**

The rotation matrix satisfies:

$$\begin{cases} {}^e\mathbf{b}_1 = \mathbf{R}_b^e \cdot {}^b\mathbf{b}_1 = \mathbf{R}_b^e \cdot \mathbf{e}_1 \\ {}^e\mathbf{b}_2 = \mathbf{R}_b^e \cdot {}^b\mathbf{b}_2 = \mathbf{R}_b^e \cdot \mathbf{e}_2 \\ {}^e\mathbf{b}_3 = \mathbf{R}_b^e \cdot {}^b\mathbf{b}_3 = \mathbf{R}_b^e \cdot \mathbf{e}_3. \end{cases} \quad (1)$$

Therefore, the rotation matrix is defined as:

$$\mathbf{R}_b^e \triangleq \begin{bmatrix} {}^e\mathbf{b}_1 & {}^e\mathbf{b}_2 & {}^e\mathbf{b}_3 \end{bmatrix} \quad (2)$$

The application of rotation matrix avoids the singularity problem. However, \mathbf{R}_b^e contains 9 unknown variables, so solving the differential equation is more computationally intensive.

- **quaternion**

The quaternion is generally expressed as

$$\mathbf{q} \triangleq \begin{bmatrix} q_0 \\ \mathbf{q}_v \end{bmatrix} \quad (3)$$

where, $q_0 \in \mathbb{R}$ is the scalar part of $\mathbf{q} \in \mathbb{R}^4$, $\mathbf{q}_v = [q_1 \ q_2 \ q_3]^T \in \mathbb{R}^3$ is the vector part. For real numbers $s \in \mathbb{R}$, the corresponding quaternion representation is of the form $\mathbf{q} = [s \ \mathbf{0}_{1 \times 3}]^T$. For a pure vector $\mathbf{v} \in \mathbb{R}^3$, the corresponding quaternion representation takes the form $\mathbf{q} = [0 \ \mathbf{v}^T]^T$.

2.3.3 Nonlinear mathematical model for quadcopter control

As shown in Figure 6, the quadrotor modeling consists of three main parts as follows:

1. Rigid Model of Quadcopter Flight Control

The rigid model of quadcopter flight control consists of two parts: the rigid kinematic model and the rigid dynamic model of the quadcopter.

(a) Rigid Kinematic Model

The input of the rigid kinematic model of the quadcopter is the velocity and angular velocity, and the output is the position and attitude. Kinematics are independent of mass and force, and only study variables such as position, velocity, attitude, and angular velocity.

(b) Rigid Dynamic Model

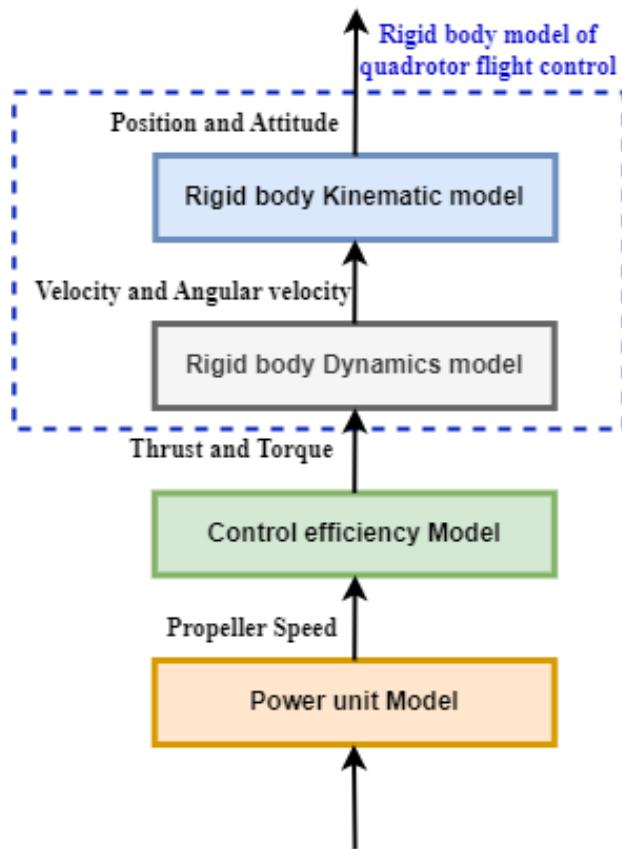


Figure 6 — Quadrrotor modeling hierarchy

The input of the rigid dynamic model of the quadcopter is the thrust and torque (pitch torque, roll torque, and yaw torque), and the output is the velocity and angular velocity. Dynamics involve both motion and force, and are related to the mass and moment of inertia of the object.

2. Control Efficiency Model

The input of the control efficiency model is the propeller speed, and the output is the thrust and torque.

The inverse process of the control efficiency model is called the control allocation model (when the desired thrust and torque are obtained through controller design, the required propeller speed can be solved through the control allocation model).

3. Power Unit Model

The input of the power unit model is the throttle command between 0 $\tilde{1}$, and the output is the propeller speed.

Now, let's provide a detailed explanation of the models for each of these three parts:

* Rigid Model of Quadcopter Flight Control

1. Assumption

During the flight of a quadcopter, it is subject to various external forces and other uncertain factors in a complex aerodynamic environment. Therefore, it is difficult to establish an accurate mathematical model for a quadcopter, and modeling can become too cumbersome, leading to complex control algorithm designs. To simplify the mathematical model of a quadcopter while ensuring its applicability, this article makes the following reasonable assumptions:

- **Assumption 1:** The quadcopter is a rigid body.
- **Assumption 2:** The mass and moment of inertia of the quadcopter are constant.
- **Assumption 3:** The geometric center of the quadcopter is the same as its center of mass.
- **Assumption 4:** The quadcopter is only subject to gravity and propeller thrust, where gravity acts along the $o_e z_e$ axis in the positive direction, and propeller thrust acts along the $o_b z_b$ axis in the negative direction.
- **Assumption 5:** The propellers with odd-numbered labels rotate counterclockwise, while those with even-numbered labels rotate clockwise.

2. Quadcopter Rigid Kinematic Model

Let the center of gravity vector of the quadrotor be ${}^e\mathbf{p} \in \mathbb{R}^3$, then:

$${}^e\dot{\mathbf{p}} = {}^e\mathbf{v} \quad (4)$$

where ${}^e\mathbf{v} \in \mathbb{R}^3$, denotes the speed of the quadrotor.

(a) Euler Angle Model:

If the angular velocity of body rotation is ${}^b\omega = [\omega_{x_b} \omega_{y_b} \omega_{z_b}]^T$, then the rate of attitude change $\dot{\theta}$ is related

to the angular velocity of body rotation ${}^b\omega$ as follows[25]:

$$\dot{\Theta} = \mathbf{W} \cdot {}^b\omega \quad (5)$$

where:

$$\Theta \triangleq \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, \quad \mathbf{W} \triangleq \begin{bmatrix} 1 & \tan \theta \sin \phi & \tan \theta \cos \phi \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi / \cos \theta & \cos \phi / \cos \theta \end{bmatrix} \quad (6)$$

Based on Assumption 1, combining Equation (4) and Equation (5), it is obtained that:

$$\begin{cases} {}^e\dot{\mathbf{p}} = {}^e\mathbf{v} \\ \dot{\Theta} = \mathbf{W} \cdot {}^b\omega. \end{cases} \quad (7)$$

(b) *Quaternion Model:*

The rate of change of quaternion $\dot{\theta}$ is related to the angular velocity of body rotation ${}^b\omega$ as follows [25]:

$$\begin{cases} \dot{q}_0 = -\frac{1}{2}\mathbf{q}_v^T \cdot {}^b\omega \\ \dot{\mathbf{q}}_v = \frac{1}{2}(q_0\mathbf{I}_3 + [\mathbf{q}_v]_{\times}) \cdot {}^b\omega \end{cases} \quad (8)$$

Based on Assumption 1, combining Equation(4) and Equation (8), it is obtained that:

$$\begin{cases} {}^e\dot{\mathbf{p}} = {}^e\mathbf{v} \\ \dot{q}_0 = -\frac{1}{2}\mathbf{q}_v^T \cdot {}^b\omega \\ \dot{\mathbf{q}}_v = \frac{1}{2}(q_0\mathbf{I}_3 + [\mathbf{q}_v]_{\times}) \cdot {}^b\omega \end{cases} \quad (9)$$

3. Quadcopter Rigid Dynamic Model

(a) *Quadrocopter position dynamics model:*

Based on assumption 4, this paper only considers the quadcopter with horizontal propeller disc, and then conducts force analysis on the quadcopter, we can obtain:

$${}^e\dot{\mathbf{v}} = {}^e\mathbf{F}/m \quad (10)$$

where $m > 0$ denotes the total mass of the Quadrocopter. The total thrust ${}^e\mathbf{F}$ consists of gravity \mathbf{G} , propeller pull ${}^b\mathbf{T}$ and aerodynamic force ${}^b\mathbf{F}_d$, which can be described as:

$$\begin{cases} {}^e\mathbf{F} = m\mathbf{G} + \mathbf{R}_b^e ({}^b\mathbf{T} + {}^b\mathbf{F}_d) \\ \mathbf{G} = \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T = g\mathbf{e}_3 \\ {}^b\mathbf{T} = \begin{bmatrix} 0 & 0 & -f \end{bmatrix}^T = -f\mathbf{b}_3 \end{cases} \quad (11)$$

where, $f \geq 0$ represents the magnitude of the total propeller pull, which is unidirectional (the case of negative pull due to variable pitch is not considered here); $g > 0$ is the acceleration of gravity. Further, since:

$${}^e\mathbf{v} = \mathbf{R}_b^e \cdot {}^b\mathbf{v} \quad (12)$$

Combining equation (11) and equation (12) yields:

$$\begin{cases} {}^b\dot{\mathbf{v}} = -[{}^b\omega]_x {}^b\mathbf{v} + {}^b\mathbf{F}/m \\ {}^b\mathbf{F} = (\mathbf{R}_b^e)^{-1} \cdot {}^e\mathbf{F} = m(\mathbf{R}_b^e)^{-1}\mathbf{G} + {}^b\mathbf{T} + {}^b\mathbf{F}_d \end{cases} \quad (13)$$

(b) *Quadrocopter attitude dynamics model:*

Based on assumptions 1 ~ 3, the attitude dynamics equations are established in the body-fixed coordinate system as follows:

$$\begin{cases} \mathbf{J} \cdot {}^b\dot{\omega} = -{}^b\omega \times (\mathbf{J} \cdot {}^b\omega) + {}^b\mathbf{M} \\ {}^b\mathbf{M} = \mathbf{G}_a + \tau + {}^b\mathbf{M}_d \end{cases} \quad (14)$$

where $\tau \triangleq \begin{bmatrix} \tau_x & \tau_y & \tau_z \end{bmatrix}^T \in \mathbb{R}^3$ denotes the moment generated by the propeller on the body axis, $\mathbf{J} \in \mathbb{R}^{3 \times 3}$ denotes the rotational inertia of the quadrocopter, $\mathbf{G}_a = \begin{bmatrix} G_{a,\phi} & G_{a,\theta} & G_{a,\psi} \end{bmatrix}^T \in \mathbb{R}^3$ denotes the gyroscopic moment and ${}^b\mathbf{M}_d \in \mathbb{R}^3$ denotes the pneumatic moment. The gyroscopic moment is related to the direction of propeller rotation. According to Assumption 5 and the definition of the coordinate system, the angular velocity vector of a single propeller is

$(-1)^k \varpi_k^b \mathbf{b}_3$ ($k = 1, \dots, 4$), where $\varpi_k > 0$ denotes the angular velocity of the k th propeller (rad/s). Thus, for a quadrocopter, the gyroscopic moment can be expressed as[25]:

$$\mathbf{G}_a = \sum_{k=1}^4 J_{RP} (\mathbf{b}\omega \times \mathbf{e}_3) (-1)^{k+1} \varpi_k \quad (15)$$

where $J_{RP} > 0$ ($\text{N} \cdot \text{m} \cdot \text{s}^2$) denotes the total rotational inertia of the entire motor rotor and propeller winding shaft

(c) *Aerodynamic model:*

Since both aerodynamic forces and moments are approximately proportional to the square of the velocity \mathbf{v}_a and the angular velocity $\mathbf{b}\omega_a$ and in the same direction, it is assumed in this paper that there is no air vortex effect. Let the rotational speed of the air relative to the body (as opposed to the angular velocity of the aircraft) be $\mathbf{b}\omega_a \approx -\mathbf{b}\omega \triangleq \begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix}^T$, then the air resistance and moments in equations (13) and (14) can be calculated by equation (16) as follows:

$$\left\{ \begin{array}{l} {}^b\mathbf{F}_d = C_d \begin{bmatrix} u|u| \\ v|v| \\ w|w| \end{bmatrix} \\ {}^b\mathbf{M}_d = C_{dm} \begin{bmatrix} \omega_x |\omega_x| \\ \omega_y |\omega_y| \\ \omega_z |\omega_z| \end{bmatrix} \end{array} \right. \quad (16)$$

Where C_d is the air drag coefficient and C_{dm} is the air damping moment coefficient, these aerodynamic parameters can be obtained by experimental measurement calculations.

4. Rigid Model of Quadcopter Control

Combining equation (7), equation (9), equation (13), equation (14), equation (15), and equation (16), the Rigid Model of Quadcopter Control can be obtained as follows:

(a) Euler Angle Model(using velocity in the Earth-fixed coordinate system):

$$\left\{ \begin{array}{l} {}^e\dot{\mathbf{p}} = {}^e\mathbf{v} \\ {}^e\dot{\mathbf{v}} = {}^e\mathbf{F}/m \\ {}^e\mathbf{F} = m\mathbf{G} + \mathbf{R}_b^e ({}^b\mathbf{T} + {}^b\mathbf{F}_d) \\ \mathbf{G} = \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T = g\mathbf{e}_3 \\ {}^b\mathbf{T} = \begin{bmatrix} 0 & 0 & -f \end{bmatrix}^T = -f\mathbf{b}_3 \\ \dot{\Theta} = \mathbf{W} \cdot {}^b\omega \\ \mathbf{J} \cdot {}^b\dot{\omega} = -{}^b\omega \times (\mathbf{J} \cdot {}^b\omega) + {}^b\mathbf{M} \\ {}^b\mathbf{M} = \mathbf{G}_a + \tau + {}^b\mathbf{M}_d \\ {}^b\mathbf{F}_d = C_d \begin{bmatrix} u|u| \\ v|v| \\ w|w| \end{bmatrix}, {}^b\mathbf{M}_d = C_{dm} \begin{bmatrix} \omega_x |\omega_x| \\ \omega_y |\omega_y| \\ \omega_z |\omega_z| \end{bmatrix} \end{array} \right. \quad (17)$$

(b) Quaternion Model:(using velocity in the body-fixed coordinate system):

$$\left\{ \begin{array}{l} {}^e\dot{\mathbf{p}} = {}^e\mathbf{v} = \mathbf{R}_b^e \cdot {}^b\mathbf{v} \\ {}^b\dot{\mathbf{v}} = -[{}^b\omega]_\times \cdot {}^b\mathbf{v} + {}^b\mathbf{F}/m \\ {}^b\mathbf{F} = m(\mathbf{R}_b^e)^{-1}\mathbf{G} + {}^b\mathbf{T} + {}^b\mathbf{F}_d \\ \dot{q}_0 = -\frac{1}{2}\mathbf{q}_v^T \cdot {}^b\omega \\ \dot{\mathbf{q}}_v = \frac{1}{2} (q_0 \mathbf{I}_3 + [\mathbf{q}_v]_\times) {}^b\omega \\ \mathbf{J} \cdot {}^b\dot{\omega} = -{}^b\omega \times (\mathbf{J} \cdot {}^b\omega) + {}^b\mathbf{M} \\ {}^b\mathbf{M} = \mathbf{G}_a + \tau + {}^b\mathbf{M}_d \\ {}^b\mathbf{F}_d = C_d \begin{bmatrix} u|u| \\ v|v| \\ w|w| \end{bmatrix}, {}^b\mathbf{M}_d = C_{dm} \begin{bmatrix} \omega_x |\omega_x| \\ \omega_y |\omega_y| \\ \omega_z |\omega_z| \end{bmatrix} \end{array} \right. \quad (18)$$

The Euler angle model is intuitively appealing and can be easily linearized for designing controllers. However, at large angles (such as pitch angles of $\theta = \pm\pi/2$), the Euler angle's attitude representation

can encounter singular problems. In contrast, quaternion's attitude representation maintains linearity of equations and eliminates singular problems. Compared to Euler angles, quaternion calculations are simpler, capable of achieving full attitude control of aircraft, and can be used to establish nonlinear mathematical models of quadcopters for simulation and modeling purposes.

* Control Efficiency Model

1. *Single propeller pull and counter-torque model:*

When a multi-rotor hovers in the absence of wind, its propeller pull can be expressed as[25]:

$$T_i = c_T \varpi_i^2 \quad (19)$$

where $c_T = 1/(4\pi^2) \cdot \rho D_p^4 C_T$ is constant and easily determined experimentally. The magnitude of the reverse torsional moment is expressed as[25]:

$$M_i = c_M \varpi_i^2 \quad (20)$$

2. *Tension and moment models:*

The flight of the quadcopter is driven by four propellers. The propeller speed $\varpi_i (i = 1, \dots, 4)$ determines the total pull f and moment τ of the quadrotor.

As shown in Figure 2, for an X-shaped

$$f = c_T (\varpi_1^2 + \varpi_2^2 + \varpi_3^2 + \varpi_4^2) \quad (21)$$

And the torque generated by the propellers is:

$$\begin{cases} \tau_x = dc_T \left(-\frac{\sqrt{2}}{2} \varpi_1^2 + \frac{\sqrt{2}}{2} \varpi_2^2 + \frac{\sqrt{2}}{2} \varpi_3^2 - \frac{\sqrt{2}}{2} \varpi_4^2 \right) \\ \tau_y = dc_T \left(\frac{\sqrt{2}}{2} \varpi_1^2 - \frac{\sqrt{2}}{2} \varpi_2^2 + \frac{\sqrt{2}}{2} \varpi_3^2 - \frac{\sqrt{2}}{2} \varpi_4^2 \right) \\ \tau_z = c_M (\varpi_1^2 + \varpi_2^2 - \varpi_3^2 - \varpi_4^2) \end{cases} \quad (22)$$

* Power Unit Model

As shown in Figure 7, the power unit model is the entire power structure consisting of a brushless motor, an electronic speed controller(ESC), and a propeller:

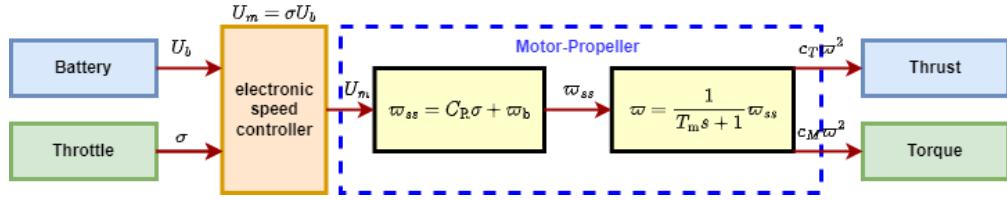


Figure 7 — Power unit signal transfer diagram

where the thrust command σ is a $0 \sim 1$ input signal, while the battery output voltage U_b is not controlled.

The ESC receives the throttle command σ and the battery output voltage U_b to produce an equivalent average voltage of $U_m = \sigma U_b$. In general, when given a thrust command, the dynamic process of a brushless DC motor can be reduced to a first-order low-pass filter with a transfer function of:

$$\varpi = \frac{1}{T_m s + 1} \varpi_{ss} \quad (23)$$

where T_m is the dynamic response time of the motor and ϖ_{ss} is the steady-state speed, which is linearly related to the input thrust command:

$$\varpi_{ss} = C_b U_m + \varpi_b = C_R \sigma + \varpi_b \quad (24)$$

where $C_R = C_b U_b, C_b$ and ϖ_b are constant parameters that can be measured experimentally.

Therefore, combining equations (35), (37) yields the complete power unit model as:

$$\varpi = \frac{1}{T_m s + 1} (C_R \sigma + \varpi_b) \quad (25)$$

2.3.4 Simplification of mathematical model of quadcopter

The nonlinear model of the quadcopter has been established in the previous section. As seen from equation (17), the quadcopter flight control system is a typical nonlinear system, and it also has the characteristics of underdrive, strong coupling, and high order, which makes the analysis and controller design of the quadcopter very complicated. Therefore, in order to facilitate the design of the controller, the nonlinear model needs to be simplified according to the

flight characteristics of the quadrotor, and then the controller is designed based on the simplified system model.

Based on the nonlinear model (17) introduced in the previous section, neglecting ${}^b\omega \times (\mathbf{J} \cdot {}^b\omega) + \mathbf{G}_{mathrma}$ and aerodynamic ${}^b\mathbf{F}_d$, aerodynamic moment ${}^b\mathbf{M}_d$, the following simplified model is obtained:

$$\left\{ \begin{array}{l} {}^e\dot{\mathbf{p}} = {}^e\mathbf{v} \\ {}^e\dot{\mathbf{v}} = g\mathbf{e}_3 - \frac{f}{m}\mathbf{R}_b^e\mathbf{e}_3 \\ \dot{\Theta} = \mathbf{W} \cdot {}^b\omega \\ \mathbf{J} \cdot {}^b\dot{\omega} = \tau \end{array} \right. \quad (26)$$

Assuming that the quadcopter's flight characteristics are characterized by extremely small pitch and roll angles, and that the total thrust generated by the propellers is roughly equal to the weight of the quadcopter. These assumptions can be further articulated as follows:

$$\sin \phi \approx \phi, \quad \cos \phi \approx 1, \quad \sin \theta \approx \theta, \quad \cos \theta \approx 1, \quad f \approx mg. \quad (27)$$

At this point, equation (26), the matrix \mathbf{W} in equation (27.2) is approximated by the unit matrix \mathbf{I}_3 , and $\mathbf{R}\mathbf{e}_3$ in equation (26) becomes:

$$\mathbf{R}\mathbf{e}_3 \approx \begin{bmatrix} \theta \cos \psi + \phi \sin \psi \\ \theta \sin \psi - \phi \cos \psi \\ 1 \end{bmatrix} \quad (28)$$

The original simplified model (26) is decoupled into three linear models, namely, the horizontal position channel model, the altitude channel model, and the attitude channel model:

1. Horizontal Position channel model

According to the small angle assumption (27) and ignoring the higher order terms, the first two equations in equation (26) simplify to:

$$\left\{ \begin{array}{l} \dot{\mathbf{p}}_h = \mathbf{v}_h \\ \dot{\mathbf{v}}_h = -g\mathbf{A}_\psi\Theta_h \end{array} \right. \quad (29)$$

where:

$$\mathbf{p}_h = \begin{bmatrix} x \\ y \end{bmatrix}, \mathbf{R}_\psi = \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix}, \mathbf{A}_\psi = \mathbf{R}_\psi \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \Theta_h = \begin{bmatrix} \phi \\ \theta \end{bmatrix}. \quad (30)$$

2. Height channel model

According to the small angle assumption (27), the third equation in equation (26) can be simplified to:

$$\begin{cases} \dot{z} = v_z \\ \dot{v}_z = g - \frac{f}{m} \end{cases} \quad (31)$$

3. Attitude channel model

Combining the last two equations in (27), the following attitude channel model can be obtained:

$$\begin{cases} \dot{\Theta} = \omega \\ \mathbf{J}\dot{\omega} = \tau \end{cases} \quad (32)$$

2.3.5 Quadcopter MATLAB simulation model

In this paper, the quadrocopter control nonlinear mathematical model is established in MATLAB/Simulink, which is used in the simulation experiment of the quadrocopter control system in the following paper to get closer to the real situation. The hardware parameters of the quadrocopter flight test measure the actual parameters of the F450 frame, and the main parameters are shown in the following table:

Table 1 — Quadcopter Parameters

parameter	R	J_x	J_x	J_y
value	0.255	0.0254	0.0298	0.0516
parameter	m	C_r	W_b	T_m
value	0.0516	1148	-141.4	0.02
parameter	C_d	J_m	C_m	C_t
value	0.0298	0.0001287	$7.7803e - 07$	$9.3144e - 06$

2.4 Chapter Summary

This chapter first introduces the two structural forms of quadcopters (cross-shaped and X-shaped) and their performance differences. X-shaped quadcopter structure is chosen as the research object for the following sections. In order to facilitate the design of quadcopter controllers, the four motion modes of quadcopters (vertical, yaw, pitch, and roll) are analyzed.

Secondly, in order to establish a mathematical model of quadcopters in the following sections, the earth-fixed coordinate system and the body-fixed coordinate system are established, and three attitude representation methods of quadcopters (Euler angles, rotation matrices, and quaternions) are introduced.

Furthermore, the force analysis of quadcopters is conducted to establish a rigid body model of quadcopter control. The control efficiency model is established by analyzing the tension and torque generated by the quadcopter propellers. The power unit model is established by analyzing the dynamic unit (motor, electronic speed controller, and propeller) of the quadcopter. The three models are coupled to obtain the quadcopter control nonlinear mathematical model.

Finally, in order to facilitate controller design, reasonable assumptions are proposed based on the flight characteristics of quadcopters. The quadcopter control nonlinear mathematical model is appropriately simplified and linearized to obtain a simplified linear model.

3 PID-BASED QUADCOPTER CONTROLLER DESIGN

With the continuous development of technology, control objects are becoming increasingly complex. However, control theory has also become more mature, solving many traditional control problems. Despite this, classic control techniques such as PID are still widely used in industrial control and are the most commonly used control technology in open-source quadcopter flight controllers. They have achieved significant results in various practical flight control tasks.

In this chapter, we first introduce PID control theory and design a cascaded PID controller based on the mathematical model of the quadcopter flight controller established in Chapter 2. We then validate the effectiveness of this controller in executing the fixed-point hovering task through MATLAB/Simulink simulation experiments. Finally, we test the controller's anti-interference ability and robustness in MATLAB/Simulink simulation experiments.

3.1 PID control theory

PID control is one of the earliest developed control strategies. It is widely applied in various industries due to its simple algorithm, good robustness, and high reliability. It is a linear control method that controls the controlled object by linearly combining the proportional (P), integral (I), and derivative (D) of the deviation to form the control quantity[26]. Therefore, it is called a PID controller. The principle block diagram of PID control is shown in Figure 8.

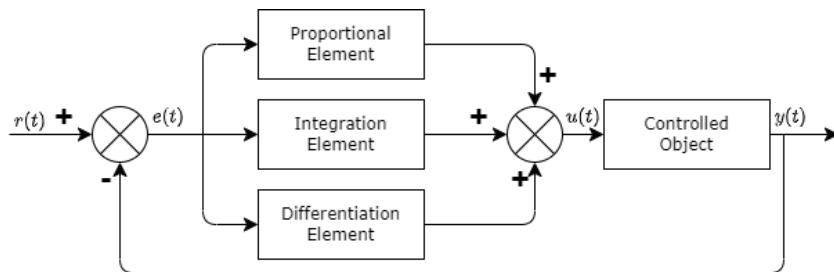


Figure 8 — PID control principle block diagram

The PID constitutes a control deviation $e(t)$ based on the given reference signal value $r(t)$ and the actual output value $y(t)$ of the controlled object:

$$e(t) = r(t) - y(t) \quad (33)$$

The mathematical expression of the PID controller is:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (34)$$

where K_p, K_i, K_d are the proportional, integral, and differential coefficients in the PID controller, respectively.

Briefly, the role of each correction link of the PID controller is as follows:

1. **Proportional Element:** The proportional element of control considers the current error and reflects the deviation signal $e(t)$ in proportion to the control system. Once the deviation occurs, the controller immediately generates a control action to reduce the error.
2. **Integral Element:** The integral element of control considers past errors and is mainly used to eliminate steady-state errors and improve system stability.
3. **Derivative Element:** The derivative component of control considers future errors and reflects the trend (rate of change) of the deviation signal $e(t)$. It can introduce an effective early correction signal into the system before the deviation signal becomes too large, thereby accelerating the system's response and reducing the tuning time. However, it is important to note that the derivative component can amplify high-frequency noise, which can decrease the controlled object's anti-interference performance[27].

3.2 Cascade PID controller design

From the (2.3.3) section, it can be seen that the quadcopter is a multi-subsystem controlled object, by the figure 6 and the simplified linear model of the quadcopter 2.3.4 attitude and position analysis of the coupling relationship in its modeling structure can be divided into $z, \psi, (x, y - \theta\phi)$ three channels, and then the controller for these three channels are designed separately. The inner loop is the control of the three attitude angles θ, ϕ, ψ , under the attitude control can adjust the current flight attitude of the aircraft, the outer loop control is the control of the three positions x, y, ψ of the aircraft. The outer loop control is the control of three positions x, y, z of the vehicle.

The cascade PID controller designed in this section uses different control methods for different channels in order to maintain a balance between control effectiveness and controller complexity. The yaw angle ψ and altitude z channels are controlled by a single PID, and the horizontal position $(x, y - \theta, \phi)$ channel is controlled by an internal and external dual-loop cascade PID. The complete Cascade PID quadrotor control system structure is shown in the following figure 9, where the variables with subscript d (x_d, y_d, z_d , etc.) denote the target values of the variables.

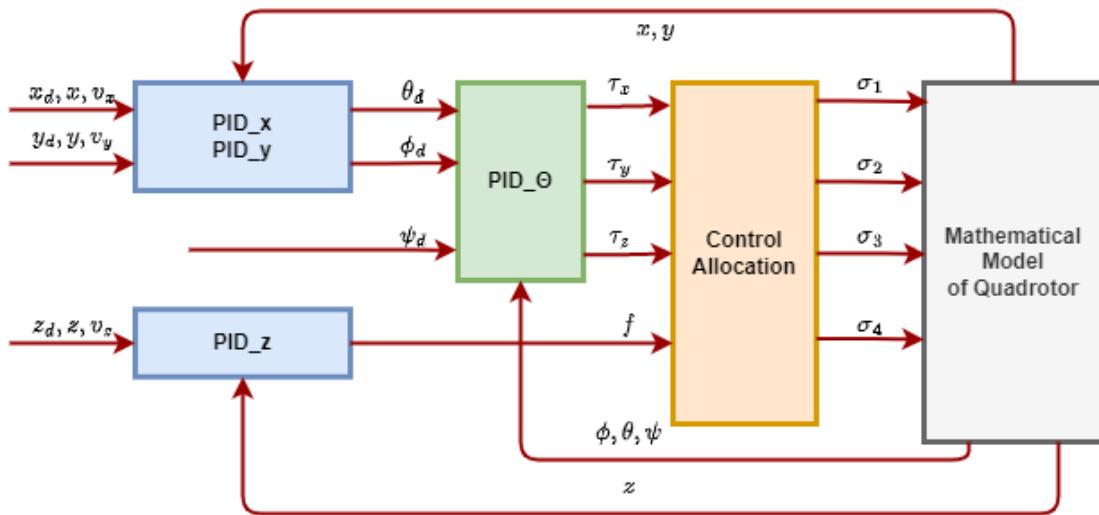


Figure 9 — pic/Cascade PID control system structure

3.3 Simulation experiment of cascade PID controller for quadcopter

3.3.1 Fixed-point hovering experiment

The quadcopter is capable of executing a series of tasks, such as aerial photography, lighting, and filming, by hovering in place. Therefore, the ability to hover in place is one of the important indicators for evaluating the performance of the quadcopter controller.

In this paper, the quadcopter takes off from the coordinate origin $(x_0, y_0, z_0) = (0, 0, 0)$ and the target position is set to $(x_d, y_d, z_d) = (1, 1, -1)$. After multiple rounds of parameter adjustments and optimizations, the simulation results of the control system's various channels are shown in Figures 10 to 11:

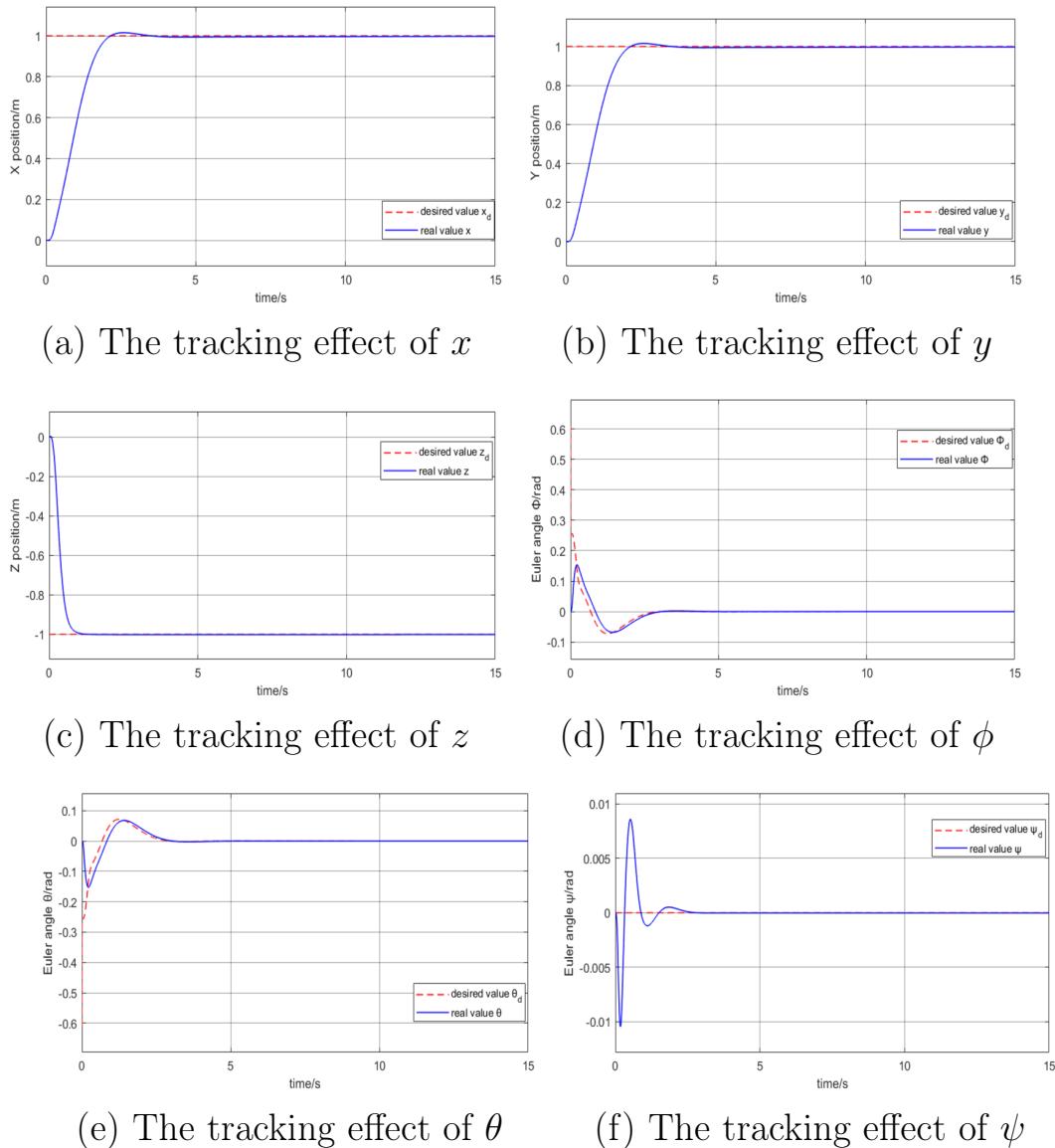


Figure 10 — Fixed-point hovering simulation experiment results graph

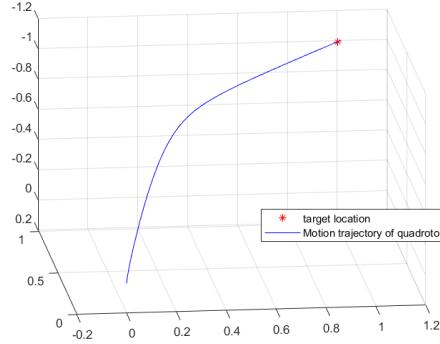


Figure 11 — Quadcopter 3D space motion trajectory

From the position and attitude simulation tracking curves Figures 10 to 11 of the quadcopter, it can be seen that under the control of the cascaded PID controller, the time required for the x, y, z positions to track the given expected values is very short. The adjustment time for position x is about $3.3s$, and the adjustment time for position y is about $3.4s$, but both channels have slight overshoot. The control effect of the z channel is relatively good, with a small adjustment time of $1.1s$ and no overshoot. The tracking effect of pitch angle, roll angle, and yaw angle is also good, especially since the expected values of pitch angle and roll angle are not fixed values, but values that change based on the expected acceleration, which makes them more difficult to control. However, they can also reach stability quickly, meeting the requirements of the quadcopter for stability. This proves that the cascaded PID controller has good fixed-point hovering ability under the condition of no interference.

3.3.2 Anti-interference ability and robustness experiments

During the flight of a quadcopter, it can be subject to internal or external uncertainties, making the controller's anti-interference and robustness important indicators for evaluating its performance.

In this section, in order to verify the anti-interference and robustness of the designed controller, each channel of the quadcopter $x, y, z, \phi, \theta, \psi$, was subjected to Gaussian white noise with different average amplitudes μ to simulate the effects of unknown external and internal interferences. Under the same conditions as the previous section, the simulation experiment yielded results for each channel of the control system, as shown in Figures 12 to 17:

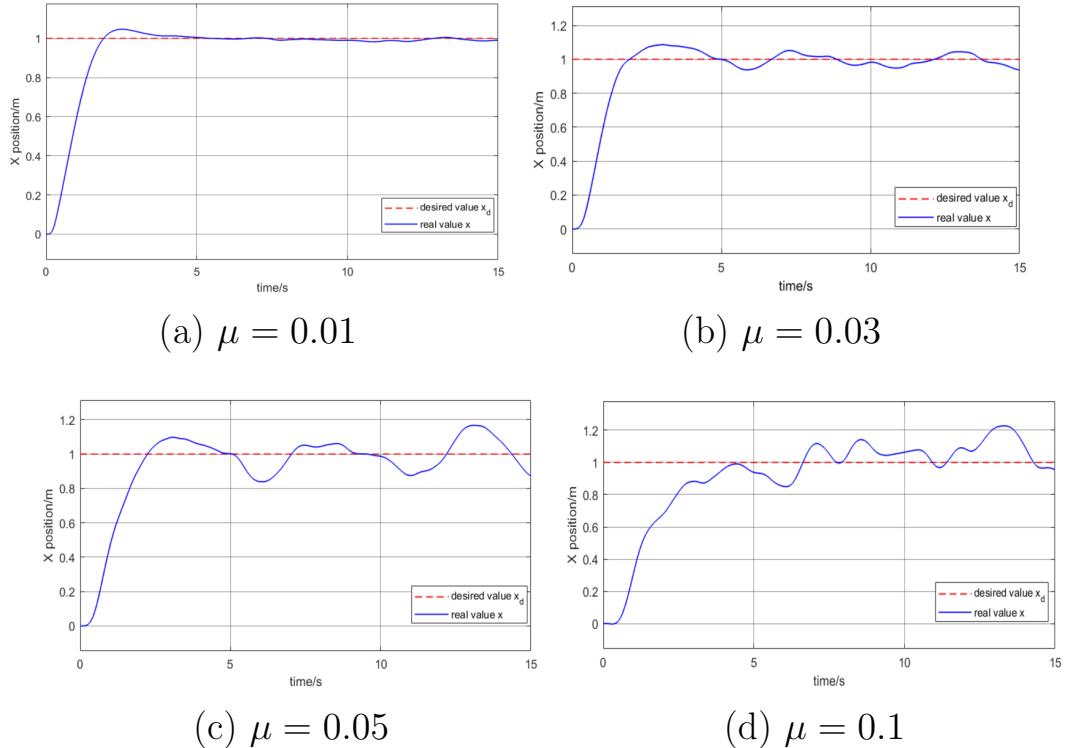


Figure 12 — The tracking effect of x under Gaussian white noise with different average amplitude μ

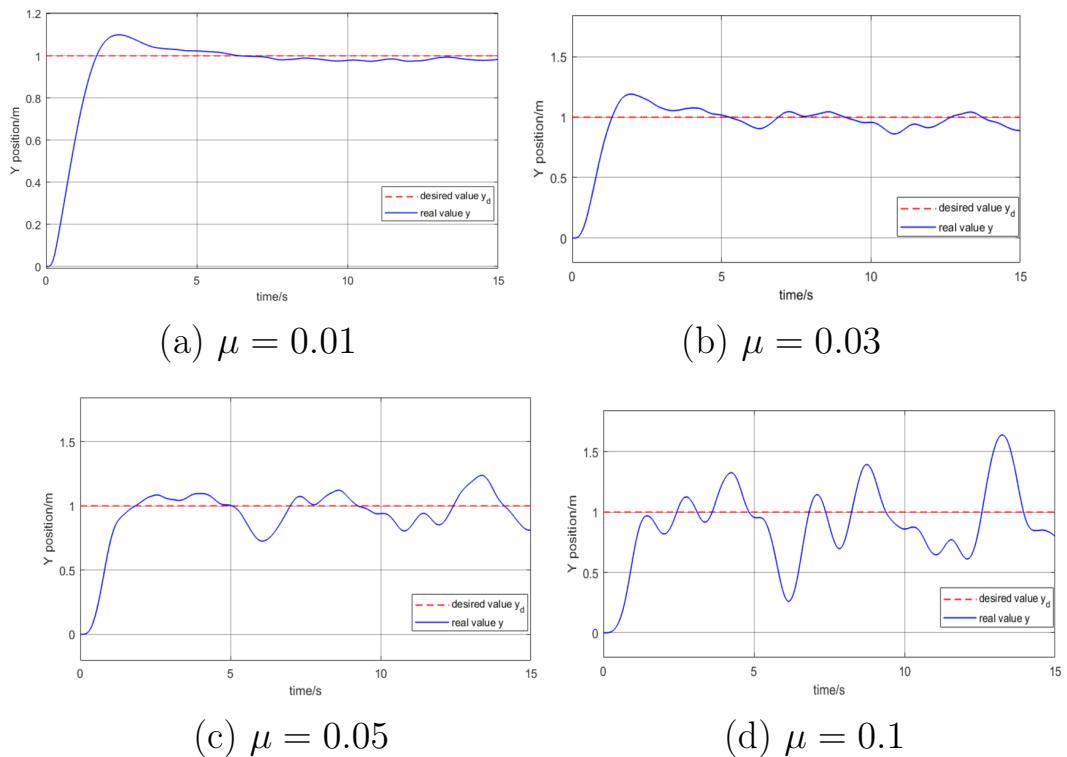


Figure 13 — The tracking effect of y under Gaussian white noise with different average amplitude μ

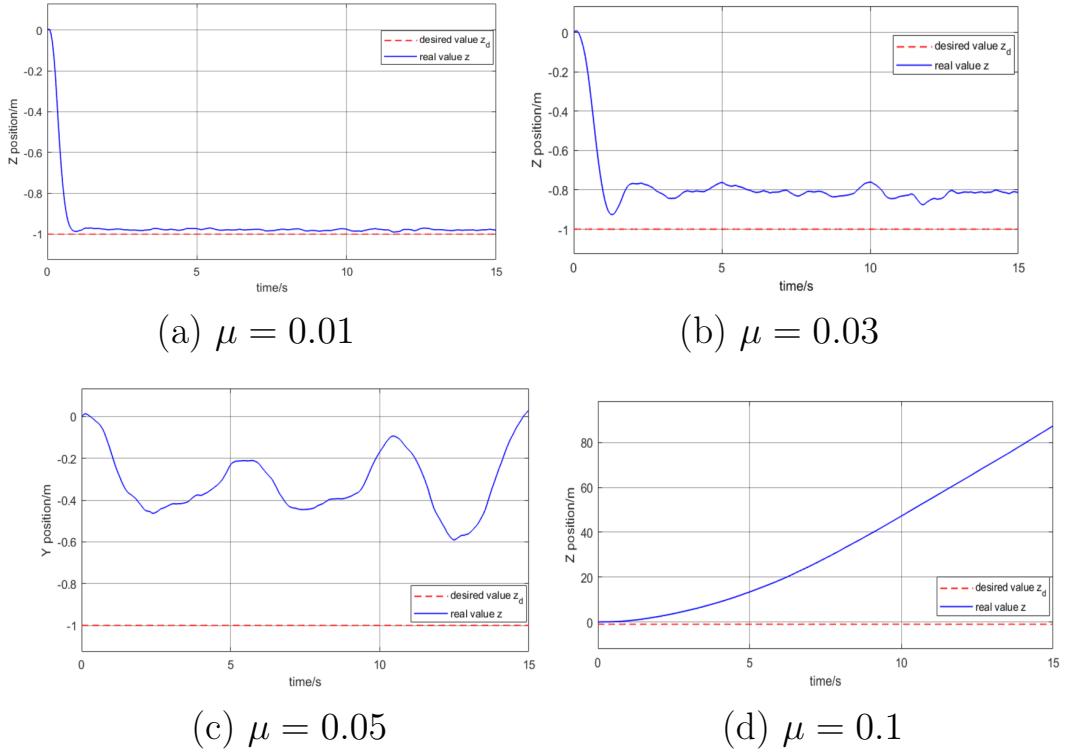


Figure 14 — The tracking effect of z under Gaussian white noise with different average amplitude μ

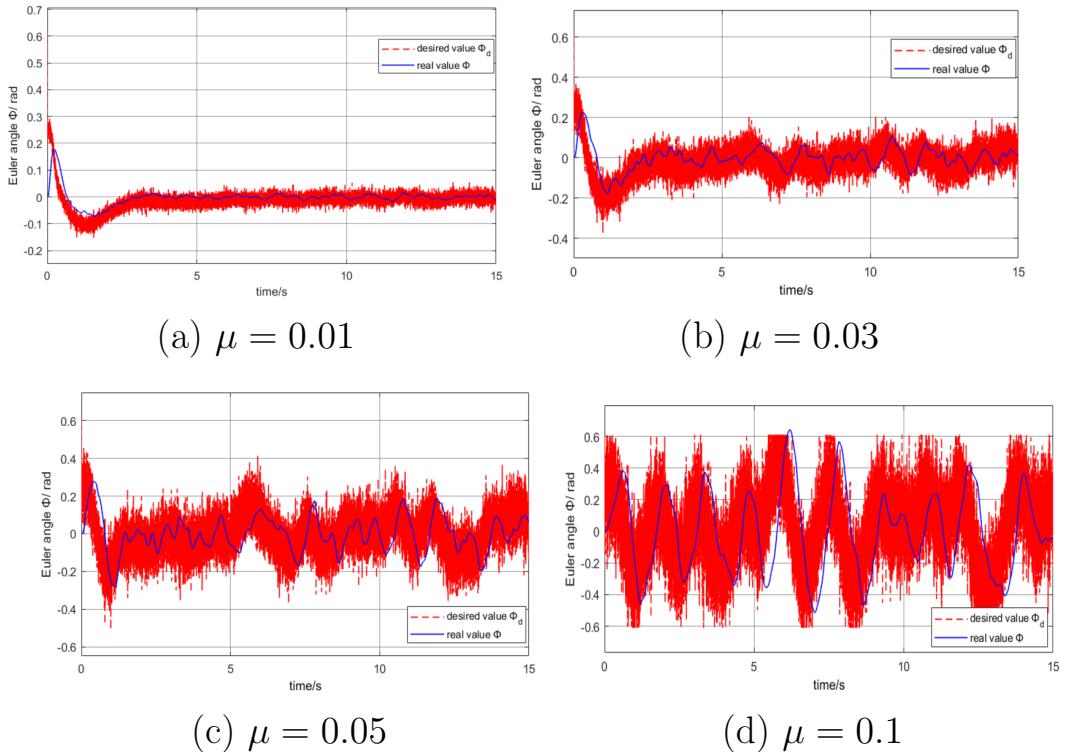


Figure 15 — The tracking effect of ϕ under Gaussian white noise with different average amplitude μ

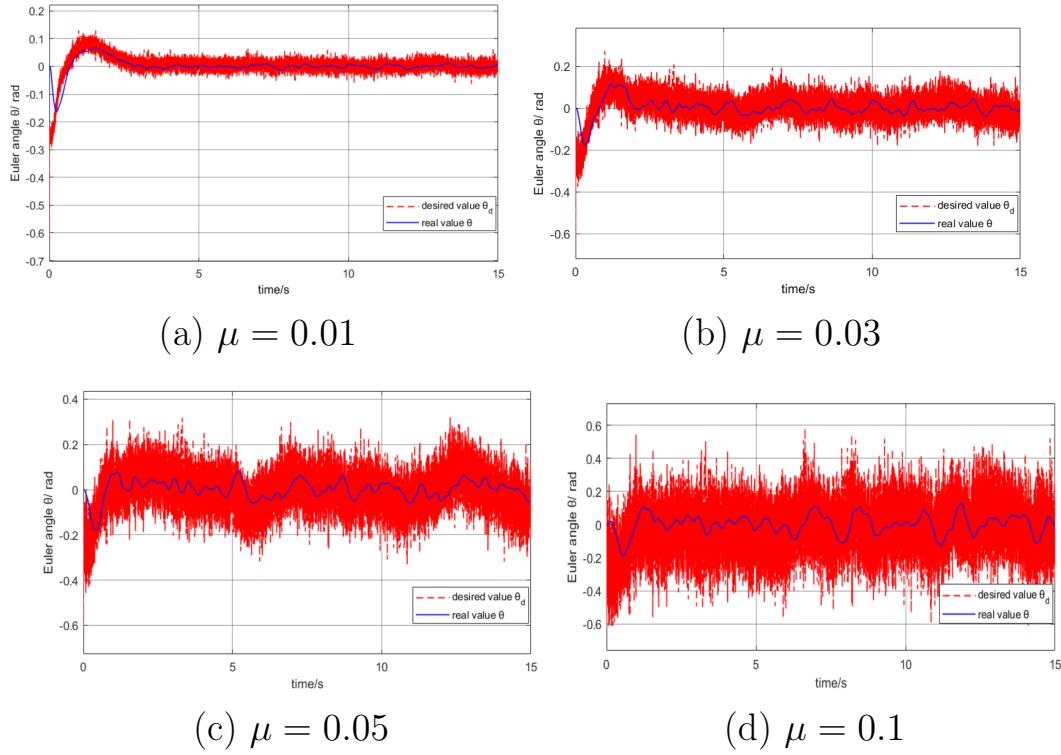


Figure 16 — The tracking effect of θ under Gaussian white noise with different average amplitude μ

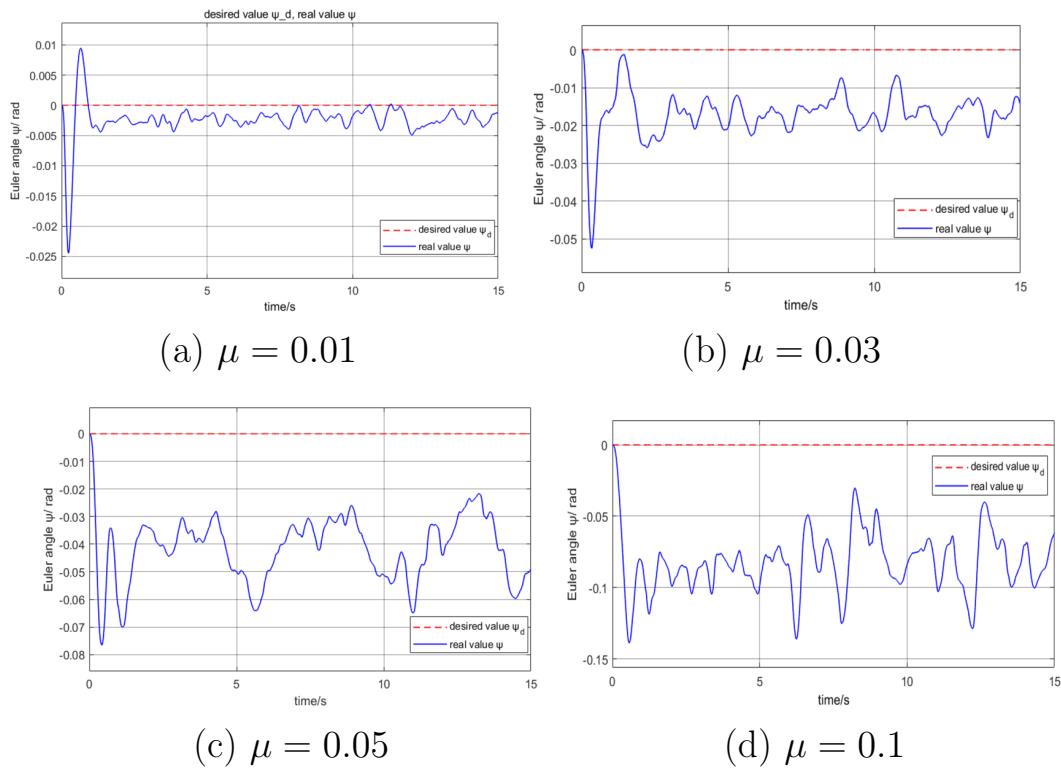


Figure 17 — The tracking effect of ψ under Gaussian white noise with different average amplitude μ

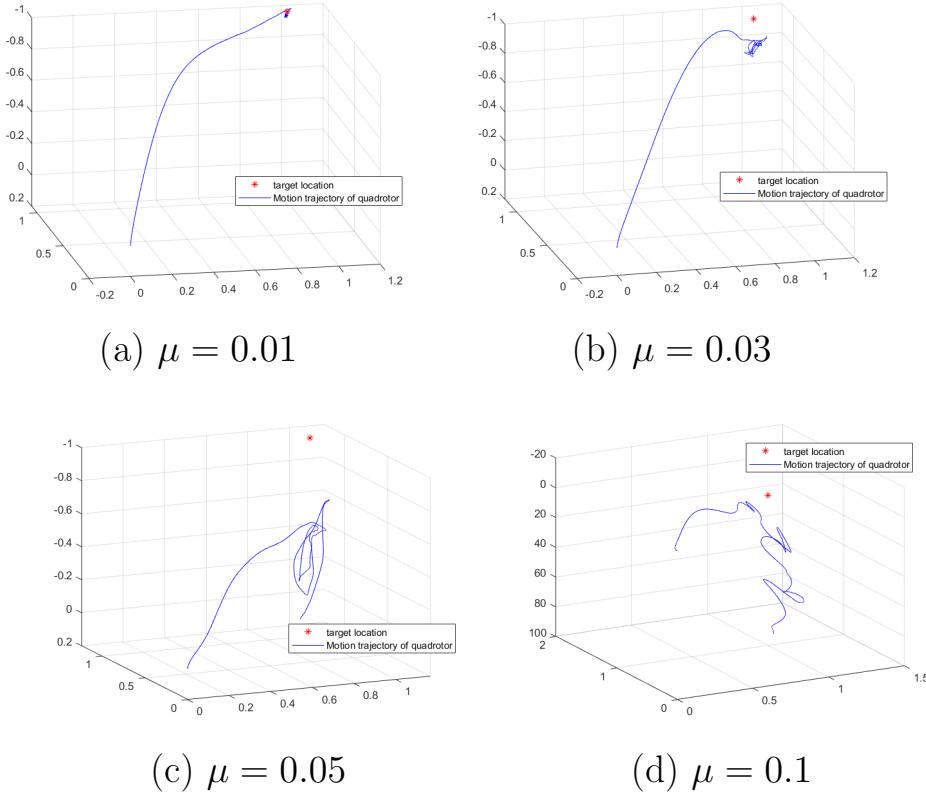


Figure 18 — 3D motion trajectory of quadcopter under Gaussian white noise with different average amplitude μ

From the simulation results in Figures 12-18 of the above anti-interference experiment, it can be seen that:

Compared with the fixed-point hovering experiment without considering interference, when the quadcopter is subjected to interference within a certain range, although the state adjustment time during its flight process is longer and the overshoot increases, its output value can still fluctuate above and below the target value. At this time, the quadcopter's state can be controlled within an acceptable range of error, thus proving that the quadcopter's cascaded PID controller has a certain anti-interference ability.

However, as the interference received by the quadcopter increases (taking the Gaussian white noise interference z channel with an average amplitude of $\mu = 0.1$ as an example), the system becomes uncontrollable and the flight state is not within an acceptable range of error. This indicates that the traditional cascaded PID control has weak anti-interference ability and cannot cope with more complex flight environments.

3.4 Chapter Summary

In this chapter, we first introduce the classical PID control principle and analyze in detail the roles of its three elements: proportional, integral, and derivative.

Next, we analyze the coupling relationship among the channels of the simplified linear model of the quadrotor aircraft and design a channel-by-channel cascaded PID controller based on PID control theory.

Then, we build a quadcopter simulation system based on the cascaded PID controller in MATLAB/Simulink and demonstrate the effectiveness of the designed cascaded PID controller by controlling the quadcopter aircraft to perform a fixed-point hovering task.

Finally, considering that each channel of the quadcopter is affected by certain unknown external and internal interferences, we add Gaussian white noise with different mean amplitudes μ to each of the six channels ($x, y, z, \phi, \theta, \psi$) of the quadrotor aircraft in MATLAB/Simulink. We test the robustness and anti-interference ability of the cascaded PID controller, and the experiment shows that when the interference on the quadrotor aircraft is small, its state can be maintained within an acceptable range of error. However, when the interference is large, the system becomes uncontrollable, indicating that the cascaded PID controller has certain anti-interference ability and robustness, but its anti-interference ability and robustness are not strong enough to cope with more complex flight environments.

4 LADRC-BASED QUADCOPTER CONTROLLER DESIGN

From the experiments on the anti-interference ability and robustness of the cascaded PID controller in Section 3.3.2, it can be seen that the anti-interference ability of the PID control algorithm is weak. Although modern control techniques such as adaptive control, robust control, and fuzzy control can improve certain control qualities of the system, these control algorithms require the mathematical model of the controlled object to be precise and its parameters to be known. However, due to the structural errors and small size and light weight of the quadcopter, it is difficult to establish an accurate model as its aerodynamic characteristics are easily affected by inherent and external factors such as motor vibration and airflow changes during flight. Therefore, this chapter will choose a control method that can solve the uncertainty problem of the system and does not depend on the controlled object: the Linear Active Disturbance Rejection Control (LADRC) method.

Firstly, this chapter studies the theoretical knowledge and composition structure of traditional Active Disturbance Rejection Control (ADRC), and elaborates on the improved linear control algorithm of ADRC - LADRC and its parameter tuning method, which is aimed at the problem of too many parameters in traditional ADRC algorithm. Secondly, based on the theoretical object and the simplified linear model of the quadcopter, a cascaded LADRC controller is designed for the quadcopter. Then, the effectiveness of this controller in performing the fixed-point hovering task is verified through MATLAB/Simulink simulation experiments. Finally, the anti-interference ability and robustness of this controller are tested through MATLAB/Simulink simulation experiments.

4.1 Overview of Active Disturbance Rejection Control (ADRC) technology

The advantage of PID control is that it is a "control strategy based on eliminating errors by using errors", which is different from modern control the-

ory and does not rely on "internal mechanism description". However, there are also several disadvantages[28, 29]:

1. The method of error signal extraction. It is unreasonable to directly take the error between the expected value and the output value as the error signal when requiring the slow-changing output to track the expected input value. This may cause the control rate of the system to be too large, resulting in large overshoot.
2. The method of extracting the differential term de/dt from the error signal $e(t)$. The differential term de/dt in the PID controller is not easy to obtain, which limits the control ability of PID and may cause the system to produce large noise.
3. The method of "weighted sum of errors" may not be the best, and there may be better control variables.
4. The integral feedback loop has many side effects. For example, in the absence of disturbances, the integral loop in the PID controller will reduce the dynamic performance of the closed-loop feedback control system, and its suppression effect on disturbances that change over time is not significant.

Through the efforts of Han Jingqing and others, a series of new practical controllers have been proposed based on the new structures that can process and control the system signals, such as the tracking differentiator and the extended state observer: nonlinear PID, Active Disturbance Rejection Controller (ADRC), and self-optimizing self-disturbance rejection controller, etc. Among these controllers, the Active Disturbance Rejection Controller (ADRC) is a typical representative, so they are collectively referred to as "Active Disturbance Rejection Control technology".[19] This technology has low dependence on the controlled object, and does not require specific measurement of the external disturbance, essentially breaking the limitations of the "absolute invariance principle" and "internal model principle"[29]. Active Disturbance Rejection Control technology inherits the core spirit of classical PID control theory, while integrating the advantages of modern control theory into the "observation + compensation" control method. It can effectively solve the problems of non-linearity and uncertainty in the controlled system, and does not rely on the

mathematical model of the controlled object, thereby exhibiting superior control performance.

The specific method of ADRC controller to overcome the "disadvantages" of PID is as follows:

1. Arrange an appropriate "transient process". The ADRC controller arranges a transient process through the tracking differentiator (TD) to solve the problem of overshoot caused by the easy jump of the expected value.
2. Reasonably extract "differential" - the ADRC controller extracts the differential signal through the "tracking differentiator" (TD) process, solving the problem that the differential term de/dt in the PID controller is not easy to obtain.
3. Explore appropriate control variable combination methods. By using the nonlinear state error feedback law (NLSEF) to nonlinearly combine the error signals, ADRC can achieve higher control performance.
4. Explore the method of "disturbance estimation". The ADRC controller estimates and compensates for the total disturbance composed of the unmodeled part of the controlled object and external random interference through the extended state observer (ESO), enabling ADRC to achieve zero steady-state error control without integral feedback loop.

4.2 Active Disturbance Rejection controller (ADRC) structure

Based on the idea of ADRC introduced in the previous section, the ADRC structure is shown in the following figure 19, taking the second-order self-rejecting controller as an example:

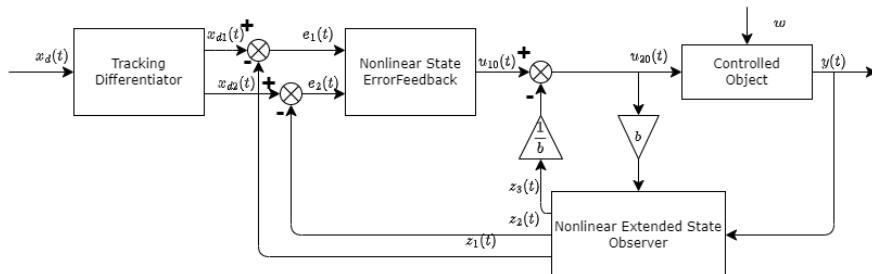


Figure 19 — Block diagram of second order ADRC structure

As can be seen from the above figure 19, the ADRC controller is mainly composed of three parts: tracking differentiator (TD), nonlinear extended state observer (NESO), and nonlinear state error feedback control law (NLSEF). The mathematical models and functions of these three parts are introduced below:

1. Second Order Tracking Differentiator (TD)

TD effectively solves the problem of overshoot caused by the fluctuation of expected values and the difficulty in obtaining the differential term de/dt in PID controllers, and also enhances the robustness of ADRC. When tracking the step signal input of $x_r(t) = 1$ with the tracking differentiator (TD), the output is softened to $x_{r1}(t), x_{r2}(t)$, where $x_{r1}(t)$ tracks the target value and $x_{r2}(t)$ approximates the differential of the target value $x_{r1}(t)$. The discrete mathematical expression of TD is as follows:

$$\begin{cases} e(k) = x_{r1}(k) - x_r(k) \\ x_{r1}(k+1) = x_{r1}(k) + h \cdot x_{r2}(k) \\ x_{r2}(k+1) = x_{r2}(k) + h \cdot fst(e(k), x_{r2}(k), r, h_0) \end{cases} \quad (35)$$

In this context, $e(k)$ represents the error; h is the step size; r is the velocity factor that affects the tracking speed of TD on the input signal. As r increases, the tracking speed becomes faster, the transition time becomes shorter, but the noise amplification also becomes stronger. h_0 is the filtering factor that has a filtering effect. The larger the h_0 , the better the filtering effect, but the phase loss of the tracking signal also increases accordingly. Generally, h_0 is taken as the step size. If there is noise input, it can be taken as an integer multiple of the step size. fst is the expression of the fastest control synthesis function, as shown below:

$$\begin{cases} d = rh_0 \\ d_0 = dh_0 \\ y = x_1 + h_0 x_2 \\ a_0 = (d^2 + 8r|y|)^{\frac{1}{2}} \\ a = \begin{cases} x_2 + \frac{1}{2}(a_0 - d), |y| > d_0 \\ x_2 + \frac{y}{h_0}, |y| \leq d_0 \end{cases} \\ fst = \begin{cases} \frac{-ra}{d}, |a| \leq d \\ -r \cdot \text{sign}(a), |a| > d \end{cases} \end{cases} \quad (36)$$

Where, the definition of $\text{rand}h_0$ is consistent with Equation (35). The function of fst is to avoid high-frequency chattering.

2. Nonlinear Extended State Observer (NESO)

The extended state observer (ESO) is a powerful tool that enables real-time tracking of state variables in a controlled system. By utilizing input and output values, the ESO can accurately estimate the size of disturbances within the system, even without direct measurement. This allows for effective compensation and improved overall control.[30].

Taking the second-order controlled object as an example, the equation of state of the system is as follows:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f(x_1, x_2, t, w) + bu \\ y = x_1 \end{cases} \quad (37)$$

Where, u represents the system input, b represents the system model parameters, y represents the system output, and w represents the external disturbance applied to the controlled object. f represents the total disturbance of the system.

By expanding the uncertain part of the second-order system into a new state variable, x_3 , we have:

$$x_3 = f(x_1, x_2, t, w) \quad (38)$$

Then equation (37) can be rewritten as:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_3 + bu \\ \dot{x}_3 = h \\ h = \dot{f}(x_1, x_2, t, w) \\ y = x_1 \end{cases} \quad (39)$$

Constructing discrete expansion state observers for the above system

equation (39):

$$\begin{cases} e(k) = z_1(k) - y(k) \\ fe_1 = fal(e(k), \alpha_1, \delta), fe_2 = fal(e(k), \alpha_2, \delta) \\ z_1(k+1) = z_1(k) + h(z_2(k) - \beta_{01} \cdot e(k)) \\ z_2(k+1) = z_2(k) + h(z_3(k) - \beta_{02} \cdot fe_1 + b \cdot u(k)) \\ z_3(k+1) = z_3(k) - h \cdot \beta_{03} \cdot fe_2 \end{cases} \quad (40)$$

In this equation, $\beta_{01}, \beta_{02}, \beta_{03} > 0$ are adjustable gains of the expanded state observer (ESO), u is the system input, and b is the coefficient of the control variable. z_1, z_2, z_3 are the estimated values of the state variables x_1, x_2, x_3 , respectively. a_1, a_2 are parameters of the FAL function, which are usually chosen to be $0 < a_1, a_2 < 1$, typically with $a_1 = 0.5$ and $a_2 = 0.25$. δ is the *fal* function [31], as expressed in the following citation:

$$fal(e, \alpha, \delta) = \begin{cases} |e|^\alpha \cdot \text{sign}(e), |e| > \delta \\ \frac{e}{\delta^{1-\alpha}}, |e| \leq \delta \end{cases} \quad (41)$$

3. Nonlinear State Error Feedback (NLSEF)

The nonlinear state error feedback (NLSEF) is the initial control quantity of the controlled object and its discrete mathematical expression is shown below:

$$\begin{cases} e_1(k) = x_{r1}(k) - z_1(k) \\ e_2(k) = x_{r2}(k) - z_2(k) \\ u_0 = \gamma_1 \cdot fal(e_1(k), \alpha_1, \delta) + \gamma_2 \cdot fal(e_2(k), \alpha_2, \delta) \end{cases} \quad (42)$$

Where, the variable u_0 represents the initial control input, while γ_1 and γ_2 are adjustable parameters. Specifically, γ_1 is used to regulate the system's response speed, while γ_2 is used to suppress excessive overshoot during operation.

To eliminate disturbances in the system, the total disturbance estimated by the ESO, denoted as z_3 , can be compensated for by using Equation (43) to obtain the actual control input.

$$u(k) = u_0(k) - \frac{z_3(k)}{b} \quad (43)$$

4.3 Linear Active Disturbance Rejection Controller (LADRC) structure

Due to the large number of parameters involved and the lack of theoretical proof for closed-loop stability in non-linear ADRC algorithms, Mr. Gao Zhi-jiang proposed an improved linear ADRC control algorithm, LADRC, based on classical ADRC principles in 2003. LADRC mainly replaces the non-linear functions in the linear extended state observer with linear functions. Additionally, Mr. Gao provided a relationship between β_{01} , β_{02} , β_{03} and the system bandwidth, reducing the number of parameters that need to be tuned and making it easier to understand and implement. The LADRC controller mainly consists of the linear extended state observer (LESO) and the linear error compensation control rate (LSEF)[32, 33]. The structure diagram of the second-order LADRC controller is shown in Figure20:

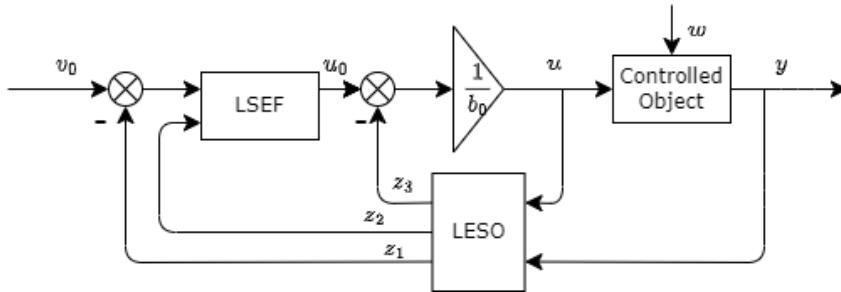


Figure 20 — Structure schematic of second-order LADRC

Let the expression for the second-order controlled object be:

$$\ddot{y} = f_1(t, y, \dot{y}, u, \dot{u}, w) + bu \quad (44)$$

where u is the system input, b is the parameter of the system model, y is the system output, and w is the external perturbation acting on the controlled object. f_1 is the total perturbation of the system.

If the parameter b in the system model is unknown, it can be roughly estimated by the estimate of b_0 , then equation (44) can be rewritten as:

$$\ddot{y} = f_1(t, y, \dot{y}, u, \dot{u}, w) + (b - b_0)u + b_0u = f + b_0u \quad (45)$$

Here, $f = f_1 + (b - b_0)u$ is referred to as a generalized disturbance or disturbance. Following the principle of Active Disturbance Rejection Control,

the uncertain part of the second-order system can be expanded into a new state variable, $x_3 = f$, and let $\dot{f} = h$ be an unknown disturbance. Then, equation (45) can be written in the form of a state equation:

$$\left\{ \begin{array}{l} \dot{x}_1 = y \\ \dot{x}_2 = \dot{y} + b_0 u \\ \dot{x}_3 = f \\ h = \dot{f} \\ y = x_1 \end{array} \right. \rightarrow \left\{ \begin{array}{l} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ b_0 \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} h \\ y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \end{array} \right. \quad (46)$$

Based on the ideas of ADRC discussed in the previous section, we can design an extended state observer for f using the state equation (46) of the expanded system, as shown below:

$$\left\{ \begin{array}{l} \dot{x} = \mathbf{A}x + \mathbf{B}u + \mathbf{E}h \\ y = \mathbf{C}x + \mathbf{D}u \end{array} \right. \rightarrow \left\{ \begin{array}{l} \dot{z} = \mathbf{A}z + \mathbf{B}u + \mathbf{L}(x_1 - z_1) \\ \hat{y} = \mathbf{C}z \end{array} \right. \quad \mathbf{L} = \begin{bmatrix} \beta_1 & \beta_2 & \beta_3 \end{bmatrix}^T \quad (47)$$

The state space model of the Linear Expanded State Observer is obtained by simplifying and expanding equation (47):

$$LESO : \left\{ \begin{array}{l} \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} -\beta_1 & 1 & 0 \\ -\beta_2 & 0 & 1 \\ -\beta_3 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} + \begin{bmatrix} 0 & \beta_1 \\ b_0 & \beta_2 \\ 0 & \beta_3 \end{bmatrix} \begin{bmatrix} u \\ y \end{bmatrix} \\ \hat{y} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} z \end{array} \right. \quad (48)$$

The observer gain vector $\mathbf{L} = \begin{bmatrix} \beta_1 & \beta_2 & \beta_3 \end{bmatrix}^T$ can be obtained through pole placement method [34].

The characteristic polynomial corresponding to the third-order linear observer in equation (48) can be expressed as:

$$s^3 + \beta_1 s^2 + \beta_2 s + \beta_3 \quad (49)$$

Configuring the system poles at ω_0 yields the expression:

$$(s + \omega_0)^3 = s^3 + 3\omega_0 s^2 + 3\omega_0^2 s + \omega_0^3 \quad (50)$$

Combining equations (49) and (50), we have:

$$\beta_1 = 3\omega_0, \beta_2 = 3\omega_0^2, \beta_3 = \omega_0^3 \quad (51)$$

Over time, z_1 for y observations converge to y , z_2 for \dot{y} observations converge to \dot{y} , and z_3 for f observations converge to f , i.e., $z_1 \rightarrow y, z_2 \rightarrow \dot{y}, z_3 \rightarrow f$.

The controller u is designed as follows:

$$u = \frac{-z_3 + u_0}{b_0} \quad (52)$$

Bringing equation (52) into equation (45), we can obtain:

$$\ddot{y} = b_0 \cdot \left(\frac{-z_3 + u_0}{b_0} \right) + f(y, \omega, t) = (f - z_3) + u_0 \quad (53)$$

Neglecting the estimation error $f - z_3$ of z_3 , we can obtain that the system is reduced to a double integrator series type system, i.e.:

$$\ddot{y} = (f - z_3) + u_0 \approx u_0 \quad (54)$$

For a simple PD control of the integrator series type system, we obtain the linear state error feedback (LSEF) as:

$$LSEF : u_0 = k_p(r - z_1) - k_d z_2 \quad (55)$$

The parameters k_p, k_d in the control rate u_0 expression (46) can be determined by the stability of the closed-loop system [35]. The closed-loop transfer function of the system is obtained as:

$$G_{cl} = \frac{y(s)}{r(s)} = \frac{k_p}{s^2 + k_d s + k_p} \quad (56)$$

where the gain can be selected as:

$$k_d = 2\xi\omega_c \text{ and } k_p = \omega_c^2 \quad (57)$$

Where, ω_c denotes the desired closed loop natural frequency, while ξ represents

the damping ratio, which is used to prevent oscillations [35]. Typically, ξ is set to 1.

According to experience and general rules, there exists a relationship between the LESO bandwidth ω_0 and the controller bandwidth ω_c , given by $\omega_0 = (3 \sim 5)\omega_c$ [35]. It can be observed that the LADRC controller requires significantly fewer parameters to be tuned compared to the ADRC controller, simplifying the process of tuning the controller parameters and making it easier to understand and implement.

4.4 Cascade LADRC controller design

Like the cascade PID controller, the cascade LADRC controller designed in this section uses different control methods for different channels to balance control effectiveness and controller complexity. The yaw angle ψ and altitude z channels are controlled by a single LADRC controller, while the horizontal position ($x, y - \theta, \phi$) channel is controlled by an inner-outer dual-loop cascade LADRC controller. The complete structure of the cascade LADRC quadrotor flight control system is shown in Figure 21:

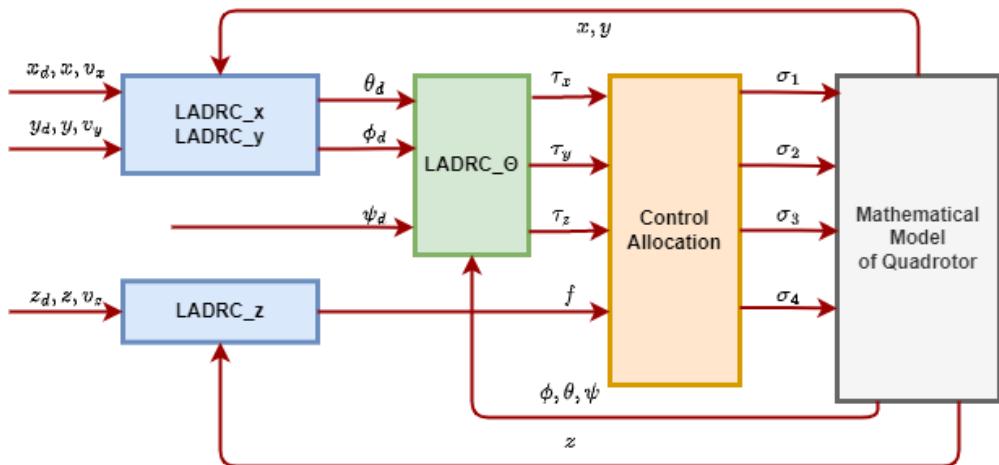


Figure 21 — Quadcopter cascade LADRC control

where the variables with subscript d (x_d, y_d, z_d , etc.) denote the target values of the variables.

The design thought process and control algorithms of the LADRC controller for each channel of the quadrotor are presented below separately:

4.4.1 Design of LADRC controller for height z channel

First, the controller for the altitude z channel is designed for the quadcopter, and the controller structure for the altitude z channel is obtained according to the structure of the LADRC controller figure 20 as shown in the following figure 22:

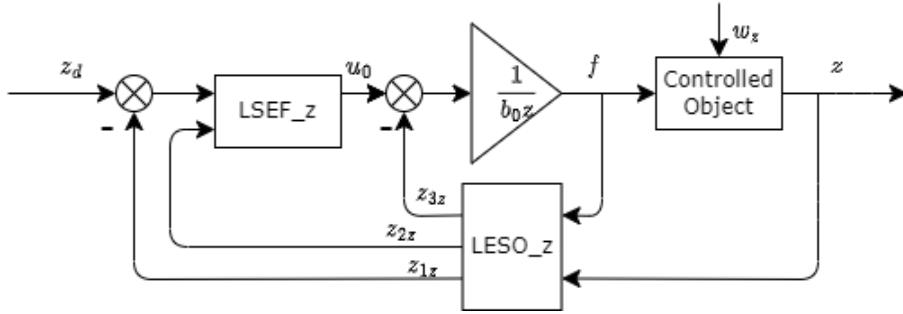


Figure 22 – z Channel LADRC

- According to the LADRC principle introduced in section (4.3), consider the external perturbation acting on the z channel of the quadrotor as w_z rewriting the linearized height z channel model (31) as

$$\ddot{z} = g - \frac{f}{m} + w_z \quad (58)$$

- From the 58 equation we know that this channel system is a second-order system, so the discrete LESO algorithm for the third-order height z channel designed according to the equation (44) to (48) is as follows:

$$\begin{cases} e_z = z_{1z}(k) - z(k), b_{0z} = -\frac{1}{m}, u_z(k) = f(k) \\ (z_{1z}(k+1) - z_{1z}(k))/h = z_{2z}(k) - \beta_{1z}e_z \\ (z_{2z}(k+1) - z_{2z}(k))/h = z_{3z}(k) - \beta_{2z}e_z + b_{0z}u_z(k) \\ (z_{3z}(k+1) - z_{3z}(k))/h = -\beta_{3z}e_z \end{cases} \quad (59)$$

where the output z_{1z} of LESO is an estimate of the actual output height z , z_{2z} is an estimate of the rate u_z at height z , and z_{3z} is an estimate of the expanded state variable $g + w_z$ (where w_z is the external perturbation acting on the quadrotor z channel); $\beta_{1z} = 3\omega_{0z}$, $\beta_{2z} = 3\omega_{0z}^2$, $\beta_{3z} = \omega_{0z}^3$; ω_{0z} is the bandwidth of the z -channel observer.

- Let the expected value of the high z channel be z_d , and the LSEF designed

according to the formula (49) to (55) is shown as follows:

$$\begin{cases} u_{0z}(k) = K_{pz} (z_d(k) - z_{1z}(k)) - K_{dz} z_{2z}(k) \\ u_z(k) = \frac{u_{0z}(k) - z_{3z}(k)}{b_{0z}} \end{cases} \quad (60)$$

Where $K_{dz} = 2\omega_{cz}$ and $K_{pz} = \omega_{cz}^2$, ω_{cz} is the bandwidth of the z channel controller.

4.4.2 Design of LADRC controller for yaw angle ψ channel

The LADRC controller for the yaw angle channel follows the same design principles and structure as the altitude channel controller mentioned earlier, and requires no further explanation.

- According to the LADRC principle introduced in section 4.3, consider the external perturbation acting on the yaw angle ψ channel of the quadrotor as w_ψ , then the linearized ψ channel model (32) can be rewritten as:

$$\ddot{\psi} = \frac{1}{J_z} \tau_z + w_\psi \quad (61)$$

- The algorithm for the design of the discrete LESO part of the yaw angle channel LADRC controller is:

$$\begin{cases} e_\psi = z_{1\psi}(k) - \psi(k), b_{0\psi} = \frac{1}{J_z}, u_\psi(k) = \tau_z(k) \\ (z_{1\psi}(k+1) - z_{1\psi}(k))/h = z_{2\psi}(k) - \beta_{1\psi} e_\psi \\ (z_{2\psi}(k+1) - z_{2\psi}(k))/h = z_{3\psi}(k) - \beta_{2\psi} e_\psi + b_{0\psi} u_\psi(k) \\ (z_{3\psi}(k+1) - z_{3\psi}(k))/h = -\beta_{3\psi} e_\psi \end{cases} \quad (62)$$

- The algorithm for the design of the discrete LSEF part of the LADRC controller for the yaw angle channel is:

$$\begin{cases} u_{0\psi}(k) = K_{\psi z} (\psi_d(k) - z_{1\psi}(k)) - K_{d\psi} z_{2\psi}(k) \\ u_\psi(k) = \frac{u_{0\psi}(k) - z_{3\psi}(k)}{b_{0\psi}} \end{cases} \quad (63)$$

4.4.3 Design of ADRC controller for horizontal position $x, y - \phi, \theta$ channel.

For the design of the $x, y - \phi, \theta$ channel controllers, an inner-outer loop double LADRC closed-loop control method is used. Based on the linearized model of the quadcopter in Section 2.3.4 and LADRC controller design principles, we obtain the structure diagram for the $x, y - \theta, \phi$ channel LADRC controller, shown in Figure 23:

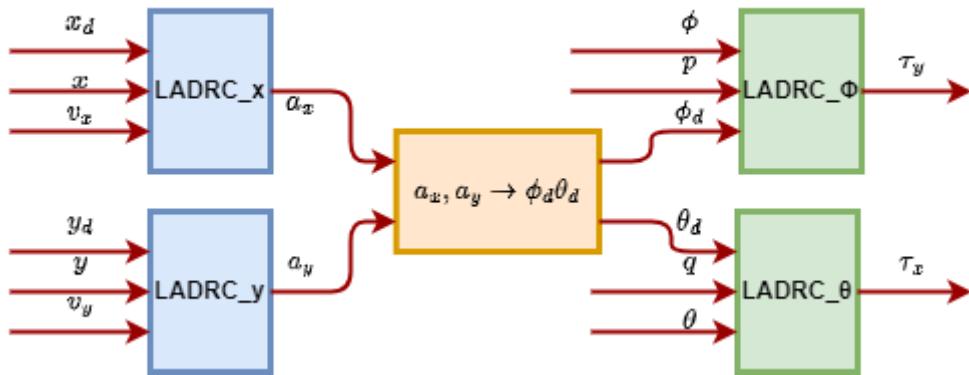


Figure 23 – Double closed loop LADRC structure diagram of $x, y - \phi, \theta$ channels

1. Based on the LADRC theory introduced in Section 4.3, taking into account the external disturbances w_x and w_y acting on the horizontal position channel, and w_ϕ and w_θ acting on the attitude channel, we can modify the linearized models of the horizontal position (29) , and attitude (32)channels to:

$$\begin{cases} \ddot{x} = a_x + w_x \\ \ddot{y} = a_y + w_y \end{cases}, \quad \begin{cases} \dot{\phi}_d = \frac{1}{g} (-\sin \psi a_x + \cos \psi a_y) \\ \dot{\theta}_d = \frac{1}{g} (-\cos \psi a_x - \sin \psi a_y) \end{cases} \quad (64)$$

$$\begin{cases} \ddot{\phi} = \frac{1}{J_x} \tau_x + w_\phi \\ \ddot{\theta} = \frac{1}{J_y} \tau_y + w_\theta \end{cases}$$

2. From the 64 equation, we know that the x, y, ϕ, θ channel system is a second-order system, so the discrete LESO algorithm for the third-order $x, y - \theta, \phi$ channels designed according to the equation (44) to (48) is as

follows:

$$x : \begin{cases} e_x = z_{1x}(k) - x(k), b_{0x} = 1, u_x(k) = a_x(k) \\ (z_{1x}(k+1) - z_{1x}(k))/h = z_{2x}(k) - \beta_{1x}e_x \\ (z_{2x}(k+1) - z_{2x}(k))/h = z_{3x}(k) - \beta_{2x}e_x + b_{0x}u_x(k) \\ (z_{3x}(k+1) - z_{3x}(k))/h = -\beta_{3x}e_x \end{cases} \quad (65)$$

$$y : \begin{cases} e_y = z_{1y}(k) - y(k), b_{0y} = 1, u_y(k) = a_y(k) \\ (z_{1y}(k+1) - z_{1y}(k))/h = z_{2y}(k) - \beta_{1y}e_y \\ (z_{2y}(k+1) - z_{2y}(k))/h = z_{3y}(k) - \beta_{2y}e_y + b_{0y}u_y(k) \\ (z_{3y}(k+1) - z_{3y}(k))/h = -\beta_{3y}e_y \end{cases} \quad (66)$$

$$\phi : \begin{cases} e_\phi = z_{1\phi}(k) - \phi(k), b_{0\phi} = \frac{1}{J_x}, u_\phi(k) = \tau_x(k) \\ (z_{1\phi}(k+1) - z_{1\phi}(k))/h = z_{2\phi}(k) - \beta_{1\phi}e_\phi \\ (z_{2\phi}(k+1) - z_{2\phi}(k))/h = z_{3\phi}(k) - \beta_{2\phi}e_\phi + b_{0\phi}u_\phi(k) \\ (z_{3\phi}(k+1) - z_{3\phi}(k))/h = -\beta_{3\phi}e_\phi \end{cases} \quad (67)$$

$$\theta : \begin{cases} e_\theta = z_{1\theta}(k) - \theta(k), b_{0\theta} = \frac{1}{J_y}, u_\theta(k) = \tau_y(k) \\ (z_{1\theta}(k+1) - z_{1\theta}(k))/h = z_{2\theta}(k) - \beta_{1\theta}e_\theta \\ (z_{2\theta}(k+1) - z_{2\theta}(k))/h = z_{3\theta}(k) - \beta_{2\theta}e_\theta + b_{0\theta}u_\theta(k) \\ (z_{3\theta}(k+1) - z_{3\theta}(k))/h = -\beta_{3\theta}e_\theta \end{cases} \quad (68)$$

3. Using the LADRC theory from Section 4.3 and equations (49) to (55), we have designed the LSEF for the $x, y - \theta, \phi$ channel as follows:

$$x : \begin{cases} u_{0x}(k) = K_{px}(x_d(k) - z_{1x}(k)) - K_{dx}z_{2x}(k) \\ u_x(k) = \frac{u_{0x}(k) - z_{3x}(k)}{b_{0x}} \end{cases} \quad (69)$$

$$y : \begin{cases} u_{0y}(k) = K_{py}(y_d(k) - z_{1y}(k)) - K_{dy}z_{2y}(k) \\ u_y(k) = \frac{u_{0y}(k) - z_{3y}(k)}{b_{0y}} \end{cases} \quad (70)$$

$$\phi : \begin{cases} u_{0\phi}(k) = K_{p\phi}(\phi_d(k) - z_{1\phi}(k)) - K_{d\phi}z_{2\phi}(k) \\ u_\phi(k) = \frac{u_{0\phi}(k) - z_{3\phi}(k)}{b_{0\phi}} \end{cases} \quad (71)$$

$$\theta : \begin{cases} u_{0\theta}(k) = K_{p\theta}(\theta_d(k) - z_{1\theta}(k)) - K_{d\theta}z_{2\theta}(k) \\ u_\theta(k) = \frac{u_{0\theta}(k) - z_{3\theta}(k)}{b_{0\theta}} \end{cases} \quad (72)$$

4.4.4 Analysis of control system stability

In this paper, we take the height z channel as an example to illustrate the analysis process of control system stability. From 4.4.1 section, we know that the generalized perturbation in the height z channel is $f_z = g + w_z$, which expands into a new state variable x_{3z} , and f_z is derivable, let $h_z = \dot{f}_z$, from 59 equation It is known that $b_{0z} = -\frac{1}{m}$, and the height z channel model established in section 2.3.4 is written in the form of the following equation of state:

$$\begin{cases} \dot{x}_{1z} = x_{2z} \\ \dot{x}_{2z} = x_{3z} + b_{0z}u_z \\ \dot{x}_{3z} = h_z \\ x_{1z} = z \end{cases} \quad (73)$$

By applying Laplace transformation to the continuous form of the observer equation (59) designed for the z channel in the LADRC controller from section 4.4.1, we can obtain the input-output transfer function form as follows:

$$\begin{aligned} Z_{1z}(s) &= \frac{3\omega_{0z}s^2 + 3\omega_{0z}^2s + \omega_{0z}^3}{(s + \omega_{0z})^3}Z(s) + \frac{s b_{0z}}{(s + \omega_{0z})^3}U_z(s) \\ Z_{2z}(s) &= \frac{3\omega_{0z}^2s^2 + \omega_{0z}^3s}{(s + \omega_{0z})^3}Z(s) + \frac{s(s + 3\omega_{0z})b_{0z}}{(s + \omega_{0z})^3}U_z(s) \\ Z_{3z}(s) &= \frac{\omega_{0z}^3s^2}{(s + \omega_{0z})^3}Z(s) - \frac{b_{0z}\omega_{0z}^3}{(s + \omega_{0z})^3}U_z(s) \end{aligned} \quad (74)$$

The linear state error feedback LSEF equation (60) in the LADRC controller designed for z channels in section 4.4.1 above is Laplace transformed into the form of a transfer function and simplified to give:

$$U_z(s) = \frac{1}{b_{0z}} [\omega_{cz}^2 (Z_d(s) - Z_{1z}(s)) - 2\omega_{cz}Z_{2z}(s) - Z_{3z}(s)] \quad (75)$$

Let the estimation error of the generalized disturbance term of the height z channel be:

$$f_z(s) - Z_3(s) = \tilde{Z}(s) \quad (76)$$

According to the above formula (73),(76) rewrite the height z as:

$$Z(s) = \frac{1}{s^2} [b_{0z}U_z(s) + Z_3(s) + \tilde{Z}(s)] \quad (77)$$

Based on equations (74), (75), and (77), we can derive the relationship between the height output $Z(s)$ and the expected value $Z_d(s)$ and $\tilde{Z}(s)$ as:

$$Z(s) = G(s)Z_d(s) + G'(s)\tilde{Z}(s) \quad (78)$$

where:

$$\begin{aligned} G(s) &= \frac{\omega_{cz}^2}{s^2 + 2\omega_{cz}s + \omega_{cz}^2} \\ G'(s) &= \frac{s^2 + (2\omega_{cz} + 3\omega_{0z})s + 6\omega_{cz}\omega_{0z} + 3\omega_{0z}^2 + \omega_{cz}^2}{s^4 + c_1s^3 + c_2s^2 + c_3s + c_4} \\ c_1 &= 3\omega_{0z} + 2\omega_{cz} \\ c_2 &= 6\omega_{cz}\omega_{0z} + 3\omega_{0z}^2 + \omega_{cz}^2 \\ c_3 &= 6\omega_{cz}\omega_{0z}^2 + 3\omega_{cz}^2\omega_{0z} \\ c_4 &= 3\omega_{cz}^2\omega_{0z}^2 \end{aligned} \quad (79)$$

Ignoring the estimation error $\tilde{Z}(s)$ of the total disturbance f_z , the closed-loop transfer function of the high z channel with the expected value z_d of z as the input and the high z as the output is as follows:

$$\Phi(s) = \frac{\omega_{cz}^2}{s^2 + 2\omega_{cz}s + \omega_{cz}^2} \quad (80)$$

Obviously, the above equation represents the transfer function of a standard second-order system. Since $\omega_{cz} > 0$, all the poles of the transfer function are located in the left half-plane. According to the Routh stability criterion [36], it can be concluded that the entire z channel system is stable.

Similarly, the stability of the closed-loop system in other channels can be proven, and thus the stability of the entire control system can be demonstrated.

4.5 Simulation experiment of cascade LADRC controller for quadcopter

4.5.1 Fixed-point hovering experiment

To test the hovering capability of a quadcopter equipped with a cascaded LADRC controller designed in Section 4.4 and compare it with the hovering capability of a cascaded PID controller presented in Section 3.3.1, this section takes the quadcopter's takeoff position $(x_0, y_0, z_0) = (0, 0, 0)$ as the coordinate origin and sets the target position of the quadcopter to $(x_d, y_d, z_d) = (1, 1, -1)$. Finally, under the action of the designed cascaded LADRC controller, the simulation results of the system are shown in Figures 24 to 25:

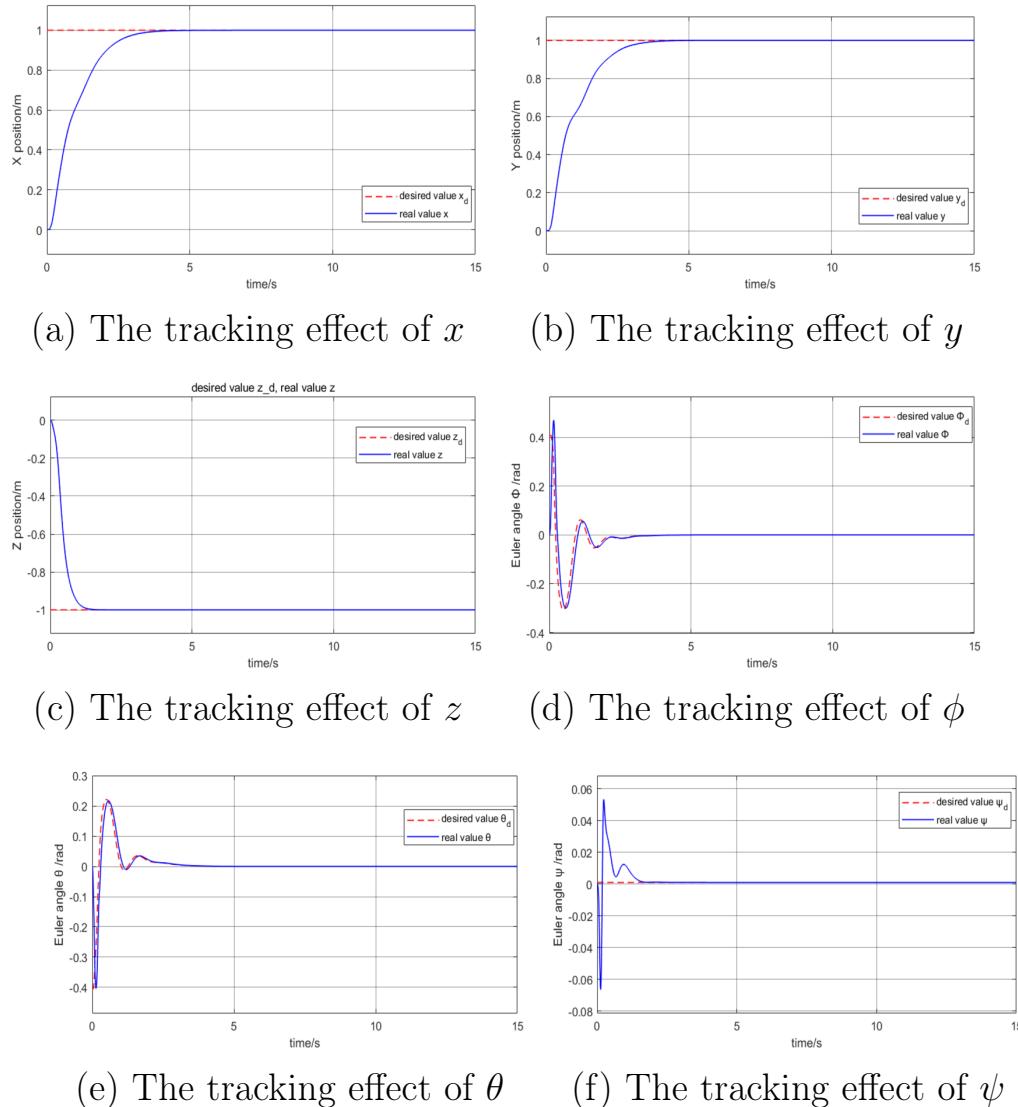


Figure 24 — Fixed-point hovering simulation experiment results graph

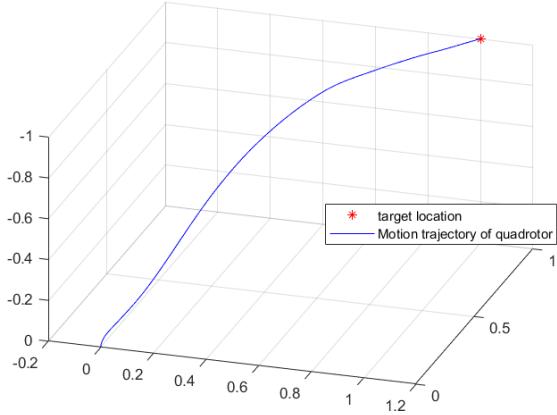


Figure 25 — Quadcopter 3D space motion trajectory

From the simulated tracking curves of the quadcopter's position and attitude shown in Figures 24 to 25, it can be observed that under the control of the cascaded LADRC controller, the positions x , y , and z reach their desired values in a short time, with an adjustment time of approximately $3.9s$ for x and y and about $1.5s$ for z , all with minimal overshoot. The tracking effect of pitch, roll, and yaw angles is also satisfactory, especially considering that the expected values of pitch and roll angles are not fixed values but instead change based on the expected acceleration, which makes them even more difficult to control. Nevertheless, the quadcopter can reach stability quickly, satisfying the requirements for stability of the aircraft. This demonstrates that the cascaded LADRC controller has a good hovering capability in the absence of disturbances.

4.5.2 Anti-interference ability and robustness experiments

To test the robustness of the quadcopter equipped with the cascaded LADRC controller designed in Section 4.4 and compare it with the robustness and disturbance rejection ability of the cascaded PID controller presented in Section 3.3.2, this section takes the quadcopter's takeoff position $(x_0, y_0, z_0) = (0, 0, 0)$ as the coordinate origin and sets the target position of the quadcopter to $(x_d, y_d, z_d) = (1, 1, -1)$. Additionally, Gaussian white noise with different average amplitudes μ was added to all six channels of the quadcopter's x , y , z , ϕ , θ , and ψ signals. Finally, under the action of the designed cascaded LADRC controller, the simulation results of the system are shown in Figures 26 to 32:

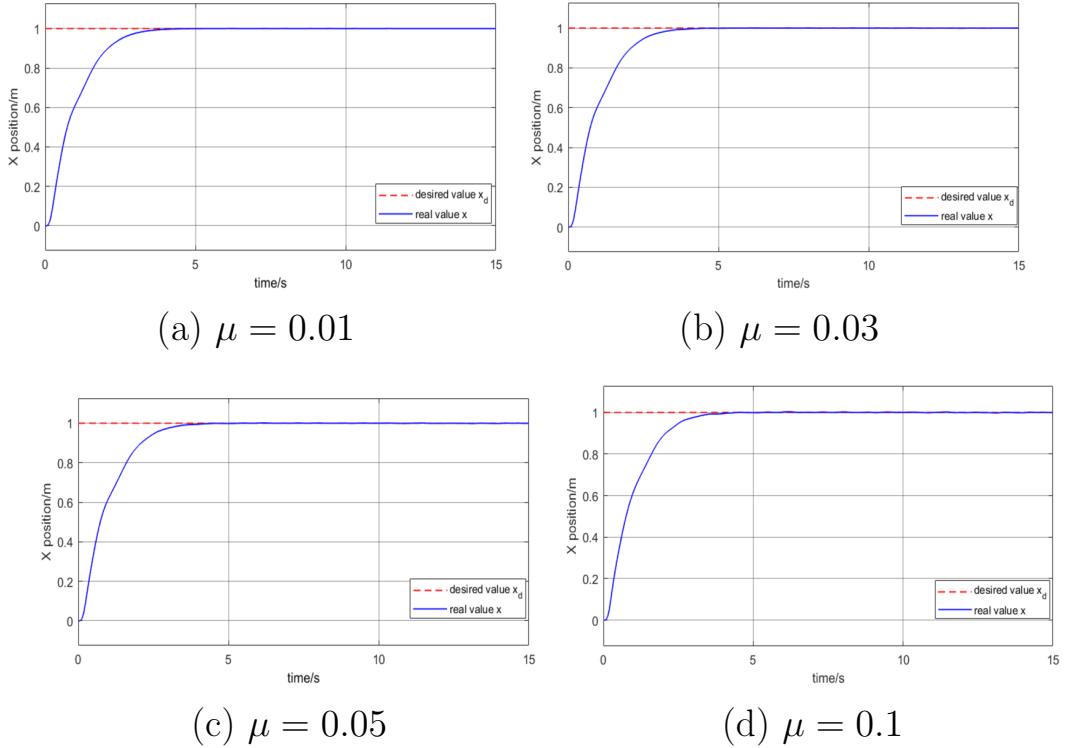


Figure 26 — The tracking effect of x under Gaussian white noise with different average amplitude μ

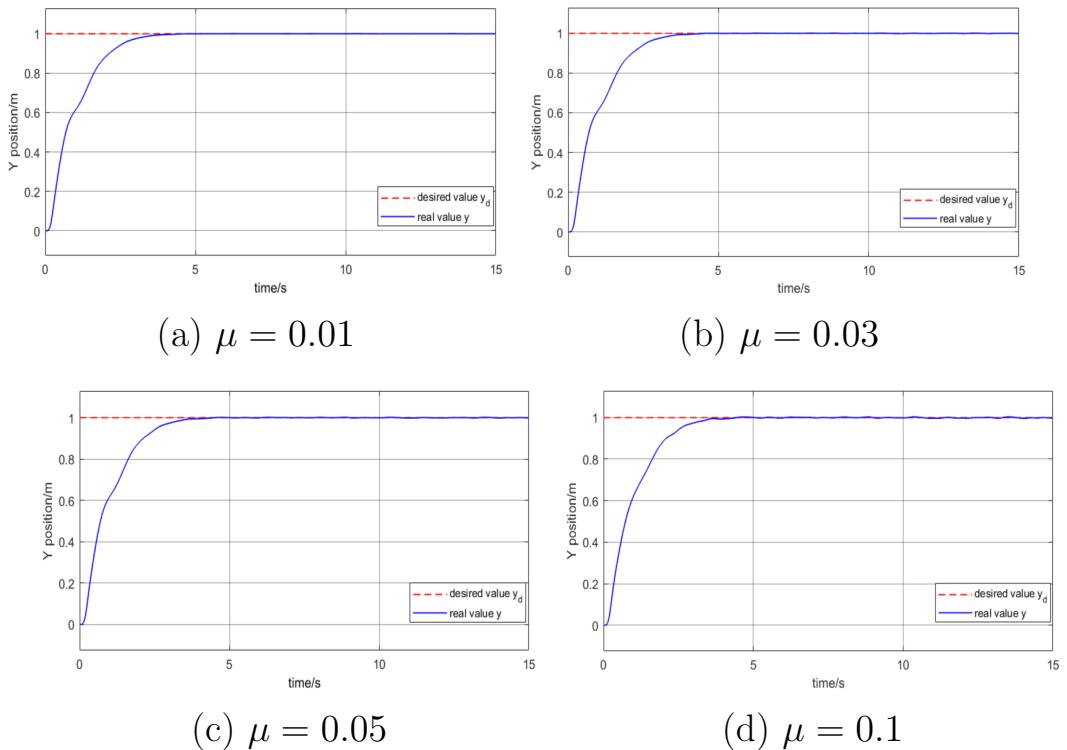


Figure 27 — The tracking effect of y under Gaussian white noise with different average amplitude μ

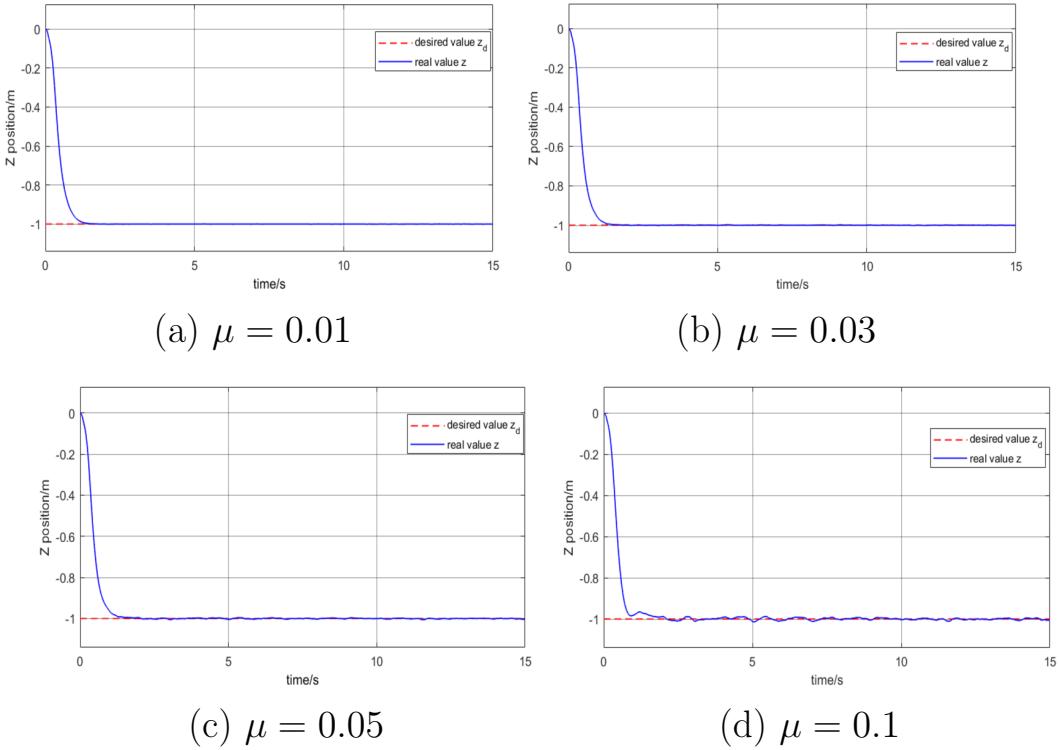


Figure 28 — The tracking effect of z under Gaussian white noise with different average amplitude μ

From the above simulation results of the disturbance rejection and robustness experiments shown in Figures 26 to 32, it can be observed that:

Compared with the fixed-point hovering experiment without considering disturbances, when the quadcopter is subjected to disturbances within a certain range, although the adjustment time and overshoot increase during the flight process, the output values can still fluctuate around the target values. At this time, the quadcopter's state can be controlled within an acceptable error range, demonstrating that the cascaded LADRC controller of the quadcopter has a certain disturbance rejection ability.

Compared with the simulation results of the cascaded PID controller in terms of disturbance rejection and robustness, under the condition of internal and external disturbances with uncertainties, the quadcopter controlled by the cascaded LADRC controller exhibits smaller oscillations, faster tracking speed, and stronger ability to suppress internal and external disturbances. This indicates that compared to the cascaded PID controller designed based on traditional PID theory, the cascaded LADRC controller designed based on LADRC

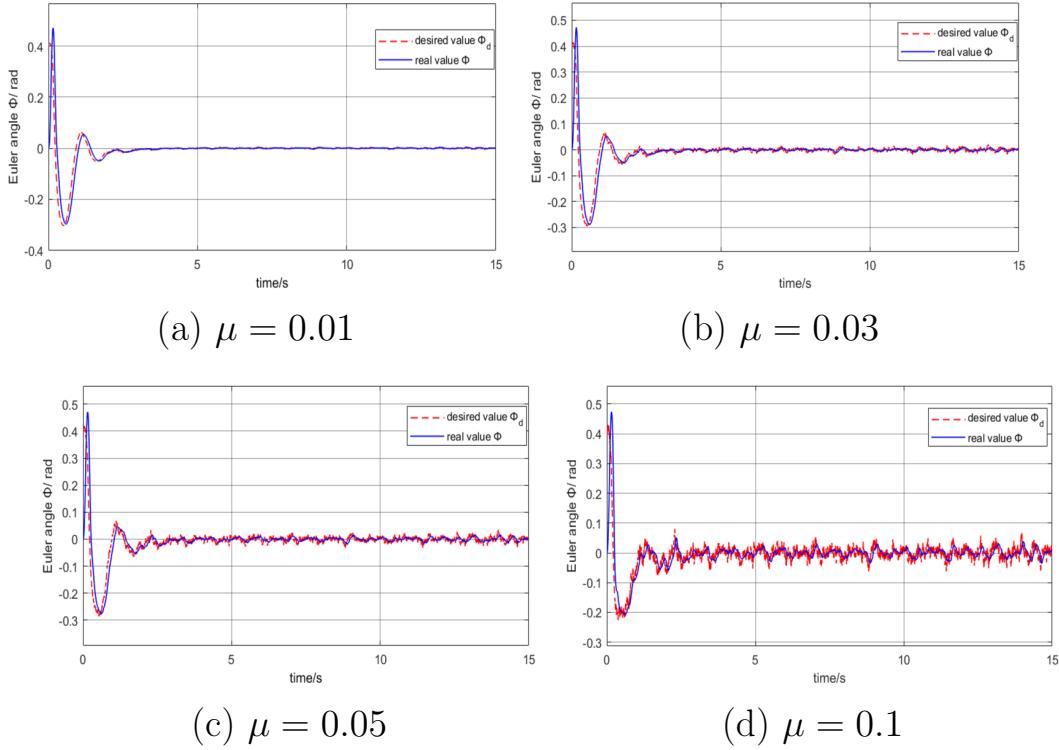


Figure 29 — The tracking effect of ϕ under Gaussian white noise with different average amplitude μ

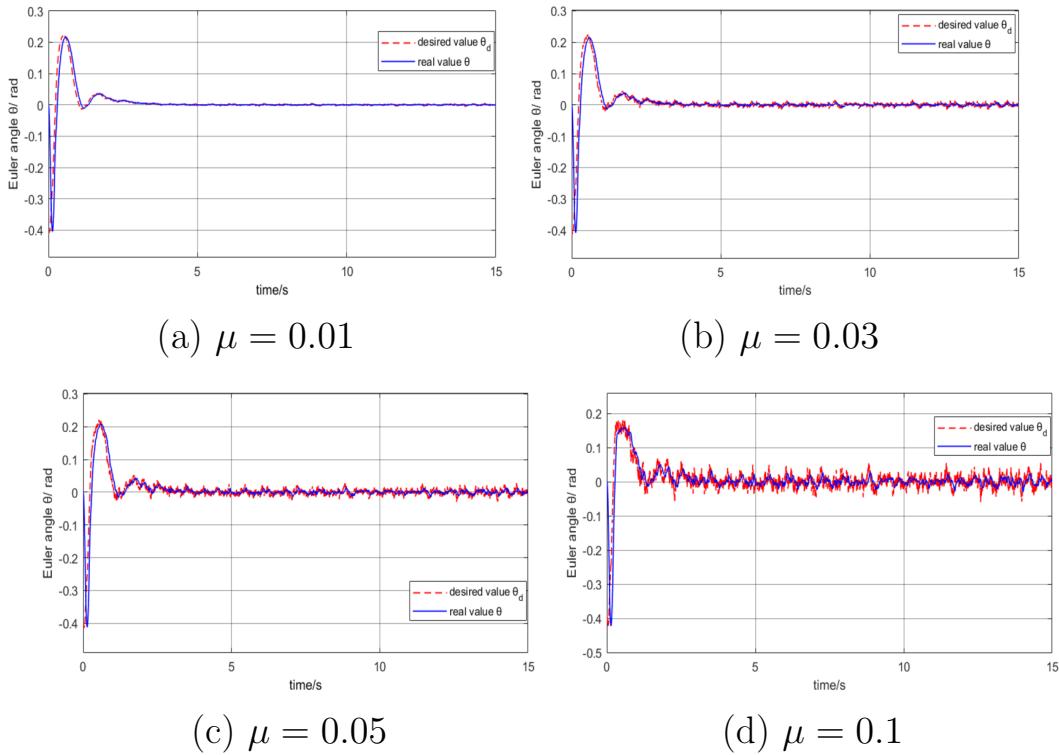


Figure 30 — The tracking effect of θ under Gaussian white noise with different average amplitude μ

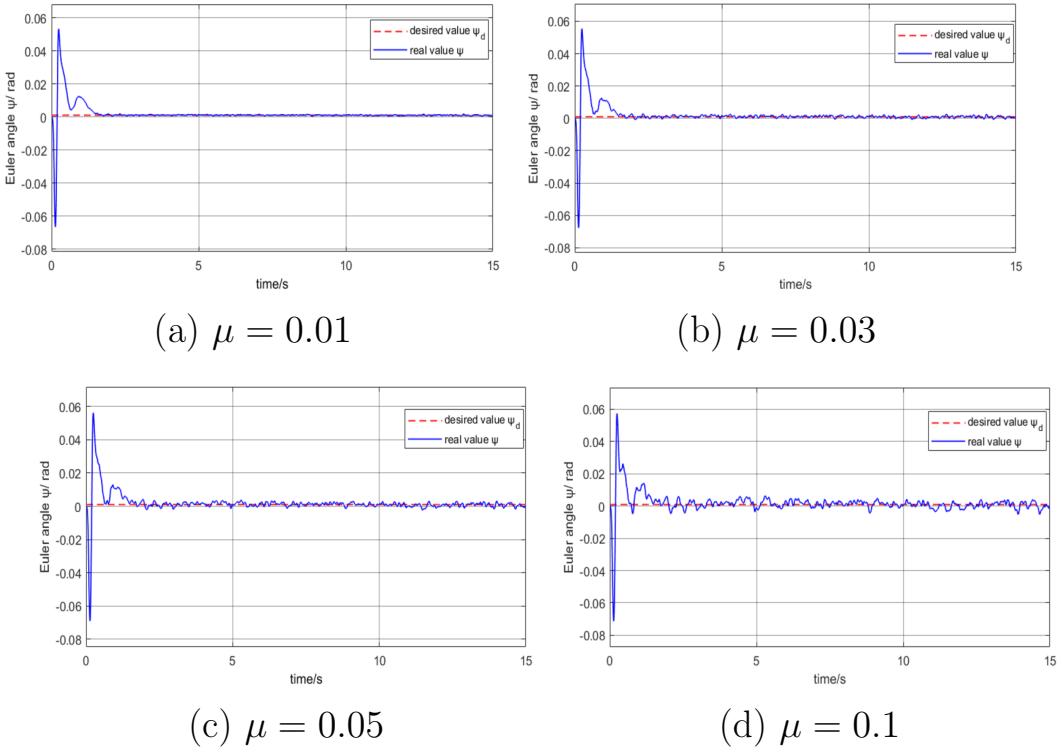


Figure 31 — The tracking effect of ψ under Gaussian white noise with different average amplitude μ

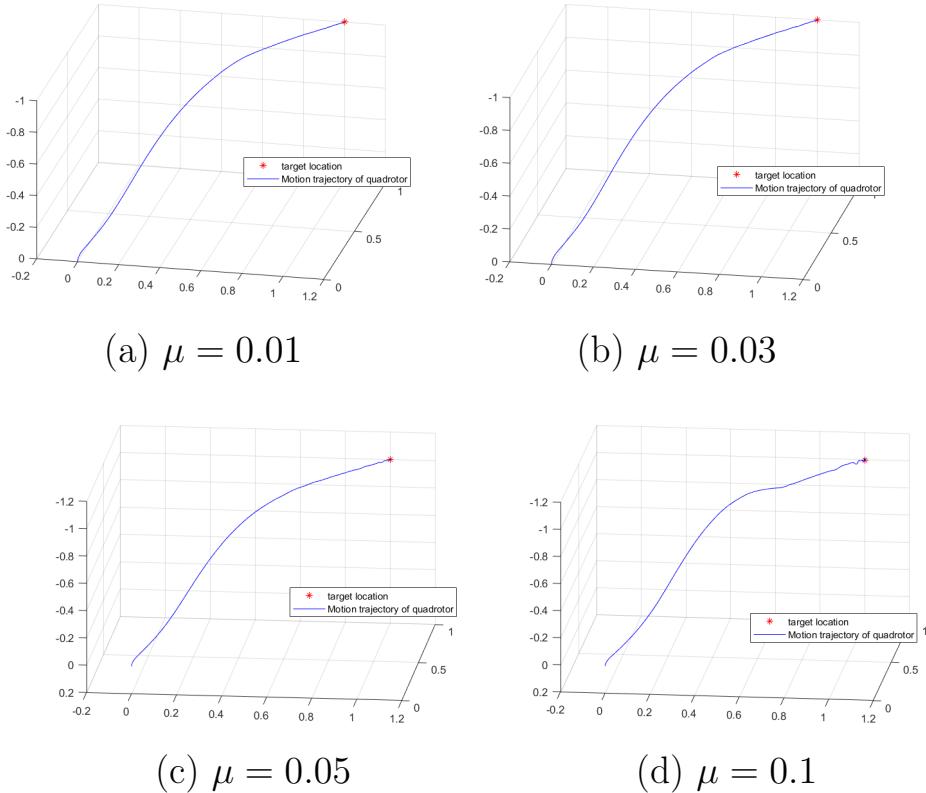


Figure 32 — 3D motion trajectory of quadcopter under Gaussian white noise with different average amplitude μ

theory has stronger robustness and disturbance rejection ability and can adapt to more complex flight environments.

4.6 Chapter Summary

This chapter firstly introduce the theoretical knowledge and composition structure of the traditional Active Disturbance Rejection Controller (ADRC), and then provide a detailed study of the improved control algorithm of ADRC - Linear Active Disturbance Rejection Control (LADRC) and its parameter tuning method, which is aimed at the complex parameter tuning of ADRC due to its many parameters.

Secondly, the coupling relationship between each channel of the linear model of the simplified quadcopter is analyzed. Based on the LADRC control theory, LADRC controllers are designed for each channel respectively and coupled into a complete cascaded LADRC controller.

Thirdly, a quadcopter simulation system based on the cascaded LADRC controller is built in MATLAB/Simulink. The effectiveness of the cascaded LADRC controller is demonstrated by controlling the quadcopter to perform a fixed-point hovering task.

Finally, considering that each channel of the quadcopter is affected by some unknown external and internal disturbances, and to compare with the anti-interference and robustness experiment in Chapter 3, Gaussian white noise with different average amplitudes μ is added to each of the six channels of the quadrotor (x , y , z , ϕ , θ , ψ) in MATLAB/Simulink. The experiment tests the robustness and anti-interference ability of the cascaded LADRC controller. Compared with the results of the anti-interference and robustness experiment in Chapter 3, the experiment clearly shows that the cascaded LADRC controller has better robustness and anti-interference ability, and can cope with more complex flight environments.

5 QUADCOPTER HARDWARE-IN-THE-LOOP(HIL) SIMULATION EXPERIMENTS

In actual production, for the model-based development mode of the controller, hardware-in-the-loop(HIL) simulation testing is a typical testing method that connects the real controller to a simulated controlled object in an efficient and cost-effective way to comprehensively test the controller. In the previous two chapters, the cascaded PID controller and cascaded LADRC controller for the quadrotor have been designed, and the effectiveness of the controllers has been preliminarily verified through simulation experiments using MATLAB/Simulink. In order to further verify the feasibility of the control algorithm, this chapter will conduct HIL simulation testing on the designed controllers.

Firstly, this chapter introduces the composition of the hardware-in-the-loop(HIL) simulation platform used in the experiment. Secondly, the controller algorithm code designed is downloaded to the Pixhawk system using MATLAB/Simulink software for cascaded PID controller and cascaded LADRC controller HIL simulation testing to verify the feasibility of the two control algorithms.

5.1 Introduction of hardware-in-the-loop simulation platform for quadcopters

For this experiment, we will use the open-source autopilot hardware Pixhawk in combination with the flight simulator software AirSim to perform hardware-in-the-loop simulation of the controller designed earlier. Below, we will introduce the hardware and software systems used in the experiment, as well as the experimental process:

5.1.1 Hardware system: Autopilot system (flight control system)

Pixhawk is an open-source flight controller hardware developed by 3DR, one of the most famous manufacturers in the world. It uses a main control based

on STM32F427 (180MHZ) and a STM32F100 co-processor, making it a powerful and reliable open-source hardware favored by many users. In quadcopter flight systems, Pixhawk autopilot system is mainly used to perceive the motion state of the multi-rotor aircraft and generate motor control signals. For this experiment, we have selected Pixhawk2.4.8 flight controller shown in Figure 33 for the control system design:



Figure 33 — Pixhawk Flight Control 2.4.8

5.1.2 Software system: Real-time motion simulation platform Airsim

AirSim is an open-source flight simulator software developed by Microsoft, based on the Unreal Engine game engine. It can be used for physical and visual simulation of robots such as drones and unmanned vehicles. The simulation interface of AirSim is shown in Figure 34:



Figure 34 — airsim simulation interface

5.1.3 Hardware-in-the-loop simulation experiment process

Chapters 3 and 4 have designed two cascaded controllers for quadcopter flight. This experiment will generate code for the controller algorithm in Simulink and download it to the Pixhawk system, which will then be connected to the computer via USB. The Airsimm software will run on the computer and send sensor data (such as accelerometer, barometer, magnetometer, etc.) to the Pixhawk system via the USB data cable. The PX4 autopilot software in the Pixhawk system will receive sensor data for filtering and state estimation, and then send the estimated state information to the controller via the internal uORB message bus. The controller will then send PWM control commands for each motor back to Airsim via the USB data cable, forming a hardware-in-the-loop(HIL) simulation closed loop.

5.2 Fixed-point hovering experiment

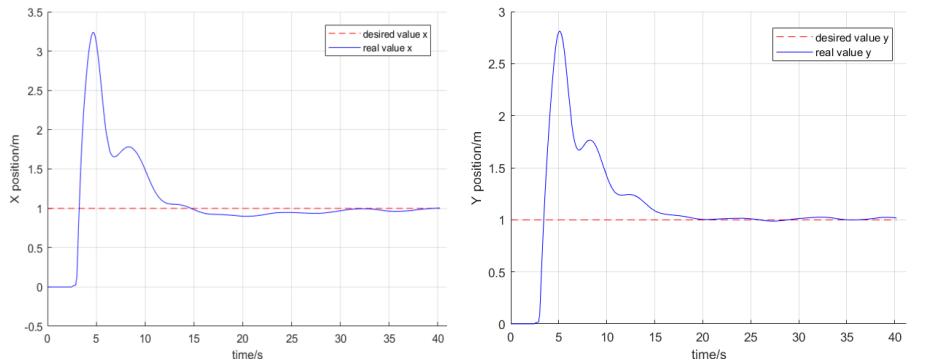
In the previous section, we introduced the hardware-in-the-loop simulation platform used in this paper: the Pixhawk 2.4.8 hardware combined with the Airsim flight simulation software. This section's experimental environment applies the controller designed in chapter 3 and 4 to an actual flight platform to achieve a fixed-point hovering experiment for a quadcopter. Taking the takeoff position of the quadcopter as the coordinate origin $(x_0, y_0, z_0) = (0, 0, 0)$, the target position of the quadcopter is set to $(x_d, y_d, z_d) = (1, 1, -1)$.

5.2.1 Experiment on HIL Simulation with Cascaded PID Controller

Under the action of cascade PID controller, the simulation results of the system are shown in the following figure 35 to 36 :

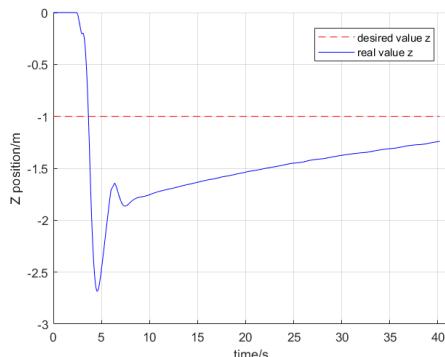


Figure 35 — 3D Motion Trajectory of quadrotor Aircraft (Airsim)

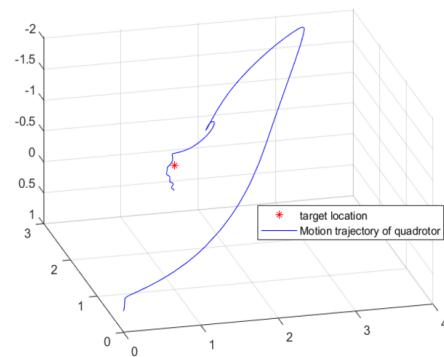


(a) HIL *x*tracking effect

(b) HIL *y*tracking effect



(c) HIL *z*tracking effect



(d) 3D space motion path (HIL)

Figure 36 — HIL tracking effect

5.2.2 Experiment on HIL Simulation with Cascaded LADRC Controller

Under the action of cascade LADRC controller, the simulation results of the system are shown in the following figure 37 to 38 :

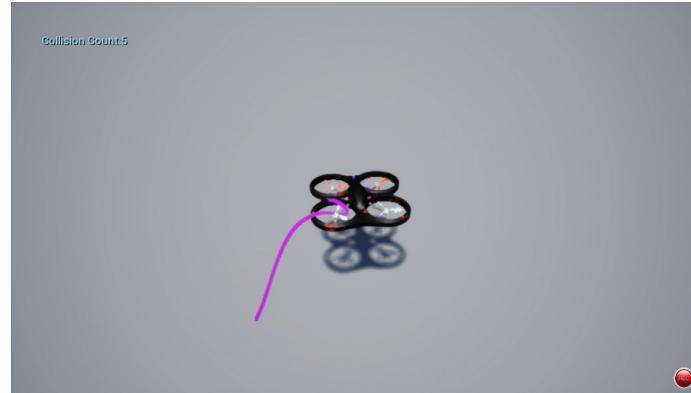


Figure 37 – 3D Motion Trajectory of quadrotor Aircraft (Airsim)

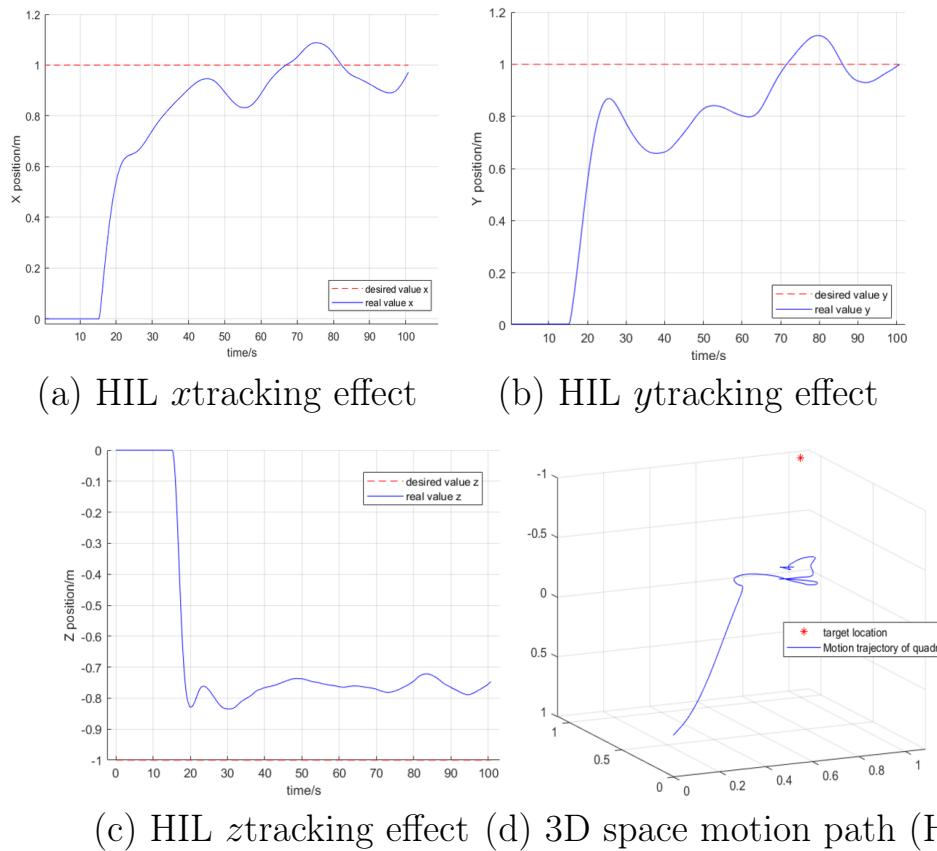


Figure 38 – HIL tracking effect

Based on the hardware-in-the-loop simulation results shown in Figures 35 to 38, it can be observed that:

Under the control of the cascade PID controller, the quadcopter's x and y channels can achieve steady-state tracking with small errors relatively quickly, but with large overshoots. The z channel takes a longer time to reach steady-state. Nonetheless, this controller can still preliminarily meet the control requirements.

Under the control of the cascade LADRC controller, the quadcopter's x and y channels exhibit large oscillations, and the z channel has a large steady-state error. However, this controller can still preliminarily meet the control requirements.

The underlying reason for these results is that when the control algorithm is applied to hardware, the computing power and data transmission speed of the hardware must be taken into account. In actual applications, further adjustments to the controller's parameters must be made to achieve better control performance.

5.3 Chapter Summary

In order to better verify the feasibility of the control scheme designed in this paper and to better integrate theory with practice, this chapter applied the cascade PID controller and cascade LADRC controller designed for quadcopters to the open-source flight control Pixhawk hardware and the Airsim flight simulator to conduct fixed-point hovering flight experiments. Based on the actual flight test, the control effect of the designed cascade PID controller and cascade LADRC controller on the quadcopter was analyzed. The experiment showed that the designed cascade PID controller and cascade LADRC controller can achieve preliminary fixed-point hovering effect, which verifies the rationality and effectiveness of the designed controller. However, the precision and stability of the control are not very high, indicating that the designed controller still needs further adjustment of parameters to achieve better control effect when applied to the hardware system.

CONCLUSION AND FUTURE WORK

5.4 Conclusion

In recent years, quadcopter have been widely used in various civilian and military fields due to their superior structure, good maneuverability, and low production costs. Because it involves multiple disciplines such as automatic control, advanced sensor technology, electronic engineering, three-dimensional path planning, and multi-aircraft collaboration, quadcopter also provide a test platform for the integration research of researchers at home and abroad, with extremely high application and research value. Autonomous flight control algorithm is an essential part of the stability of quadcopter. This paper focuses on the research of the control algorithm of quadcopter, and the main work is as follows:

- This paragraph introduces the development process of quadcopter in recent years, analyzes the current methods for establishing mathematical models for quadcopter and the research status of the main control methods for quadcopter control systems. The structure and working principle of quadcopter are analyzed, and based on this, a nonlinear mathematical model for simulating quadcopter is established using mechanics theory. Then, to facilitate the design of the controller, reasonable assumptions are made based on the flight characteristics of quadcopter, and the non-linear model is simplified into a linear model for the controller design in the following section.
- Furthermore, based on the principles of two control methods, PID and Linear Active Disturbance Rejection Control (LADRC), corresponding cascade PID and cascade LADRC controllers are designed according to the coupling relationship and characteristics of the mathematical model for quadcopter. Through MATLAB/Simulink simulation, the effectiveness of the two controllers in point hovering, anti-interference ability, and robustness is evaluated and compared, demonstrating that both controllers are effective under no disturbance, and the cascade LADRC con-

troller has better robustness and anti-interference ability than the cascade PID controller.

- Finally, by combining the open-source Pixhawk flight controller with the Airsim real-time motion simulation platform, a quadcopter experimental platform is built. By combining hardware and software, the effectiveness of the designed cascaded PID and cascaded LADRC controllers has been validated through hardware-in-the-loop simulation experiments.

5.5 Future Work

This paper is a study on the design of controllers for quadcopters. It mainly applies PID theory and LADRC to the experimental platform of quadcopters, analyzes and compares the performance of two designed controllers, and finally conducts hardware-in-the-loop(HIL) simulation experiments, which has certain practical significance. However, there are still many shortcomings in this paper, and there are many issues worth further thinking and research, mainly including the following aspects:

1. This article did not consider safety-related situations such as communication failure, sensor failure, and power system abnormality in quadcopter flight. Subsequent research can try to combine fault diagnosis fault-tolerant control and linear active disturbance rejection control(LADRC) to ensure flight safety to a greater extent.
2. In addition to the control methods studied in this paper, there are many other linear/nonlinear control methods with superior control performance that need to be applied to the quadcopter control system. Moreover, it is possible to combine the linear active disturbance rejection control(LADRC) algorithm with other algorithms to achieve more complex tasks and optimize various dynamic performance of the quadcopter.
3. The parameters of the series PID controller and series LADRC controller designed in this paper are manually adjusted and not the optimal parameters. Subsequent research can try to combine intelligent algorithms such as particle swarm optimization and genetic algorithm to optimize the control parameters.

4. The control parameters were further adjusted when the control algorithm was applied to hardware, but its performance was not very good. In actual applications, the parameters should be further adjusted to adapt to the control rate of the embedded discrete system to improve control performance.

ACKNOWLEDGEMENTS

At the end of this paper, I would like to express my heartfelt gratitude to my supervisor, Ms. Yin Ke. Ms. Yin is witty, humorous, and has unique insights and deep research in the field of robotics. After several academic exchanges with her, I gained a deeper understanding and recognition of the field of robotics, which not only broadened my research thinking but also greatly inspired my enthusiasm for in-depth research in this field. She patiently listened and carefully guided me through academic and experimental problems, which provided me with great help. When I was confused about my future academic path, she gave me great spiritual encouragement. She strictly required and frequently supervised every detail and aspect of my graduation design, which infected me with her serious attitude towards work. Ms. Yin created a good scientific research environment and a relaxed and pleasant atmosphere, which made me realize that doing experiments is actually a very interesting thing, and failed experiments and experiences are not worthless but valuable accumulation. I am very lucky to have met Ms. Yin in college, and it is also a pity that we only had a few months of time together. She is not only my supervisor but also a friend with common topics and interests. The time we spent together in the laboratory is also a precious memory of my future life.

Secondly, I would like to express my gratitude to all the teammates and friends I met in the university. There were laughter and tears, and the time we spent together is a precious memory of my college life. I also want to thank all the teachers who taught me during my college years, especially the competition teachers. It is your careful guidance that gave me a good foundation of professional knowledge, which is the basis for the completion of this paper.

Finally, I want to express my deep gratitude to my family who has always been by my side, giving me support and encouragement. They are my forever solid backing, giving me the courage to face the difficulties in life and studies, and the confidence to make my own choices and bravely face the unknown and challenging future.

Lastly, I would like to express my deep gratitude to the experts and teachers who reviewed this paper.

REFERENCES

1. Kimon P Valavanis. Advances in unmanned aerial vehicles: state of the art and the road to autonomy. 2008.
2. Farid Kendoul. Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems. *Journal of Field Robotics*, 29(2):315–378, 2012.
3. Jun Dai, Yufeng Guan, and Shuhong Ren. Research status and development trend of multi-rotor unmanned aerial vehicle (in chinese). *Journal of Chifeng University: Natural Science Edition*, 32(16):22–24, 2016.
4. Ahmed F El-Sayed. *Fundamentals of aircraft and rocket propulsion*. Springer, 2016.
5. B Yao, P Lu, S Yang, and J Ji. An overview of flight control technology for quadrotor aircrafts (in chinese). *Aero Weaponry*, 27(1):9–16, 2020.
6. Yibo Li and Shuxi Song. A survey of control algorithms for quadrotor unmanned helicopter. In *2012 IEEE fifth international conference on advanced computational intelligence (ICACI)*, pages 365–369. IEEE, 2012.
7. Bernard Mettler. *Identification modeling and characteristics of miniature rotorcraft*. Springer Science & Business Media, 2013.
8. Wenkai Liang, Xiaolong Zhang, Xiaoquan Xie, Peng Gao, and Leiqing Yang. Four-rotor control simulation and anti-interference verification based on matlab(in chinese). *Computer Engineering & Software*, 41(01):143–147, 2020.
9. Laifu Shi, Jianxin Shen, Qisheng Wang, and Junsheng Jiang. Fuzzy active disturbance rejection attitude control of quadrotor aircraft (in chinese). *Machine Building & Automation*, 50(03):157–162, 2021.
10. Cheng Zhang, Yuying Guo, and Zhengwei Zhu. Trajectory tracking control for quad-rotor uav based on linear active disturbance rejection (in chinese). *Flight Dynamics*, 39(03):75–81, 2021.

11. N Koksal, H An, and B Fidan. Backstepping-based adaptive control of a quadrotor uav with guaranteed tracking performance. *ISA transactions*, 105:98–110, 2020.
12. Wenjing Cai, Jinhua She, Min Wu, and Yasuhiro Ohyama. Disturbance suppression for quadrotor uav using sliding-mode-observer-based equivalent-input-disturbance approach. *ISA transactions*, 92:286–297, 2019.
13. K McCaney. Tiny drone with brain-like chip learns on the fly. *Defense Systems*, 2016.
14. Dario Floreano and Robert J Wood. Science, technology and the future of small autonomous drones. *nature*, 521(7553):460–466, 2015.
15. Yoshito Okada, Takuma Ishii, Kazunori Ohno, and Satoshi Tadokoro. Real-time restoration of aerial inspection images by recognizing and removing passive rotating shell of a uav. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5006–5012. IEEE, 2016.
16. Shaojie Shen, Yash Mulgaonkar, Nathan Michael, and Vijay Kumar. Multi-sensor fusion for robust autonomous flight in indoor and outdoor environments with a rotorcraft mav. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4974–4981. IEEE, 2014.
17. Stella Latscha, Michael Kofron, Anthony Stroffolino, Lauren Davis, Gabrielle Merritt, Matthew Piccoli, and Mark Yim. Design of a hybrid exploration robot for air and land deployment (herald) for urban search and rescue applications. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1868–1873. IEEE, 2014.
18. Georgios Darivianakis, Kostas Alexis, Michael Burri, and Roland Siegwart. Hybrid predictive control for aerial robotic physical interaction towards inspection operations. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 53–58. IEEE, 2014.
19. Mingxia Bao, Wuxi Shi, and Jianchuan Guo. Attitude control of four-rotor aircraft based on adrc (in chinese). *Automation & Instrumentation*, 32(2):10–13, 2017.

20. Renying Cao. Design and research of control system for quadrotor aircraft (in chinese). Master's thesis, East China Jiaotong University, 2020.
21. Jingqing Han. *Self-Disturbance Attenuation Control Technology (in Chinese)*. National Defense Industry Press, Beijing, 2013.
22. Xiaolu Ye, Li Yu, Wen'an Zhang, and Dan Zhang. Cascade adrc-based hover control for quadrotor air vehicles (in chinese). *Journal of Central South University: Science and Technology*, (8):2079–2087, 2017.
23. Mohammad Reza Mehranpour, Alireza Mohammad Shahri, Omid Emamgholi, and Mohammad Farrokhi. A new fuzzy adaptive control for a quadrotor flying robot. In *2013 13th Iranian Conference on Fuzzy Systems (IFSC)*, pages 1–5. IEEE, 2013.
24. Eric W Weisstein. Euler angles. <https://mathworld.wolfram.com/>, 2009.
25. Quan Quan. *Introduction to multicopter design and control*. Springer, 2017.
26. Liuping Wang. *PID control system design and automatic tuning using MATLAB/Simulink (in Chinese)*. John Wiley & Sons, 2020.
27. Tong Wu, Xin-Jun Sheng, Wei Dong, and Han Ding. Hardware-in-the-loop simulation design and experiment of a quadrotor (in chinese). *Mechatronics*, pages 205–205, 2014.
28. Han Jingqing. Auto-disturbances-rejection controller and it's applications (in chinese). *Control and decision*, 13(1):19–23, 1998.
29. JQ Han. From pid technique to active disturbances rejection control technique (in chinese). *Control engineering of China*, 9(3):13–18, 2002.
30. Jiahao Li. Improvement of self-disturbance attenuation control and its application in quadrotor uav (in chinese). Master's thesis, Xiamen University, 2018.
31. Jing Qing Han. Nonlinear state error feedback control law-nlsef (in chinese). In *Chinese Univ Pr*, page 211, 1995.

32. Zhiqiang Gao and Panos J Antsaklis. New methods for control system design using matrix interpolation. In *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, volume 3, pages 2506–2511. IEEE, 1994.
33. C Wu, WH Wang, Y Zhang, J Tan, and X Ni. Ladrc-based trajectory tracking for unmanned helicopter (in chinese). *Acta Aeronautica et Astronautica Sinica*, 36(2):473–483, 2015.
34. Zhiqiang Gao, Yi Huang, and Jingqing Han. An alternative paradigm for control system design. In *Proceedings of the 40th IEEE conference on decision and control (Cat. No. 01CH37228)*, volume 5, pages 4578–4585. IEEE, 2001.
35. Zhiqiang Gao et al. Scaling and bandwidth-parameterization based controller tuning. In *ACC*, pages 4989–4996, 2003.
36. Edward John Routh. *A treatise on the stability of a given state of motion, particularly steady motion: being the essay to which the Adams prize was adjudged in 1877, in the University of Cambridge*. Macmillan and Company, 1877.