Trajectory Tracking and Payload Dropping of an Unmanned Quadrotor Helicopter Based on GS-PID and Backstepping Control

Jing Qiao

A Thesis

in

The Department

of

Mechanical, Industrial and Aerospace Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of Master of Applied Science at

Concordia University

Montreal, Quebec, Canada

June 2018

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: Jing Qiao

Entitled: Trajectory Tracking and Payload Dropping of an Unmanned Quadrotor Helicopter Based on GS-PID and Backstepping Control

and submitted in partial fulfilment of the requirements for the degree of

Master of Applied Sciences (Mechanical Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. Javad Dargahi

_____ Examiner
Dr. Wen-Fang Xie

_____ External Examiner
Dr. Wei-Ping Zhu
Electrical and Computer Engineering

_____ Thesis Supervisor
Dr. Youmin Zhang

Approved by _____

Dr. Mamoun Medraj, MASc Program Director
Department of Mechanical, Industrial and Aerospace Engineering

_____ 2018

June 19th, *Dr. Amir Asif, Dean*
Faculty of Engineering and Computer Science

ABSTRACT

Two useful control techniques, the Gain-Scheduled Proportional-Integral-Derivative (GS-PID) control and backstepping control, have been applied by using quadrotor Unmanned Aerial Vehicle (UAV) in the applications of trajectory tracking and payload dropping operations in this thesis. These control algorithms are analyzed and verified through software simulations and experimental tests.

The dynamic model of the quadrotor UAV is firstly established using Newton-Euler laws. The quadrotor comes with a symmetric, nonlinear and multiple-input-multiple output (MIMO) dynamic model.

The GS-PID control algorithm is implemented firstly in take-off, trajectory tracking, payload dropping, and landing periods of flight in trajectory tracking and payload dropping scenarios.

Unlike other control algorithms that tend to linearize nonlinear systems, backstepping works without cancelling the nonlinearities in the system. This leads to more flexible designs of the control model. The backstepping control is implemented in this thesis for better performance of the quadrotor UAV for the two scenarios as well. Both control algorithms are implemented on the parameters of an unmanned quadrotor helicopter platform known as Qball-X4 available at the Networked Autonomous Vehicles Lab (NAVL) of Concordia University.

Using MATLAB/Simulink to build the simulation control model, the flight simulation of the Qball-X4 is carried out for the trajectory tracking and the payload dropping. In order to further investigate these two control approaches, the Qball-X4 is used for experimental verification on payload dropping performance. The results indicate that both algorithms can obtain acceptable performance, but the backstepping controller proves to be a better performed one.

Key words: Quadrotor UAV; Trajectory tracking; Payload dropping; Gain-Scheduled (GS) PID; Backstepping control

PAPERS AND PUBLICATIONS

(1) J. Qiao, Z. X. Liu, and Y. M. Zhang, Gain scheduling PID control of a quadrotor, 2017 IEEE International Conference on Unmanned System (IEEE ICUS 2017), October 27-29, 2017, Beijing, China

(2) J. Qiao, Z. X. Liu, and Y. M. Zhang, Modeling and GS-PID control of a quadrotor UAV, 10th International Conference on Computer and Automation Engineering (ICCAE 2018), February 24-26, 2018, Brisbane, Australia

ACKNOWLEDGMENTS

I would like to take this moment to express my deep gratitude to those who have made it possible for me to achieve this accomplishment.

Firstly, I would like to thank my supervisors Professor Youmin Zhang at the Department of Mechanical, Industrial and Aerospace Engineering of Concordia University for his continued guidance and assistance. His excellent supervision and valuable experience and support in both academic and personal matters were keys to making the accomplishments of this thesis possible.

I would also like to express my appreciation to Dr. Zhixiang Liu for providing support with experiment, computation, and computer software. Special thanks to Mr. Ban Wang, my colleague in the Diagnosis, Flight Control and Simulation Lab (DFCSL) and Networked Autonomous Vehicles Lab (NAVL) of Concordia University, for their help when I worked in the Lab. Thanks also to all the members of the DFCSL and NAVL of Concordia University for providing the pleasant atmosphere.

Finally, I wish to express my gratitude and love for my family. Each and every one of them has been a tremendous support to me through my entire life. From my parents, I have received the most valuable gifts of their strength, their courage and their love.

CONTENTS

NOMENCLATURE ix

LIST OF FIGURES xii

LIST OF TABLES xv

1 Introduction 1

1.1 Background 1

1.2 Review of Researches on Quadrotor Helicopter 2

1.2.1 Some Typical Quadrotor Helicopters 3

1.2.2 The UAVs Developed in Some Universities 5

1.3 The Control Principle and Key Technologies of Quadrotor UAV 10

1.3.1 Quadrotor UAV Structure and Control Principle 10

1.3.2 Key Technologies of Quadrotor UAV Control 11

1.4 A Review of Control Methods for Quadrotors 13

1.4.1 Survey of Control Methods 13

1.4.2 GS-PID Control and Backstepping Control for Quadrotor UAV 17

1.5 Motivation of This Study 19

1.5.1 GS-PID Controller 19

1.5.2 Backstepping Controller 20

1.6 Thesis Structure 20

2 Flight Principle and System Modeling of Quadrotor Helicopter 23

2.1 Flight Principle of Quadrotor Helicopter 23

2.2 Coordinate System and Coordinate Transformation Matrix 26

2.2.1 Coordinate System 26

2.2.2 Attitude Description Method (Coordinate Transformation Matrix) 27

2.3 Dynamic Modeling of a Quadrotor Helicopter 29

2.3.1 Linear Motion Model 29

2.3.2 Angular Motion Model 31

2.4 Summary..... 33
3 Qball-X4 UAV System and MATLAB/Simulink Control Simulation Platform..... 34
3.1 Introduction to Quanser Unmanned Experimental System..... 34
3.1.1 Qball-X4 System Configuration..... 34
3.1.2 Qball-X4 Quadrotor..... 36
3.1.3 OptiTrack Motion Tracking System for Localization..... 37
3.1.4 The Payload Releasing Mechanism..... 38
3.2 Qball-X4 DC Motors and Geometry..... 39
3.2.1 Electronic Speed Controller, DC Motor, and Propeller of Qball-X4..... 39
3.2.2 Geometry..... 40
3.3 MATLAB/Simulink Flight Control Simulation Platform..... 41
3.3.1 Introduction of Simulation Software..... 41
3.3.2 Simulation Platform Building for Quadrotor Helicopter..... 43
3.4 Summary..... 44
4 Gain Scheduling Based PID Controller of the Quadrotor Helicopter..... 45
4.1 Introduction..... 45
4.2 The PID Controller for Quadrotor UAV..... 46
4.2.1 The PID Control Approach..... 46
4.2.2 Altitude Control of Quadrotor UAV..... 51
4.2.3 Attitude Control of Quadrotor UAV..... 51
4.2.4 Position Controller of Quadrotor UAV..... 52
4.3 Gain Scheduling Controller for Quadrotor UAV..... 53
4.3.1 Introduction to GS-PID Control..... 53
4.3.2 Gain Scheduling Control Approach..... 54
4.3.3 Gain Scheduling PID Controller Design..... 55
4.4 Summary..... 57
5 Trajectory Tracking and Payload Dropping of Qball-X4 with GS-PID Controller..... 58
5.1 Introduction..... 58
5.2 Step Response Analysis..... 60
5.3 Fixed-Point Control..... 62

5.4 Trajectory Tracking Control with GS-PID Controller..... 68
5.5 Payload Dropping Control with PID and GS-PID Controller.....75
5.5.1 Simulation Experimental Results Analysis..... 75
5.5.2 Flight Test Experimental Results Analysis.....81
5.6 Summary..... 83
6 Backstepping Controller of the Quadrotor Helicopter.....84
6.1 Introduction..... 84
6.2 Backstepping Control Principle..... 85
6.2.1 Lyapunov Stability Theory..... 86
6.2.2 Backstepping Approach.....87
6.3 Backstepping Controller Design Approach.....89
6.4 Backstepping Controller Design for Quadrotor UAV.....90
6.4.1 Attitude Loop (Inner Loop) Controller Design..... 91
6.4.2 Position Loop (Outer Loop) Controller Design.....93
6.4.3 Nonlinear Constraint Conditions.....94
6.5 Summary..... 96
7 Trajectory Tracking and Payload Dropping of Qball-X4 with Backstepping Controller.....97
7.1 Introduction..... 97
7.2 Trajectory Tracking with Backstepping Controller.....99
7.3 Payload Dropping with Backstepping Controller..... 108
7.3.1 Simulation Experimental Results Analysis..... 108
7.3.2 Flight Test Experimental Results Analysis.....111
7.4 Summary..... 112
8 Conclusions and Future Works..... 114
8.1 Conclusions..... 114
8.2 Future Works..... 115
Reference..... 116

NOMENCLATURE

A:	Continuous time state matrix
A_d:	Discrete time state matrix
B:	Continuous time input matrix
B_d:	Discrete time input matrix
C:	Continuous time output matrix
C_d:	Discrete time output matrix
d:	Level arm
d_x:	Unit disturbance force added to the system in x direction.
d_y:	Unit disturbance force added to the system in y direction.
d_z:	Unit disturbance force added to the system in z direction.
F_B:	Body fixed reference frame
F_E:	Earth fixed reference frame
F_i:	Force generated by *i*th rotor
T:	Thrust, N
u_i:	Pulse Width Modulation (PWM) Signal
K:	DC Motor's Gain
ω:	DC Motor's Bandwidth, N
ϕ:	Roll Angle, rad
ϕ:	Pitch Angle, rad
ψ:	Yaw Angle, rad
m:	Mass, kg
g:	Gravity, m/s^2
L:	Quadrotor's Arm Length, m
τ:	Torque, N _m^2
Kp:	Proportional Gain of PID Controller
K_i:	Integral Gain of PID Controller

K_d: Derivative Gain of PID Controller

A_m: Augmented State Matrix

B_m: Augmented Input Matrix

C_m: Augmented Output Matrix

D_m: Augmented Feedforward Matrix

N_p: Prediction Horizon

N_c: Control Horizon

R_s: Setpoint Signal

R: Weighting Matrix

M_p: Maximum Overshoot

t_d: Delay Time

t_r: Rise Time

t_p: Pick Time

t_s: Settling Time

g: Gravitational acceleration

$I_{n\times n}$: *n*th order identity matrix

I_x: Moment of inertia along x axis

I_y: Moment of inertia along y axis

I_z: Moment of inertia along z axis

J: Body inertia matrix

J_d: Discrete-time performance index (cost function) of LQT controller

J(u): Continuous time cost function of LQR controller

K: Fixed (time-invariant) optimal control gain of LQR controller

k: Discrete time step

K_t: Aerodynamic drag matrix for translational motion

k_f: Final time

k_g: Motor gain

k_m: Torque factor (Constant that relates F_i and T_i)

k_n: Thrust factor (Constant that relates F_i and T_i)

m: Mass of the quadrotor

p; q; r: Body angular velocities

P_i: Power consumption of *i*th motor.

$P_{back;\, i}$: Power consumption of *i*th motor for backstepping controller.

Q: State weight matrix

R: Control weight matrix

T_i: Torque generated by *i*th rotor

U: Input vector

X: State vector

X_d: Desired state vector

Y: Output vector

S: Laplace variable

$\dot{f}$: First order time derivative

$\ddot{f}$: Second order time derivative

LIST OF UNITS

deg: Degree

Kg: Kilogram

m/s: Meter per seconds

N: Newton

rad: Radians

rpm: Revolutions per minute

LIST OF FIGURES

Figure 1.1 The first quadrotor helicopter

Figure 1.2 Appearance of the commercial Drangonflyer

Figure 1.3 Appearance of the EADS Quattrocopter

Figure 1.4 Appearance of the X-4 Flyers

Figure 1.5 Appearance of the quadrotor designed in UPenn

Figure 1.6 Appearance of the quadrotor designed in Cornell

Figure 1.7 Appearance of the quadrotor designed in ETH

Figure 1.8 Hovering quadrotor designed in Standford

Figure 1.9 Appearance of the quadrotor and its control system design in UBC

Figure 1.10 The scheme of the rotating direction of the quadrotors

Figure 2.1 The principle diagram of quadrotor UAV

Figure 2.2 Flight-attitude of quadrotor

Figure 2.3 Body-fixed Frame and Earth-fixed Inertial Frame

Figure 3.1 The Qball-X4 quadrotor UAV

Figure 3.2 Qball-X4 communication hierarchy and communication diagram.

Figure 3.3 The OptiTrack System Camera.

Figure 3.4 Servo-based payload releasing mechanism

Figure 3.5 The UAV system block diagram

Figure 3.6 Schematic representation of the Qball-X4.

Figure 3.7 The system diagram of MATLAB

Figure 3.8 The Module composition of Simulink

Figure 3.9 Schematic figure of simulation system module

Figure 4.1 Structure of PID control system

Figure 4.2 Plot of PV vs time, for three values of K_p (K_i and K_d held constant)

Figure 4.3 Plot of PV vs time, for three values of K_i (K_p and K_d held constant)

Figure 4.4 Plot of PV vs time, for three values of K_d (K_p and K_i held constant)

Figure 4.5 Principle of GS-PID and two rules for the interpolation of the gain K_p

Figure 4.6 Gain scheduling control structure

Figure 5.1 GS-PID controller Simulink model

Figure 5.2 Simulation results of pitch angle

Figure 5.3 Simulation results of roll angle

Figure 5.4 Simulation results of raw angle

Figure 5.5 Simulation results of altitude

Figure 5.6 Horizontal position output x, expect d_x and its error curve

Figure 5.7 Horizontal position output y, expect d_y and its error curve

Figure 5.8 System position output z, expected d_z and its error curve

Figure 5.9 System yaw angle ψ, and their expectations ψ_d and error curve e_ψ

Figure 5.10 Rolling angle curve

Figure 5.11 Elevation curve

Figure 5.12 3D trajectory of fixed point control

Figure 5.13 Control force F_1 curve

Figure 5.14 Control force F_2 curve

Figure 5.15 Control force F_3 curve

Figure 5.16 Control force F_4 curve

Figure 5.17 Composition of forces curve F_i $(i = 1,2,3,4)$

Figure 5.18 Position x under trajectory tracking, desired trajectory x_d and its error curve

Figure 5.19 Position y under trajectory tracking, desired trajectory y_d and its error curve

Figure 5.20 Position z under trajectory tracking, desired trajectory z_d and its error curve

Figure 5.21 Yaw angle ψ under trajectory tracking, desired yaw angle ψ_d and its error curve

Figure 5.22 Curve of rolling angle ϕ under trajectory tracking

Figure 5.23 Curve of pitching angle θ under trajectory tracking

Figure 5.24 3D rendering of trajectory tracking

Figure 5.25 Trajectory tracking control force curve F_1

Figure 5.26 Trajectory tracking control force curve F_2

Figure 5.27 Trajectory tracking control force curve F_3

Figure 5.28 Trajectory tracking control force curve F_4

Figure 5.29 F_i (i=1,2,3,4) under trajectory tracking control

Figure 5.30 Torque T curve following trajectory Tracking

Figure 5.31 Torque T_θ curve following trajectory tracking

Figure 5.32 Torque T_ψ curve following trajectory tracking

Figure 5.33 The flying path of Qball-X4 for payload dropping with PID controller

Figure 5.34 The flying attitude of Qball-X4 for payload dropping with PID controller

Figure 5.35 The total thrust of Qball-X4 with PID controller

Figure 5.36 The flying path of Qball-X4 for payload dropping with GS-PID controller

Figure 5.37 The flying attitude of Qball-X4 for payload dropping with GS-PID controller

Figure 5.38 The total thrust of Qball-X4 with GS-PID controller

Figure 5.39 Payload dropping experiment with GS-PID controller altitude output results

Figure 6.1 Control relationship of quadrotor UAV system

Figure 6.2 Double closed loop structure diagram of quadrotor unmanned aerial vehicle

Figure 6.3 Quadrotor system simulation model in Simulink

Figure 7.1 Backstepping controller Simulink model

Figure 7.2 Quadrotor unmanned aerial vehicle actual trajectory

Figure 7.3 Position x under trajectory tracking, desired trajectory x_d and its error curve

Figure 7.4 Position y under trajectory tracking, desired trajectory y_d and its error curve

Figure 7.5 Position z under trajectory tracking, desired trajectory z_d and its error curve

Figure 7.6 Yaw angle ψ, rolling angle θ and pitching angle θ under trajectory tracking

Figure 7.7 Controller output U1 (T), U2 (T), U3 (T) and U4 (T)

Figure 7.7 Propeller propeller force F_1, F_2, F_3, F_4

Figure 7.9 Estimation parameters

Figure 7.10 The flying path of Qball-X4 for payload dropping with backstepping controller

Figure 7.11 The flying attitude of Qball-X4 for payload dropping with backstepping controller

Figure 7.12 The total thrust of Qball-X4 with backstepping controller

Figure 7.13 Payload dropping experiment with Backstepping controller altitude output result

LIST OF TABLES

Table 3-1 Qball-X4 components

Table 5-1 Physical parameters of the quadrotor UAV Qball-X4

Table 5-2 PID controller parameters of quadrotor UAV

Table 5-3 GS-PID controller parameters for simulation

Table 5-4 GS-PID controller gains for flight test

Chapter 1

Introduction

1.1 Background

Unmanned Aerial Vehicles, or commonly known as UAVs, are gaining increasing popularities for multiple future applications, either in civilian or military fields. Among which UAV helicopters are the most noticeable type [1,2]. They have widely expanded the future possibilities of applications for commercial and military alike, with the aid of Vertical Take-Off and Landing, or known as VTOL abilities. The fields of current and future applications of the unmanned helicopter include aerial surveillance, border patrol, environmental monitoring [3,4,5,6], and search and rescue in natural disasters and factitious accidents such as earthquake, volcano eruption, flood, or accidents related to nuclear and radiations [7,8]. The detection and prevention of forest fire or urban high-level building fire [9,10], surveillance in order to prevent leakage of pipelines [11], and other highly risky circumstances for human pilots are also providing the necessities for the implementing of UAVs for such applications, especially for unmanned helicopters.

Flight missions of trajectory tracking and payload dropping is typical for a UAV helicopter in various applications, such as express delivery in urban, suburb or rural areas, fire detection and firefight, as well as sending payloads into areas of nuclear hazard. Helicopters were introduced for dumping seawater on a malfunctioned nuclear reactor for cooling down core temperature in Japan's Fukushima Daiichi Nuclear Power Plant during the earthquake in 2011. Considering the potential danger to the helicopter pilot caused by radioactive hazard, it would be avoidable for such risk if unmanned helicopters possessed with large carrying capacity could be deployed in such operation. The Joint Precision Airdrop System (JPAS), which is a national defense program proposed by the Joint Forces Command of the United States, provides new approaches of delivering supplies to

ground infantries while minimizing operation casualties [12]. The introducing of UAVs such as unmanned helicopters is an apparent approach in securing the success of such operations. All of these applications mentioned above have motivated the study and development of new and good performed control strategies for unmanned helicopters, especially a special model of helicopter with 4 symmetric propellers known as “quadrotor”, considering their significant advantages over other UAVs due to their unique VTOL and hovering capabilities and their relatively simple structure, simple dynamic model, as well as low cost. Control models implemented started with mature and commonly-used control algorithms, such as PID [13,14] or LQR controllers [15]. However, these simple linear methods were proved not to be promising in a number of certain scenarios when implemented on the nonlinear dynamic model of an unmanned quadrotor helicopter. Therefore, nonlinear controller designs have been put forward by implementing several different algorithms such as feedback linearization [16], sliding mode control [17], model predictive control [18], and backstepping control [19].

Nevertheless, the total masses of the control objects are always assumed to be a constant value in the flight process, and the mass changes during the flight periods have not been studied widely among the existing literature. During its payload dropping process, mass change is a factor that can not be neglected for a good performance of the flight. The study of a robust controller to optimize the performance of such a practical application of the quadrotor is the main motivation for this thesis work.

1.2 Review of Researches on Quadrotor Helicopter

The quadrotor helicopter was first invented in Europe in the early twentieth century, which was manufactured by the French Breguet brothers and other researchers [20]. This quadrotor helicopter needed a large number of personnel cooperation when taking off. Its stability is poor, and its flight height and time are not satisfied with the expected results. However, the successful manufacture and flight test of this quadrotor helicopter have laid a good foundation for the miniaturization and functional diversification of the subsequent multi-rotor aircraft. Figure 1.1 shows the structure of the first quadrotor helicopter.

With the rapid development of IT industry, the use of integrated circuit, micro inertial navigation system, intelligent control theory and other related technologies become more feasible.

More and more enterprises and research institutions have begun focusing on the research of the quadrotor helicopter. Many recent researches on the control of the quadrotor helicopter focus on either the modeling of aircraft and have verified the effectiveness of the control strategy by simulation, including the modeling of the aircraft and different control strategies, or whether the aircraft can execute certain missions successfully in a real environment.

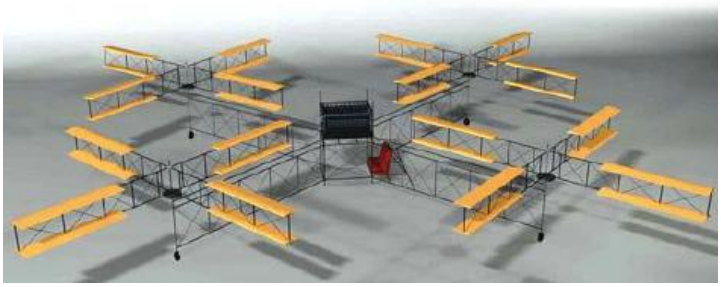


Figure 1.1 The first quadrotor helicopter [20]

1.2.1 Some Typical Quadrotor Helicopters

(1) Dragonflyer

Dragonflyer is a commercial product developed by RC toy company. It is controlled by R/C signal transmitting device and onboard control chip. Chip on board includes a leader for receiving input device, sensors measuring angular velocities of the vehicle in three directions, a micro controller to implement the control algorithm. The latest Dragonflyer also includes four infrared thermal sensors to help the aircraft balance during outdoor flight. The aircraft is made of carbon fiber and high density nylon material, with a distance of 30 feet at the two ends, and weighs 17 ounces [21] (Figure 1.2).

Figure 1.2 Appearance of the commercial Drangonflyer

(2) Quattrocopter

EADS Quattrocopter was originally used as a test platform for micro air vehicle control unit, and now it is produced by industry because of its good performance. The aircraft onboard chip developed is composed of MEMS Inertial Measurement Unit (IMU), changes of airflow sensor and a GPS receiving device, R/C signal receiver, 16 bit analog-to-digital converter and drive rotor power [22]. The lithium battery used by Quattrocopter can maintain 20 minutes of flight after one time full of electricity. This aircraft is about 65cm long and weighs about 0.5 kg, the body can be removed (Figure 1.3). Electric rotor reduces the aircraft noise to a very low level.

Figure 1.3 Appearance of the EADS Quattrocopter

(3) X-4 Flyers

Development of X-4 Flyers is in Australia. One of the two IMUs is the commercial IMU produced by Crossbow Co. and weighing about 475 grams. Another IMU called EiMU has been developed by a robotic team in Australia, weighing about 100 grams [23] which was finally used due to the light weight. A dual core board computer is implemented for recording the input command from R/C receiver. The serial interface on the IMU data is recorded at a frequency of 120Hz. Another serial interface is used to implement the communication between IMU and ground computing system.

The development of X-4 Flyers put forward a nonlinear model of the aircraft. This model is linear when aircraft is in hover. The transfer function of the pilot's input command to Euler angle output is obtained based on a linearized model. The sensing signal from the IMU is used to high pass filtering angular velocity measurement as required for inner loop control. Outer loop control is eventually abandoned, because it is difficult to obtain aircraft altitude values from the IMU.

X-4 Flyers is of a total weight about 2 kg, support is of 70 cm long, and the diameter of rotor is 11 inch. The flight test indicated that the lift of the aircraft is insufficient. In order to provide

enough lift and reduce weight, a new X-4 Flyers has been developed. The onboard power of this new plane could maintain 2 minutes flight, and the X-4 Flyers is equipped with wireless serial connection and camera system. The reformed X-4 Flyers adopts the anti-rotating rotor and the rotor rotating fixing device [24] with spring (Figure 1.4). The simulation results show that the new X-4 Flyers can be better controlled with very slow fluctuations. Since the dynamics of the new X-4 Flyers is more desirable, the control algorithm of the aircraft can be completely implemented by onboard processors without the assistance of the ground computing system.

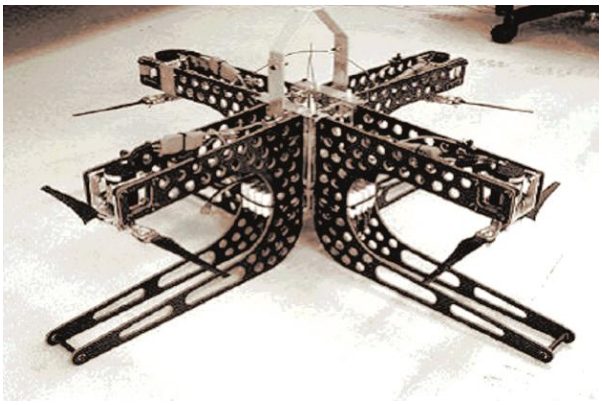


Figure 1.4 Appearance of the X-4 Flyers

1.2.2 The UAVs Developed in Some Universities

(1) The UAV Developed in the University of Pennsylvania

A research team at the University of Pennsylvania used a commercial model, HMX-4, to develop its own quadrotor UAV. The first paper published by them proposed that the control of the quadrotor UAV is implemented by onboard computers and ground computing systems to process data from sensors and cameras to implement [25]. Three onboard sensors provide a stable vehicle inner loop control. The camera placed on the ground is as the main sensor to be used. The five colors marking with 2.5cm in the aircraft base is used for providing location information for the camera. The relative position changes of the markers on the aircraft can be calculated by using the tag localization algorithm. Then, the pitching angle, roll angle, yaw angle and translational position coordinate of the quadrotor are calculated (Figure 1.5). Due to the weight limitation of the HMX-4 model, the aircraft could not be assembled with the GPS system and the acceleration measuring instrument.

The ground computing system is used to receive and process image information transmitted by

the ground camera, and to set target command values and transfer the calculated rotor drive inputs. The onboard computer stabilizes the aircraft through sensor signals, and obtains the control signals transmitted by the ground computing system through the onboard R/C receiving device.

They presented a dynamic model of the quadrotor UAV and two control strategies in their papers, which are respectively the feedback linearization method (feedback linearization) and backstepping method (backstepping). The simulation results show that backstepping method is better.

Recently, this quadrotor UAV has added an onboard camera to combine the ground camera to estimate the position of the aircraft [26]. On the basis of five markers that were previously observed by ground cameras, another marker was taken on the ground camera for onboard camera observation.

In the first paper of the team, the HMX-4 aircraft experiment [25] using double camera method was recorded. The onboard and ground calculator is used for video signal processing. Three onboard sensors are used for aircraft attitude control. Using ground camera to calculate the flight height of the aircraft is proved to be effective in flight control experiment, but it can only be used in specific environment. For example, the method can be used for aircraft cameras at the specified location for takeoff and landing.

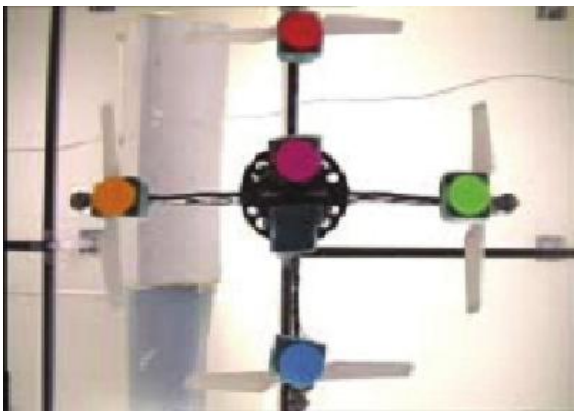


Figure 1.5 Appearance of the quadrotor designed in UPenn

(2) The UAV Developed in Cornell University

The Cornell University has launched two quadrotor UAV projects. The objective of the first project is to calculate the flight height of the aircraft by using three onboard sensors and a ground video system [27]. The four LED is placed at the four ends of the cross arm of the aircraft to provide data for the video system composed of three cameras on the ground. The onboard computer

transmits sensing data to the ground computing system and adjusts the rotor speed according to the instruction data from the ground computing system. The ground computing system calculates the input of the rotor according to the sensing data and the video signal and transmits it to the on-board computer (Figure 1.6). In this project, a Carle filter is implemented to estimate the height of the quadrotor UAV in real-time. This filter is mainly used to retain high frequency sensor data (300Hz) and low frequency video signals (10Hz), and filter out interference signals of other frequency components. The experimental results show that the filter successfully eliminated the adverse effects of sensor offset.

The second project in Cornell University on the quadrotor UAV is completed in a Master's thesis [28]. This project focuses on the four lift generating devices of aircraft and the overall structure of the aircraft. Compared with the previous UAV, the new aircraft increases its weight to 6.2kg. They use MATLAB and ANSYS finite element software to design the bracket of the quadrotor UAV to determine the size and stress intensity of its structural unit. Although the drive circuit of brushless rotor is very complicated, it is still used to achieve a higher power weight ratio (power to weight ratios). The large diameter rotor is used to ensure the stability of the system.

Figure 1.6 Appearance of the quadrotor designed in Cornell

The quadrotor UAV uses onboard power supplies and sensors, which occupy two of the mass of the aircraft, and the inertial measurement unit is produced by Systron-Donner. Although IMU has a certain degree of drift, but it is enough to ensure the stability of aircraft hover.

The simulation of the nonlinear dynamics model of the aircraft includes external disturbance, sensor offset and noise. Simulation is the key for the controller and state estimator design, modulation and detection. In many flight experiments, several times succeeded, but UAV fall in one

of the experiments and caused IMU damage.

(3) The UAV Developed in Swiss Confederation Institute of Technology

A team from ETH studies angular velocity control and flight height control of the quadrotor UAV (Figure 1.7). In the simulation of aircraft system dynamics, two control methods, namely PID and LQ, are adopted and validated by [29]. The controller developed according to the two control algorithms is equipped with a test platform which can limit the flight height and only allow the rotational angle change. A commercial IMU and a Kalman filter that use accelerometer and magnetometer to eliminate sensor drift are used to determine the position and direction of the quadrotor UAV. A ground PC is used to send signals to the test platform, and a PIC microcontroller is used to perform the calculation of the control quantities. The experiments show that the classical PID control is better than the LQ control, and the PID controller allows the power supply from the ground power supply to the aircraft through the wire. One possible reason for the poor performance of the LQ controller is that the transfer function does not consider the dynamic characteristics of the drive device (actuator).

Figure 1.7 Appearance of the quadrotor designed in ETH

(4) The UAV Developed in Stanford University

Stanford University uses an improved Drangonflyer as its autonomous aircraft Multi-Agent Control Research (STARMAC) test platform [30]. After a number of outdoor experimental flights and validation of the multi aircraft control scheme, the quadrotor UAV is selected from the traditional helicopter and fixed-wing aircraft. Drangonflyer's original chip was replaced by a printed

circuit board designed by the University of Stanford, which performs all sensor and communications tasks. It consists of the following components: a commercial IMU called MicroStrain, two pieces of PIC micro control chip, a supersonic sonar sensor, a GPS5 unit and two Bluetooth devices with an effective distance of 150-300 feet. The ground computing system consists of several PC machines and a notebook computer equipped with a standard joystick for remote control aircraft.

IMU gets the data from the sensor, estimates the height of the current aircraft and its speed of change, and then outputs the two data. Due to the strong vibration of the support of the aircraft when the lift is large, the two data are likely to mix with large noise, so the accuracy is low. Because the data read from a supersonic sensor is unreliable, a Kalman filter is implemented to estimate the height of the aircraft. The system also adds an infrared range sensor to help track the trajectory of the aircraft in a specific area. Another Kalman filter is used for GPS measurements, velocity measurements, together with altitude information to calculate the position and velocity of the aircraft.

Figure 1.8 Hovering quadrotor designed in Standford

Experiments show that the aircraft performs well in outdoor hover (Figure 1.8). In the experiment, some artificial inputs are needed to suppress the wind disturbance, while the height input commands remain constant. The team's future goal is to use multi-agent technology to make the flying behavior of a four multi-rotor aircraft as multiple agents in a system.

(5) The UAV Developed in University of British Columbia (UBC)

The quadrotor UAV developed by the Institute of Electrical and Computer Engineering at University of British Columbia focuses on the nonlinear modeling of aircraft [31]. The

experimental equipment consists of a flight turntable, a DSP control system, a microprocessor and a set of wireless signal transmission devices (Figure 1.9). Based on the nonlinear model of aircraft, the research team designed a H_∞ loop controller to achieve the stability control of the stability, speed and yaw.

The PIC16F877 microcontroller is used to transfer the control data to the pulse width modulation signal to reduce the operating load of the CPU. Through 4 channel Futaba wireless transmission equipment, the modulation signal is further used to control the quadrotors of the Dragonflyer. Based on the nonlinear model of the aircraft, the H∞ loop controller is used to control the stability, velocity and yaw-angle of the aircraft. A model predictive controller, or known as MPC, is used to control the position of the aircraft.

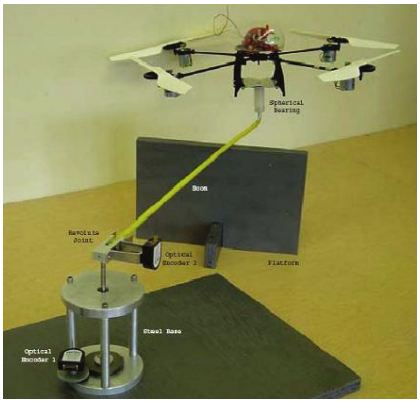


Figure 1.9 The appearance of the quadrotor and its control system design in UBC

1.3 The Control Principle and Key Technologies of Quadrotor UAV

1.3.1 Quadrotor UAV Structure and Control Principle

The traditional rotary helicopter system consists of a main rotor and a tail rotor structure. The traditional helicopter changes the lift of the aircraft by changing the cutting angle of the rotor blade rotation. In contrast, because the rotational angle of the rotor blade of the small quadrotor UAV is fixed, quadrotor UAV is controlled by changing the angular speed of each rotor rotation, making the control of quadrotor UAVs relatively simpler and easier.

The rotors of a quadrotor UAV are located in the front, back, left, right ends of a geometrically symmetric crossed bracket as shown in Figure 1.10. The quadrotor is controlled by four rotors, and the whole aircraft is controlled by altering the speed of each rotor to achieve flight attitude control.

Figure 1.10 is a typical structure of the quadrotor UAV. The front rotor 1 and the rear rotor 3 rotate counterclockwise, while the left rotor 2 and the right rotor 4 rotate clockwise to balance the torque generated by the rotor rotation. By adding a group of rotors (front/rear rotor 1, 3 or left/right rotor 2, 4), one end of the rotor increasing its speed while the other end reducing the same speed can produce different thrust for the maneuvering of the aircraft. The rotor speed difference between the left and the right rotor 2 and 4 is used to control rolling motion, while the front and rear rotor 1, 3 speed difference is used to control pitch motion. The yaw rate is controlled by rotating clockwise and counterclockwise relative rotation rate.

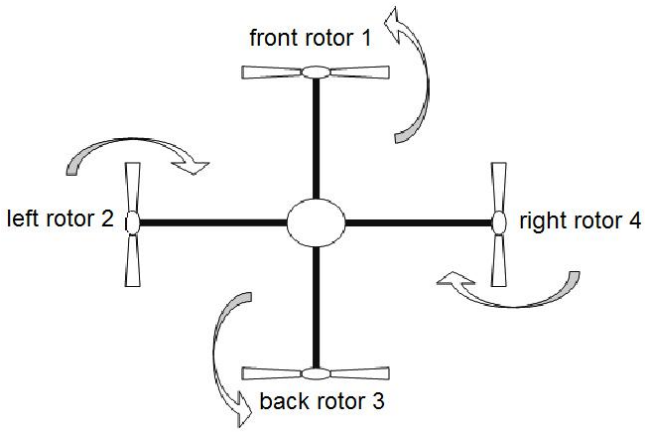


Figure 1.10 The scheme of the rotating direction of the quadrotors

In terms of dynamics, the quadrotor UAV system itself is unstable [31], [32]. Therefore, it is particularly critical for the design of control algorithm to guarantee the system stability. Because the quadrotor UAV is a system with six degrees of freedom (three angular motion displacements and three linear motion displacements) and the control variables are only four, this means that there is a coupling relationship between the controlled volumes [32]. Therefore, the designed control algorithm must be efficient enough for the underactuated system in order to achieve steady control for the three angular motion displacements and three linear motion displacements.

1.3.2 Key Technologies of Quadrotor UAV Control

So far, the basic theory and experimental research of the quadrotor unmanned aerial vehicle have made great progress, though there are still challenges that need further investigation and improvements.

(1) Optimization of Overall Design

In the overall design, the following principles should be followed: small wingspan, light weight, fast flight, low energy consumption, and low cost. But there is a constraint for these design principles. For example, when the weight of an aircraft keeps constant, its wingspan is inversely proportional to the speed and power consumption of [33]. Therefore, in the overall design, the aircraft weight should be reduced as much as possible, and the appropriate aircraft materials should be selected according to the performance and price. The overall design optimization can be achieved by combining the constraints of weight, size, flight speed and power consumption.

(2) Power and Energy

The energy of rotor, DC motor / brushless DC motor, deceleration device and drive module is supplied by lithium battery. Main factors affecting vehicle size is the weight, and lithium battery weight occupies a large proportion in the aircraft total weight. Sometimes the battery weight accounted for the UAV total weight of 75% [34]. Therefore, the key of quadrotor unmanned aerial vehicle is to develop a lighter, more efficient power and energy. In addition, the rotor to drive propellers for providing lift will consume a lot of energy, sometimes as high as 91% of energy is consumed by rotor [34]. The key to improve the efficiency of power is to improve the rotors efficiency. Also in the premise of ensuring optimal efficiency and power output, make the rotor work in the recommended high efficiency area.

(3) Dynamics Modeling

In order to effectively control the quadrotor unmanned aerial vehicle, it is necessary to establish its mathematical model accurately in a variety of flight states. But the flight is always affected by such as air resistance, gravity, inertia, gyroscopic rotor and air flow and so on many physical effects and external factors, it is difficult to accurately and effectively build the dynamic model. In addition, due to the small size, light weight and easy deformation of the rotor, the aerodynamic parameters could not be accurately obtained, and the accurate modeling is also affected. Moreover, the aerodynamic problems of rotor at low Reynolds number should be further studied and solved.

(4) Flight Control

The quadrotor unmanned aerial vehicle (UAV) is an underactuated system with 6 degrees of freedom and 4 variable inputs [35]. The difficulty of flight control system design is mainly due to its characteristics of multivariable, nonlinear, strong coupling and interference sensitivity. In addition, the modeling accuracy and precision of sensor also affects the performance of the

controller. The key of flight control is attitude control, because its attitude and position are directly coupled. Because flight control is dependent on the accuracy of the model, the adaptive controller which is suitable for the specifying environment is needed.

(5) Navigation, Positioning and Communication

Quadrotor unmanned aerial vehicle is mainly used in urban, forest, tunnel and indoor environment of flight. In these complex environments, the current navigation, positioning and communication could not fully meet the needs. In order to develop accurate and reliable navigation system, integrated navigation, GPS, light, sound, radar and terrain matching technologies are needed. In addition, due to the complex terrain and interference near the ground, there is no communication chain technology that meets the actual needs for security, reliability and anti-interference. Therefore, the key for the development of the quadrotor aircraft is to develop a communication chain of small size, light weight, low power consumption, and with good stability and reliability.

1.4 A Review of Control Methods for Quadrotors

1.4.1 Survey of Control Methods

Due to the requirements of autonomous flight under different flight conditions without a pilot onboard, control of UAV is much more challenging compared with manned aerial vehicle since all operations have to be carried out by the automated flight control, navigation and guidance algorithms embedded on the onboard flight microcomputer/microcontroller or with limited interference by a ground pilot if needed. Successful application of the quadrotor UAV is closely tied with existence of autopilot control systems. The controller design is the research hotspot of the quadrotor unmanned aerial vehicle (UAV) in the world today. Many researches have designed different controllers to control the position and attitude of the quadrotor UAV. In general, before the design of the controller, it is necessary to understand the model information of the system or the main characteristics of the system, then to establish the quadrotor UAV system model. Each parameter of the system model must be acquired. This involves the parameter identification problem of the model. When the above work is completed, the software simulation platform should be built. Finally, the control algorithm is designed, and the simulation and hardware in the loop

simulation are carried out, and the control effect is analyzed.

The characteristics of multivariable, nonlinear, underactuated and strong coupling make the quadrotor UAV become a complex unstable system. How to analyze the complicated system, to design a reasonable flight controller, and to achieve safe and satisfactory UAV flight control are an extremely critical problem. As mentioned earlier, the quadrotor UAV is a typical underactuated unstable object, and its dynamic characteristics are strong coupling, strong nonlinear, multivariable. The safety, reliability and performance acceptability are important indicators to measure a control system.

Several control techniques have been proposed for the quadrotor helicopter flight control, and with the development of control theory, more and more control algorithms are applied to the flight control of UAV. However, as mentioned above, linear methods cannot have a good performance for the nonlinear quadrotor dynamic model in all the cases. The problem of nonlinear controller has been addressed using multiple methods such as feedback linearization [16], sliding mode control [17,36,37], backstepping control [19,38], and adaptive control [39].

A proportional-integral-derivative controller (PID controller) is a generic control loop feedback mechanism (controller) widely used in industrial control systems; a PID is the most commonly used feedback controller. A PID controller calculates an "error" value as the difference between a measured process variable and a desired set-point. The controller attempts to minimize the error by adjusting the process control inputs. PID control algorithm was developed in late 1930s. From the analog controller to the digital controller development process, PID controller has achieved great success through in-depth theoretical research and rich application practice. The PID controller contains profound philosophical ideas. The integral reflects the historical change of the input signal, and the proportion reflects the current state of the input signal, while the differential represents the future trend of the input signal. PID controllers are widely used in industry and aviation applications because of their small dependence, relative reliability, simplicity and ease of adjustment. Because the accurate mathematical model of the quadrotor UAV is difficult to obtain, so the control of the quadrotor UAV is dominated by PID control. However, the classical PID is not suitable for the nonlinear, multivariable, underactuated and symmetric characteristics of the quadrotor UAV control system. Therefore, many researchers have improved the conventional PID algorithm and successfully applied it to the control of the quadrotor UAV.

Prof. Youmin Zhang in Concordia University and his research groups [40-42] proposed for multiple control algorithms including Model Predictive Control (MPC), LQR controller, sliding mode controller, and backstepping controller, etc. on an unmanned quadrotor helicopter for robustness in multiple flight missions. His works and contributions mainly focus on the fault-tolerant control aspect of the UAV control, which handles faults that could occur in a certain flight mission. Based on the nonlinear mathematical model of the quadrotor unmanned aerial vehicle, Yang et al [43] proposed that the adaptive sliding mode control method is used to directly repair the UAV actuator damage, and the effectiveness of the proposed method is verified in a simulation environment. Cen Z H et al. [44] proposed a fault diagnosis method based on nonlinear observer, and applied it to a real UAV platform, which verified that the method has certain accuracy in detection, isolation and estimation of faults. The fault diagnosis and fault-tolerant control of UAV sensor based on hardware redundancy are proposed in the paper [45]. This hardware redundancy fault-tolerant control method is more reliable. The Swiss company Meteomatics put forward a method of at least two enhance passive components in the radial symmetrically arranged around the quadrotor UAV yaw axis to prevent the quadrotor from an abrupt failure. These torsional lifting elements can start rotating around the yaw rotation, thereby increasing dynamic lift, and reducing the UAV crash and the collision speed [46]. However, the practical application for many of these control approaches concerning fault-tolerant still have a lot of work to do. Gain scheduling PID active fault tolerant control proposed in the literature [47] need to repeatedly adjust PID parameters online, increasing the workload. It is difficult to accurately estimate the size and occurrence time of the fault by using the residual error generated by the height measurement and altitude instruction of the OptiTrack camera. The fault tolerant control effect is limited.

Due to its predominant prediction and constraints handling capabilities, MPC is one of the most widely used advanced model-based control techniques in process industries, in particular chemical and petro-chemical industries as examples, in addition to the model-free PID controllers. MPC is capable of systematic handling of operational constraints existing on amplitude of the control signals, rate of change of the control signals, and system's output/state variables [48]. However, due to the specific structure of MPC, its successful implementation is highly dependent on the availability of sufficient computational power. Such a difficulty has inhibited the use of MPC for aerospace systems due to the inherently fast and complex dynamics of most aerospace systems.

Based on the use of mathematical model of robot manipulators and Lyapunov stability theory, global asymptotic stability of particular cases of the class of nonlinear global PID regulators, originally proposed by Santibanez and Kelly, have been extended in many researches [50-52] for several different cases. However, all stability analyses need system's mathematical model and by making use of Lyapunov stability theory. Yarza et al [53] gave a proof for the stability of two degree of freedom rigid robot arms controlled by PID algorithms based on a model of the robot where the nominal decoupled (diagnonal system) linear part is emphasized as investigated by Rocco [54], a priori sufficient conditions for asymptotic stability in the Lyapunov sense and input-to-state stability (ISS) are provided. Also a stochastic Lyapunov theory to perform stability analysis of MPC controllers for nonlinear deterministic systems is put forward where the underlying optimization algorithm is based on Markov Chain Monte Carlo (MCMC) or other stochastic methods [55].

Because of the utility in engineering practice, adaptive control of nonlinear systems is always the most attractive research direction in control theory. Since backstepping adaptive control scheme for a class of nonlinear systems is first proposed, the research on the backstepping adaptive control of nonlinear systems has made much progress [56]. In 1995, Richard et al. [57] first discussed the backstepping control scheme for the missile longitudinal motion controller; In 1998, Steinberg et al [58] designed a backstepping adaptive flight control law for a complex high performance aircraft and carried out a simulation study. The results show that even if there is a large model error, even in the case of a flat tail failure of a plane, the backstepping adaptive control shows fast convergence and good robustness. Steinberg [59] has carried out a preliminary study on the backstepping adaptive control of the flight control system, and compared different design methods of flight control (fuzzy control, backstepping adaptive control, variable structure control, nonlinear dynamic inverse control, indirect adaptive control, etc.). The results show that the backstepping method has outstanding advantages in the design of future flight control system. In 2002, Krstic et al. developed the theory of backstepping adaptive control for steady linear systems, and achieved some good results [60]. In the design of backstepping controller, the most difficult problem is how to deal with the differential problem of the virtual control signal. Krstic et al. put forward a method of constructing Lyapunov function to avoiding the differentiation of signals. In 2012, Lee and Kim [61] use neural network and backstepping adaptive method to discuss the design of flight controller.

The Lyapunov function plays an important role in the controller design of nonlinear control

systems. For a long time, although Lyapunov stability theory has achieved many results, there is no general method to construct Lyapunov functions. By 1980s, Saberi et al put forward backstepping design method for partially linear strict feedback systems. This method uses the reverse thrust design. In each step, the change of the state coordinates, the adaptive adjustment function of the uncertain parameters and the stabilization function of a virtual control system of a known Lyapunov function are connected, and the stabilization controller is designed by the gradual correction algorithm to achieve the global adjustment or tracking of the system. This method can be applied to uncertain systems with state linearization or strict parameter feedback, and it can be conveniently achieved by symbolic algebra software. Therefore, the backstepping control method has attracted the attention of many scholars in recent years.

The GS-PID controller and the backstepping controller will be implemented in this thesis. So, The GS-PID and backstepping control used for quadrotor developed in recent year will be briefly presented in the following sections.

1.4.2 GS-PID Control and Backstepping Control for Quadrotor UAV

(1) Gain-Scheduled PID Control for Quadrotor UAV

The PID controller has been applied to a broad range of controller applications. It is indeed the most applied controller in industry [62]. The classical PID linear controller has the advantage that parameter gains are easy to adjust, is simple to design and has good robustness. However some of the major challenges with the quadrotor control include the non-linearity associated with the mathematical model and the imprecise nature of the model due to unmodeled or inaccurate mathematical modeling of some of the dynamics. Therefore, applying PID controller to the quadrotor limits its performance.

A PID controller was used for the attitude control of a quadrotor by Lee et al [63], while a dynamic surface control (DSC) was used for the altitude control. Applying Lyapunov stability criteria, Lee et al. were able to prove that all signals of the quadrotor were uniformly ultimately bounded. This means that the quadrotor was stable for hovering conditions. However, the simulation and experimental results reveal the PID controller to have performed better in the pitch angle tracking, whereas large steady state errors could be observed in the roll angle tracking.

In another work by Li and Li [64], a PID controller was applied to regulate both position and

orientation of a quadrotor. The PID parameter gains were chosen intuitively. The performance of the PID controller indicated relatively good attitude stabilization. The response time was good, with almost zero steady state error and with a slight overshoot.

It is generally established in the literature that the PID controller has been successfully applied to the quadrotor though with some limitations. The tuning of the PID controller could pose some challenges as this must be conducted around the equilibrium point, which is the hover point, to give good performance.

However, classical PID control sometimes could not meet the flight safety of the aircraft in special circumstances when the parameters or the state of the UAV change rapidly, such as UAV with a flight mission of payload dropping, or flying at abnormal state. In recent years, Prof. Zhang in Concordia University proposed to use Gain-Scheduling based PID (GS-PID) controller for fault tolerant control and payload dropping application using a quadrotor UAV [1]. It is known that GS-PID is a versatile technique which can be used for situations in which the parameters or the operating conditions of the plant can change largely and rapidly [13,14]. In aerospace applications, different portions of the flight envelope must be considered in control system design for different flight conditions. Various phases of flight need proper tuning of the controller gains under different flight conditions [15,16].

(2) Backstepping Control for Quadrotor UAV

Backstepping control is a recursive algorithm that breaks down the controller into steps and progressively stabilizes each subsystem. Madaniand co-researchers [65] applied backstepping control to stabilize a quadrotor system consisting of an underactuated, fully actuated and propeller subsystems.

The backstepping approach was also applied in reference [66] for attitude stabilization of a quadrotor. Using Lyapunov stability analysis, the closed-loop attitude system was found to be asymptotically stable with all states uniformly ultimately bounded in the presence of external disturbance. It was also implicit that the Quaternion formulation also helped in the computational side for stabilization in addition to avoiding singularity.

To increase robustness (to external disturbances) of the general backstepping algorithm, an integrator is added and the algorithm becomes Integrator backstepping control as articulated by Fang and Gao in [67]. The integral approach was shown to eliminate the steady-state errors of the

system, reduce response time and restrain overshoot of the control parameters.

1.5 Motivation of This Study

As it has been indicated, quadrotor UAV is a very advantageous VTOL concept especially in terms of aerodynamic and mechanical simplicity. However, control of the vehicle is not straightforward and requires solving some problems due to under-actuated, highly nonlinear and unstable dynamics of the vehicle. Therefore, it is really motivating and challenging to design and test autonomous control systems for quadrotor UAVs.

In this thesis, the objective is to obtain trajectory tracking and payload dropping of a quadrotor UAV by controlling its attitude and position simultaneously. Two different control methods are used independently. One of the control methods is the GS-PID controller. The other method is the backstepping nonlinear control technique and it has been used by various researchers and experimentally proven for the control of quadrotors.

1.5.1 GS-PID Controller

In view of popularity of GS-PID and advantages of such simple and model-free control design strategy, a GS-PID controller that assumes a separate set of gains for an actuator fault injected procedure is designed and implemented for the trajectory tracking and payload dropping of a quadrotor UAV in this thesis. The interpolation algorithm is also used to optimize the Gain Scheduling PID tuning process, which reduces the workload of GS-PID controller tuning, as well as optimizes the fault-tolerant control performance. The proposed control law stabilizes the quadrotor UAV system and drives the system states, including three-dimensional positions and yaw angle to track the desired path, while keeps the closed loop system stable. In this thesis, first a single PID controller is implemented on a dynamic model of a unmanned quadrotor helicopter. Then a GS-PID controller presetting several sets of controller gains for each phase of flight during the trajectory tracking and payload dropping missions. Over the course of flight the controller switches between each set of gains in order to maintain the desired height and minimize the overshoot of the quadrotor. Maintaining the height within small deviations from the desired set-point is intended.

1.5.2 Backstepping Controller

As indicated above, because of its flexibility in nonlinear processing, backstepping is the most commonly used method for the design of nonlinear system controller. It is a regression design method combining the selection of Lyapunov functions with the design of the controller. Due to the abandonment of the deterministic equivalence principle and the normalized parameter regulation, the backstepping adaptive controller designed has the following advantages over the traditional adaptive controller. 1) The transition process of the system tracking error can be greatly improved, and the definite relation between the performance index of the system tracking error and the design parameters is given. When the high frequency gain is known, the definite relationship between the performance index of the system tracking error and the design parameters can be given, thus the concrete way is pointed out for improving the tracking precision. 2) The influence of the tuning law on the system is weakened. 3) Under the condition of no parameter adjustment, backstepping control is still robust to parameter uncertainties.

Although the backstepping method has been used by many researchers, with some improvements, the backstepping control is investigated and used in this thesis for more accurate and efficient tracking of the quadrotor UAV compared with PID or GS-PID controllers.

1.6 Thesis Structure

This thesis focuses on the design of control system for the trajectory tracking and payload dropping of a small unmanned quadrotor helicopter. Firstly, a dynamic model of quadrotor UAV is established in this thesis. Then, the quadrotor UAV flight controller based on Gain Scheduling based PID approach (GS-PID) and backstepping approach are presented. Finally, the trajectory tracking and payload dropping of a quadrotor UAV were simulated with Quanser unmanned experimental system in the Networked Autonomous Vehicles Lab (NAVL) of Concordia University.

This thesis includes this first chapter and following 7 chapters:

Chapter 2 presents flight principle and system modeling of quadrotor helicopter. Equations of motion governing dynamics of the Qball-X4 are presented in this chapter.

Chapter 3 shows the Qball-X4 UAV system and MATLAB/Simulink control simulation platform. The overall system hardware and simulation and experimental software were described in

this chapter.

Chapter 4 presents gain scheduling based PID controller of the quadrotor helicopter. This chapter describes the GS-PID control algorithm and discusses how it can be applied to the control of the Qball-X4 for the trajectory tracking and payload dropping.

Chapter 5 shows the simulation and experimental results of the trajectory tracking control and payload dropping control of Qball-X4 helicopter with GS-PID Controller.

Chapter 6 presents backstepping controller of the quadrotor helicopter. This chapter describes backstepping control principle based on Lyapunov stability theory, and discusses how it can be applied to the control of the Qabll-X4 for the trajectory tracking and payload dropping.

Chapter 7 shows the simulation and experimental results of the trajectory tracking control and payload dropping control of Qball-X4 helicopter with backstepping control Approach.

Chapter 8 shows the conclusions and future works of this thesis.

Chapter 2

Flight Principle and System Modeling of Quadrotor Helicopter

2.1 Flight Principle of Quadrotor Helicopter

The mechanical structure of the quadrotor helicopter is fairly simple. Its basic structure consists of a rigid cross symmetrical bracket and a rotor drive system connected to each shaft. The four rotors are respectively fixed on the four vertex positions on the cross structure. These rotors are symmetrically distributed in the front, back, left and right directions of the airframe, and at the same level. The quadrotors have the same structure and radius. As shown in Figure 2.1, rotor 1 and rotor 3 rotating counterclockwise, the rotor 2 and the rotor 4 clockwise. The quadrotors and speed control devices are symmetrically distributed below the quadrotors, and the middle part of the bracket is equipped with flight controllers, batteries and other external devices.

Three positive direction axes are determined according to the right-hand rule. The four rotors are used as direct power source of quadrotor UAV. The two rotors on the diagonal are the same group, the same group of rotors turns the same, and the rotors in different groups turn opposite. The quadrotor is connected with the cross structure of the resulting anti-torque can offset each other in the case of the same speed.

The quadrotor UAV is different from the conventional UAV. The flight states of the aircraft can be controlled by controlling the rotational speed of the quadrotors. It changes the speed of the rotor by adjusting the speed of the quadrotors, and achieves the change of lift, so as to control the attitude and position of the UAV. The quadrotor UAV has 6 degrees of freedom (6 DOF) in space (moving and rotating along 3 coordinates), and the 6 degrees of freedom are controlled by the rotation speed of 4 rotors.

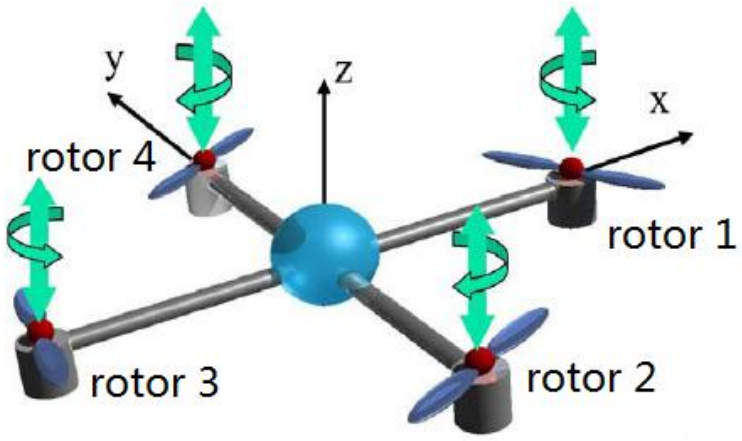
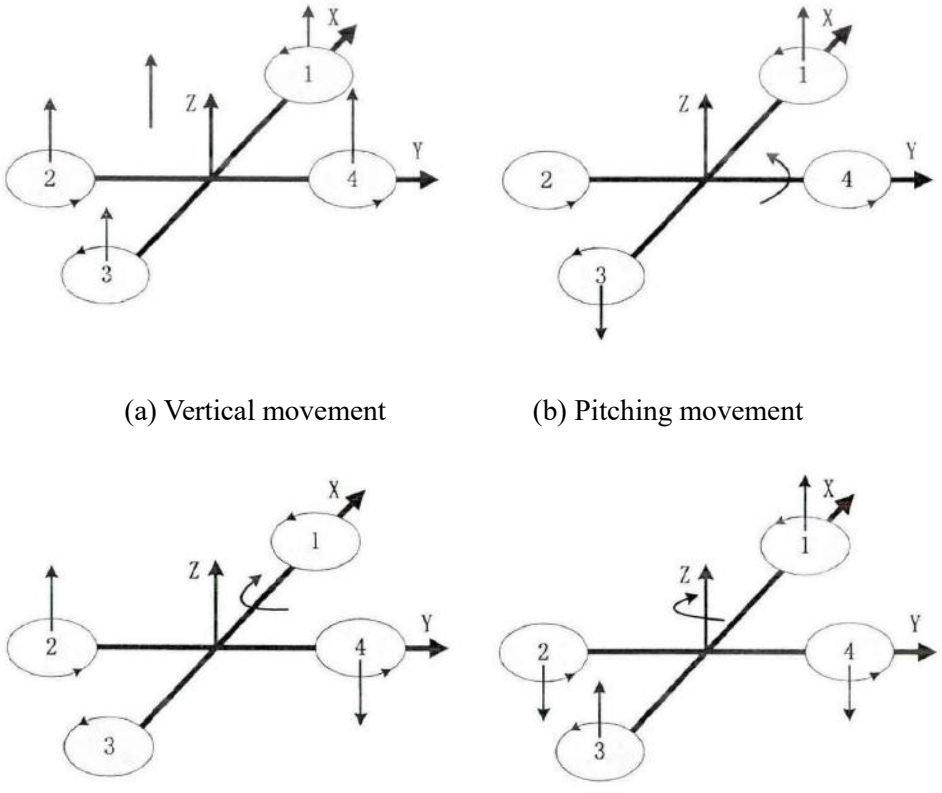


Figure 2.1 The principle diagram of quadrotor UAV

The basic movements of UAV include vertical motion, front and rear motion, left and right motion, pitching motion, roll motion, and yaw motion. Each flight attitude is shown in Figure 2.2, in which the upward arrow represents the increase of the speed, and the downward arrow represents the decrease of the speed.



(a) Vertical movement (b) Pitching movement

(c) Rolling movement (d) Yawing movement

Figure 2.2 Flight-attitude of a quadrotor helicopter

(1) Vertical Motion:

As shown in Figure 2.2 (a), the two groups of rotor reversal, balanced rotor torque on the fuselage, while increased four PWM rotor input, the rotor speed increases the total lift increase, when the total lift large enough W to overcome the body gravity, the UAV will leave ground vertical flight.

On the contrary, when the PWM input of the quadrotors is reduced, the quadrotor will drop vertically; when the external disturbance is zero and the total lift of the quadrotor is equal to the gravity, the UAV can achieve hovering flight mode.

(2) Pitching and Forward Motion:

When the speed of rotor 1 increases, the speed rotor 3 is reduced and that of rotor 2 and 4 keep constant. At the same time, in order not to produce speed changes caused by the change of the UAV overall torque and lift, keeping the rotor 1 and the rotor 3 equal in the amount of change, and creating the imbalance torque which causes the body to rotate around the Y axis is necessary. The UAV's pitching motion can be realized this way, and the horizontal force component can be generated at the same time in order to make backward flight. Similarly, when the speed of the rotor 1 drops and that of the rotor 3 rises, the UAV will rotate in the other direction of the Y axis. Pitching and front and rear motion can be made as well. The flight principle of pitching and forward motion is as shown in Figure 2.2 (b).

(3) Rolling, Left and Right Motion:

Because of the special symmetrical structure of the quadrotor UAV, similar to Figure 2.2 (b), the principle of rolling and left and right motion is the same as that of the pitching motion, change the rotor 2 and rotor 4 speed, while keeping the rotor 1 and a rotor 3 speed constant, can make the body to rotate around the X axis forward or reverse, so as to achieve the rotation and lateral rolling exercise.

(4) Yawing Motion:

Yaw is achieved by changing the rotor torque on the body. In the process of rotor rotation, the air resistance will produce the reverse torque which is opposite to the rotation direction. In order to overcome the influence of the anti-torque, the two rotors of the quadrotors are inverted and inverted respectively. The anti-torque is decided by the size of the quadrotor speed, when the quadrotor at the same speed, the rotor anti torque offset each other, the quadrotor not yaw; once changed the balance, anti-torque will produce yaw movement. As shown in Figure 2.2, when the rotation speed

of the motor 1 and 3 motor decreases, and rotor 4 and 2 extra rotation speed increase, the anti-torque produced by the motor 1 and the motor 3 is less than the torque generated by the motor 2 and the motor 4. At this time, the counter torque difference will act on the body rotating around the Z axis, so as to achieve the yaw movement.

2.2 Coordinate System and Coordinate Transformation Matrix

2.2.1 Coordinate System

The basis of flight control is the precise definition of a series of Descartes reference frames. It is meaningless to discuss the motion of a single object without reference. In the course of flight, the position, attitude angle and velocity direction of the quadrotor UAV are relative values, so in order to describe the motion accurately, the appropriate coordinate system must be selected first. To establish the mathematical model of the aircraft and design the flight control system correctly, it is necessary to make reasonable selection of different coordinate systems, and give a complete description of various flight parameters.

The flight of the quadrotor has both its own body movement and the relative ground motion, so we define the body coordinate system and the ground coordinate system, and the schematic diagram is shown in Figure 2.3.

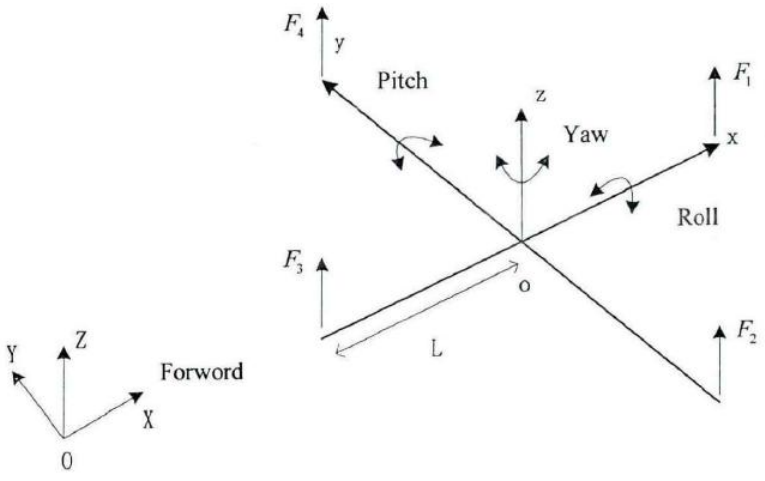


Figure 2.3 Body-fixed frame and Earth-fixed inertial frame

(1) Body Coordinate System (*oxyz*)

The origin of the body coordinate system is the center of gravity of the quadrotor UAV. The rotor 1 rotor 3 axis is *x*, rotor 2 and 4 axis is the *y*, perpendicular to the *x*-*o*-*y* plane direction as *z*

axis, and *x* axis pointe rotor 1, *y* axis direction points 4, *z* axis vertical.

(2) Ground Coordinate System (*OXYZ*)

The ground coordinate system is used to observe the motion state of the UAV relative to the ground and determine the coordinate of the space position of the UAV. The origin of ground coordinate is selected on UAV takeoff point. In order to establish the model conveniently, the coordinates of the determined coordinates are in accordance with the coordinates of the body, which conforms to the right-hand rule.

Based on the coordinate system described above, six spatial dimensions that can represent the dynamics of the aircraft are introduced. The body position coordinates [*x, y, z*] is used to describe the relative position of the vehicle mass center in space with the ground coordinate system is used as reference. The attitude angles [θ, ϕ, ψ] represent the transformation relation between two coordinate systems.

2.2.2 Attitude Description Method (Coordinate Transformation Matrix)

Because of the low flying height and low flying speed, the quadrotor aircraft can ignore the influence of the earth rotation and other factors. The navigation coordinate system and the carrier coordinate system are used. The navigation coordinate system is defined as the reference coordinate system in space. In the course of flight, the reference coordinate system of the quadrotor aircraft remains unchanged, and the carrier coordinate system is rotated relative to the reference coordinate system to obtain a new carrier coordinate system.

The attitude representation of the aircraft is the basis of the modeling of the quadrotor aircraft system, and the chosen attitude representation parameters determine the expression of the system model. The attitude of the quadrotor aircraft in space is represented by the relative angle between the carrier coordinate system and the geographic coordinate system. In this thesis, the pitch angle, roll angle and yaw angle are used to represent the attitude angle of the carrier.

(1) Pitch angle: the angle between the Y axis and the horizontal plane of the body which is defined as the horizontal axis of Y is zero, the upward is positive, the downward is negative, and the definition domain is [-90 ~ 90].

(2) Roll angle: the angle between the X axis and the horizontal plane of the body which is defined as the right side is positive from the vertical surface, the left is negative, and the definition

domain is [-180 ~180].

(3) Yaw angle: the angle between the direction of projection in the horizontal plane of the Y axis and the body surface parameters of line, according to the geographic north to 0 degrees clockwise, counterclockwise direction for positive, negative direction, and the domain is [0 ~ 360].

The attitude of aircraft is represented by Euler angle. The physical meaning is intuitive, simple and explicit. It is a widely used attitude representation method in flight control. In the Euler kinematics, the description of the rigid body of the fixed axis rotation requires only one independent coordinate variable, i.e., the rotation angle. When describing the position of a rigid body rotating at a fixed point, the process of turning a fixed point is decomposed into three independent fixed axis rotations, and three independent coordinate variables are needed. In describing the rotation of an aircraft, the coordinate system can be rotated three times independently around the fixed point, which can coincide with the carrier coordinate system, and the three independent rotation angles are Euler angles. The rotation matrix describing the transformation relationship between the carrier coordinate system and the geographic coordinate system is related to the three-rotation order of the Euler angle, and can be obtained by the sequential product of the three rotation matrices.

Define Euler angle as $\eta=[\phi,\theta,\psi]^T$. The transformation from Earth-frame to Body-frame is often described in accordance with the rotation sequence of Z-Y-X. The angles are in turn ψ -θ -ϕ. Thus, the rotation matrix can be obtained by the order product of the three coordinate rotation matrices.

The angle of rotation around the Z axis is defined as the angle ψ,

$$C_1 = R(z,\psi) = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2-1}$$

The angle of rotation around the Y axis is defined as the angle θ,

$$C_2 = R(y,\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \tag{2-2}$$

The angle of rotation around the X axis is defined as the angle ϕ,

$$C_3 = R(x,\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \tag{2-3}$$

Therefore, the transformation relations from the geographic coordinate system to the carrier coordinate system can be represented by the order product of the 3 direction cosine matrices:

$$C_E^B = C_3 C_2 C_1 \tag{2-4}$$

Similarly, the transformation relations from the carrier coordinate system to the geographic coordinate system can be represented by the matrices:

$$C_B^E = (C_E^B)^T = C_1^T C_2^T C_3^T \tag{2-5}$$

Let R denote the transformation relation from carrier coordinates to geographical coordinates, $R(\phi,\theta,\psi)$ is a rotation matrix,

$$R = C_B^E = \begin{bmatrix} \cos\theta\cos\varphi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \sin\psi\cos\theta & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\sin\psi \\ 0 & \cos\theta\sin\phi & \cos\phi\cos\theta \end{bmatrix} \tag{2-6}$$

The rotation matrix is an orthogonal matrix, and the attitude angle can be calculated in the case of the known rotation matrix:

$$\phi = \arctan\frac{C_{32}}{C_{33}}, \theta = -\arcsin C_{31}, \psi = \arctan\frac{C_{21}}{C_{11}} \tag{2-7}$$

2.3 Dynamic Modeling of a Quadrotor Helicopter

2.3.1 Linear Motion Model

The lift force generated in the quadrotors is F_i $(i=1,2,3,4)$, and the force acting on the UAV in the airframe coordinate system is:

$$F_B = [0,0,U_1]^T \tag{2-8}$$

where

$$U_1 = F_1 + F_2 + F_3 + F_4 \tag{2-9}$$

According to the transformation matrix $R(\phi,\theta,\psi)$, the force acting on the UAV in the ground coordinate system can be calculated:

$$F_E = [F_{E1}, F_{E2}, F_{E3}]^T = R(\phi,\theta,\psi)F_B = U_1 \begin{bmatrix} \sin\phi\sin\psi + \cos\phi\cos\psi\sin\theta \\ \cos\phi\sin\theta\sin\psi - \cos\psi\sin\phi \\ \cos\phi\cos\theta \end{bmatrix} \tag{2-10}$$

By Newton's second law:

$$\begin{cases} m\ddot{X} = F_{E1} - K_1\dot{X} \\ m\ddot{Y} = F_{E2} - K_1\dot{Y} \\ m\ddot{Z} = F_{E3} - K_2\dot{Z} - mg \end{cases} \tag{2-11}$$

Combining the equations (11) and (12), the line motion model of the quadrotor UAV in the earth coordinates can be obtained:

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \\ \ddot{Z} \end{bmatrix} = \frac{1}{m} \begin{bmatrix} U_1(\sin\phi\sin\psi + \cos\phi\cos\psi\sin\theta) - K_1\dot{X} \\ U_1(\cos\phi\sin\theta\sin\psi - \cos\psi\sin\phi) - K_1\dot{Y} \\ U_1(\cos\phi\cos\theta) - K_2\dot{Z} - mg \end{bmatrix} \tag{2-12}$$

K_i is the coefficient of air resistance, where K_1 is the coefficient of air drag in the horizontal direction, K_2 is the coefficient of air drag in the vertical direction.

The angular velocity component on each axis of the body coordinate is

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = R(x,\phi)R(y,\theta) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} + R(x,\phi) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{2-13}$$

This equation can be transformed as,

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \tan\theta\sin\phi & \tan\theta\cos\phi \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2-14}$$

In the model simulation and the actual flight control, the attitude angle on the horizontal axis is controlled in a very small range. In the model, it can be approximated to 0, therefore，

$$\begin{aligned} &\phi \approx 0 \Rightarrow \sin\phi \approx 0, \tan\phi \approx 0, \cos\phi \approx 1 \\ &\theta \approx 0 \Rightarrow \sin\theta \approx 0, \tan\theta \approx 0, \cos\theta \approx 1 \end{aligned} \tag{2-15}$$

Then the equation (2-15) can be further simplified as:

$$[\dot{\phi}, \dot{\theta}, \dot{\psi}]^T = [p, q, r]^T \tag{2-16}$$

2.3.2 Angular Motion Model

The quadrotor UAV has a good symmetry structure. Assuming that the center of mass is located in the center of the body, its inertia product is 0. Around x, y, z axis rotation inertia is not 0, then the body of the inertia matrix is,

$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \tag{2-17}$$

With the Euler equation of rigid body motion, angular motion equation can be expressed as UAV,

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times (I \begin{bmatrix} p \\ q \\ r \end{bmatrix}) = \begin{bmatrix} I_x \dot{p} + (I_z - I_y) qr \\ I_y \dot{q} + (I_x - I_z) pr \\ I_z \dot{r} + (I_y - I_x) pq \end{bmatrix} \tag{2-18}$$

M_x, M_y and M_z represent the moment component of the rigid body on three axes respectively. Equation (19) can be re-written as,

$$\begin{cases} \dot{p} = \left[M_x + qr(I_y - I_z) \right] / I_x \\ \dot{q} = \left[M_y + pr(I_z - I_x) \right] / I_y \\ \dot{r} = \left[M_z + pq(I_x - I_y) \right] / I_z \end{cases} \tag{2-19}$$

Set the lift generated by each rotor is respectively F_1, F_2, F_3 *and* F_4, the control amount generated by the four lifting forces in the roll angle is F_4-F_2, the control amount generated by the four lifting forces in the pitch angle is F_3-F_1. The amount of control on the yaw angle is τ_2+τ_4-τ_1-τ_3. 1, and τ_1, τ_2, τ_3, τ_4 are torque generated by each motor respectively. So the moment of rotation is,

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} L(F_4 - F_2) \\ L(F_3 - F_1) \\ \left(\tau_2 + \tau_4 - \tau_1 - \tau_3 \right) \end{bmatrix} \tag{2-20}$$

The angular motion equation of the quadrotor can be calculated as,

$$\begin{cases} \ddot{\varphi} = \left[L(F_4 - F_2) + \dot{\theta}\dot{\psi}(I_y - I_z) \right] / I_y \\ \ddot{\theta} = \left[L(F_3 - F_1) + \dot{\varphi}\dot{\psi}(I_z - I_{\mathrm{x}}) \right] / I_y \\ \ddot{\psi} = \left[(\tau_2 + \tau_4 - \tau_1 - \tau_3) + \dot{\varphi}\dot{\theta}(I_x - I_y) \right] / I_z \end{cases} \tag{2-21}$$

The quadrotor is a system with 4 inputs and 6 outputs, now it has summed up its input and output model. But the coupling of 4 inputs to 6 outputs is too strong, so, we set intermediate inputs

as:

$$\begin{cases} U_1 = F_1 + F_2 + F_3 + F_4 \\ U_2 = F_4 - F_2 \\ U_3 = F_3 - F_1 \\ U_4 = \tau_2 + \tau_4 - \tau_1 - \tau_3 \end{cases} \tag{2-22}$$

Using formula (2-13), formula (2-22) and formula (2-23), the nonlinear dynamic model of the quadrotor can be obtained as:

$$\begin{aligned} \ddot{X} &= \left[U_1(\cos\varphi\sin\theta\cos\psi + \sin\varphi\sin\psi) - K_1\dot{X}\right]/m \\ \ddot{Y} &= \left[U_1(\sin\psi\sin\theta\cos\varphi - \cos\psi\sin\varphi) - K_2\dot{Y}\right]/m \\ \ddot{Z} &= \left[U_1(\cos\varphi\cos\theta) - mg - K_3\dot{Z}\right]/m \\ \ddot{\varphi} &= \left[U_2L + \dot{\theta}\dot{\psi}(I_y - I_z)\right]/I_x \\ \ddot{\theta} &= \left[U_3L + \dot{\varphi}\dot{\psi}(I_z - I_x)\right]/I_y \\ \ddot{\psi} &= \left[U_4 + \dot{\varphi}\dot{\theta}(I_y - I_z)\right]/I_z \end{aligned} \tag{2-23}$$

The dynamic equation (2-23) shows that the quadrotor UAV flight dynamics model has the following important characteristics，

(1) Underactuated: The actual input of the quadrotor is the four force generated by the quadrotors (F_1, F_2, F_3 , F_4), while the output is in six states (x, y, z, ϕ, θ, ψ), so the quadrotor UAV is a typical underactuated system.

(2) Nonlinear: It is obvious that the equation (2-23) satisfies neither the homogeneous theorem nor the superposition theorem, so the system is also a nonlinear system.

(3) Uncertainty: Due to the influence of various factors, the uncertainty of parameters and models will inevitably affect the establishment of the quadrotor unmanned maneuver model. The uncertainty of model controller will be discussed later.

(4) Strong coupling: The position model of formula (2-23) shows that the change of attitude angle (ϕ, θ, ψ) has a direct impact on the position (x, y, z) changes, so there is coupling in the position model, which will affect the effect of the controller.

In this thesis, the quadrotor UAV flight test is completed indoors, the experimental space and flight speed is relatively small. At the same time attitude control in a small range of flight, the change rate is 0. Due to the relatively stable air flow in the room, the air resistance can also be ignored. The above model can be simplified as:

$$
\begin{aligned}
\ddot{X} &= \left[U_1(\cos\varphi\sin\theta\cos\psi+\sin\varphi\sin\psi)\right]/m \\
\ddot{Y} &= \left[U_1(\sin\psi\sin\theta\cos\varphi-\cos\psi\sin\varphi)\right]/m \\
\ddot{Z} &= \left[U_1(\cos\varphi\cos\theta)-mg\right]/m \\
\ddot{\varphi} &= U_2 L/I_x \\
\ddot{\theta} &= U_3 L/I_y \\
\ddot{\psi} &= U_4/I_z
\end{aligned}
\tag{2-24}
$$

Equation (2-24) is the basis of the UAV dynamic model in this thesis, and the model is used in the simulation experiment as the dynamic model of the quadrotor UAV.

2.4 Summary

In this chapter, the structure of the quadrotor UAV with cross symmetry is introduced firstly, and the flight principle of its vertical takeoff and landing is analyzed. With the introduction of the earth coordinate system and body coordinate, and the conversion relationship between these systems is presented. By using the Newton-Euler laws, the force and moment of the UAV are analyzed, and the linear motion equations and angular motion equations of the quadrotor UAV are derived by proper simplification. At last, the overall state equation of the UAV is constructed by combining the two kinds of equations. On this basis, the dynamic model of the quadrotor's line motion and angular motion is established. Some characteristics of the model are analyzed, and the model is presented according to the characteristics of the research environment.

Chapter 3

Qball-X4 UAV System and MATLAB/Simulink Control Simulation Platform

The unmanned quadrotor helicopter, Qball-X4, which is available at the Networked Autonomous Vehicles Lab (NAVL) of Concordia University, will be investigated in this thesis. The Quanser Qball-X4 is a quadrotor UAV experimental platform developed by the Quanser company in Canada. Based on this quadrotor UAV platform, many kinds of flight control studies can be done, such as motion planning, remote obstacle control, multi-information fusion and formation control. In this thesis, the control algorithms as well as the simulation and tests concerning the control of the trajectory tracking and payload dropping of quadrotor UAV are investigated. The overall system hardware and simulation and experimental software will be described in this chapter. Firstly, the main components of the platform will be introduced. Then, the DC motors and geometry of Qball-X4 UAV are described. At last, the MATLAB/Simulink system, as well as the Simulink/QUARC system will all be presented.

3.1 Introduction to Quanser Unmanned Experimental System

3.1.1 Qball-X4 System Configuration

The quadrotor UAV available at the Network Autonomous Vehicle Lab (NAVL) in the Department of Mechanical and Industrial Engineering of Concordia University is the Qball-X4 as shown in Figure 3.1, which was developed by Quanser Inc. partially under the financial support of NSERC (Natural Sciences and Engineering Research Council of Canada) in association with an NSERC Strategic Project Grant led by Concordia University since 2007. The quadrotor UAV is enclosed within a protective carbon fiber ball-shape cage (therefore a name of Qball-X4) to ensure

safe operation. It uses four 10*4.7 inch propellers and standard RC motors and speed controllers.

The Qball-X4's proprietary design ensures safe operation and opens the possibilities for a variety of novel applications. The protective cage is a crucial feature since this unmanned aerial vehicle was designed mainly for use in an indoor environment/laboratory, where there are typically many close range hazards (including other vehicles) and personnel doing flight tests with the Qball-X4. To obtain the measurement from onboard sensors and to drive the motors connected to the four propellers, the Qball-X4 utilizes Quanser's onboard avionics Data Acquisition Card (DAQ), the HiQ, and the embedded Gumstix single-board microcomputer. The HiQ DAQ is a high-resolution Inertial Measurement Unit (IMU) and avionics Input/Output (I/O) card designed to accommodate a wide variety of applications. QuaRC, Quanser's real-time control software, allows researchers and developers to rapidly develop and test controllers on actual hardware through a MATLAB/Simulink interface. The open-architecture of QuaRC and the extensive Simulink blocksets provide users with powerful control development tools. QuaRC can target the Gumstix embedded computer automatically to generate code and execute controllers on board the vehicle. During flights, while the controller is operating on the Gumstix, users can tune parameters in real time and observe sensor measurements from a ground station host (PC or laptop).

Figure 3.1 The Qball-X4 quadrotor UAV

The user interface to the Qball-X4 is MATLAB/Simulink with QuaRC. The controllers are developed in Simulink. Next, the simulated control system comprised of model(s) and controller(s) is compiled into an executable code and downloaded on the onboard micro-computer (Gumstix)

seamlessly. The communication hierarchy and the Qball-X4 system configuration are shown in following Figure 3.2.

For Qball-X4, the following hardware and software are embedded:

- Qball-X4: as shown in Figure 3.1;
- HiQ: QuaRC aerial vehicle Data Acquisition Card (DAQ);
- Gumstix: The QuaRC target single-board micro-computer.
- An embedded, Linux-based system with QuaRC runtime software installed;
- Batteries: Two 3-cell, 2650 mAn Lithium-Polymer batteries;
- Real-Time Control Software: QuaRC, a specially designed built-in blockset that was designed to integrate with the MATLAB/Simulink for real-time controller(s) implementation.

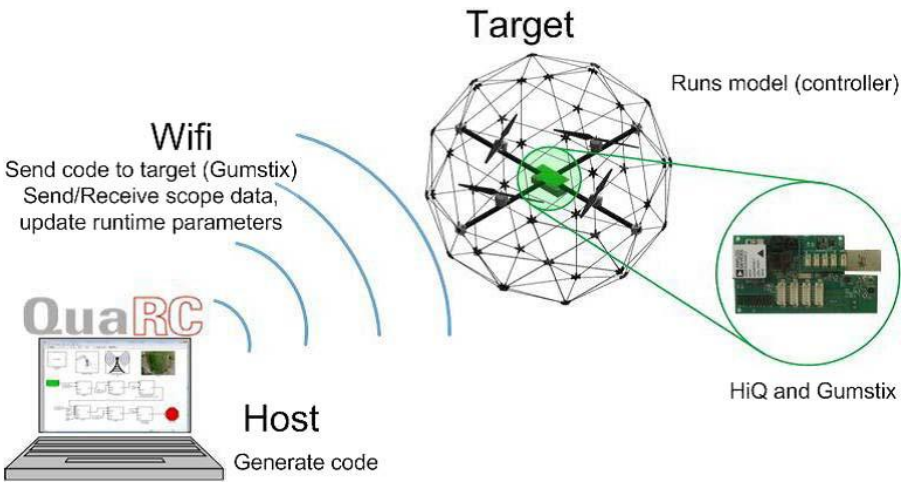


Figure 3.2 Qball-X4 communication hierarchy and communication diagram

3.1.2 Qball-X4 Quadrotor

There are four 10 inch propellers, standard motor and speed controller, quipped with Quanser embedded control module (QECM). This module consists of a Quanser HiQ data acquisition card and a Gumstix embedded single chip computer with Quanser QUARC real-time control software, as shown in Figure 3.2.

HiQ can not only collect sensor signals of high precision accelerometers, gyroscopes and magnetometer inertial measurement components, but also collect servo output signals that drive four motors. The main components and specifications are shown in Table 3-1.

Table 3-1 Qball-X4 components

Components	Specification parameter	Number
Protection cover	Spherical carbon fiber, diameter 0.7m	1
Battery	2500mA lithium battery, Charging voltage 10.6V~12.0V	2
Electric motor	12 pole brushless DC motor	4
Propeller	APC material, diameter 10 inch, pitch 4.7inch	4
Sonar	Effective detection range 20~645cm	1
Electron speed regulator	Maximum output speed 35000rpm	4
Joystick	B type	1
Flight control board	Gumstix VerdexPro, CPU: MarvellPXA270, dominant frequency: 600MHz, Memory: 128M, Integrated accelerometer, gyroscope, servo motor PWM output, Sonar sensor input, GPS serial entry, Signal receiver and Wireless communication module, size 80mmX20mmX5.3mm	1
Mass	1.4Kg	1

Gumstix could run the QUARC, and QUARC supports the real-time control of the Qball-X4 developed by the MATLAB/Simulink environment. In the process of controlling UAV flying, the controller supports sensor measurement, data recording and parameter tuning in operation, and supports direct data transfer between the PC host and the UAV. It can achieve the purpose of real-time control.

3.1.3 OptiTrack Motion Tracking System for Localization

A set of six or more V100:R2 cameras which offers integrated image capture, processing, and motion tracking in a compact package constitute the OptiTrack's optical motion tracking system. The capability of customizing cameras with user-changeable M12 lenses, and OptiTrack's exclusive Filter Switcher technology has let V100 cameras deliver one of the world's premier optical tracking value propositions.

Each V100:R2 camera is capable of capturing fast moving objects with its global shutter imager at 100 FPS capture speed. By maximizing its 640*480 Video Graphic Array (VGA) resolution through advanced image processing algorithms, the V100:R2 can also track markers down to submillimeter movements with maintainable accuracy [30].

A variety of V100:R2 settings are customized with any of OptiTrack's software applications such as the one employed in this study, i.e. Tracking Tool, for greater control over what cameras

capture and what information they report to the personal computer set up as the ground station. Available settings include: image processing type, frame rate, exposure, threshold, illumination, filter switching, and status LED control. OptiTrack's application software, named Tracking Tool, interfaces with MATLAB via specific blocks inside MATLAB/Simulink within the library of QuaRC.

Figure 3.3 illustrates one of the six cameras employed constituting the system of OptiTrack.

Figure 3.3 The OptiTrack System Camera

3.1.4 The Payload Releasing Mechanism

For the purpose of payload drop, a servo motor is used in a simple configuration and is installed under the quadrotor battery bay, as shown in Figure 3.4. The PWM signal generated by Gumstix computer controls the servo motor to turn the rod attached to servo horn. On the other hand, the payload, an empty battery weighing 300 g, is hooked to the above-mentioned rod and can be dropped at the desired time.

The satisfaction on fine tuning of controller gains as well as healthy and fully charged Li-Po batteries are two essential factors affecting system's behavior. For instance, not fully charged battery packs can have a direct effect on the performance of fine-tuned controller gains and deviate them from the desired values.

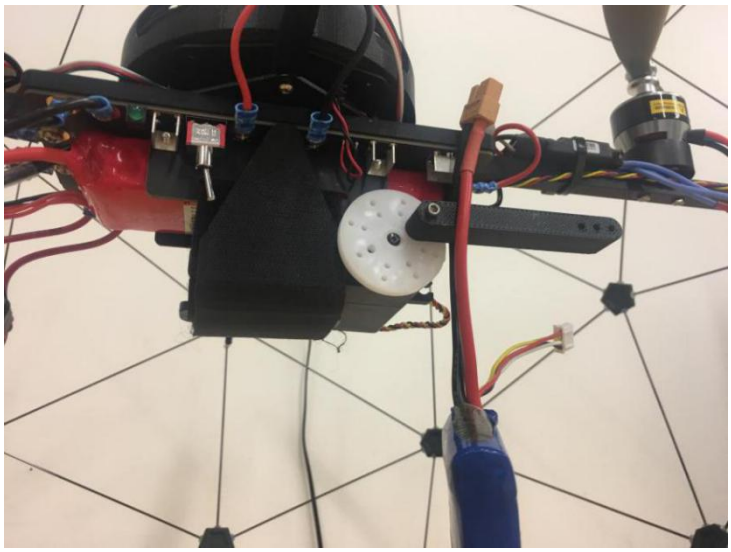


Figure 3.4 Servo-based payload releasing mechanism

3.2 Qball-X4 DC Motors and Geometry

3.2.1 Electronic Speed Controller, DC Motor, and Propeller of Qball-X4

The motors of the quadrotor UAV are usually out-runner brushless motors. The generated thrust T_i of the *No.i* motor is related to the *No.i* PWM input u_i by a first-order linear transfer function as follows:

$$T_i = k\frac{\omega}{s+\omega}u_i \tag{3-1}$$

where $i = 1, 2, 3, 4$ and k is a positive gain and ω is the motor bandwidth. k and ω are theoretically the same for the four motors but this may not be the case in practice. It should be noted that $u_i=0$ corresponds to zero thrust and $u_i=0.05ms$ corresponds to the maximal thrust that can be generated by the *No. i* motor for Qball-X4 UAV which will be investigated (Qball-X4 quadrotor helicopter will be described in detail at the following chapter).

The block diagram of the UAV system is illustrated in Figure 2.4. It is composed of three main parts. The first part represents the Electronic Speed Controllers (ESCs), the motors, and the propellers in a set of four. The input to this part is $u = \begin{bmatrix} u_1 & u_2 & u_3 & u_4 \end{bmatrix}^T$ which are Pulse Width Modulation (PWM) signals. The output is the thrust vector $T = \begin{bmatrix} T_1 & T_2 & T_3 & T_4 \end{bmatrix}^T$ generated by four individually-controlled motor-driven propellers. The second part is the geometry that relates the generated thrusts to the applied lift and torques to the system. This geometry corresponds to the

position and orientation of the propellers with respect to the center of mass of the Qball-X4. The third part is the dynamics that relate the applied lift and torques to the position (P), velocity (V) and acceleration (A) of the Qball-X4.

Figure 3.5 shows the Qball-X4 UAV system block diagram which will be investigated in this thesis.

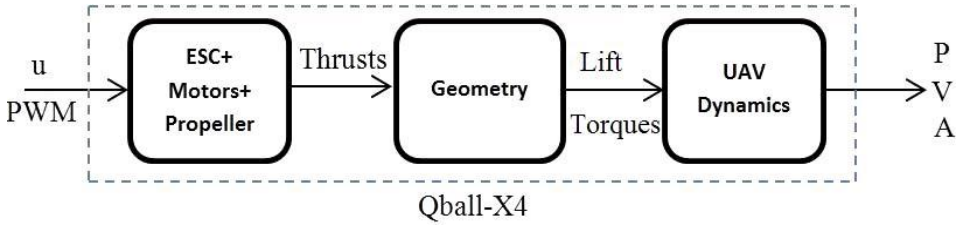


Figure 3.5 The UAV system block diagram

3.2.2 Geometry

A schematic representation of the Qball-X4 is given in Figure 3.6. The motors and propellers are configured in such a way that the back and front (1 and 2) motors spin clockwise (thus inducing two counterclockwise torques on the body) and the left and right (3 and 4) spin counterclockwise (thus inducing two clockwise torques on the body). Each motor is located at a distance L from the center of mass o (0.2 m) and when spinning, a motor produces a torque τ_i which is in the opposite direction of motion of the motor as shown in Figure 3.6.

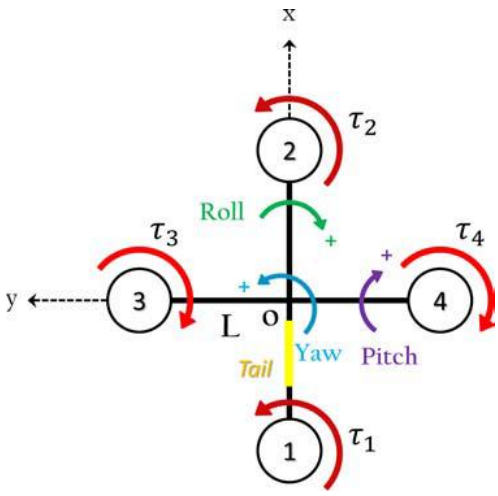


Figure 3.6 Schematic representation of the Qball-X4

The origin of the body-fixed frame is the system's center of mass o with the x-axis pointing from back to front and the y-axis pointing from right to left. The thrust T_i generated by the i_{th} propeller is always pointing upward in the z-direction in parallel to the motor's rotation axis. The thrusts T_i and the torques τ_i result in a lift in the z-direction (body-fixed frame) and torques about the x-, y- and z-axis.

The relations between the lift/torques and the thrusts are

$$\begin{aligned} u_z &= T_1 + T_2 + T_3 + T_4 \\ u_\theta &= L(T_1 - T_2) \\ u_\phi &= L(T_3 - T_4) \\ u_\psi &= \tau_1 + \tau_2 + \tau_3 + \tau_4 \end{aligned} \tag{3-2}$$

The torque $\tau_i (i = 1,2,3,4)$ produced by the ith motor is directly related to the thrust T_i via the relation of $\tau_i = K_\psi T_i$ with K_ψ as a constant. In addition, by setting $T_i \approx K u_i$, from equation (3-2), the relation (3-3) reads,

$$\begin{aligned} u_z &= K(u_1 + u_2 + u_3 + u_4) \\ u_\theta &= KL(u_1 - u_2) \\ u_\phi &= KL(u_3 - u_4) \\ u_\psi &= KK_\psi(u_1 + u_2 - u_3 - u_4) \end{aligned} \tag{3-3}$$

where u_z is the total lift generated by the four propellers and applied to the quadrotor UAV in the z-direction (body-fixed frame). u_θ, u_ϕ, u_ψ are the applied torques in θ, ϕ and ψ directions, respectively.

3.3 MATLAB/Simulink Flight Control Simulation Platform

3.3.1 Introduction of Simulation Software

MATLAB is a mathematical software developed by the American MathWorks company, which has powerful numerical computing function. With the MATLAB updating the toolbar add many toolboxes (Toolbox). The function of MATLAB becomes more powerful, and the application field is more and more extensive. In the aspect of system control, it can be used for simulation and analysis, and the image processing toolbox supports the basic image processing type. The system diagram of MATLAB is as Figure 3.7.

MATLAB software has the features as follows:

(1) It has powerful numerical calculation and symbolic calculation function, so it greatly reduces the amount of users' mathematical calculation;

(2) It has a powerful graphics processing function, the results of the calculation, programming visualization in the form of expression;

(3) Its user interface is very friendly to users, and the mathematical expressions of naturalized language forms are easy to understand;

(4) It has a lot of application toolbox (such as signal processing toolbox, communication toolbox, etc.), the function has been well developed.

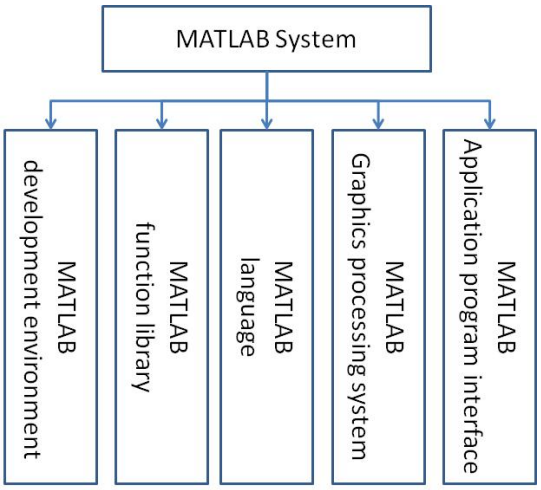


Figure 3.7 The system diagram of MATLAB

Simulink is one of the most important components of MATLAB as shown in Figure 3.8. It is an integrated environment platform. Using this platform, we can build the simulation platform of the established mathematical model, and then carry on the simulation, and use the simulation results to carry on the comprehensive analysis. Simulink uses a modular toolbox, the user does not need to write a large number of programs, only through the mouse operation, click on the required modules, self-constructing simulation system. Simulink has many advantages, and has been widely used in complex simulation and design fields of control theory and digital signal processing.

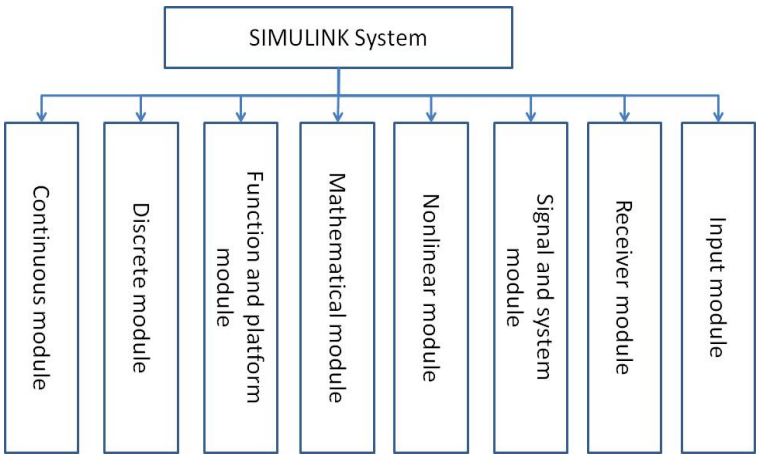


Figure 3.8 The module composition of Simulink

3.3.2 Simulation Platform Building for Quadrotor Helicopter

In this thesis, a simulation module of quadrotor unmanned aerial vehicle (UAV) is built by MATLAB/Simulink, and the system simulation is carried out to verify the design result of Gain Scheduling PID controller and backstepping controller. The module diagram of the simulation system designed in this thesis is as shown in Figure 3.9.

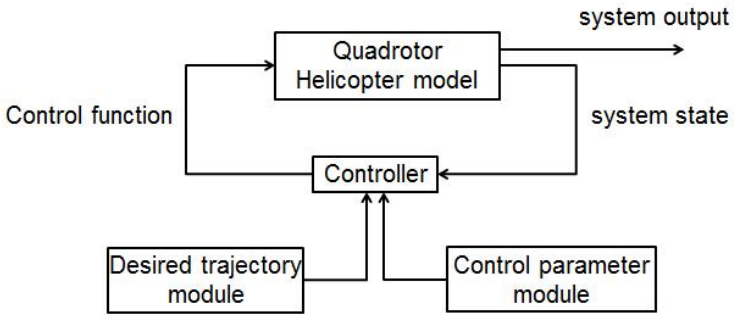


Figure 3.9 Schematic figure of simulation system module

It can be seen from the figure that the input of the controller can be divided into three parts: the system state, the control parameter and the desired trajectory. The quadrotor aircraft model, the controller module, and the desired trajectory module can be written using the S function. The control parameter module is a subsystem composed of a series of constants. In the simulation process, the control parameters can be adjusted so as to achieve the best simulation effect. The parameter of input for the quadrotor helicopter system is $U=[U_1,U_2,U_3,U_4]^T$, which respectively are throttle, roll, pitch and yaw movements. System output is $Y=[Z,\phi,\theta,\psi]^T$, which are respectively z axis speed, pitch angle, roll angle, and yaw angle.

3.4 Summary

In this chapter, Qball-X4 UAV system and MATLAB/Simulink control simulation platform which will be used in this thesis are described. The overall system hardware and simulation and test software are all described. The main components of the platform are introduced. The MATLAB/Simulink system, as well as the Simulink/QUARC system are also mentioned in this chapter.

Chapter 4

Gain Scheduling Based PID Controller of the Quadrotor Helicopter

4.1 Introduction

Flight control, as an important part of unmanned aerial vehicle (UAV) system, plays an important role in improving the performance of UAV. In the design of the control system of the quadrotor aircraft, in addition to the complete control system hardware design and reasonable fuselage structure, the correct software algorithm is the core of stable flight. In order to achieve a stable flight of quadrotor helicopter, we need to carry out researches on the flight control algorithm, and use the flight control algorithm to achieve stable flight control of quadrotor unmanned aerial vehicle. In addition, we need to validate the flight control algorithm with the MATLAB control algorithm simulation.

As indicated in the Chapter 2, the quadrotor UAV has six degrees of freedom in space, but only by adjusting the speed of the four servo motors to achieve its flight control. The system of independent control variables (the system 4) smaller than the number of degree of freedom (the system 6), that is to say the space dimension of the input vector control system which is less than the configuration space dimension, this system is called underactuated system. The UAV system is characterized by a relatively small number of control inputs to determine its motion in a larger space than that of the control input space. The difficulty is that the degree of freedom of the direct excitation part and the degree of freedom of the under actuated part are nonlinear coupling. Because of the characteristics of multi variable, nonlinear, underactuated and strong combination, the design of flying control system becomes relatively difficult.

A GS-PID control strategy approach is used for a multivariable nonlinear unmanned aerial

vehicle, a quadrotor helicopter in this thesis. In this chapter, theory of PID controller and Gain Scheduling based PID controller are explained. The Gain Scheduling based PID controller of the quadrotor UAV is designed and each control system is formulated and modeled in MATLAB/Simulink in this chapter.

4.2 The PID Controller for Quadrotor UAV

4.2.1 The PID Control Approach

A **Proportional-Integral-Derivative controller (PID controller)** is a generic control loop feedback mechanism (controller) widely used in industrial control systems – a PID is the most commonly used feedback controller. A PID controller calculates an "error" value as the difference between a measured process variable and a desired set-point. The controller attempts to minimize the error by adjusting the process control inputs.

The PID controller calculation (algorithm) involves three separate constant parameters, and is accordingly sometimes called three-term control: the proportional, the integral and derivative values, denoted *P, I,* and *D*. Heuristically, these values can be interpreted in terms of time: *P* depends on the present error, *I* on the accumulation of past errors, and *D* is a prediction of future errors, based on current rate of change. The weighted sum of these three actions is used to adjust the process via a control element such as the position of a control valve, or the power supplied to a heating element.

In the absence of knowledge of the underlying process, a PID controller has historically been considered to be the best controller. By tuning the three parameters in the PID controller algorithm, the controller can provide control action designed for specific process requirements. The response of the controller can be described in terms of the responsiveness of the controller to an error, the degree to which the controller overshoots the set-point and the degree of system oscillation. Note that the use of the PID algorithm for control does not guarantee optimal control of the system or system stability.

Some applications may require using only one or two actions to provide the appropriate system control. This is achieved by setting the other parameters to zero. A PID controller will be called a *PI, PD, P* or *I* controller in the absence of the respective control actions. *PI* controllers are fairly

common, since derivative action is sensitive to measurement noise, whereas the absence of an integral term may prevent the system from reaching its target value due to the control action.

The PID control scheme is named after its three correcting terms, whose sum constitutes the manipulated variable (MV). The proportional, integral, and derivative terms are summed to calculate the output of the PID controller. The key idea of PID control algorithm is to eliminate the deviation according to the deviation. The algorithm can predict the future of the control system in the past, the present and the controlled system effectively. Control systems are usually composed of system set values, controlled objects, and PID controllers. The structure of the PID control system is shown in Figure 4.1. According to the figure, the system consists of two parts, the forward channel and the feedback channel. The output of the controlled object *y(t)* is used as the input of the feedback channel. The value and control system for a given value of $y_d(t)$ value of the deviation control system (error), $error(t) = y_d(t) - y(t)$, the deviation by proportional integral differential three after adding *u(t)* as a controlled object input.

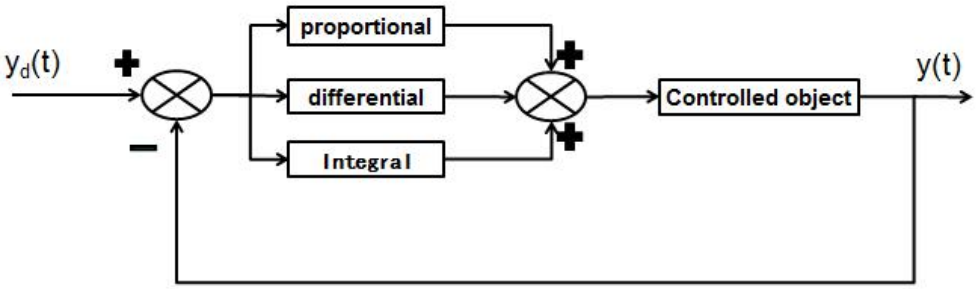


Figure 4.1 Structure of PID control system

Defining *u(t)* as the controller output, the final form of the PID algorithm is:

$$u(t) = MV(t) = K_p error(t) + K_i \int_0^t error(t)dt + K_d \frac{derror(t)}{dt} \quad (4\text{-}1)$$

where,

K_p: Proportional gain, a tuning parameter

K_i: Integral gain, a tuning parameter

K_d: Derivative gain, a tuning parameter

e: Error

t: Time or instantaneous time (the present)

(1) Proportional Term

The proportional term produces an output value that is proportional to the current error value.

The proportional response can be adjusted by multiplying the error by a constant K_p, called the proportional gain.

The proportional term is given by:

$$P_{out} = K_p e(t) \tag{4-2}$$

A high proportional gain results in a large change in the output for a given change in the error. As shown in Figure 4.2, if the proportional gain is too high, the system can become. In contrast, a small gain results in a small output response to a large input error, and a less responsive or less sensitive controller. If the proportional gain is too low, the control action may be too small when responding to system disturbances. Tuning theory and industrial practice indicate that the proportional term should contribute the bulk of the output change.

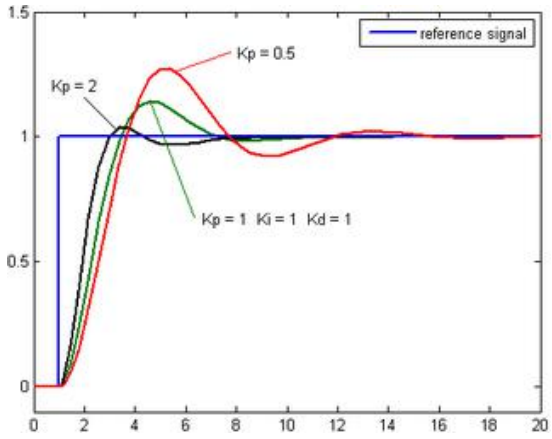


Figure 4.2 Plot of PV vs time, for three values of K_p (K_i and K_d held constant)

Because a non-zero error is required to drive the controller, a pure proportional controller generally operates with a steady-state error, referred to as droop. Droop is proportional to the process gain and inversely proportional to proportional gain. Droop may be mitigated by adding a compensating bias term to the setpoint or output, or corrected by adding an integral term.

(2) Integral Term

The contribution from the integral term is proportional to both the magnitude of the error and the duration of the error. The integral in a PID controller is the sum of the instantaneous error over time and gives the accumulated offset that should have been corrected previously. The accumulated error is then multiplied by the integral gain and added to the controller output.

The integral term is given by:

$$I_{out} = K_i \int_0^t e(\tau) d\tau \tag{4-3}$$

The integral term accelerates the movement of the process towards setpoint and eliminates the residual steady-state error that occurs with a pure proportional controller. However, since the integral term responds to accumulated errors from the past, it can cause the present value to overshoot the setpoint value, as shown in Figure 4.3.

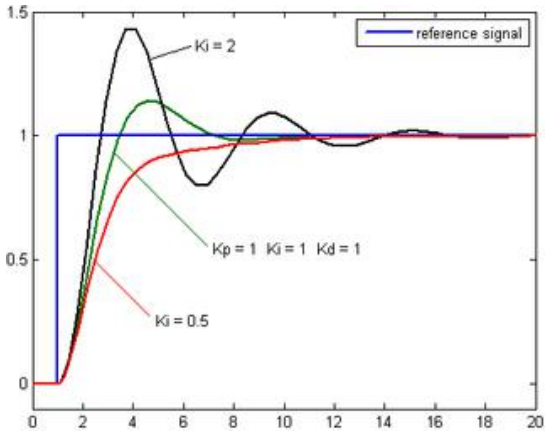


Figure 4.3 Plot of PV vs time, for three values of K_i (K_p and K_d held constant)

(3) Derivative Term

The derivative of the process error is calculated by determining the slope of the error over time and multiplying this rate of change by the derivative gain K_d. The magnitude of the contribution of the derivative term to the overall control action is termed the derivative gain, K_d.

The derivative term is given by:

$$D_{out} = K_d \frac{d}{dt} e(t) \tag{4-4}$$

The derivative term slows the rate of change of the controller output. Derivative control is used to reduce the magnitude of the overshoot produced by the integral component and improve the combined controller-process stability. However, the derivative term slows the transient response of the controller as shown in Figure 4.4.

Also, differentiation of a signal amplifies noise and thus this term in the controller is highly sensitive to noise in the error term, and can cause a process to become unstable if the noise and the derivative gain are sufficiently large. Hence an approximation to a differentiator with a limited bandwidth is more commonly used. Such a circuit is known as a phase-lead compensator.

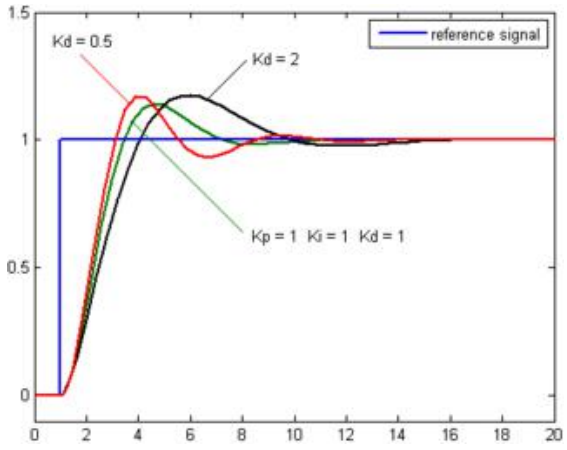


Figure 4.4 Plot of PV vs time, for three values of K_d (K_p and K_i held constant)

(4) Ideal Versus Standard PID Form

The form of the PID controller most often encountered in industry, and the one most relevant to tuning algorithms is the standard form. In this form the K_p gain is applied to the I_{out}, and D_{out} terms, yielding:

$$MV(t) = K_p\left[e(t) + \frac{1}{T_i}\int_0^t e(t)dt + T_d\frac{d}{dt}e(t)\right] \tag{4-5}$$

where T_i is the *integral time*, and T_d is the *derivative time.*

In this standard form, the parameters have a clear physical meaning. In particular, the inner summation produces a new single error value which is compensated for future and past errors. The addition of the proportional and derivative components effectively predicts the error value at T_d seconds (or samples) in the future, assuming that the loop control remains unchanged. The integral component adjusts the error value to compensate for the sum of all past errors, with the intention of completely eliminating them in T_i seconds (or samples). The resulting compensated single error value is scaled by the single gain K_p.

In the ideal parallel form, shown in the controller theory section

$$MV(t) = K_p e(t) + K_i\int_0^t e(t)dt + K_d\frac{d}{dt}e(t) \tag{4-6}$$

The gain parameters are related to the parameters of the standard form through $K_i = \frac{K_p}{T_i}$ and $K_d = K_p T_d$. This parallel form, where the parameters are treated as simple gains, is the most general and flexible form. However, it is also the form where the parameters have the least physical interpretation and is generally reserved for theoretical treatment of the PID controller. The standard

form, despite being slightly more complex mathematically, is more common in industry.

4.2.2 Altitude Control of Quadrotor UAV

After the mathematical model of the quadrotor is verified, a PID controller was developed. The PID controller generates the desired control inputs for the quadrotor. The block diagram for a PID controller is shown in Figure 3.1.

As indicated above, to achieve stable operation control or autonomous control of quadrotor aircraft, position control is the key, and attitude control is the basis for achieving stability of aircraft. For the position control of quadrotor aircraft, four control variables, U_1, U_2, U_3, U_4 is introduced in the dynamic model. The complex nonlinear coupling model is decoupled and decomposed into four independent control channels. The whole system can be considered as two independent subsystems of line motion and angular motion. The line motion belongs to the position control of the aircraft. Angular motion belongs to the attitude control of the aircraft. The angular motion is not affected by the line motion, while the line motion is affected by the angular motion, that is, the position control needs to be controlled through the attitude control.

A PID controller is developed to control the altitude of the quadrotor. It generates the control input U1 which is responsible for the altitude for the quadrotor as per Equation (2.10). The derived control law is as follows:

$$U_1 = k_p(z - z_d) + k_d(\dot{z} - \dot{z}_d) + k_i \int (z - z_d)dt \tag{4-7}$$

where k_p is proportional gain, z_d is desired altitude, k_d is derivative gain, $\dot{z}_d$ is desired altitude rate of change, k_i is integral gain.

4.2.3 Attitude Control of Quadrotor UAV

(1) Roll Controller

Another PID controller is developed to control the roll angle ϕ of the quadrotor. The derived control law generates the input U2 that controls the roll angle as follows:

$$U_2 = k_p(\phi_d - \phi) + k_d(\dot{\phi}_d - \dot{\phi}) + k_i \int (\phi_d - \phi)dt \tag{4-8}$$

where k_p is proportional gain, ϕ_d is desired roll angle, k_d is derivative gain, $\dot{\phi}_d$ is desired roll

angle of change, k_i is integral gain.

(2) Pitch Controller

A PID controller is developed to control the pitch angle θ of the quadrotor. The derived control law generates the input U3 that controls the pitch angle as follows,

$$U_3 = k_p(\theta_d - \theta) + k_d(\dot{\theta}_d - \dot{\theta}) + k_i\int(\theta_d - \theta)dt \tag{4-9}$$

where k_p is proportional gain, θ_d is desired pitch angle, k_d is derivative gain, $\dot{\theta}_d$ is desired pitch angle rate of change, k_i is integral gain.

(3) Yaw Controller

Similar to the pitch and roll controllers, a yaw controller was developed to generate the control input U4 based on the following control law,

$$U_4 = k_p(\psi_d - \psi) + k_d(\dot{\psi}_d - \dot{\psi}) + k_i\int(\psi_d - \psi)dt \tag{4-10}$$

where k_p is proportional gain, ψ_d is desired pitch angle, k_d is derivative gain, $\dot{\psi}_d$ is desired pitch angle rate of change, k_i is integral gain.

4.2.4 Position Controller of Quadrotor UAV

After acquiring stable controllers for the altitude and the attitude of the quadrotor, a complete position controller is developed. PID controllers are used to calculate the desired accelerations $\ddot{x}_d$ and $\ddot{y}_d$

$$\ddot{x}_d = k_p(x_d - x) + k_d(\dot{x}_d - \dot{x}) + k_i\int(x_d - x)dt \tag{4-11}$$

$$\ddot{y}_d = k_p(y_d - y) + k_d(\dot{y}_d - \dot{y}) + k_i\int(y_d - y)dt \tag{4-12}$$

where k_p is proportional gain, x_d is desired x position, k_d is derivative gain, $\dot{x}_d$ is desired x position rate of change, y_d is desired y position, $\dot{y}_d$ is desired y position rate of change, k_i is integral gain.

Plugging the values of the desired accelerations $\ddot{x}_d$ and $\ddot{y}_d$ into Equations (2-23) and (2-24), the desired roll and pitch angles ϕ_d and $\dot{\theta}_d$ can be calculated which are in turn fed to the attitude

controller previously expressed in Equations (4-8) and (4-9).

4.3 Gain Scheduling Controller for Quadrotor UAV

4.3.1 Introduction to GS-PID Control

While PID controllers are applicable to many control problems, and often perform satisfactorily without any improvements or even tuning, they can perform poorly in some applications, and do not in general provide optimal control. The fundamental difficulty with PID control is that it is a feedback system, with constant parameters, and no direct knowledge of the process, and thus overall performance is reactive and a compromise, while PID control is the best controller with no model of the process, better performance can be obtained by incorporating a model of the process.

The most significant improvement is to incorporate feed-forward control with knowledge about the system, and using the PID only to control error. Alternatively, PIDs can be modified in more minor ways, such as by changing the parameters (either gain scheduling in different use cases or adaptively modifying them based on performance), improving measurement (higher sampling rate, precision, and accuracy, and low-pass filtering if necessary), or cascading multiple PID controllers.

PID controllers, when used alone, can give poor performance when the PID loop gains must be reduced so that the control system does not overshoot, oscillate or *hunt* about the control setpoint value. They also have difficulties in the presence of nonlinearities, may trade-off regulation versus response time, do not react to changing process behavior (say, the process changes after it has warmed up), and have lag in responding to large disturbances.

To overcome the shortcomings of the linear PID controller in its ability to only operate in the linear near hover region, a Gain Scheduling based PID controller is proposed for the control of quadrotor UAV. The theory behind Gain Scheduling is developing a set of controllers for different operating points and switching between these controllers depending on the operating point of the system. In this work, a family of PID controllers will be developed, each PID controller having different controller gains and will be able to stabilize the quadrotor system in a certain range of operation. Gain Scheduling will then be used to choose an appropriate controller from the family of developed PID controllers. This approach renders the classical PID controller an adaptive controller since the controller's parameters are adapting to different operating conditions. The acquired gains

were used in a look up table fashion in the developed MATLAB/Simulink model.

4.3.2 Gain Scheduling Control Approach

Although one single PID controller can handle a wide range of system nonlinearities, better performance can be obtained when using multiple PIDs to cover the entire operation range of a non-linear process/system. This is known as gain-scheduled PID (GS-PID). The operating principle of GS-PID is shown in Fig 4.5 (a) where the controlled system may have varying dynamic properties as for example a varying gain. The adjustment/scheduling of the PID controller gains is performed using a gain scheduling variable GS. This is some measured process variable which at every instant of time expresses or represents the dynamic properties of the process.

The basic idea of gain scheduling is: First of all, choose to cover the operating point of the flight envelope, and get the work point linear model by small perturbation linearization. Then, using the linear control theory to design the corresponding controller which satisfies some flight standards; finally, the controller adopts the method of switching or interpolation quasi global controller synthesis with a variable parameter adjustment, and real-time controller gain to adapt to changes in the aircraft dynamics model.

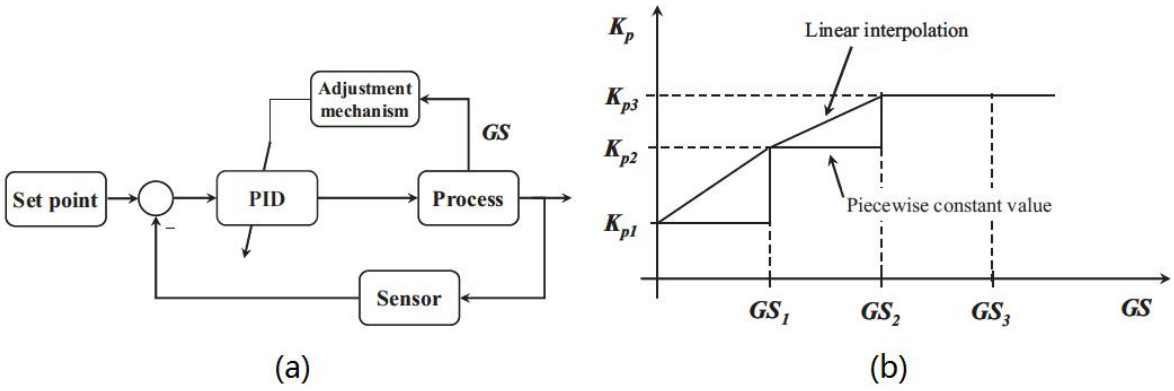


Figure. 4.5 Principle of GS-PID and two rules for the interpolation of the gain Kp. (a) Operating principle of GS-PID. (b) Constant and linear interpolation

It is well known that the Gain-Scheduled PID (GS-PID) is a versatile technique which can be used for situations in which the parameters or the operating conditions of the plant can change largely and rapidly. In aerospace applications, different portions of the flight envelope must be considered in control system design for different flight conditions. Various phases of flight need proper tuning of the controller gains under different flight conditions. In view of popularity of

GS-PID and advantages of such simple and model-free control design strategy, in this thesis, a GS-PID controller that assumes a separate set of gains for each and every phase of flight is designed and implemented on a quad-rotor helicopter UAV. The piecewise constant value is used in this study to optimize the gain scheduling PID tuning process, which reduces the workload of gain scheduling PID controller tuning, and optimizes the fault-tolerant control performance (Figure 3.5(b)). The proposed control law stabilizes the quad-rotor UAV system and drives the system states, including three-dimensional positions and yaw angle to track the desired path, while keep the closed loop system stable.

4.3.3 Gain Scheduling PID Controller Design

Generally, if the change of dynamics in a system/process with the operating condition is known, then it is possible to change the parameters of the controller by monitoring the operating conditions of the process. This approach is called Gain Scheduling because the scheme was originally used to accommodate changes in process gain only. A block diagram of a control system with gain scheduling mechanism is shown in Figure 4.6.

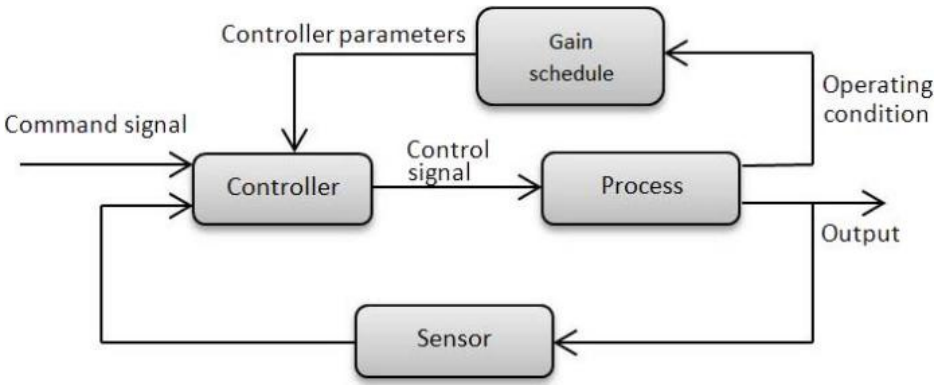


Figure 4.6 Gain scheduling control structure

The idea of relating the controller parameters to auxiliary variables is not new, but the implementation of gain scheduling is still challenging in practice. The adjustment/scheduling of the PID controller gains is performed using a gain scheduling variable GS. This is some measured process variable which at every instant of time expresses or represents the dynamic properties of the process. But it is difficult to give a general rate of control in gain scheduling control. Each fault case or parameters change must be handled separately in gain scheduling control.

There are several ways to express the PID parameters as functions of the GS variable using deterministic and fuzzy rules. The deterministic GS-PID was widely used and was also used in this thesis. In the deterministic rule-based GS-PID, one may parametrize the PID parameters as functions of the GS variable using the piecewise constant controller parameters or the piecewise interpolation. In the former method, an interval is defined around each GS value and the controller parameters are kept constant as long as the GS value is within the interval (see Fig. 4.5(b) for an example of the proportional gain Kp). When the GS variable changes from one interval to another, the controller parameters are changed abruptly. The same idea applies for the integral and derivative gains. In the second method also shown in Fig. 4.5(b), a linear function is found relating the controller parameter (output variable) and the GS variable (input variable). For the proportional gain, the linear function is of the form Kp = aGS + b, where a and b are two constants to be calculated.

As indicated above, a potential application of the GS-PID controller is the controlling of the trajectory tracking with payload dropping and payload carrying of an unmanned quadrotor. The gain scheduling based PID control was used in this thesis by exploiting the advantages for PID controllers and gain scheduling control strategy. In this work, a set of pre-tuned PID controller gains are applied to the controllers for situations in which the parameters or the operating conditions of the UAV can change largely and rapidly (such as payload dropping and payload carrying). The flight condition was used for switching pre-tuned PID controllers for obtaining best performance for each considered operating conditions.

It must be indicated that, in order to obtain the best stability and performance of Qball-X4 under both payload carrying and payload drop conditions, especially in trajectory tracking with payload drop scenario, the switching action from one set of pre-tuned PID gains to another set plays a vital role in performance of the helicopter at the moment of releasing the payload. In other words, if this transient (switching) time is held long (more than one second) it can cause the Qball-X4 to overshoot abruptly and even a crash.

In order to overcome this limitation, the interpolation algorithm is proposed to optimize the gain scheduling. Assume that proper controller gains of the PID controllers have been found for the normal and different situations, these values can be stored in a parameter table which will be picked up based on a gain scheduling variable based on the different fault types and magnitudes. In the

actual operation of the system, the region of the compensator is determined according to the situation. For example, when different faults occur, the PID controllers will be switched to the corresponding fault case, and the parameters of the compensator are determined by interpolation between the operation points.

With gain scheduling table, the PID parameter can be reset by interpolation method according to different fault. Lagrange interpolation formula is used in this study.

$$K = \sum_{i=1}^{N} \left[K \prod_{j=1, i \neq j}^{N} \frac{f - f_j}{f_i - f_j} \right] \tag{4-13}$$

where K is controller gains associated with proportional (P), integral (I), and derivative (D) action respectively. f is the failure rate parameter for specified fault types. The subscript i and j are the specified failure rate in the parameter table. For convenience, the fault estimate value of the total lift of the quadrotors is used as the scheduling variable.

4.4 Summary

Gain scheduling (GS) is one of the most popular approaches to nonlinear control design and it is known that GS controllers have a decent performance in many circumstances. Based on the quadrotor mathematical model established in the previous chapter, and considering the limitations of PID control, a GS-PID controller is proposed in this chapter to control quadrotor flight.

Chapter 5

Trajectory Tracking and Payload Dropping of Qball-X4 with GS-PID Controller

5.1 Introduction

Because PID controller has little dependence on the model, it is relatively reliable, simple and easy being adjusted, it is widely used in industrial and aviation applications. PID controller is currently dominated in the control of the quadrotor UAV. In this chapter, the trajectory tracking simulation and payload dropping simulation of Qball-X4 UAV with GS-PID controller will be presented. The MATLAB/Simulink simulation tool is used to simulate the control algorithm for quadrotor UAV. Some of the basic parameters for Qball-X4 quadrotor UAV model are shown in Table 5-1.

The MATLAB Control System Toolbox provides complete commands and algorithms for solving linear two-order optimal control problems. Normally, if the desired input signal is small, select the larger R matrix, which forces the input signal to be smaller. Otherwise, the objective function will increase and will not achieve the optimization requirements. Step signals are added to the control system, and then the parameters of the PID are adjusted. With repeatedly adjusting the PID parameter, the most desirable PID parameter could be obtained. System output is $Y=[z,\phi,\theta,\psi]^T$, which are respectively z axis speed, pitch angle, roll angle, yaw angle.

The Simulink model for trajectory tracking and payload dropping control of quadrotor helicopter using PID and GS-PID controller is shown in Figure 5.1.

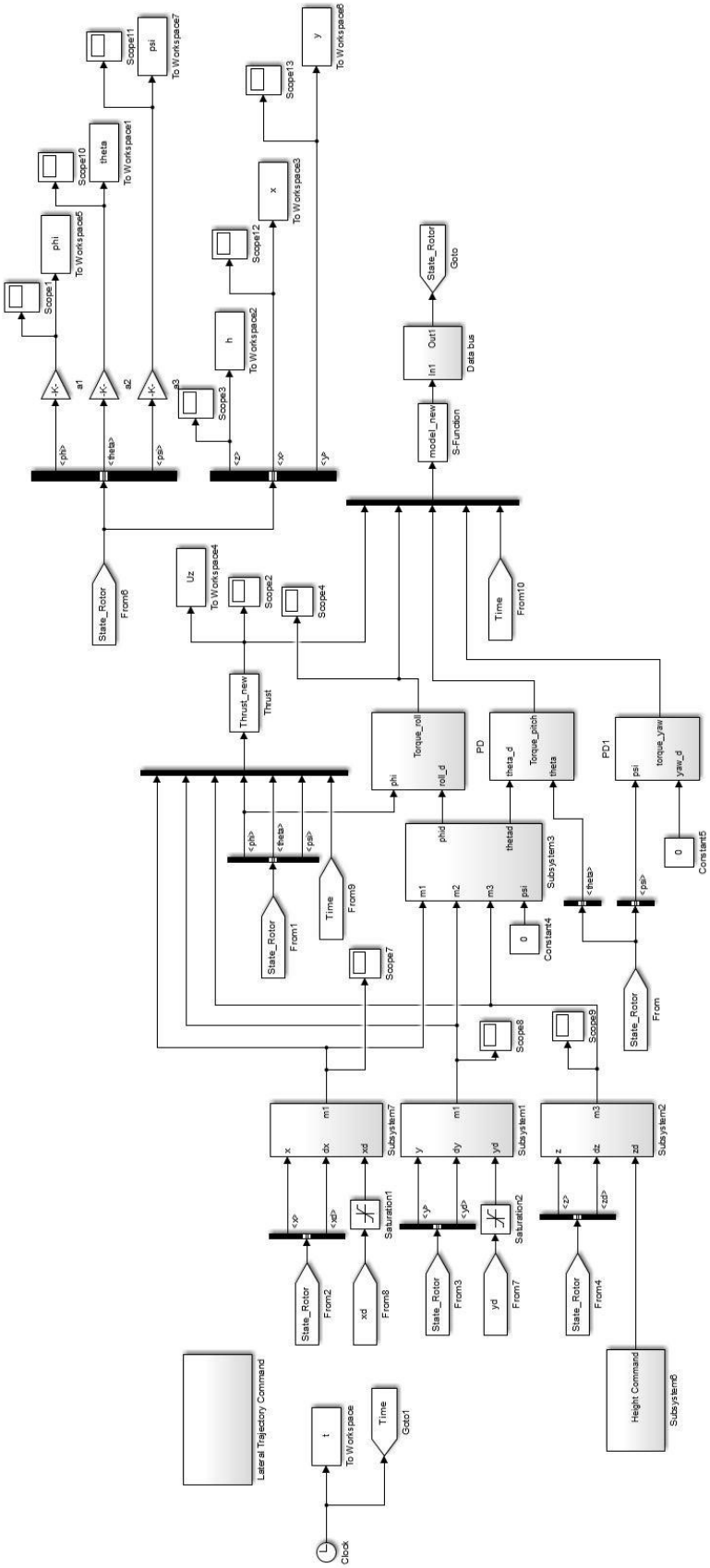


Figure 5.1 GS-PID controller Simulink model

Table 5-1 Physical parameters of the quadrotor UAV Qball-X4

Parameter	Value
K_{motor}	120N
ω	15rad/sec
I_x	0.03kg.m^2
I_y	0.03kg.m^2
I_z	0.04kg.m^2
m	1.4Kg
K'	4N.m
l	0.2m
g	9.8N/kg

5.2 Step Response Analysis

At the beginning, step input was applied to the quadrotor UAV system to demonstrate the effectiveness of the controller. The step response of the system is defined by the characteristics of overshoot, rise time and stability time.

The step response is generated at zero time, the initial value is 0, and the end of the step is 1. By adjusting the PID parameters several times, the best PID parameters can be obtained. The PID controller parameters of the quadrotor unmanned aerial vehicle are shown in Table 5-2. The simulation results of the pitch angle, roll angle, yaw angle and altitude were shown in Figure 5.2 to Figure 5.5. In these figures, the transverse coordinates are time and the longitudinal coordinates are the desired attitude angles. The comparison of the simulation results with PID controller and GS-PID controller was also shown in these figures.

Table 5-2 PID controller parameters of quadrotor UAV

PID parameters	K_P	K_I	K_D
Pitching angle	4.5	0.3	0.15
Pitching speed	0.45	0.01	0
Roll angle	4.5	0.3	0.15
Roll speed	0.45	0.01	0
Yaw angle	3.5	0.2	0.12
Yaw speed	0.35	0.5	0
altitude	35	0.3	0.4

It could be found from the results of Figure 5.2 to Figure 5.5 that the stability, rapidity and accuracy of gain scheduling PID controller are better than simple PID controller. It could be found

from these figures that the quadrotor unmanned aerial vehicle with GS-PID control algorithm, the response time is about 0.2s for the step signal input, and the system basically has no overshoot. The response time of the single PID control algorithm is about 0.2s, but the system has overshoot phenomenon. This result shows that the GS-PID control system can achieve better control effect on the quadrotor UAV platform. Because the motion of the Z axis has no angular velocity, the rapidity and stability of the reaction could not catch up with the control of the other three angles, but it can still be stable in the 0.5s, and the overshoot of the system is within 10%. This fully meets the control needs of the vast majority of the quadrotor UAV.

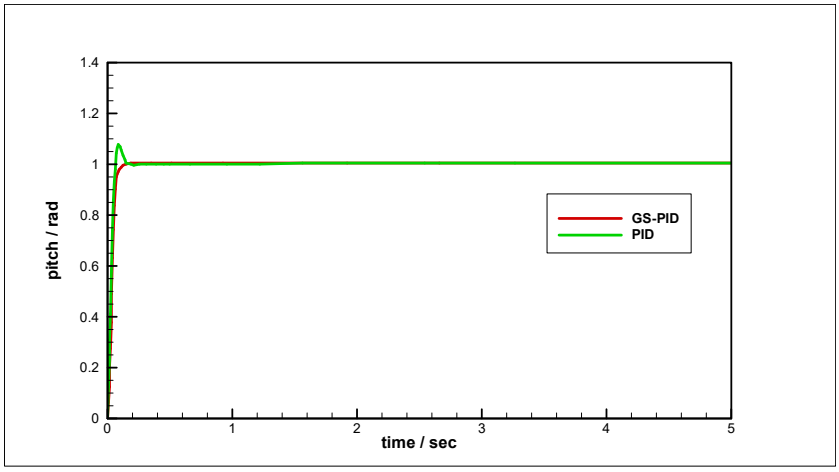


Figure 5.2 Simulation results of pitch angle

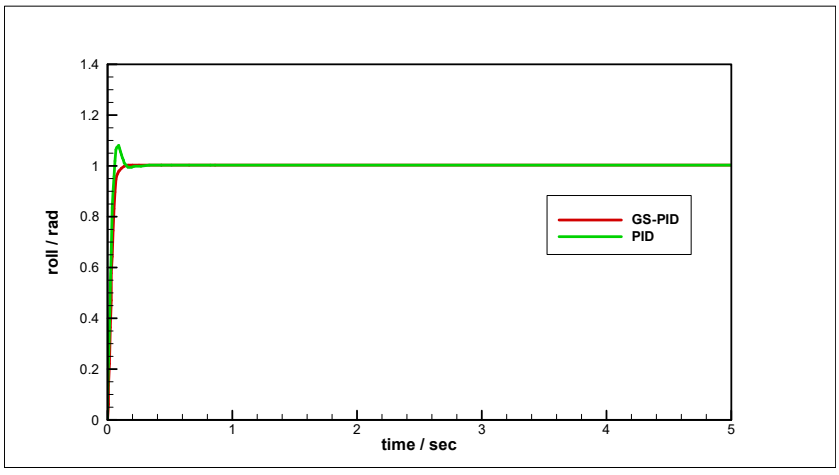


Figure 5.3 Simulation results of roll angle

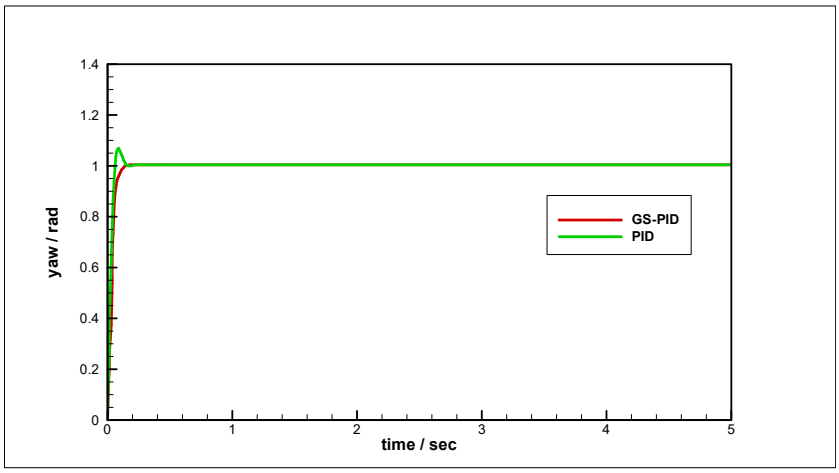


Figure 5.4 Simulation results of raw angle

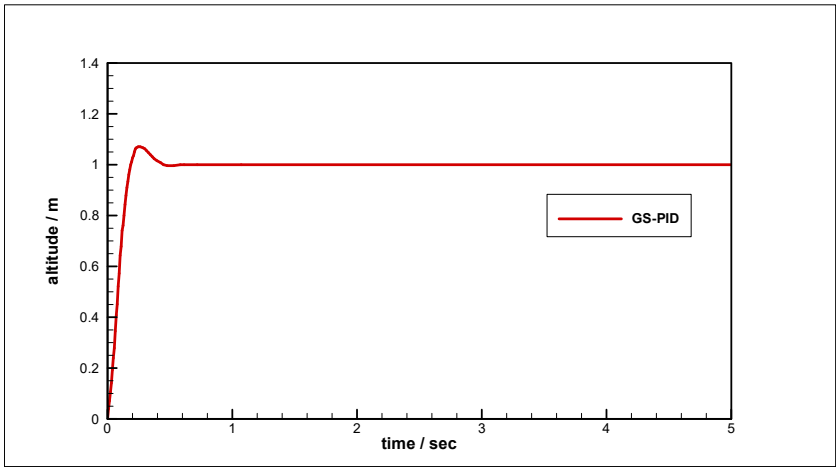


Figure 5.5 Simulation results of altitude

Through the above simulation experiments, it can be seen that the stability, rapidity and accuracy of gain scheduling PID controller are better than simple PID controller. It also shows that the gain scheduling controller can be well applied to the quadrotor UAV control system platform.

5.3 Fixed-Point Control

A fixed-point control is carried out in this study. Set the desired position for the fixed-point control as follows:

$$x_d = 1 - e^{-t^3}(m)$$
$$y_d = 1 - e^{-t^3}(m)$$
$$z_d = 2 \times (1 - e^{-t^3})(m)$$

Desired yaw angle is

$$\psi_d = \frac{\pi}{6} \times (1 - e^{-t^3})(rad) = 30 \times (1 - e^{-t^3})(\deg)$$

and desired output x_d, y_d, z_d will soon rise to 1 meters, d and ψ will soon rise to 30 degrees. The attenuation term is followed to ensure that the derivative of the desired trajectory is zero and to prevent the simulation from occurring at zero time, and reduce the impact of control. Fixed-point control takes the controller parameters as $A_1 = A_2 = A_3 = A_4 = diag[1,1]$, $A_5 = A_6 = A_7 = A_8 = 1$, $A_9 = diag[1,1,1,1]$. The corresponding adaptive gain parameters are $\Gamma_1=5$, $\Gamma_2=0.005$, $\Gamma_4=0.5$, $\Gamma_5=5$. The parameters of projective operators are taken as $\bar{c} = 10, c = 0.5, \varepsilon = 0.2$. The initial state of the 6 degrees of freedom of the quadrotor aircraft is calculated as: $x(0)=y(0)=z(0)=\phi(0)=\theta(0)=\psi(0)$. The initial value of adaptive parameter estimation is taken as:

$$\hat{s}_1(0) = \hat{s}_3(0) = \hat{s}_4(0) = \hat{s}_6(0) = 0$$

$$\hat{s}_2(0) = \hat{s}_5(0) = 0.5$$

In Figures 5.6, 5.7, and 5.8, x, y, and z represent the system's position output curve, x_d,y_d z_d represent related expectations, e_x, e_y, e_z showed the error curve. As can be seen from the figures, the position outputs x, y and z can all reach the desired set point, and the error can eventually be zero, and the control effect is good. The adjustment time is less than 5 seconds, the response speed is relatively fast.

In Figures 5.9 and 5.10, φ and θ represent the output curve of the system roll angle and pitch angle. As can be seen from the figure, the roll angle and pitch angle will have an oscillation at the beginning of the adjustment, and the maximum amplitude is about 10 degrees, which is acceptable to the system. Moreover, these two angles are rapidly approaching zero, the adjustment time is less than 5 seconds.

In Figure 5.11, ψ showed the yaw angle, ψ_d showed the expectation, e_ψ showed the error. As can be seen from the figure, the yaw angle can reach the desired set point, and the error eventually tends to zero. The adjustment time is less than 5 s, the response speed is relatively fast.

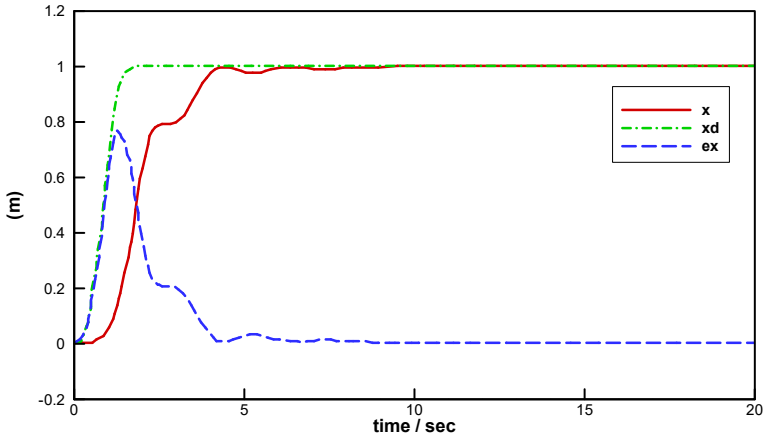


Figure 5.6 Horizontal position output x, expect d_x and its error curve

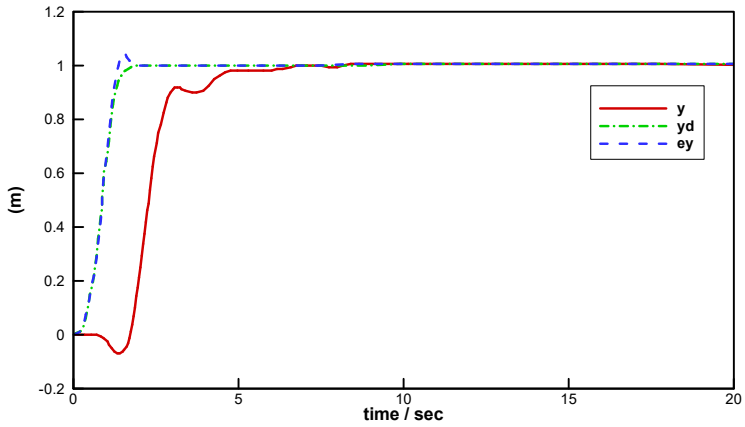


Figure 5.7 Horizontal position output y, expect d_y and its error curve

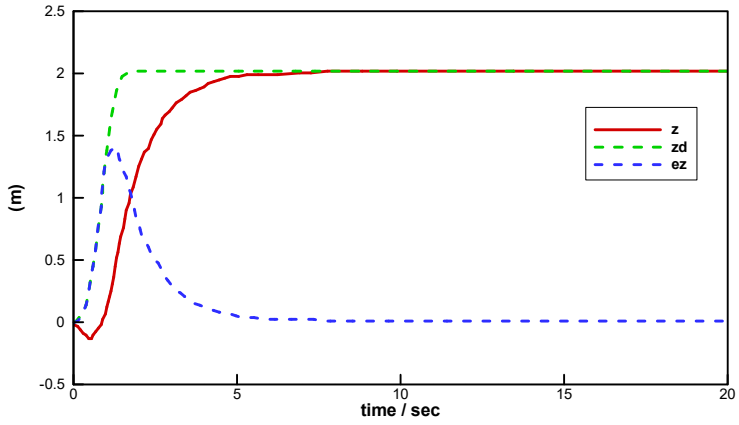


Figure 5.8 System position output z, expected d_z and its error curve

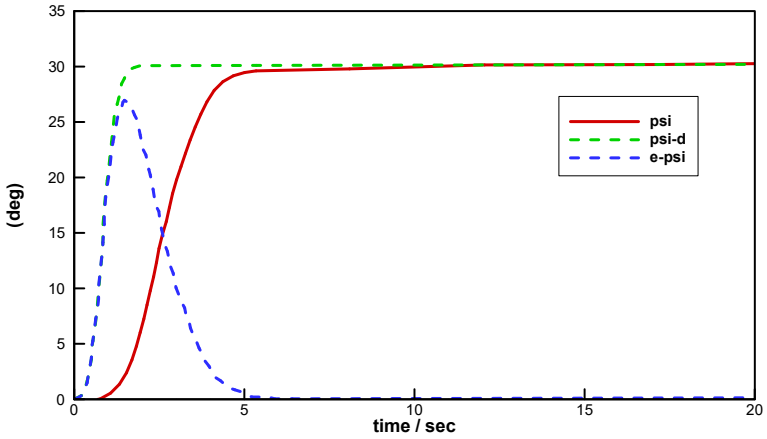


Figure 5.9 System yaw angle ψ, and their expectations ψ_d and error curve e_ψ

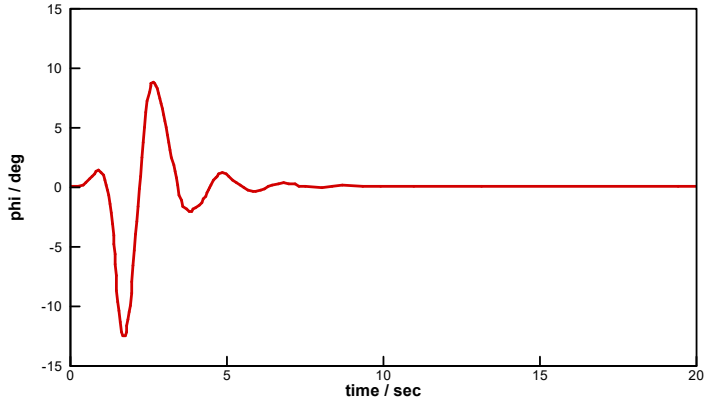


Figure 5.10 Rolling angle curve

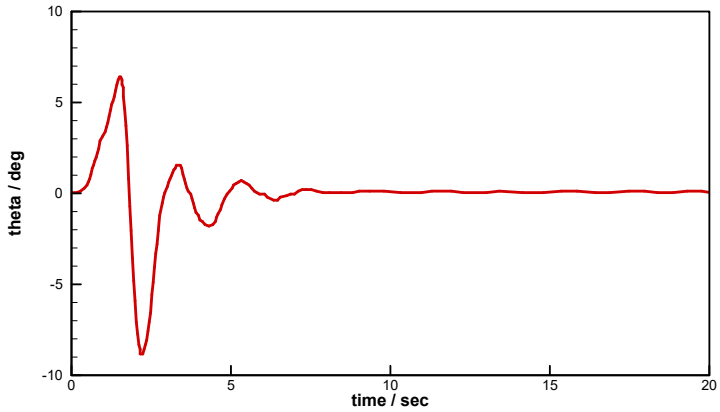


Figure 5.11 Elevation curve

It can be seen from the output curve of the six degrees of freedom of the integrated system that the controller can make the system state of $\{x,y,z,\psi\}$ reached the set point, while maintaining the

attitude angle $\{\phi,\theta\}$ stability.

Figure 5.12 is a three-dimensional trajectory of point control of a quadrotor unmanned aerial vehicle. As can be seen from the figure, the aircraft starts at points (0, 0, 0), then spirals up to point (1, 1, 2), that is, the control set point, and achieves the effect of point control.

The following are the investigation of the control force diagram of the system, mainly to see whether they are stable and the system is in line with the actual situation.

It can be seen from Figure 5.13, Figure 5.14, Figure 5.15 and Figure 5.16 that, the control forces of the system, F_1,F_2,F_3,F_4, eventually stabilized at 5 s, equal to the gravity of 0.5 kilograms of objects. This lift is easy for the propeller of the Quadrotor aircraft.

Figure 5.17 shows the composition of forces F_1,F_2,F_3,F_4, it can be seen that after the system is stable, the rising force of the quadrotor is about 19.6 newton, which is equal to the weight of the aircraft. This is in line with the actual situation.

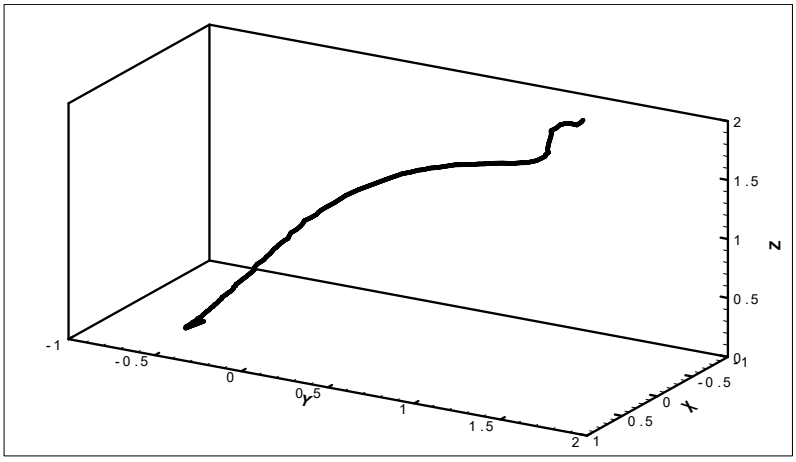


Figure 5.12 3D trajectory of fixed point control

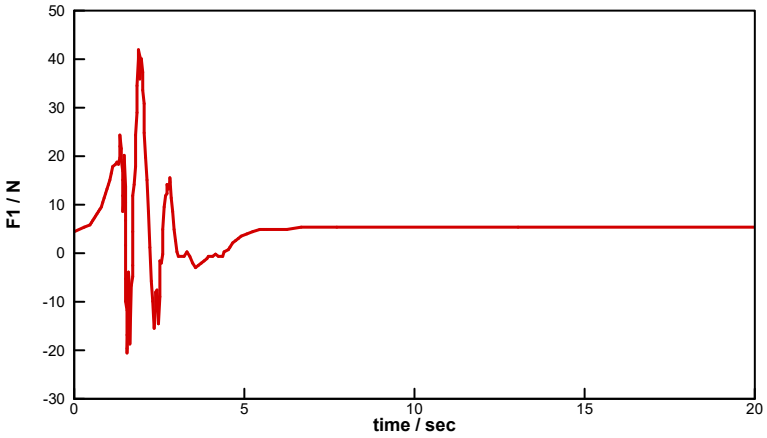


Figure 5.13 Control force F_1 curve

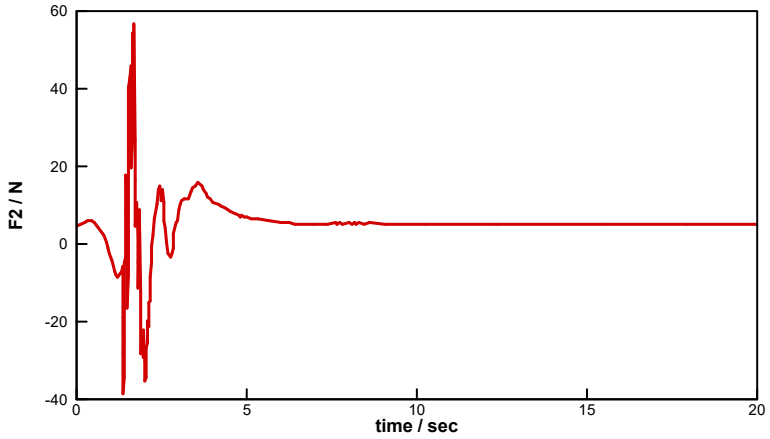


Figure 5.14 Control force F_2 curve

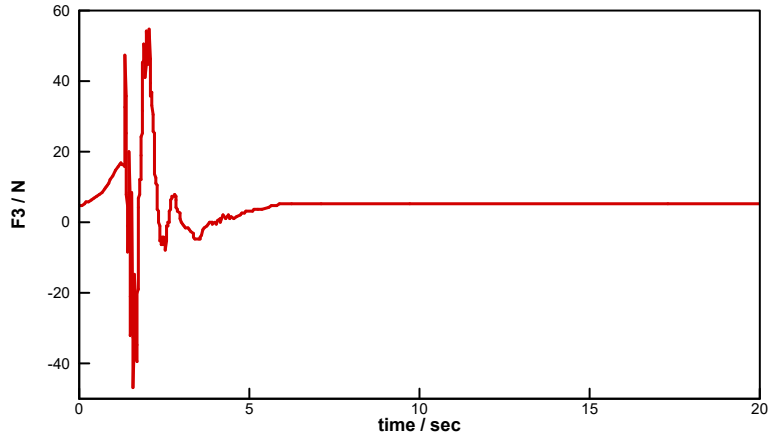


Figure 5.15 Control force F_3 curve

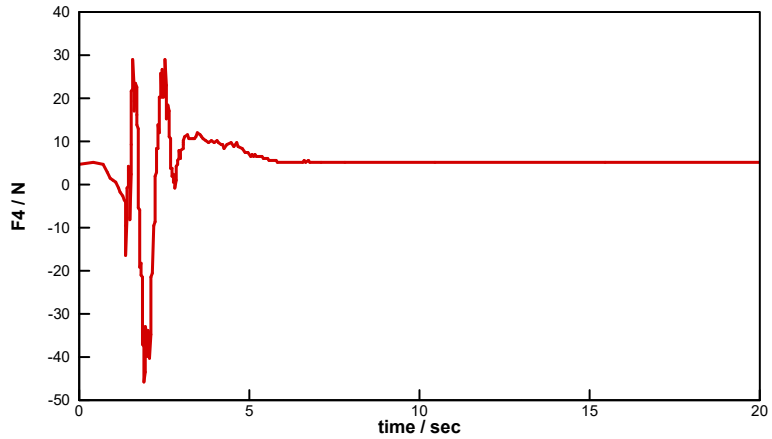


Figure 5.16 Control force F_4 curve

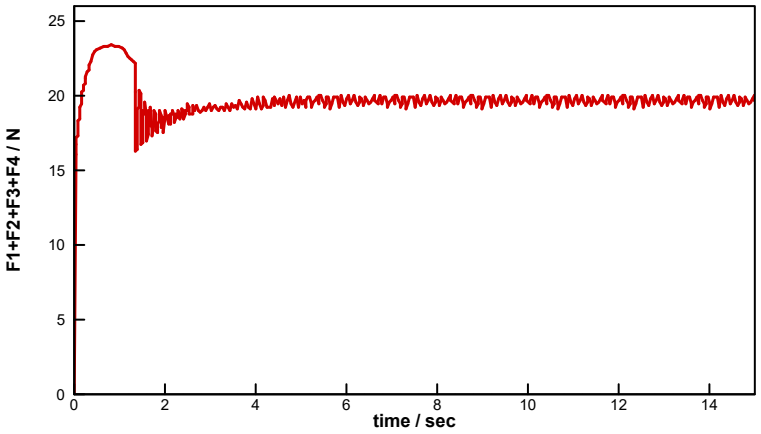


Figure 5.17 Composition of forces curve F_i $(i = 1,2,3,4)$

5.4 Trajectory Tracking Control with GS-PID Controller

A trajectory tracking control is performed by the GS-PID controller Simulink simulation system. Set the desired trajectory for the trajectory tracking control as,

$$x_d = \sin t(1 - e^{-t^3})\ (m)$$

$$y_d = \cos t(1 - e^{-t^3})(m)$$

$$z_d = 1 \cdot (1 - e^{-t^3})(m)$$

and desired yaw angle trajectory is

$$\psi_d = 30 \sin t(1 - e^{-t^3})\ (\deg)$$

The attenuation term is used to ensure that the derivative of the desired trajectory is zero at zero time. In a very short period of time, the expected output is attenuated:

$$x_d = \sin t(m),\ y_d = \cos t(m),\ z_d = 1(m),\ \psi_d = 30 \sin t(\deg)$$

Take the controller parameters as:

$$A_1 = A_2 = A_3 = A_4 = diag[2,2],\ A_5 = A_6 = A_7 = A_8 = 2,\ A_9 = diag[2,2,2,2]$$

In Figures 5.18, 5.19, and 5.20, curves *x, y,* and *z* represent position output curves under trajectory tracking control, x_d, y_d, z_d represent the corresponding desired trajectory, and, θ_x, θ_y, θ_z showed the error curve. As can be seen from the figures, the final error of the position output tends to zero, that is to say, the controller ensures the stability of the position output and achieves

trajectory tracking. The adjustment time is about 3 seconds.

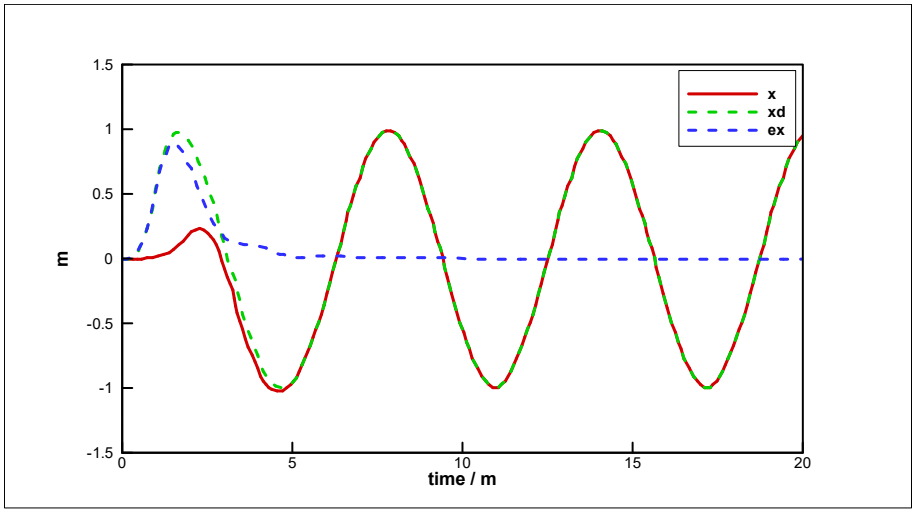


Figure 5.18 Position x under trajectory tracking, desired trajectory x_d and its error curve

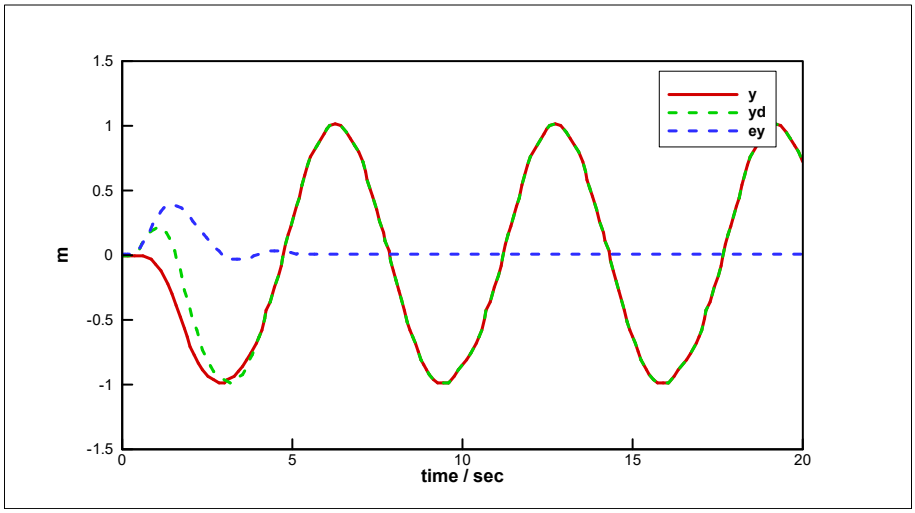


Figure 5.19 Position y under trajectory tracking, desired trajectory y_d and its error curve

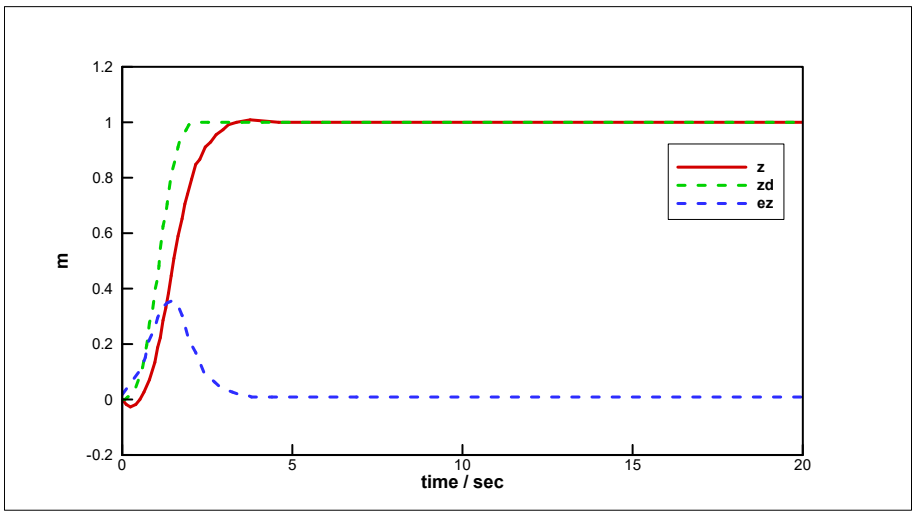


Figure 5.20 Position z under trajectory tracking, desired trajectory z_d and its error curve

In Figure 5.21 showed yaw trajectory, ψ_d is the desired trajectory yaw angle, e_ψ showed the corresponding error. It can be seen that after the system is stable, e_ψ is vibration up and down at 2 degrees that is to say the system tracking error is less than 6.7%. The trajectory tracking of yaw angle is achieved.

Figures 5.22 and 5.23 show the output curve of the system roll angle ϕ and pitch angle θ. As you can see from the figures, the roll angle and pitch angle tend to be stable in 5 seconds and vibrate at a maximum of 5 degrees. The magnitude of this vibration is acceptable in practice.

The simulation results indicate that in the trajectory tracking control, the state of system$\{x,y,z,\psi\}$ is to track the desired trajectory, while keep the attitude angle $\{\phi,\theta\}$ steady. The simulation results also indicated that the system control force for trajectory tracking is eventually stabilized at about 5 second, no more than 40 Newton within normal range, and the resultant composition forces is eventually stabilized near 19.6 N.

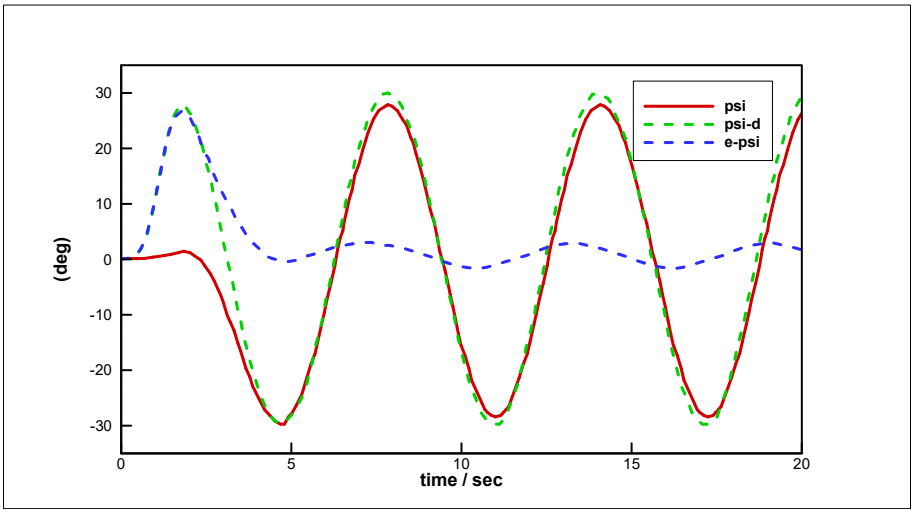


Figure 5.21 Yaw angle ψ under trajectory tracking, desired yaw angle ψ_d and its error curve

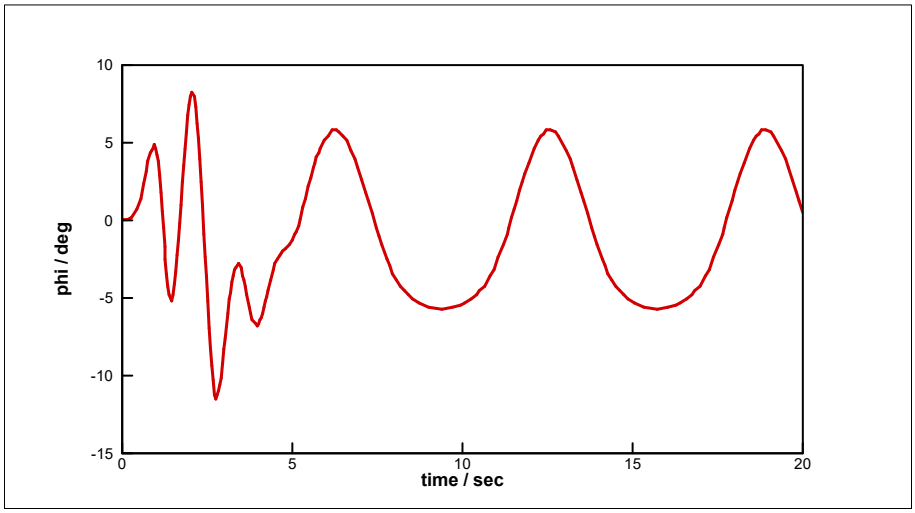


Figure 5.22 Curve of rolling angle ϕ under trajectory tracking

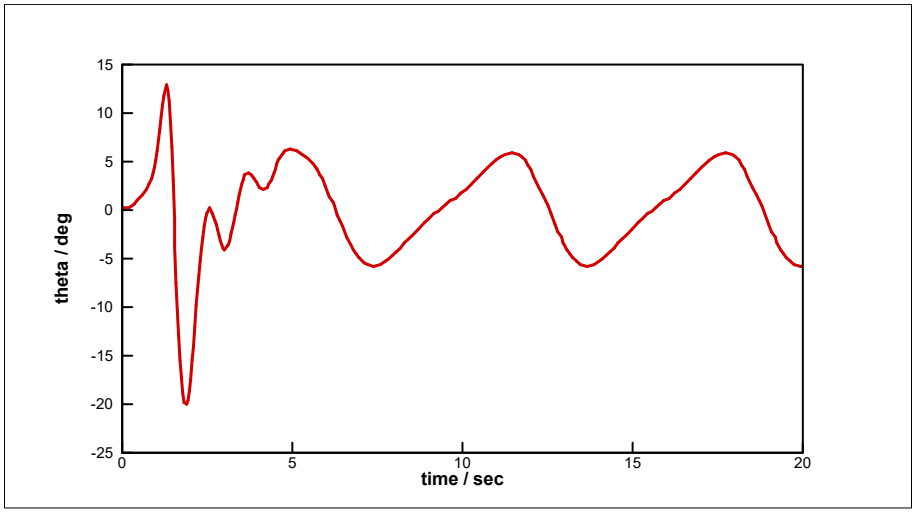


Figure 5.23 Curve of pitching angle θ under trajectory tracking

Figure 5.24 is a three-dimensional trajectory map of a quadrotor UAV trajectory tracking. As can be seen from the figure, the aircraft starts at points (0, 0, 0), then spirals up to the plane of z= 1, and this plane travels along a circle of 1 meters in diameter. Therefore, the control design makes the quadrotor UAV achieve the trajectory tracking effect.

In order to further illustrate the dynamics control of the quadrotor UAV with present GS-PID controller, the control force and the torque under trajectory tracking are all shown in the following figures.

Figures 5.25, 5.26, 5.27, and 5.28 show the results of the system control force for trajectory tracking. It can be seen from these figures that, the control forces of the system, F_1,F_2,F_3,F_4, eventually stabilized at about 5 second, no more than 40 Newton within normal range.

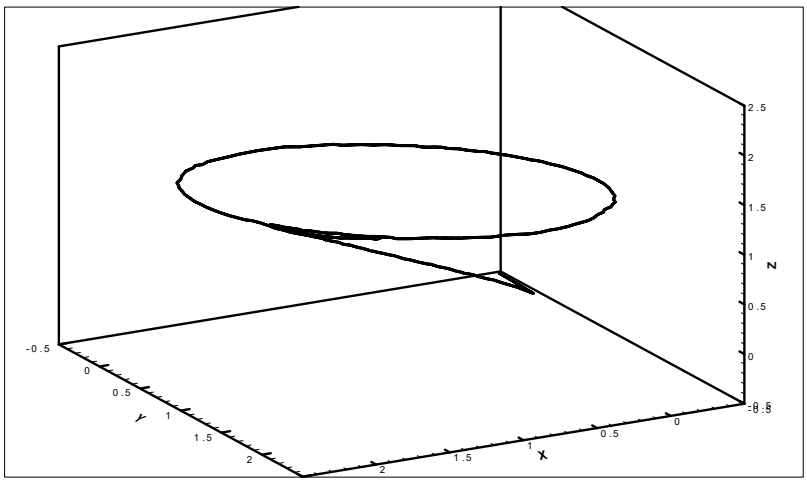


Figure 5.24 3D rendering of trajectory tracking

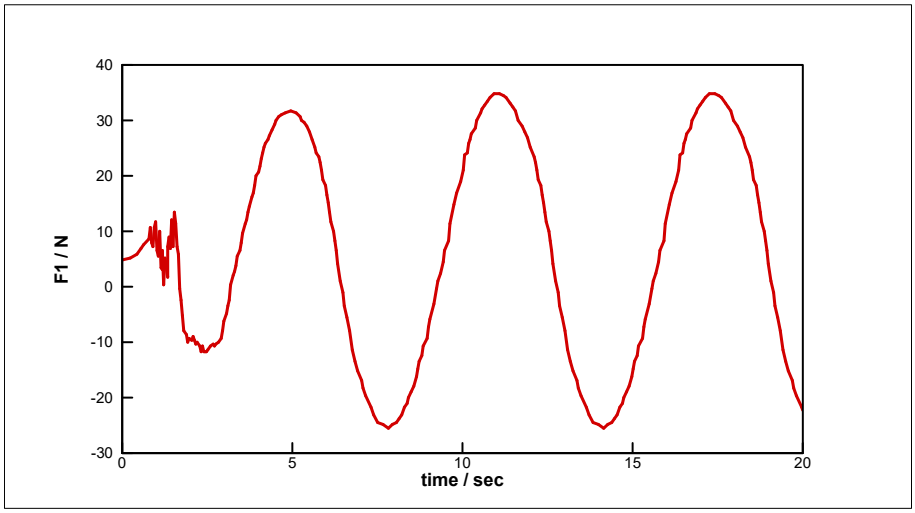


Figure 5.25 Trajectory tracking control force curve F_1

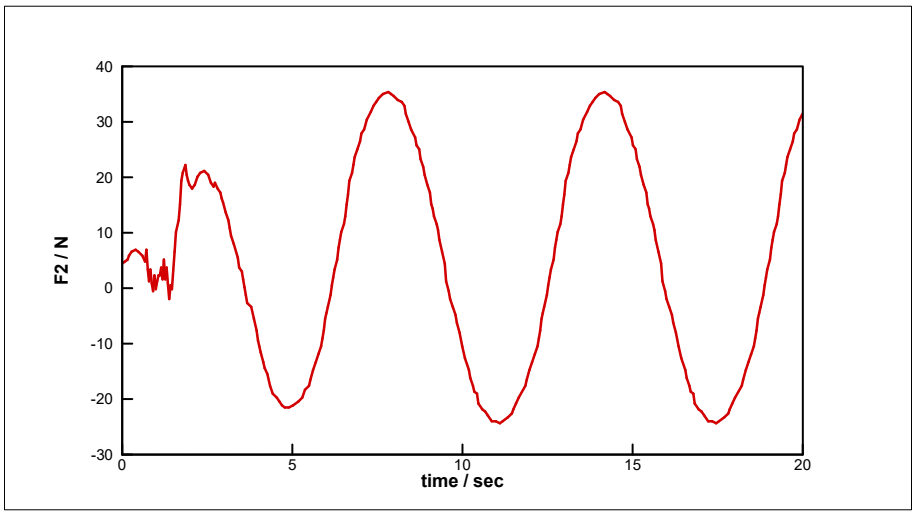


Figure 5.26 Trajectory tracking control force curve F_2

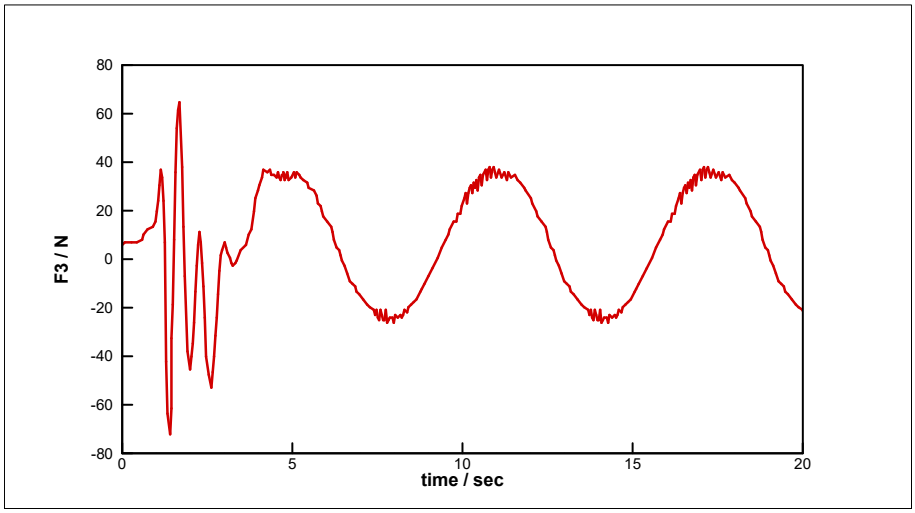


Figure 5.27 Trajectory tracking control force curve F_3

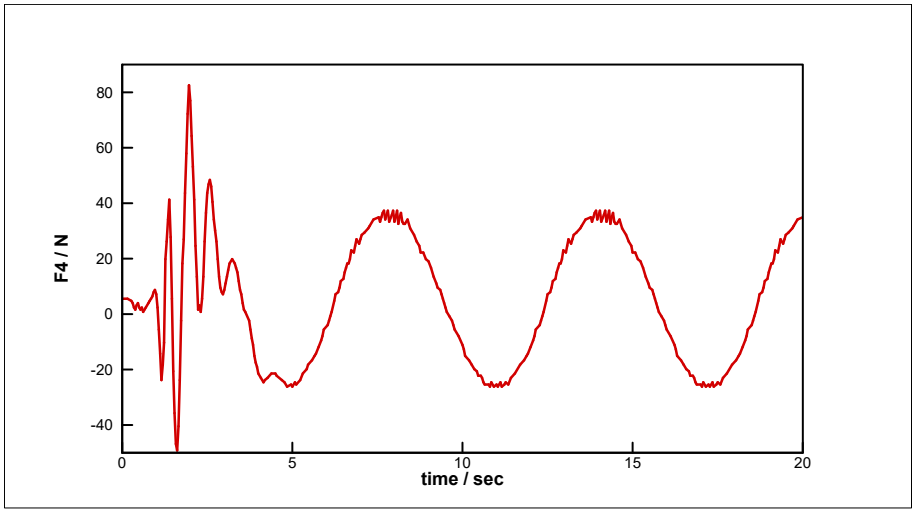


Figure 5.28 Trajectory tracking control force curve F_4

Figure 5.29 is the resultant of composition of forces curve on quadrotor F_i (i=*1,2,3,4*) under trajectory tracking control. As seen from the figure, the resultant force is eventually stabilized near 19.6 N, which is exactly the weight of the aircraft.

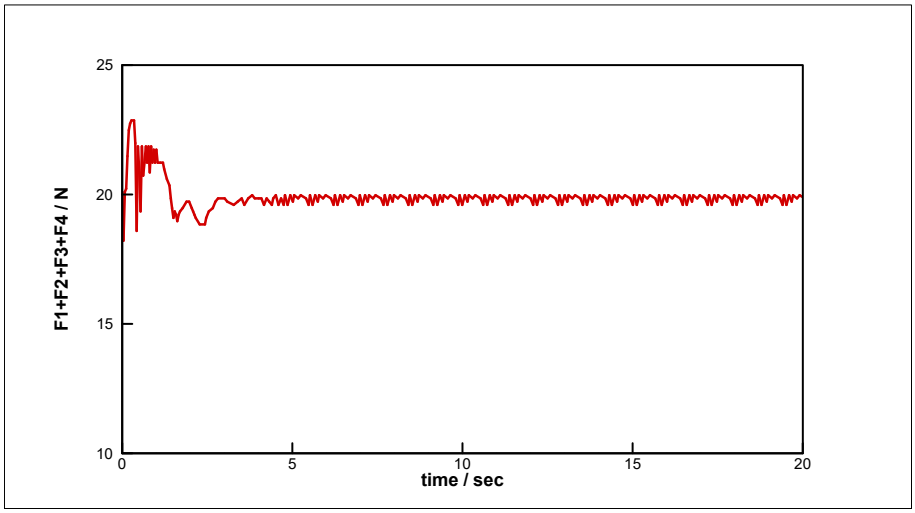


Figure 5.29 F_i (i=1,2,3,4) under trajectory tracking control

Figure 5.30, Figure 5.31 and Figure 5.32 are the torques $\{T_\phi, T_\theta, T_\psi\}$ in the trajectory tracking process. As can be seen from the figures, three torsional moments eventually become stable. After stabilization, $\{T_\phi, T_\theta\}$ oscillate in a small range near zero. The torque of T along the direction of the yaw angle is close to sine wave, which is within the normal range.

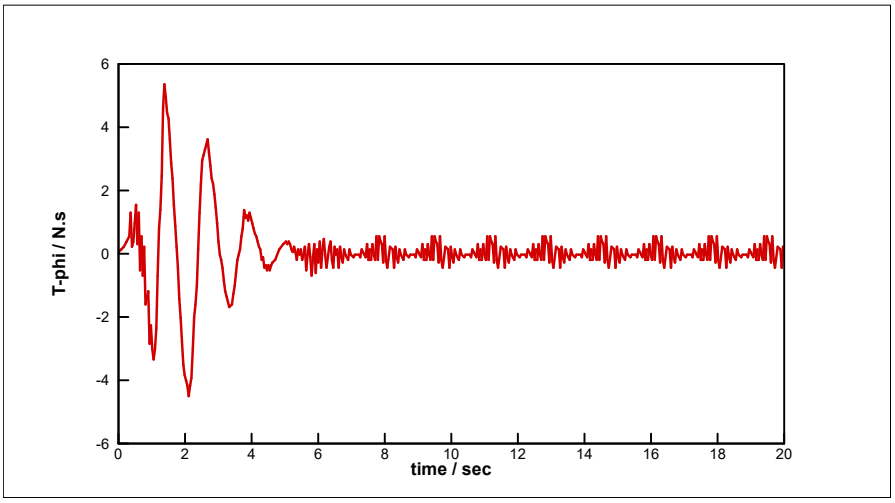


Figure 5.30 Torque T curve following trajectory tracking

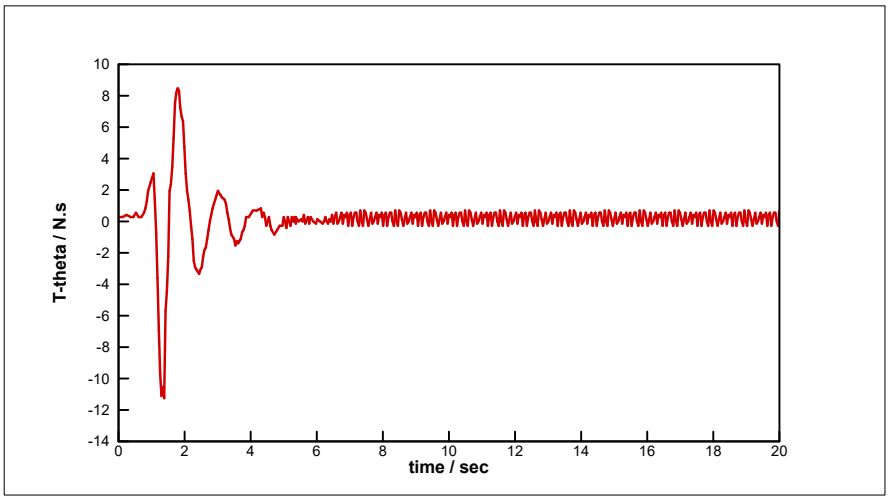


Figure 5.31 Torque T_θ curve following trajectory tracking

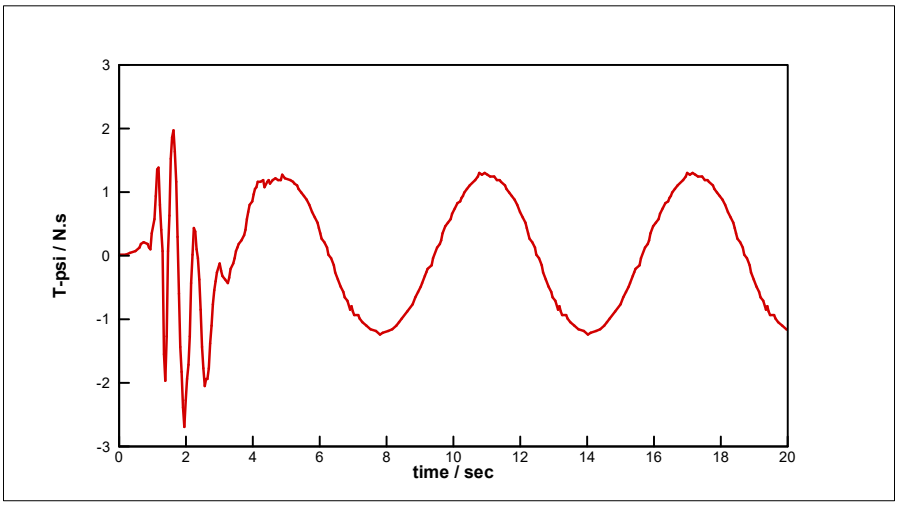


Figure 5.32 Torque T_ψ curve following trajectory tracking

5.5 Payload Dropping Control with PID and GS-PID Controller

Qball-X4 quadrotor helicopter payload dropping control with GS-PID controller is performed using simulation experiment and flight test in this chapter. Simulation experimental and flight test were carried out in the Networked Autonomous Vehicles Lab (NAVL) of Concordia University.

5.5.1 Simulation Experimental Results Analysis

In the simulation experiment, the initial state of the Qball-X4 is set as, altitude z is 0 m, roll angle ϕ is 0 degree, pitch angle θ is 0 degree, and yaw angle ψ is 0 degree. And while the Qball-X4 finishes climbing and is in a rotating trajectory, the desired altitude z_d is 1m and the rotating diameter is 1m. The payload of 300g in the Qball-X4 is dropping at the 20 second.

Simulation time is set to 45 seconds. The focus of this study is mainly on the flight stability when the payload is dropped from the Qball-X4.

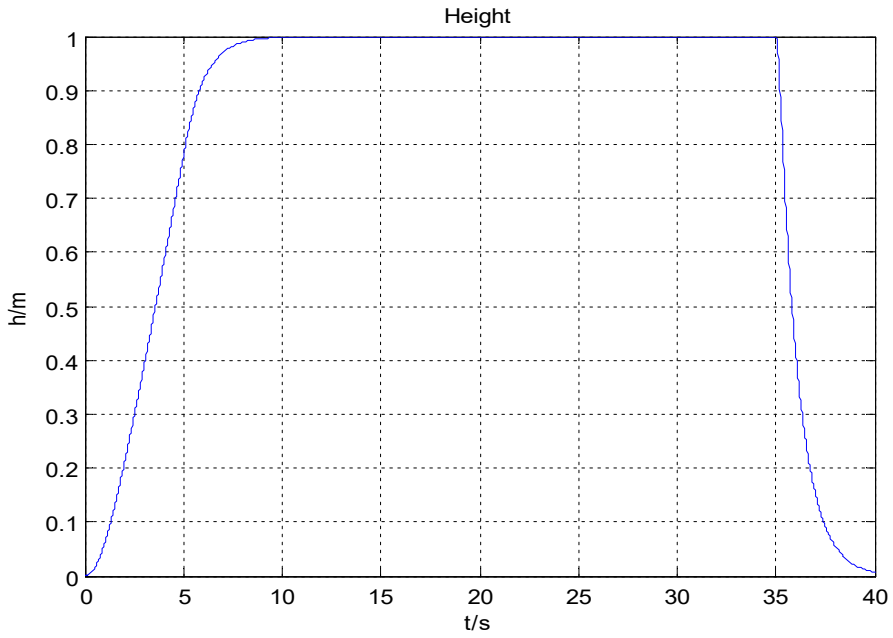
Firstly, the PID controller was used to control the flight of the Qball-X4 helicopter over the phases of take-off, hover with payload onboard, payload dropping, and landing. The assigned gains for PID controller set as K_p=0.6, K_i=1 and K_d=0.1. Figure 5.33 to Figure 5.35 show the flying path, flying attitude and the total thrust of Qball-X4 when payload drops. It could be seen from these results that the single PID controller was capable of keeping the desired height. The total thrust of Qball-X4 shows the effect of the abrupt mass change when payload drops.

Secondly, the single PID controller was replaced by the GS-PID controller to make improvements. In this simulation, three sets of PID gains are used for take-off, hovering, and payload dropping phases as shown in Table 5-3. The first set of gains is designed for short rise time in take-off. The second set is assigned to be switched prior to the desired height of 1m. The third set of gains is assigned for the payload release phase of flight. After payload dropping, the controller switches back to gains at hover and the landing is performed with the same gains as hovering controller.

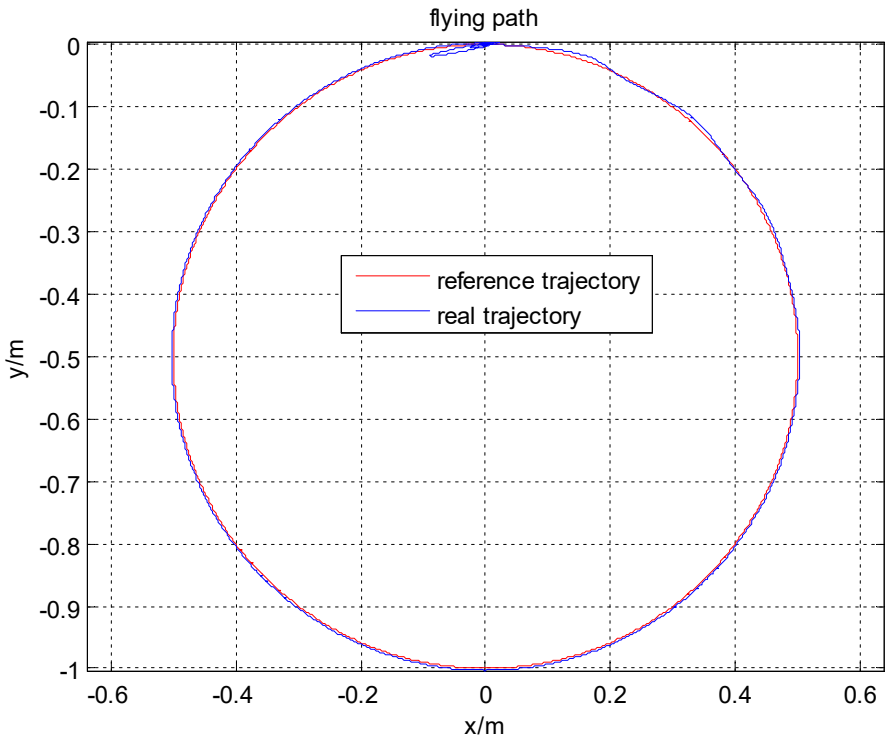
Figure 5.36 to Figure 5.38 show the flying path, flying attitude and the total thrust of Qball-X4 when payload dropping. It could also be seen from these results that the GS-PID controller was capable of keeping the desired height as well. The total thrust of Qball-X4 shows the effect of the abrupt mass change when payload drops at 20s.

Table 5-3 GS-PID controller gains for simulation

Gains	K_p	K_i	K_d
Take off	1	0.8	3
Hover	0.36	0.2	0.75
Payload drop	1.2	1.3	0.5



(a) The height of Qball-X4



(b) The X-Y position of Qball-X4

Figure 5.33 The flying path of Qball-X4 for payload dropping with PID controller

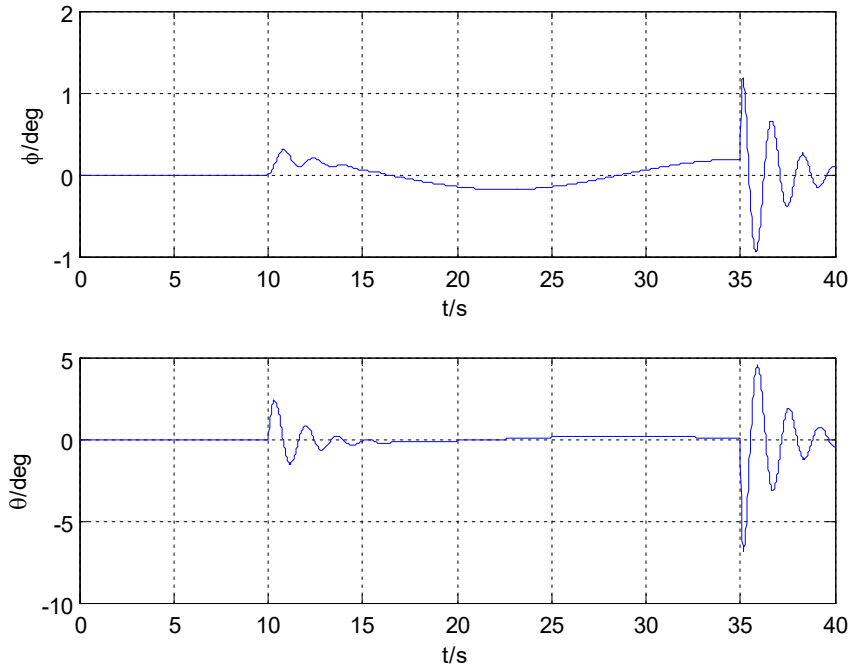


Figure 5.34 The flying attitude of Qball-X4 for payload dropping with PID controller

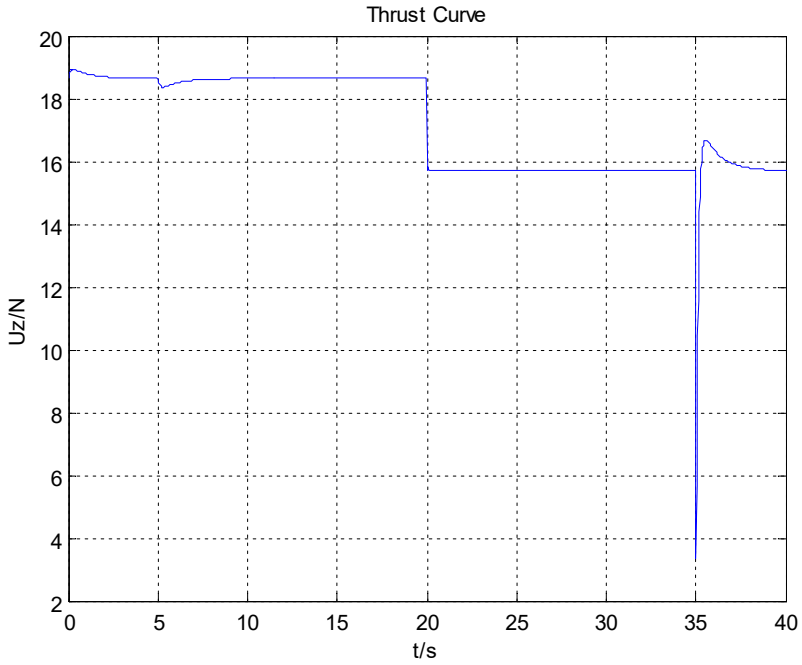
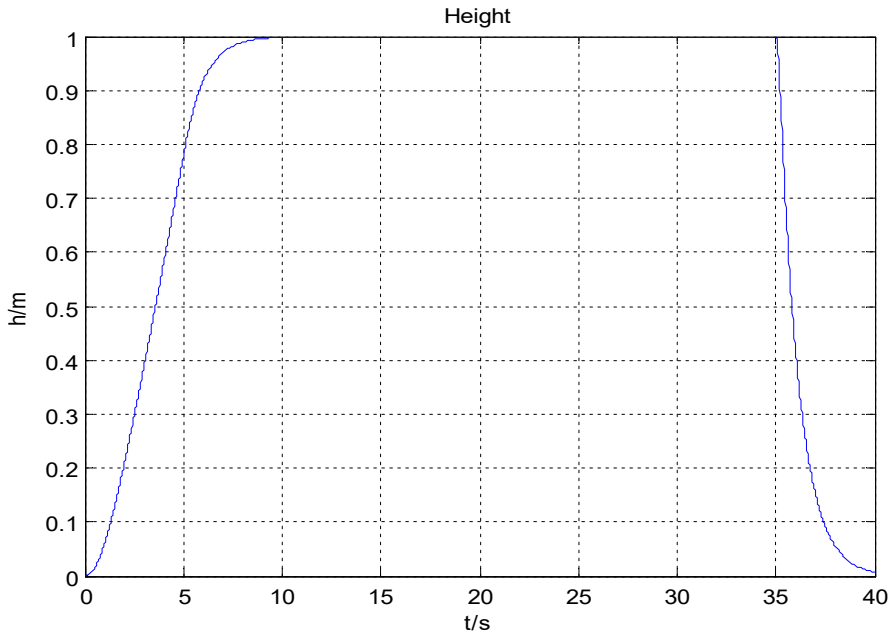
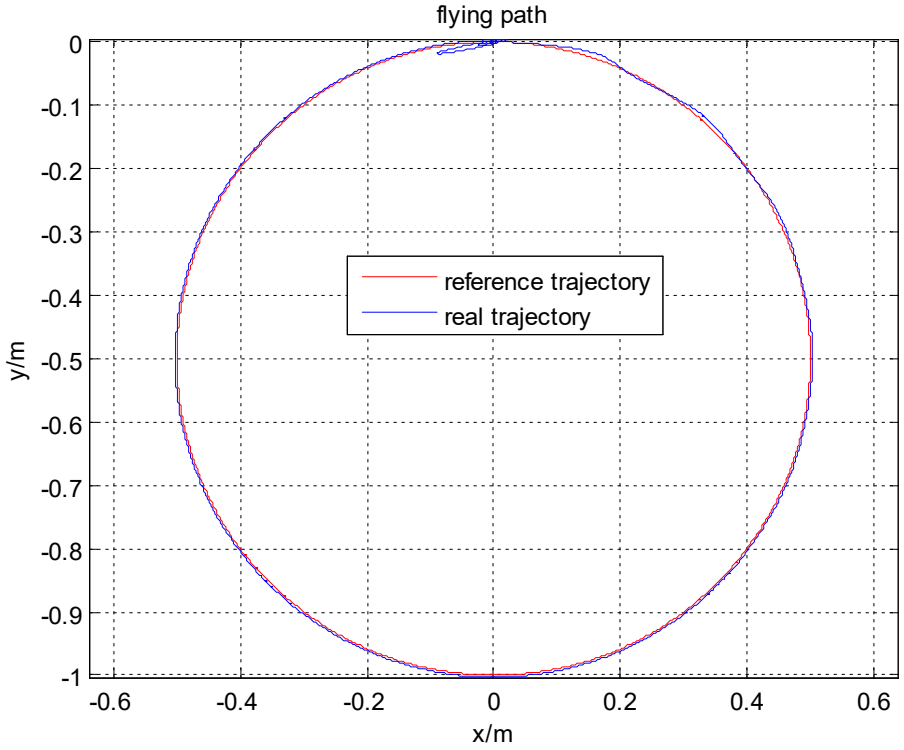


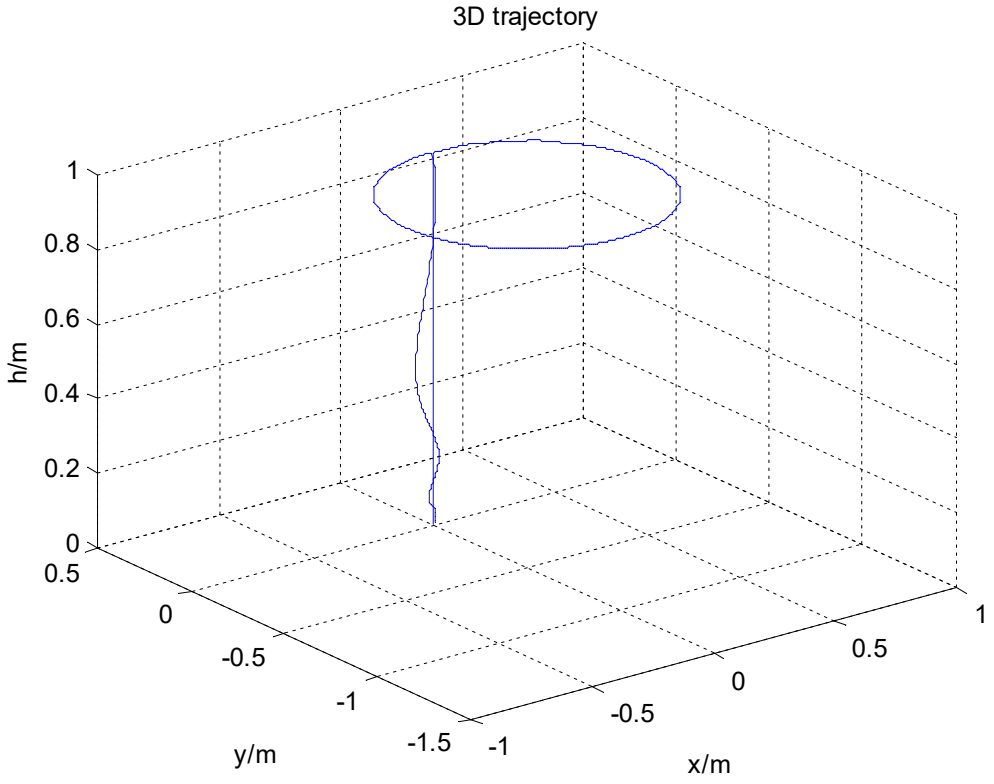
Figure 5.35 The total thrust of Qball-X4 with PID controller



(a) The height of Qball-X4



(b) The X-Y position of Qball-X4



(c) The 3D trajectory of Qball-X4

Figure 5.36 The flying path of Qball-X4 for payload dropping with GS-PID controller

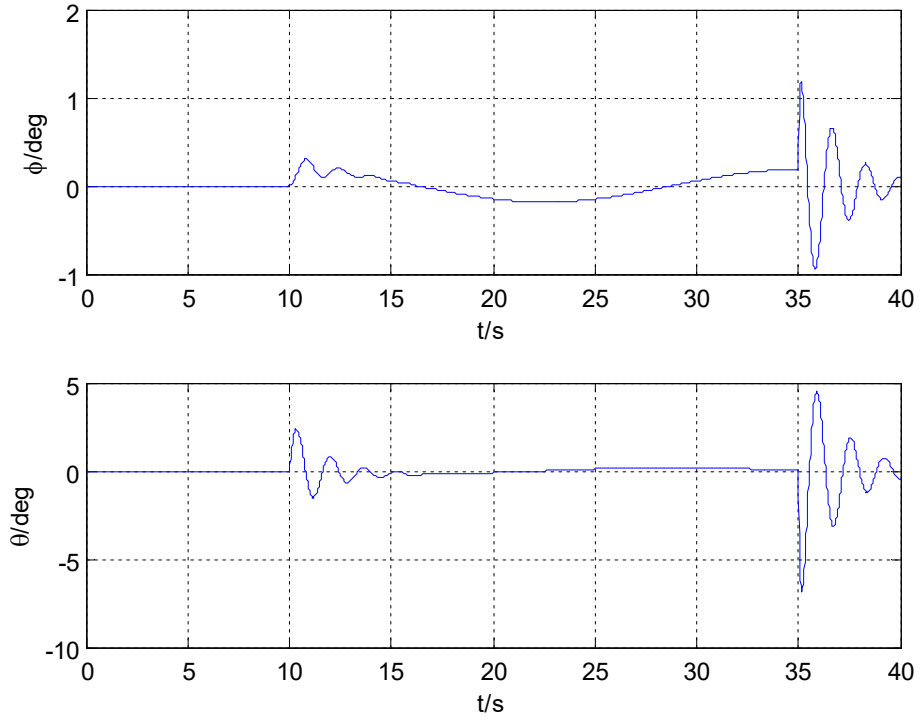


Figure 5.37 The flying attitude of Qball-X4 for payload dropping with GS-PID controller

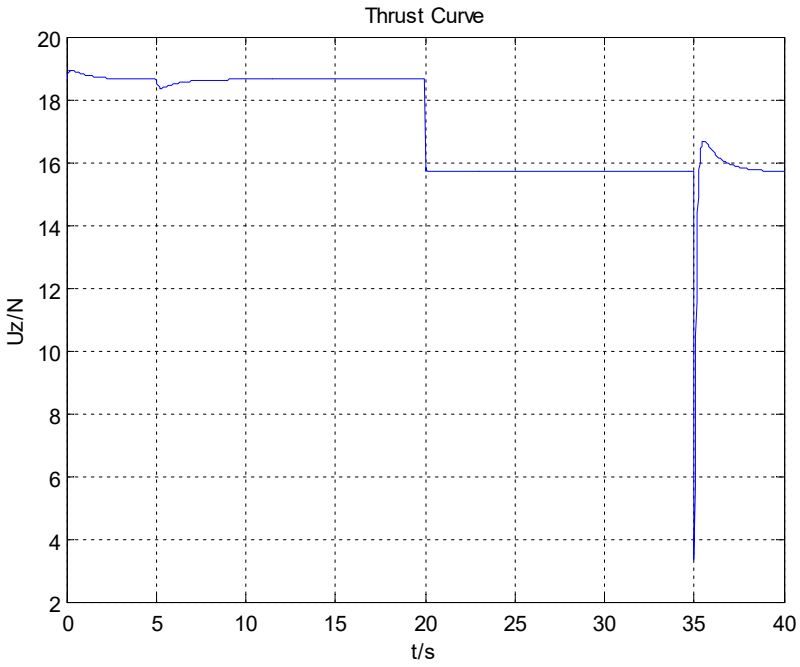


Figure 5.38 The total thrust of Qball-X4 with GS-PID controller

5.5.2 Flight Test Experimental Results Analysis

In this section, GS-PID controller is implemented in real time on the Qball-X4 quadrotor helicopter for performance analysis with payload dropping application. The flight test was carried out for the payload dropping control with Qball-X4 platform mentioned in Chapter 3 in NAVL. As indicated above, the Qball-X4 quadrotor helicopter works in the indoor environment.

Before the flight test, the control platform was analyzed and adjusted on the simulation platform. In consideration of the safety and stability of the platform, certain control parameters have been altered from the ones implemented in the simulation.

The desired altitude was changed into 0.7 m and the rotating diameter remains 1m. Flight time is changed to 100 seconds. When the Qball-X4 hovers on its rotating trajectory, the payload of 300g in the Qball-X4 is dropping at the 50 second.

Three sets of altered PID gains are used for take-off, hovering, and payload dropping phases. The first set of gains is designed for short rise time in take-off. The second set is assigned to be switched prior to the desired height of 0.7m to reduce the overshoot and finally the third set of gains

is assigned for payload release phase of flight.

Through multiple manual tuning of PID parameters, Table 5-4 shows the GS-PID controller gains for each phase of flight test.

Table 5-4 GS-PID controller gains for flight test

Gains	K_p	K_i	K_d
Take off	2.2	3	5
Hover	1.2	0.5	2.5
Payload drop	1.8	1.9	0.9

The output results of the altitude of the Qball-X4 helicopter payload dropping with GS-PID controller is shown in Figure 5.39.

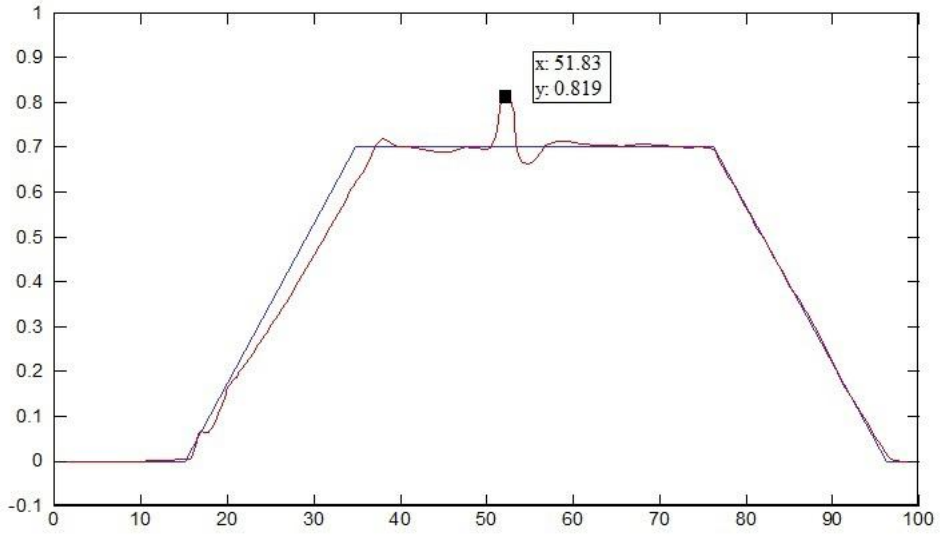


Figure 5.39 Payload dropping experiment with GS-PID controller altitude output results

It can be seen from Figure 5.39 that the Qball-X4 quadrotor with GS-PID control method comes with a small overshoot and an overall satisfying performance. At the 50 second, the payload is dropped from Qball-X4 helicopter, there is a fluctuation in the trajectory of the Qball-X4. The overshoot of Qball-X4 helicopter at the payload dropping is about 17% with the GS-PID controller. However, the GS-PID controller is capable of keeping the desired height and keeping stability after around 10 seconds of payload dropping. With GS-PID control method implied, the control performance of quadrotor helicopter and the ability of interference suppression have been successfully achieved.

5.6 Summary

In this chapter, a robust PID control strategy was presented for the control of quadrotor UAV. The proposed controller has been implemented in a nonlinear six-degree of freedom simulation model of a quadrotor helicopter.

Using the simulation experiment as well as the flight test, it is found that the GS-PID controller can achieve an acceptable performance, and the model analysis, point selection method and GS-PID controller design presented in this thesis is effective for the path tracking of the quadrotor helicopter. The simulation results show that the GS-PID control structure for the quadrotor UAV could increase the performance of the quadrotor UAV in tracking the desired trajectory and increase reliability of the quadrotor UAV. Based on the proposed pre-tuned PID controller design, advantages (for example, robust properties, easy design and implementation) of PID controller can be employed.

It is well known that it is difficult to accurately put a quadrotor dynamic model into experimental practice. The uncertainty can often occur during the flight test. The common uncertainties of the quadrotor UAV model are mass uncertainty, inertia uncertainty, and the uncertainty of the traction constant, especially for the abrupt mass and some inertia change scenario of payload dropping. The focus of this work is to maintain the height of quadrotor UAVs for payload drop applications with objective to reduce the overshoot of a quadrotor helicopter at the moment of payload dropping. The control design of this thesis solved the problem of aircraft control when there is mass uncertainty.

Chapter 6

Backstepping Controller of the Quadrotor Helicopter

In this chapter, a nonlinear backstepping controller will be designed for the quadrotor UAV. The objective of the controller is tracking of a desired trajectory asymptotically.

6.1 Introduction

It is well known that the quad-rotor UAV system has the characteristics of nonlinearity, multivariable and under-actuation. Because of its complex aerodynamic environment and the uncertainty of systems susceptible to external disturbances, it is usually necessary to build models on the basis of some engineering assumptions, which leads to a large difference between the simulation and experimental results of controller design. Therefore, the design of control system requires that when the object is uncertain, it can not only maintain closed-loop stability, but also have better robust performance.

As indicated in the Chapter 4, in order to obtain the best stability and performance of Qball-X4 with fault input, the switching action from one set of pre-tuned PID gains to another set plays a vital role at the moment of releasing the payload. If this transient (switching) time is held long (more than one second) it can cause the Qball-X4 to overshoot abruptly and even a crash. This can be avoided with the using of the nonlinear backstepping control.

Backstepping is a recursive control algorithm that works by designing intermediate control laws for some of the state variables. These state variables are called "virtual controls" for the system. Unlike other control algorithms that tend to linearize nonlinear systems such as the feedback linearization algorithm, backstepping does not work to cancel the nonlinearities in the system. This

leads to more flexible designs since some of the nonlinear terms can contribute to the stability of the system. An example of such terms that add to the stability of the system are state variables taking the form of negative terms with odd powers (e.g. $-x^3$), they provide damping for large values of x.

Since nonlinear backstepping control scheme for a class of nonlinear systems was first proposed, the research on the backstepping control of nonlinear systems has made much progress. The backstepping control can be simply understood as: the final control signal is obtained recursively through a series of "virtual" signals. The virtual signal, in fact, does not need to pass through the hardware, and can be obtained directly by calculation, so that the structure of the controller can be simplified; each step of recursion only needs to deal with a relatively simple error system, so that it can be compared. Flexible selection of control signals can effectively improve the transition process.

In the academic papers published in recent years [66,67], the backstepping method is often used in the control design of the quadrotor unmanned aerial vehicle. This method requires that the dynamic model of the system satisfies the strict feedback form, that is, the form of the inverted triangle. The dynamic model of the horizontal position subsystem of the quadrotor unmanned aerial vehicle shows that the control input of the system directly affects the roll angle and pitch angle of the aircraft, and the position and line velocity signals are not included in the dynamic model of the attitude angle. The dynamic model of the aircraft's horizontal position is directly related to the attitude angle of the aircraft, so it can be shown that the horizontal position subsystem of the quadrotor unmanned aerial vehicle satisfies the strict feedback form.

Modern UAVs need to be designed to achieve the desired performance under both normal and fault conditions. Therefore, the main objective and contribution of this chapter are to analyze the system responses based on backstepping control theory in quadrotor UAV. In this chapter, the backstepping control principle will be discussed firstly. Then, the backstepping control approach for quadrotor UAV will be presented in this chapter.

6.2 Backstepping Control Principle

The backstepping method is a control method based on Lyapunov stability theory. The

derivation of the backstepping method is also a proof of Lyapunov stability. The Lyapunov stability theory is presented firstly before introducing the backstepping method.

6.2.1 Lyapunov Stability Theory

Lyapunov stability theory is a theoretical system established by Russian mathematician Lyapunov in 1892 to analyze system stability. Two ways to judge whether a system is stable were given in this theory, that are indirect method and direct method. The basic idea of the Lyapunov direct method is to directly construct the Lyapunov function to determine the stability of the system, and there is no need to solve the state equation of the system. Even for complex nonlinear systems, the direct method can still accurately determine its stability.

Usually, a nonlinear system can be represented by the following equations:

$$\dot{x} = f(x,t)$$

where $x = \left[x_1, x_2, \cdots, x_n\right]^T$ representing the system state variables. According to the Lyapunov direct method, the stability of the system under the condition of equilibrium state could be estimated according to the symbolic characteristics of constructed function V (x) and its derivatives,

It is assumed that the state equation of a nonlinear system is as follows:

$$\dot{x} = f(x,t), f(0,t) = 0, \forall_t$$

Theorem 1: If the function $V(x)$ exists and has a continuous first order partial derivative, it also satisfies: 1) $V(x)$ is positive definite, 2) $\dot{V}(x)$ is negative definite. The equilibrium state of the system at its origin is asymptotically stable.

Theorem 2: If the function $V(x)$ exists and has a continuous first order partial derivative, it also satisfies: 1) *V(x)* is positive definite, 2) $\dot{V}(x)$ is semi negative definite. The equilibrium state of the system at the origin is uniformly stable.

Theorem 3: If the function $V(x)$ exists and has a continuous first order partial derivative, it also satisfies: 1) *V(x)* is positive definite, 2) $\dot{V}(x)$ is positive definite. The equilibrium state of the system at the origin is unstable.

6.2.2 Backstepping Approach

The backstepping approach, also known as inversion, is a new control system design method proposed by Professor Kokotovic of the University of California at Santa Barbara in 1991. The essence of this method is: Firstly, to reduce the order processing for the system, a high level system is split into several low order subsystems. Then, for each of the lower order subsystems, select a virtual control variable to design the corresponding controller and to ensure that the system before the subsystem can be asymptotically stable. The Lyapunov functions of all subsystems can be obtained by recursive construction and have to make sure that the derivatives of the function are all negative. Finally, after gradually revising the algorithm, we ensure that the system can achieve the purpose of tracking. From the above, we can see that the theoretical basis of the backstepping method is the Lyapunov stability principle. The design process of the backstepping method controller is also the proof of the Lyapunov stability of the system.

This method can be applied not only to the design of linear system controller, but also to the design of nonlinear system controller. The premise of system controller design using backstepping is that the system model satisfies the strict feedback structure. The system structure can be expressed as follows:

$$\begin{cases} \dot{x}_i = f_i(\overrightarrow{x_i}) + g_i(\overrightarrow{x_i}) \cdot x_{i+1} \\ i = 1, \cdots, n-1 \\ \dot{x}_n = f_n(\overrightarrow{x_n}) + g_n(\overrightarrow{x_n}) \cdot u \end{cases} \tag{6-1}$$

where $\vec{x}_i = [x_1, \cdots, x_i], i = 1, 2, \cdots, n$ is the state variable vector of the system, u is the control input, $f_i(\vec{x}_i), g_i(\vec{x}_i)$ are known functions. Here the second-order system model is taken as an example to illustrate the design process of backstepping.

It is assumed that the system model is shown as follows:

$$\begin{cases} \dot{x}_1 = f(x_1) + g(x_1) \cdot x_2 \\ \dot{x}_2 = u \end{cases} \tag{6-2}$$

Step 1: Suppose the stable point of the second order system is x_d, and the tracking error of the system state variable x_1 is defined as

$$\dot{e}_1 = \dot{x}_d - \dot{x}_1 \tag{6-3}$$

That is,

$$\dot{e}_1 = \dot{x}_d - f(x_1) - g(x_1) \cdot x_2 \tag{6-4}$$

Take the Lyapunov function $V_1 = 1/2e_1^2$, and get derivative of it as,

$$\dot{V}_1 = e_1 \cdot \dot{e}_1 \tag{6-5}$$

That is

$$\dot{V}_1 = e_1 \cdot (\dot{x}_d - f(x_1) - g(x_1) \cdot x_2) \tag{6-6}$$

Introducing virtual control variable α_1, defining error variable $e_2 = x_2 - \alpha_1$. To substitute it into the upper equation (6-2), we obtain,

$$\dot{V}_1 = e_1 \cdot [\dot{x}_d - f(x_1) - g(x_1) \cdot (e_2 + \alpha_1)] \tag{6-7}$$

Take

$$\alpha_1 = [\dot{x}_d - c_1 e_1 - f(x_1)] / g(x_1) \tag{6-8}$$

Then we have,

$$\dot{V}_1 = -c_1 e_1^2 - g(x_1) e_1 e_2 \tag{6-9}$$

That is,

$$\dot{e}_1 = -c_1 e_1 - g(x_1) e_2 \tag{6-10}$$

$$e_2 = (-\dot{e}_1 - c_1 e_1) / g(x_1) \tag{6-11}$$

where $c_1 > 0$

Step 2: Take the Lyapunov function $V_2 = 1/2e_2^2$, and get derivative of it as,

$$\dot{V}_2 = \dot{V}_1 + e_2 \dot{e}_2 \tag{6-12}$$

To get the derivative of error variable,

$$\dot{e}_2 = \dot{x}_2 - \partial \alpha_1 / \partial x_1 \dot{x}_1 \tag{6-13}$$

To substitute equation (6-2) into the equation (6-13), we obtain,

$$\dot{e}_2 = u - \partial \alpha_1 / \partial x_1 [f(x_1) + g(x_1) x_2] \tag{6-14}$$

To substitute equation (6-14) into the equation (6-12), we have,

$$\dot{V}_2 = -c_1 e_1^2 + e_2[u - \partial\alpha_1 / \partial x_1(f(x_1) + g(x_1)x_2) - g(x_1)e_1] \tag{6-15}$$

Take the controller as,

$$u = -c_2 e_2 + g(x_1)e_1 + \partial\alpha_1 / \partial x_1(f(x_1) + g(x_1)x_2) \tag{6-16}$$

where $c_2 > 0$.

To substitute equation (6-16) into the equation (6-15), we have,

$$u = -c_2 e_2 + g(x_1)e_1 + \partial\alpha_1 / \partial x_1(f(x_1) + g(x_1)x_2) \tag{6-17}$$

According to the Lyapunov stability theorem, the system will be globally asymptotically stable under the action of controller Equation (6-16).

6.3 Backstepping Controller Design Approach

As discussed in Chapter 2, due to the strict symmetry of the quadrotor helicopter structure, the quadrotor height control channel and yaw control channel are relatively independent, and its horizontal control is controlled by the underdrive of rolling angle and pitch angle. The relationship between the virtual input of the system $U = (U_1, U_2, U_3, U_4)$ and the 6 degrees of freedom of the system $(x, y, z, \phi, \theta, \psi)$ is like that of Figure 6.1.

$$\begin{pmatrix} U_2 \\ U_3 \\ U_4 \end{pmatrix} \rightarrow \begin{pmatrix} \phi & \dot{\phi} \\ \theta & \dot{\theta} \\ \psi & \dot{\psi} \end{pmatrix} \Bigg\} \rightarrow \begin{pmatrix} x & \dot{x} \\ y & \dot{y} \\ z & \dot{z} \end{pmatrix}$$
$$U_1 \rightarrow \Bigg\}$$

Figure 6.1 control relationship of quadrotor UAV system

When virtual input vector $U = (U_2, U_3, U_4)$ is known, three attitude angles and their angular velocities can be obtained. If the U_1 is also again, then the position and speed of the aircraft will be obtained. Obviously, the attitude loop and the position loop are semi-coupled. The position loop is the inner loop and the position is the outer loop. Desired positions are denoted as $(x_{cmd}, y_{cmd}, z_{cmd})$ and desired attitude angles are $(\phi_{cmd}, \theta_{cmd}, \psi_{cmd})$, and the controller structure could be given as following Figure 6.2,

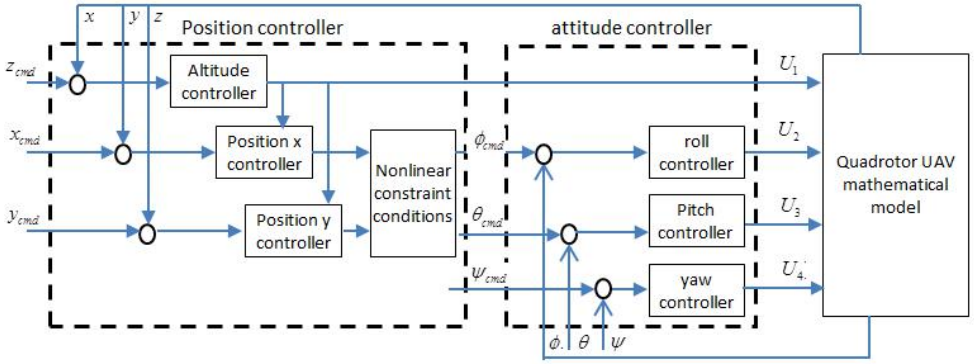


Figure 6.2 Double closed loop structure diagram of quadrotor unmanned aerial vehicle

6.4 Backstepping Controller Design for Quadrotor UAV

Take the state variables as $x=(\phi,\dot{\phi},\theta,\dot{\theta},\psi,\dot{\psi},z,\dot{z},x,\dot{x},y,\dot{y})$, that is

$$x_1=\phi,x_2=\dot{x}_1=\dot{\phi};x_3=\theta,x_4=\dot{x}_3=\dot{\theta};x_5=\psi,x_6=\dot{x}_5=\dot{\psi};$$
$$x_7=z,x_8=\dot{x}_7=\dot{z};x_9=x,x_{10}=\dot{x}_9=\dot{x};x_{11}=y,x_{12}=\dot{x}_{11}=\dot{y}$$

When virtual input is $U=(U_1,U_2,U_3,U_4)$, the mathematical model of the quadrotor is written as a strict feedback format,

$$f(x,U)=\begin{cases}\dot{x}_2=\dot{x}_1\\ \dot{x}_2=a_1\cdot x_4\cdot x_6+b_1\cdot U_2\\ \dot{x}_3=x_4\\ \dot{x}_4=a_2\cdot x_2\cdot x_6+b_2\cdot U_3\\ \dot{x}_5=x_6\\ \dot{x}_6=a_3\cdot x_2\cdot x_4+b_3\cdot U_4\\ \dot{x}_7=x_8\\ \dot{x}_8=\cos x_1\cdot\cos x_3\cdot U_1/m-g\\ \dot{x}_9=x_{10}\\ \dot{x}_{10}=U_x\cdot U_1/m\\ \dot{x}_{11}=x_{12}\\ \dot{x}_{12}=U_y\cdot U_1/m\end{cases} \tag{6-18}$$

where $a_1=(I_y-I_z)/I_x$, $b_1=L/I_x$; $a_2=(I_z-I_x)/I_y$, $b_2=L/I_y$; $a_3=(I_x-I_y)/I_z$, $b_3=L/I_z$; $U_x=(\cos\phi\sin\theta\cos\psi+\sin\phi\sin\psi)$, $U_y=(\cos\phi\sin\theta\sin\psi-\sin\phi\cos\psi)$.

The design of the traditional backstepping controller is to use the whole system as an ensemble, and finally use the Lyapunov stability theorem to derive the output of the system. In order to reduce

the difficulty of the design of the system controller, the whole system is divided into 6 relatively independent control channels. Where, position loop contains a highly controlled channel, horizontal *x, y* control channel. And the attitude loop contains the roll control channel, the pitch control channel and the yaw control channel. The backstepping method is applied in each relatively independent channel.

6.4.1 Attitude Loop (Inner Loop) Controller Design

The equations of the rolling channel are as follows:

$$\begin{cases} x_2 = \dot{x}_1 \\ \dot{x}_2 = a_1 \cdot x_4 \cdot x_6 + b_1 \cdot U_2 \end{cases} \tag{6-19}$$

The desired roll angle control variable for the control channel is $\phi_{cmd} = x_{1d}$, the error variable is defined as $e_1 = x_{1d} - x_1$, and derivative of the error variable is set as,

$$\dot{e}_1 = \dot{x}_{1d} - \dot{x}_1 = \dot{x}_{1d} - x_2 \tag{6-20}$$

Introducing the virtual control variable α_1 and defining the error variable,

$$e_2 = x_2 - a_1 \tag{6-21}$$

Implementing Lyapunov function $V_1 = 1/2e_1^2$, and derivative of which as,

$$\dot{V}_1 = e_1 \cdot \dot{e}_1 \tag{6-22}$$

From equation (6-20) and equation (6-21), the following equation can be get,

$$\dot{V}_1 = e_1 \cdot (\dot{x}_{1d} - e_2 - \alpha_1) \tag{6-23}$$

Set $\alpha_1 = \dot{x}_{1d} + c_1 \cdot e_1 (c_1 > 0)$, then,

$$\dot{V}_1 = -c_1 \cdot e_1^2 - e_1 \cdot e_2 \tag{6-24}$$

It can be deduced that

$$\dot{e}_1 = -c_1 \cdot e_1 - e_2 \tag{6-25}$$

$$e_2 = -\dot{e}_1 - c_1 \cdot e_1 \tag{6-26}$$

In the equation (6-24) contains $-e_1 \cdot e_2$ term, the system is not stable. In order to eliminate

$-e_1 \cdot e_2$, Lyapunov function $V_2 = V_1 + 1/2e_2^2$ and derivative of which are implemented as,

$$\dot{V}_2 = \dot{V}_1 + e_2 \cdot \dot{e}_2 \tag{6-27}$$

From equation (6-24) we get,

$$\dot{V}_2 = -c_1 \cdot e_1^2 + e_2 \cdot (\dot{e}_2 - e_1) \tag{6-28}$$

To get the derivatives of equation (6-20) and equation (6-26), and use these derivatives into above equation, we get,

$$\dot{V}_2 = -c_1 \cdot e_1^2 + e_2 \cdot (\dot{x}_2 - c_1\dot{e}_1 - \ddot{x}_{1d} - e_1) \tag{6-29}$$

When $\dot{x}_2 - c_1 \cdot \dot{e}_1 - \ddot{x}_{1d} - e_1 = -c_2 e_2 (c_2 > 0)$, $\dot{V}_2 = -c_1 \cdot e_1^2 - c_2 \cdot e_2^2$, based on Lyapunov stability theory, it could be seen that the system is asymptotically stable.

Using equation (6-19), we could get,

$$U_2 = 1/b_1[\ddot{x}_{1d} + (c_1 + c_2)\dot{e}_1 + (1 + c_1c_2)e_1 - a_1x_4x_6] \tag{6-30}$$

The system itself is a high order system. The introduction of higher order variables makes the system more unstable. In order to better control, the second order variable $\ddot{x}_{1d}$ of the system is omitted. The following equations are also omitted the higher order variables.

$$U_2 = 1/b_1[(c_1 + c_2)\dot{e}_1 + (1 + c_1c_2)e_1 - a_1x_4x_6] \tag{6-31}$$

where $e_1 = x_{1d} - x_1$, $e_2 = -\dot{e}_1 - c_1 \cdot e_1$, $c_1 > 0, c_2 > 0$.

In the same way, we know, when the desired pitch control variable for a pitching control channel is $\theta_{cmd} = x_{3d}$, the pitching channel is

$$U_3 = 1/b_2[(c_3 + c_4)\dot{e}_3 + (1 + c_3c_4)e_3 - a_2x_2x_6] \tag{6-32}$$

where $e_3 = x_{3d} - x_3$, $e_4 = -\dot{e}_3 - c_3 \cdot e_3$, $c_3 > 0, c_4 > 0$.

When the desired yaw control variable for a pitching control channel is $\theta_{cmd} = x_{3d}$, the yaw channel is

$$U_4 = 1/b_3[(c_5 + c_6)\dot{e}_5 + (1 + c_5c_6)e_5 - a_3x_2x_4] \tag{6-33}$$

where $e_5 = x_{5d} - x_5$, $e_6 = -\dot{e}_5 - c_5 \cdot e_5$, $c_5 > 0, c_6 > 0$.

6.4.2 Position Loop (Outer Loop) Controller Design

The equation for the altitude control channel is as follows:

$$\begin{cases} \dot{x}_7 = x_8 \\ \dot{x}_8 = \cos x_1 \cdot \cos x_3 \cdot U_1 / m - g \end{cases} \tag{6-34}$$

The desired altitude control variable for altitude control channel is $z_{cmd} = x_{7d}$, error variables is defined as $e_7 = x_{7d} - x_7$, and derivative of the error variable can be get as,

$$\dot{e}_7 = \dot{x}_{7d} - \dot{x}_7 = \dot{x}_{7d} - x_8 \tag{6-35}$$

Introducing the virtual control variable α_7 and defining the error variable，

$$e_8 = x_8 - \alpha_7 \tag{6-36}$$

Implementing Lyapunov function $V_7 = 1/2e_7^2$, and the derivative of which as,

$$\dot{V}_7 = e_7 \cdot \dot{e}_7 \tag{6-37}$$

From equation (6-35) and equation (6-36), the following equation could be get,

$$\dot{V}_7 = e_7 \cdot (\dot{x}_{7d} - e_8 - \alpha_7) \tag{6-38}$$

Set $\alpha_7 = \dot{x}_{7d} + c_7 \cdot e_7 (c_7 > 0)$, then,

$$\dot{V}_7 = -c_7 \cdot e_7^2 - e_7 \cdot e_8 \tag{6-39}$$

It can be deduced that

$$\dot{e}_7 = -c_7 \cdot e_7 - e_8 \tag{6-40}$$

$$e_8 = -\dot{e}_7 - c_7 \cdot e_7 \tag{6-41}$$

The equation (6-39) contains $-e_7 \cdot e_8$ term, thus the system is not stable. In order to term $-e_7 \cdot e_8$, using Lyapunov function $V_8 = V_7 + 1/2e_8^2$ and get the derivative of it as,

$$\dot{V}_8 = \dot{V}_7 + e_8 \cdot \dot{e}_8 \tag{6-42}$$

From equation (6-39) we get,

$$\dot{V}_8 = -c_7 \cdot e_7^2 + e_8 \cdot (\dot{e}_8 - e_7) \tag{6-43}$$

To get the derivatives of equation (6-35) and equation (6-41), and use these derivatives into

above equation, we get,

$$\dot{V}_8 = -c_7 \cdot e_7^2 + e_8 \cdot (\dot{x}_8 - \ddot{x}_{7d} - c_7 \cdot \dot{e}_7 - e_7) \tag{6-44}$$

When $\dot{x}_8 - c_7 \cdot \dot{e}_7 - \ddot{x}_{7d} - e_7 = -c_8 e_8 (c_8 > 0)$, $\dot{V}_8 = -c_7 \cdot e_7^2 - c_8 \cdot e_8^2$, based on Lyapunov stability theory, it could be seen that the system is asymptotically stable.

Using equation (6-34), we could get,

$$U_1 = m[\ddot{x}_{7d} + (c_7 + c_8)\dot{e}_7 + (1 + c_7 c_8)e_7 + g] / (\cos x_1 \cos x_3) \tag{6-45}$$

The system is a high order system. The introduction of higher order variables makes the system more unstable. In order to better control, the order variable of the system is omitted. The following equations are also omitted the higher order variables.

$$U_1 = m[(c_7 + c_8)\dot{e}_7 + (1 + c_7 c_8)e_7 + g] / (\cos x_1 \cos x_3) \tag{6-46}$$

where $e_7 = x_{7d} - x_7$, $e_8 = -\dot{e}_7 - c_7 \cdot e_7$, $c_7 > 0, c_8 > 0$

In the same way, we know, when the desired horizontal axis x variable for a pitching control channel is $x_{cmd} = x_{9d}$, it will be get,

$$U_x = m[(c_9 + c_{10})\dot{e}_9 + (1 + c_9 c_{10})e_9] / U_1 \tag{6-47}$$

where $e_9 = x_{9d} - x_9$, $e_{10} = -\dot{e}_9 - c_9 \cdot e_9$, $c_9 > 0, c_{10} > 0$

When the desired horizontal axis y variable for a pitching control channel is $x_{cmd} = x_{11d}$, it will be get,

$$U_y = m[(c_{11} + c_{12})\dot{e}_{11} + (1 + c_{11} c_{12})e_{11}] / U_1 \tag{6-48}$$

where $e_{11} = x_{11d} - x_{11}$, $e_{12} = -\dot{e}_{11} - c_{11} \cdot e_{11}$, $c_{11} > 0, c_{12} > 0$

6.4.3 Nonlinear Constraint Conditions

From the mathematical model of the quadrotor UAV, we can know that the altitude control channel and yaw control channel of the aircraft are two opposing channels. The horizontal control channel is closely related to the roll control channel and the pitch control channel. To be precise, the horizontal control channel is a typical underactuated control, which depends on the control of roll angle and pitch angle.

Therefore, in the design of double closed loop controller, the desired roll angle and desired pitch angle of the attitude loop depend on the inverse solution of the horizontal control. That is, the so-called nonlinear constraint conditions.

$$\begin{cases} U_x = \cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi \\ U_y = \cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi \end{cases} \tag{6-49}$$

The inverse solution can be obtained，

$$\begin{cases} \phi_{cmd} = \arcsin(U_x \sin\psi - U_y \cos\psi) \\ \theta_{cmd} = \arcsin[\dfrac{U_x \cos\psi + U_y \sin\psi}{\cos\phi_{cmd}}] \end{cases} \tag{6-50}$$

In view of the characteristics of fixed point flight, assuming that the desired yaw angle is $\psi_{cmd} = 0rad$, the actual yaw angle will change at a small angle near the desired value. Therefore, the influence of yaw angle on the stability of the underactuated horizontal channel can be neglected, thus by simplifying the nonlinear constraints, new inverse solution can be get as:

$$\begin{cases} \phi_{cmd} = \arcsin(-U_y) \\ \theta_{cmd} = \arcsin[\dfrac{U_x}{\cos\phi_{cmd}}] \end{cases} \tag{6-51}$$

The Simulink module design is shown in Figure 6.3. There are three modules: command signals module, backstepping controller module, and quadrotor UAV system module.

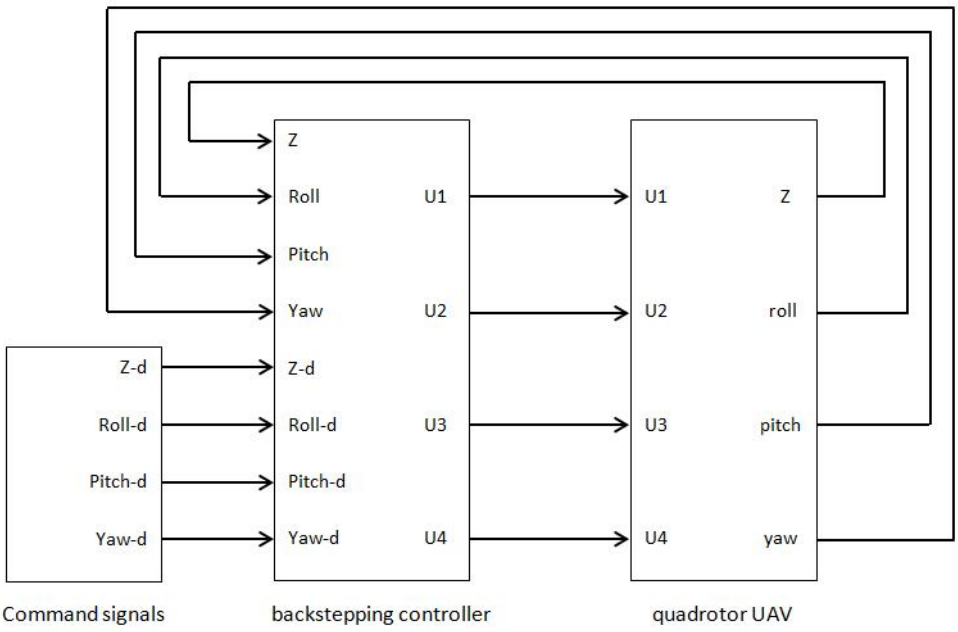


Figure 6.3 Quadrotor Simulink module design

6.5 Summary

Based on the four rotor mathematical models established in Chapter 2, considering the drawbacks of linearized control methods, the nonlinear controller, backstepping control algorithm, is designed in this chapter for the control or quadrotor UAV. The principle of backstepping control algorithm, the design of the quadrotor UAV controller based on the backstepping control algorithm and the simplification of the nonlinear constraint conditions are introduced. The backstepping controller will also be used on the path tracking as well as payload dropping of Qball-X4 UAV in next chapter.

Chapter 7

Trajectory Tracking and Payload Dropping of Qball-X4 with Backstepping Controller

7.1 Introduction

In this thesis, a GS-PID adaptive tracking controller is designed for the quadrotor unmanned aerial vehicle (UAV), which achieves the tracking control of the three directions of the aircraft and the attitude angle. As indicated in the Chapter 5, in order to obtain the best stability and performance of Qball-X4 under payload drop scenario, the switching action from one set of pre-tuned PID gains to another set plays a vital role at the moment of releasing the payload. If this transient (switching) time is held long (more than one second) it can cause the Qball-X4 to overshoot abruptly and even a crash.

In order to improve the output characteristics of the system controller, the flight controller is designed by the robust control strategy based on backstepping controller, which can effectively suppress the unknown time-varying disturbance in the system dynamics model, and make the system control input smooth. In fact, the load dropping control is essentially similar to the control problem of a helicopter with unknown mass.

In this chapter, the trajectory tracking simulation and payload dropping simulation of Qball-X4 UAV with backstepping controller will be presented. Some of the basic parameters for QballX4 quadrotor UAV model are shown in the Table 5-1 in the Chapter 5 of this thesis. The Simulink model for trajectory tracking and payload dropping control of quadrotor helicopter using backstepping controller is shown in Figure 7.1.

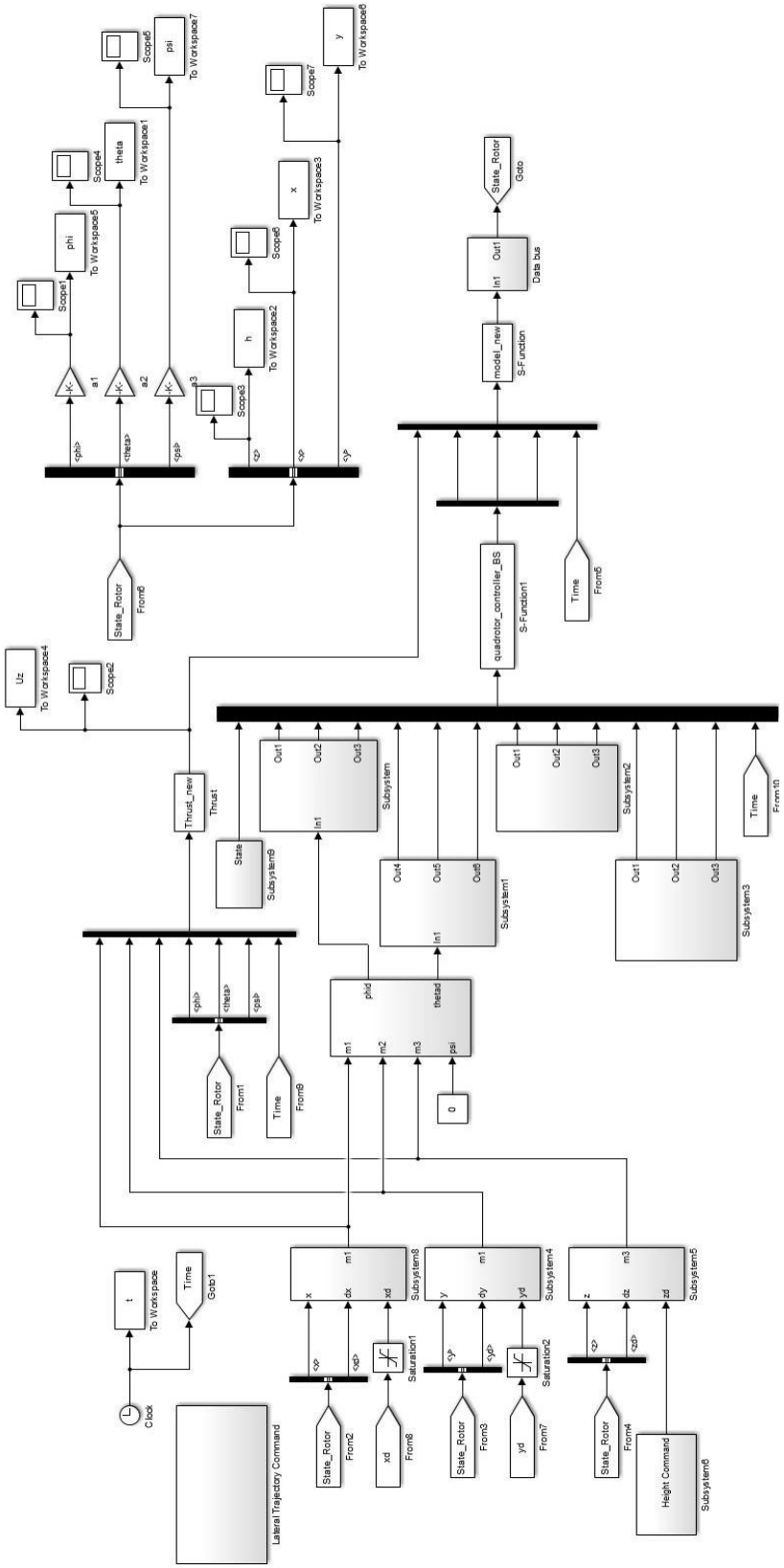


Figure 7.1 Backstepping controller Simulink model

7.2 Trajectory Tracking with Backstepping Controller

In order to verify the effectiveness of the backstepping controller, the quadrotor UAV, Qball-X4 dynamics model is used for numerical control simulation.

In order to avoid the overshoot of the output amplitude of the controller at the beginning stage, the soft start is added in the reference trajectory, and the reference trajectory is closer to the standard sine and cosine curve as the time increases. A desired trajectory for the trajectory tracking control is set as,

$$x_d = \sin t(1 - e^{-t^3})\ (m)$$

$$y_d = \cos t(1 - e^{-t^3})(m)$$

$$z_d = 1 \cdot (1 - e^{-t^3})(m)$$

and desired yaw angle trajectory is

$$\psi_d = 60 \cdot \left(1 - e^{-t^3}\right) \text{deg} = 1.047 \cdot \left(1 - e^{-t^3}\right) \text{rad}.$$

According to the design process of the backstepping controller, because the controller uses the coupling relationship between the roll and pitch angle and the horizontal position to control the position of the aircraft, the roll angle and the pitch angle will fluctuate in the course of tracking the predetermined trajectory.

The attenuation term is used to ensure that the derivative of the desired trajectory is zero at zero time. In a very short period of time, the expected output is attenuated:

$$x_d = \sin t(m),\ y_d = \cos t(m),\ z_d = 1(m),\ \psi_d = 1.047 rad$$

Compared with the results of simulation tests, a larger $c_i (i = 1,2,3,4,5,6,7,8,9,10,11,12)$ is likely to lead to a smaller overshoot. At the same time, the overshoot of position will also increase. Considering the safety of actual flight, take

$$c_i = 1.5, c_j = 1.25, (i = 1,2,3,4,5,6), (j = 7,8,9,10,11,12).$$

Figure 7.2 is the three-dimensional simulation of the reference trajectory and actual trajectory of a quadrotor UAV. From the figure, we can see that the UAV rises to a height of 1 m along the spiral track, and then does a circular motion of 1 m radius. Since the aircraft has initial error in the beginning of control, there is a certain deviation between the two tracks, and the two tracks

coincide over a period of time due to the function of the controller. This proves the effectiveness and stability of the backstepping control algorithm in the quadrotor UAV model.

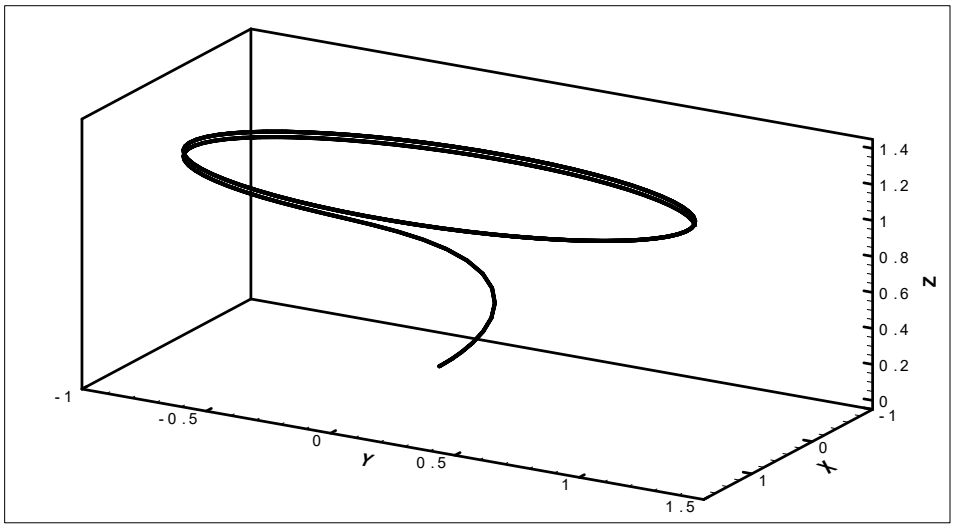


Figure 7.2 Quadrotor unmanned aerial vehicle actual trajectory

Figure 7.3 shows the simulation curve of the horizontal displacement of the quadrotor unmanned aerial vehicle. The vehicle tracks the sine curve of the amplitude of 1 m in the X direction. The simulation curve shows that the tracking error is 0.1 m at the initial time, and the tracking error tends to zero at about 6 s, and the tracking error is always kept from -0.4 m to +0.2 m during the whole control process within the range of the interval. Figure 7.4 shows the simulation curve of the horizontal displacement of the quadrotor unmanned aerial vehicle. The aircraft tracks the cosine curve of the amplitude of 1 m in the Y direction. It is known from the diagram that the tracking error is always within the range of + 0.5 m, and the aircraft tracks the predetermined reference trajectory around 6 s. Figure 7.5 shows the simulation curve of the altitude of the aircraft. According to a predetermined reference trajectory to 1 m height, the height error is close to 0 m at 4 s, and there is no negative value in the altitude direction.

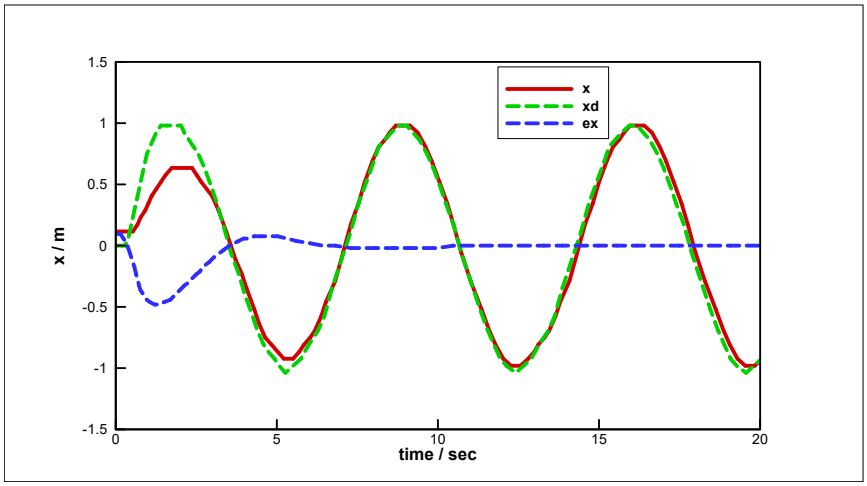


Figure 7.3 Position x under trajectory tracking, desired trajectory x_d and its error curve

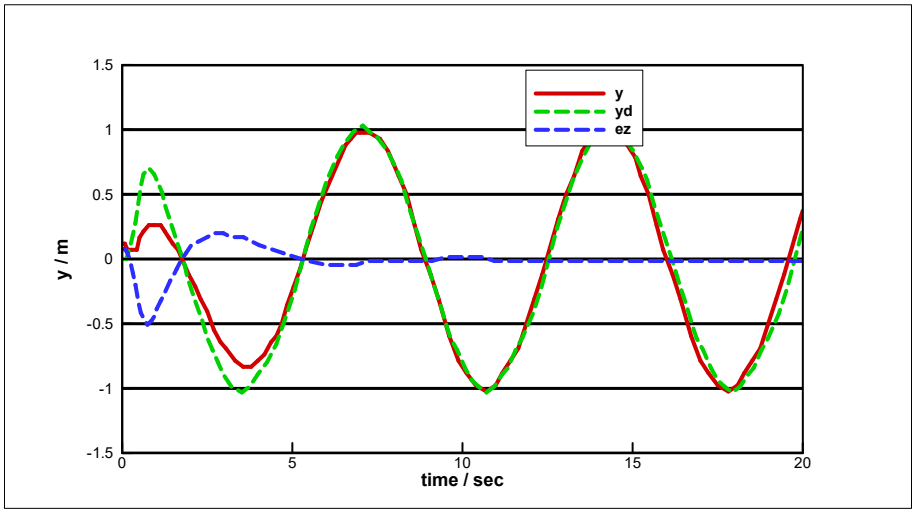


Figure 7.4 Position y under trajectory tracking, desired trajectory y_d and its error curve

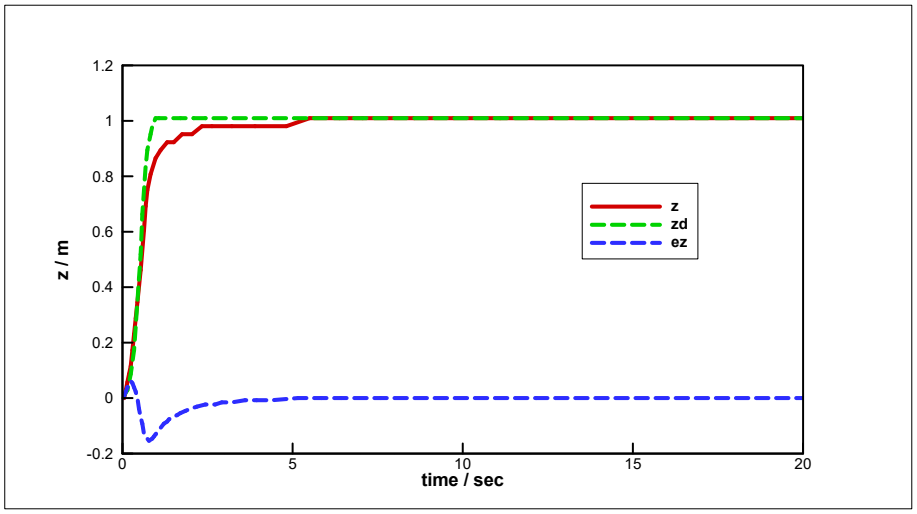
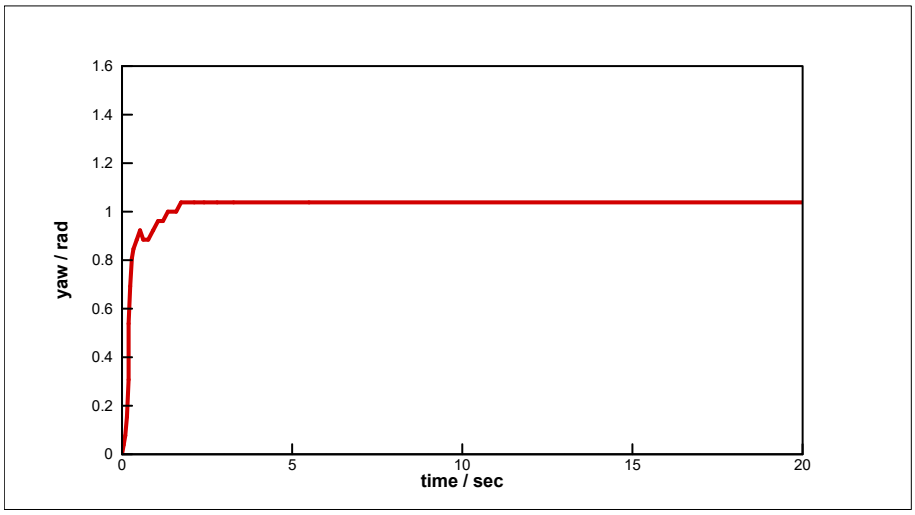


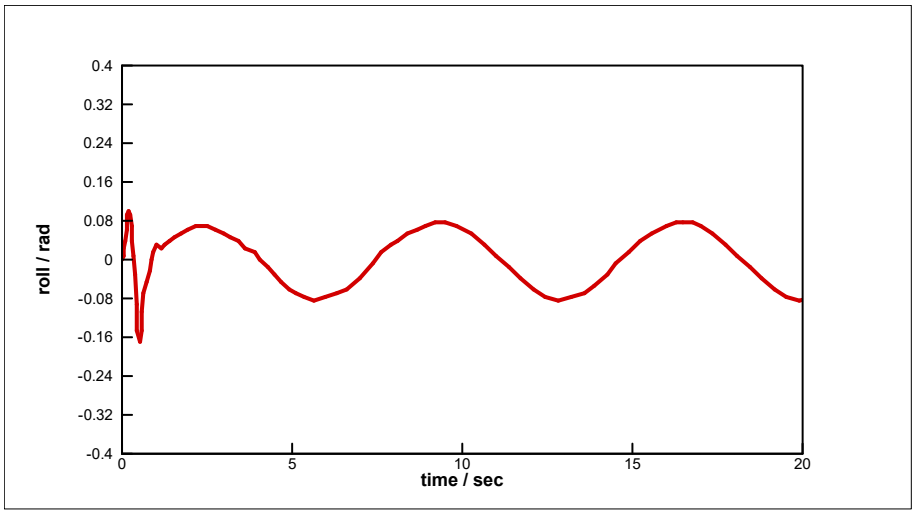
Figure 7.5 Position z under trajectory tracking, desired trajectory z_d and its error curve

Figure 7.6 shows the simulation curve of three attitude angles of the quadrotor unmanned aerial

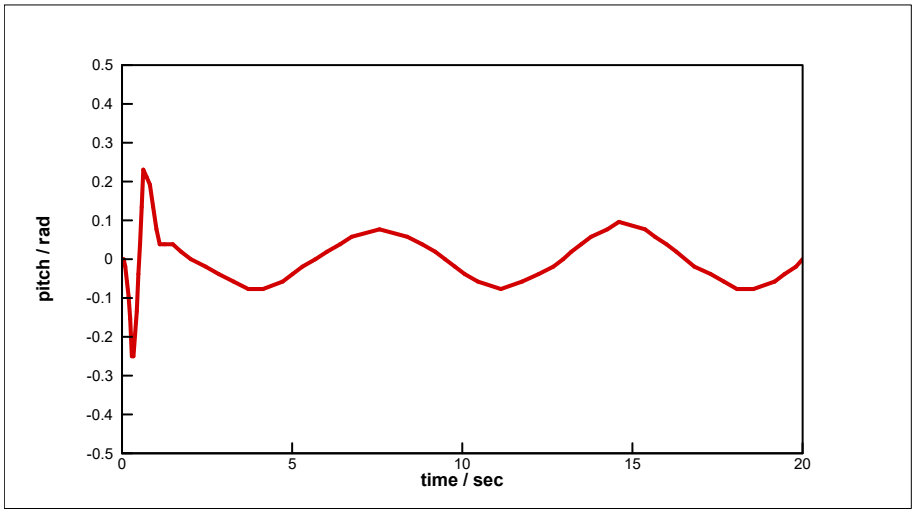
vehicle. The validity of attitude channel control is proved. It is known from the first picture that the aircraft reaches a specified yaw angle of 1.047 rad at about 2 s. According to the design process of the controller U2 (T) and the U3 (T), it is known that because the controller uses the coupling relationship between the roll angle and the pitch angle and the horizontal position to control the position of the aircraft, the roll angle and the pitch angle will fluctuate in the course of tracking the predetermined trajectory, but the two attitude angles can be seen from the simulation curve is kept within the range of 0.589 rad.



(a) Yaw angle ψ



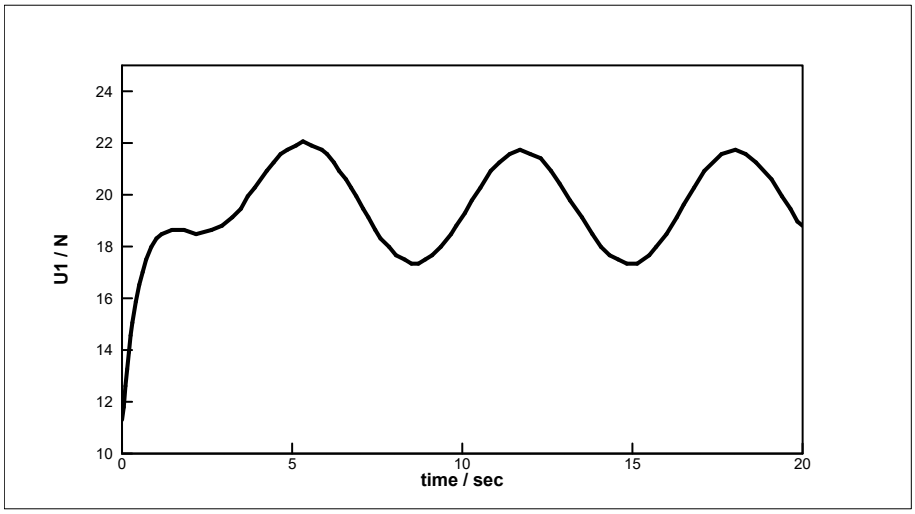
(b) Roll angle ϕ



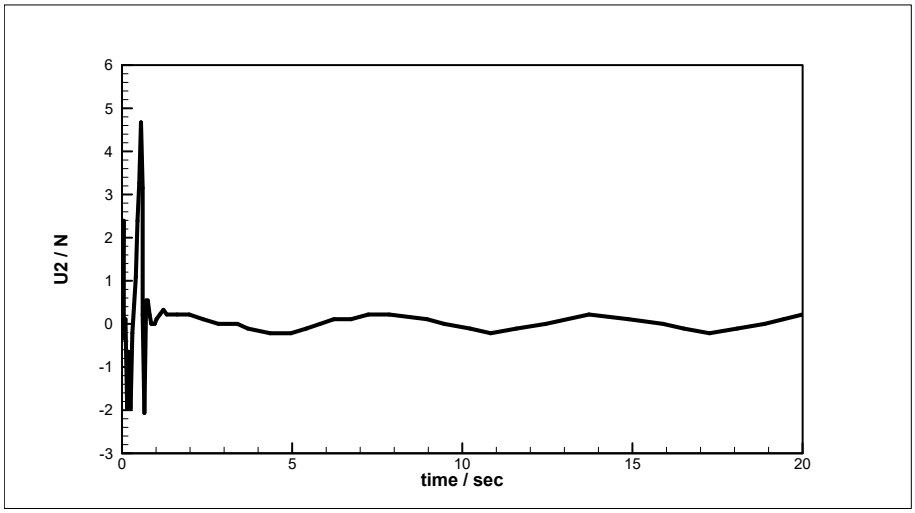
(c) Pitch angle θ

Figure 7.6 Yaw angle ψ, rolling angle ϕ and pitching angle θ under trajectory tracking

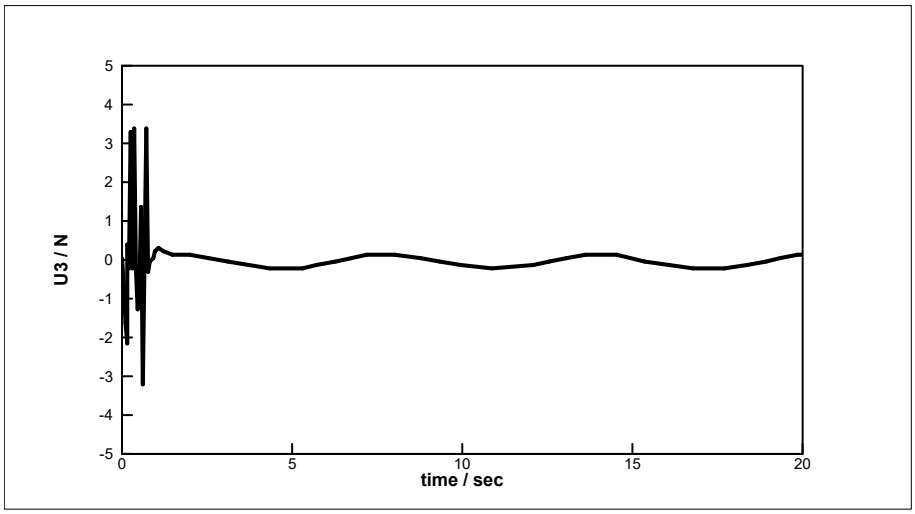
Figures 7.7 show the simulation curves of the controller output U1 (T), U2 (T), U3 (T) and U4 (T). As the diagram shows, as the quadrotor aircraft is affected by the external disturbance D3 (T) in a high direction, the ascending force U1 (T) generates a certain amplitude wave after stability, but always surrounds the gravity oscillation of the aircraft. Due to the soft start of the predetermined trajectory XD (T) and yd (T), and the derivative of the soft start will produce a certain degree of high frequency phenomenon, so the controller output U2 (T) and U3 (T) will have high frequency at the beginning stage. Because of the external disturbance in the system dynamics model, there is also a wave phenomenon in the simulation curve.



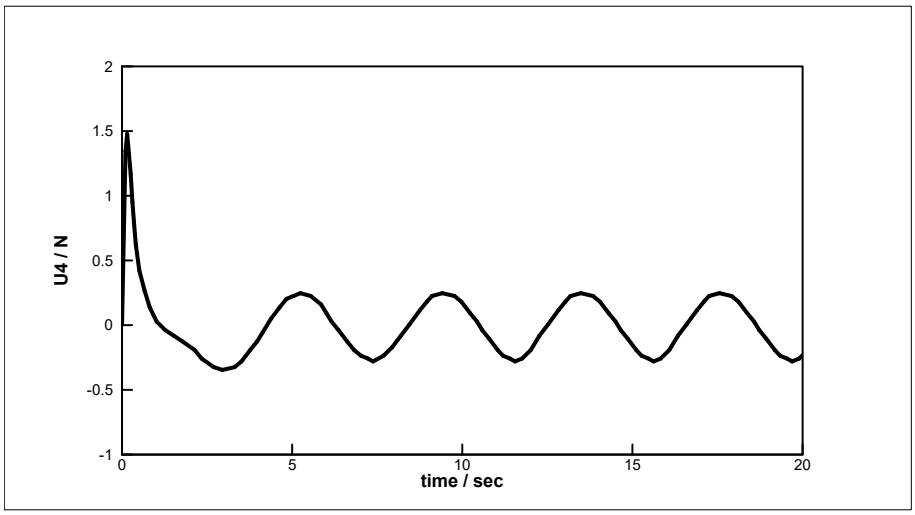
(a) Controller output U1 (T)



(b) Controller output U2 (T)



(c) Controller output U3 (T)

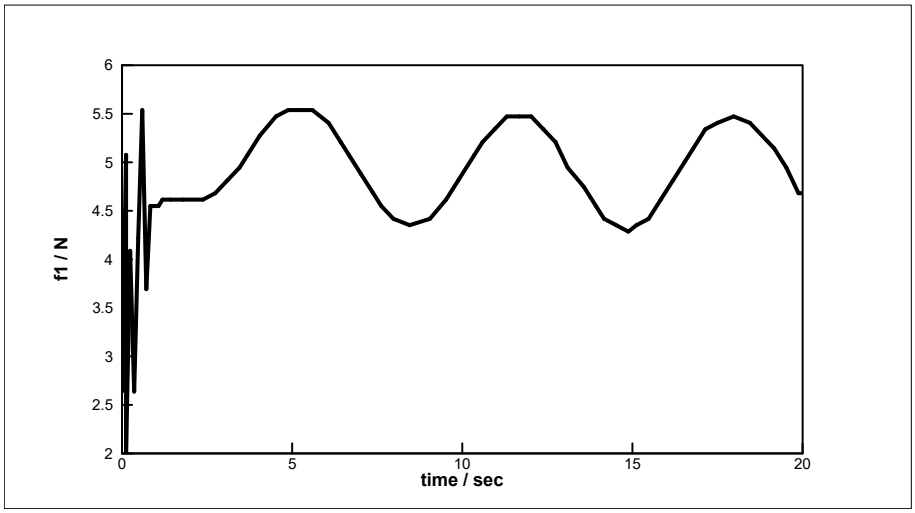


(d) Controller output U4 (T)

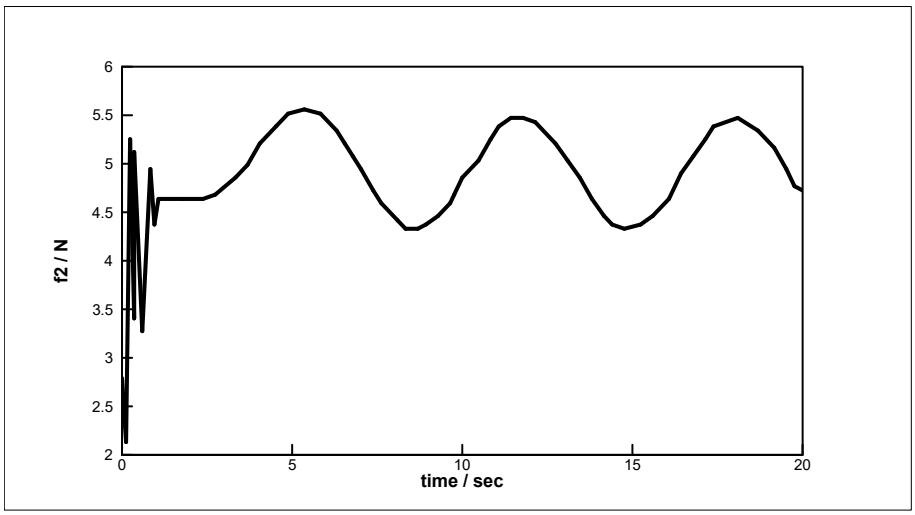
Figure 7.7 Controller output U1 (T) , U2 (T), U3 (T) and U4 (T)

In order to further verify the practicability of the controller proposed in this thesis, the controller

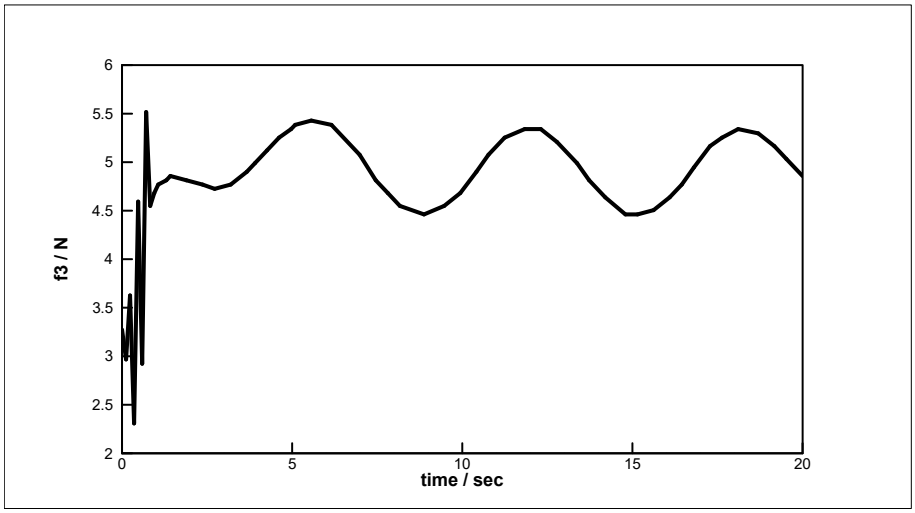
u is used in this thesis. The mapping relationship between I (T), I = 1... 4 and four fi propeller force fi (T), I = 1... 4 is calculated. From Figure 7.8, the propeller force generated by the propeller is kept within 6 N, which is consistent with the output range of the quadrotor UAV DC motor. Figure 7.9 is the online estimation curve of the unknown parameters of the controller, which is bounded by graph $S_i(T)$, $I = 1... 4$ and can be stabilized at about 2 seconds.



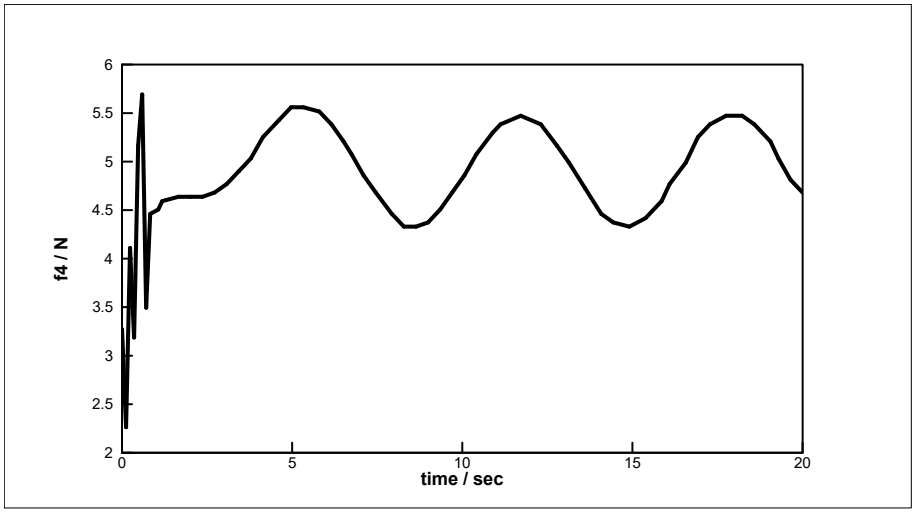
(a) Propeller force F_1



(b) Propeller force F_2

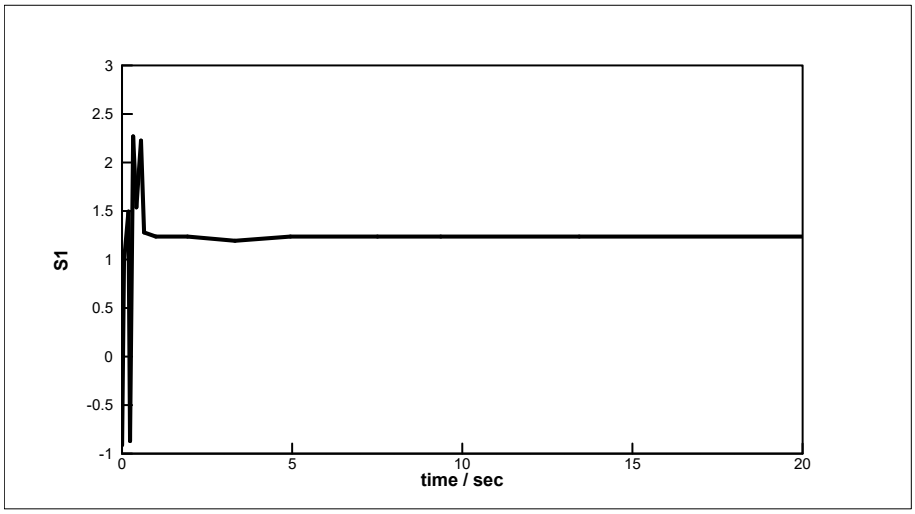


(c)Propeller force F_3

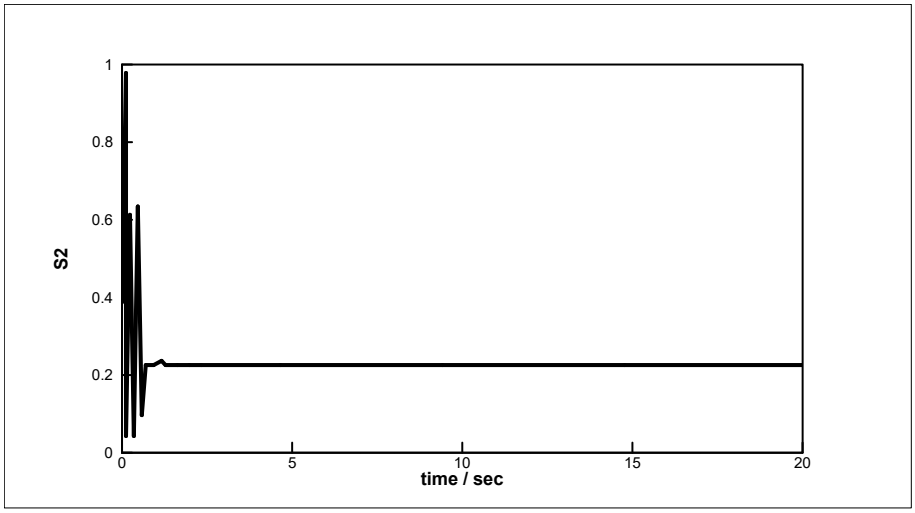


(d) Propeller force F_4

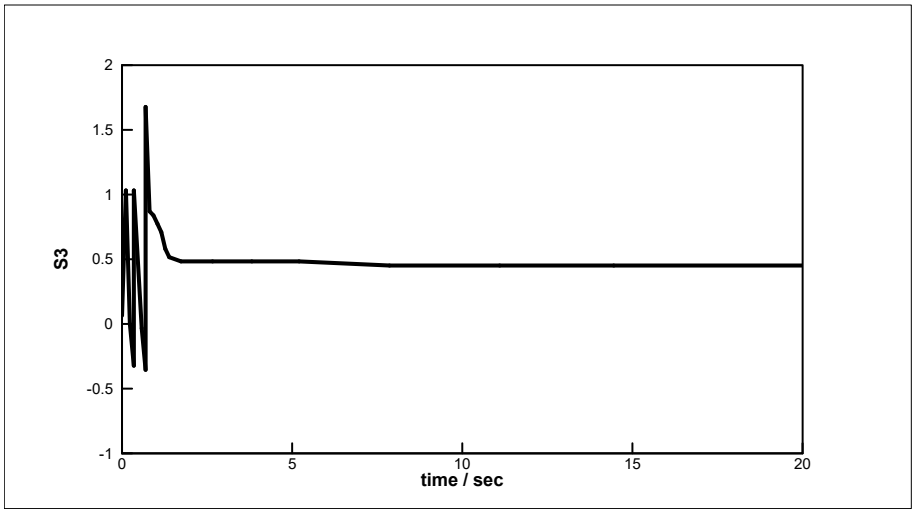
Figure 7.8 Propeller force F_1, F_2, F_3, F_4



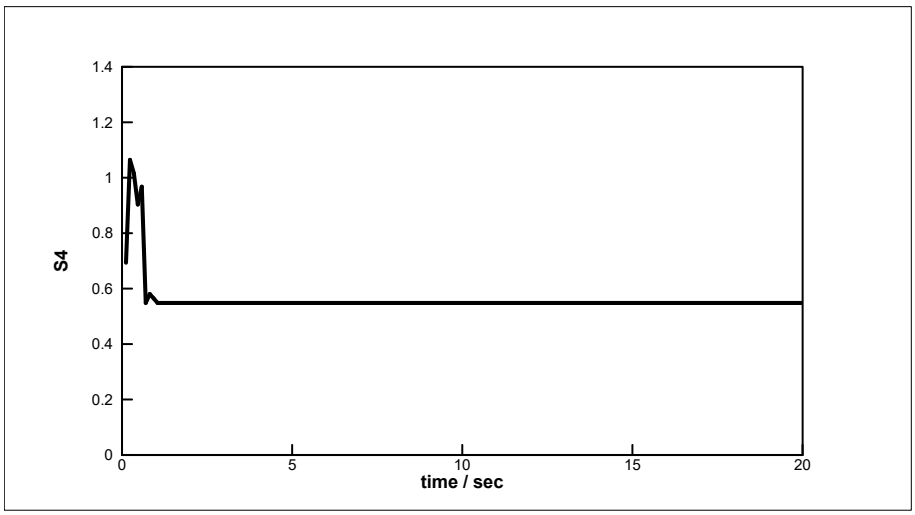
(a) Estimation parameters S_1



(b) Estimation parameters S_2



(c) Estimation parameters S_3



(d) Estimation parameters S_4

Figure 7.9 Estimation parameters

The results show that the nonlinear tracking controller designed in this chapter still can track the predetermined reference trajectory in the three directions of the quadrotor unmanned aerial vehicle (UAV) in the case of modeling uncertainty, and can effectively suppress the roll angle and pitch angle during the flight. From the control input curve, it can be seen that the propulsion and propeller speed of the four propellers of the aircraft are in a reasonable range. Therefore, the controller has good practicability and can effectively improve the safety performance of the UAV operation.

It could be seen from the results that, the tracking error of the whole system is relatively small, the tracking accuracy is high, and the tracking effect is acceptable. For the safety, in the process of the inverse solution to the motor PWM, four virtual inputs are given a first order low pass filter link $10/(s+10)$. As can be seen from above results, the trajectory tracking of the quadrotor UAV in flight is smooth and meets the requirements of the actual flight, which lays a theoretical foundation for the application of the next step controller to the actual flight platform for payload dropping.

7.3 Payload Dropping with Backstepping Controller

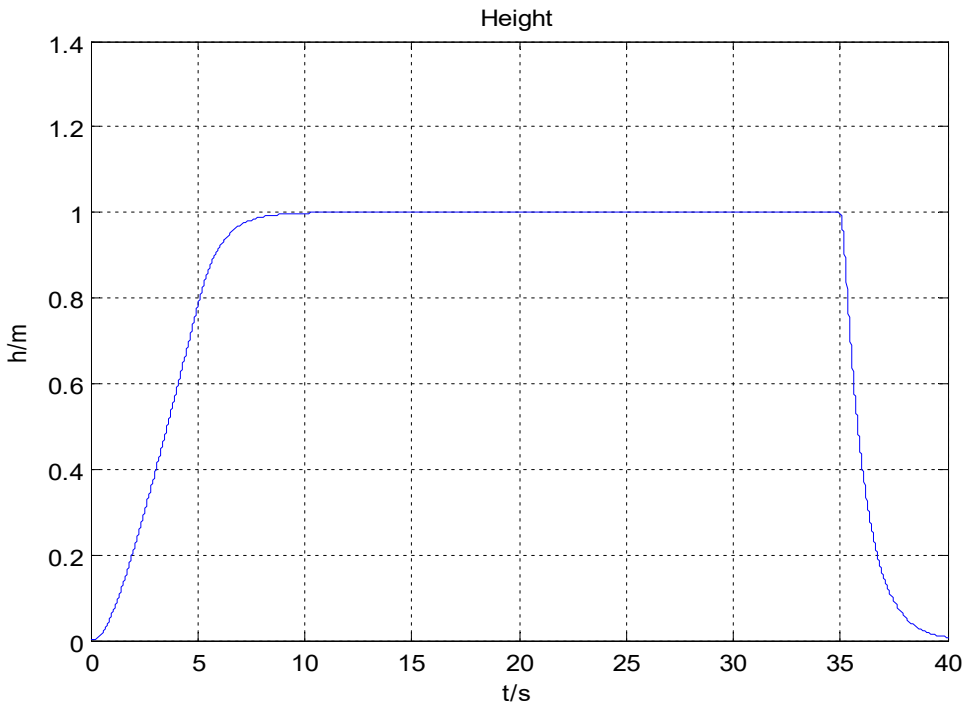
In order to reduce the overshoot at the instant of payload drop, the backstepping controller was used for the control of Qball-X4. Simulation experiment and flight test were all carried out for the payload dropping control with backstepping controller in the NAVL.

7.3.1 Simulation Experimental Results Analysis

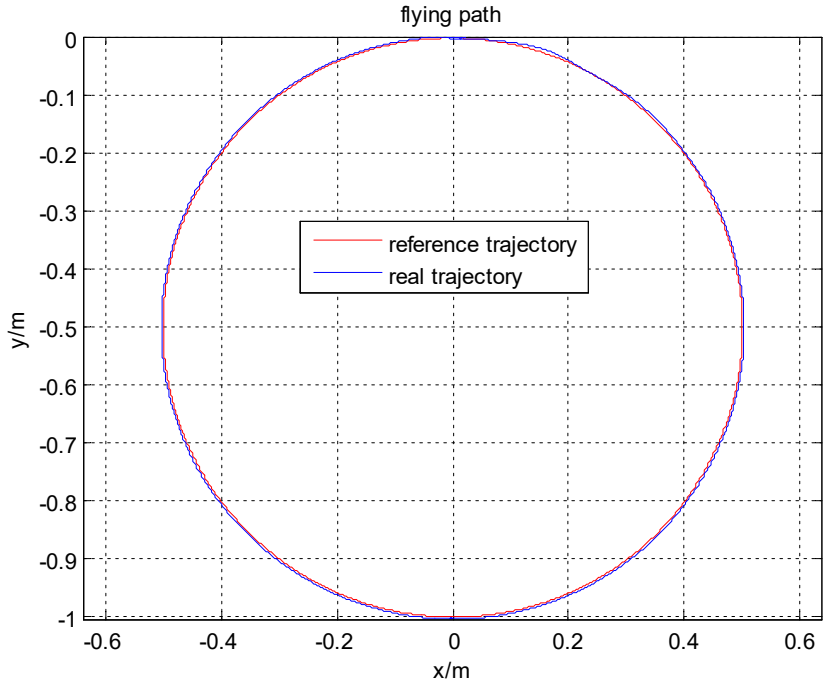
Similar to the simulation and experimental works taken in Chapter 5, a payload dropping control with backstepping controller is firstly performed by the simulation system. The same flight state for Qball-X4 helicopter is simulated in this chapter. The initial state of the Qball-X4 unmanned aerial vehicle is set. Altitude z is 0 m, roll angle ϕ is 0 degree, pitch angle θ is 0 degree, and yaw angle ψ is 0 degree. And while the Qball-X4 helicopter finishes climbing and is in a rotating trajectory, the desired altitude z_d is 1m and the rotating diameter is 1m. The payload with 300g is dropping at the 20 second. Simulation time is set to 45 seconds. This study focuses on the flight stability of the Qball-X4 helicopter when its payload is dropped.

Figure 7.10 to Figure 7.12 show the flying path, flying attitude and the total thrust of Qball-X4 when payload is dropping with the backstepping controller. It could be seen from simulation results

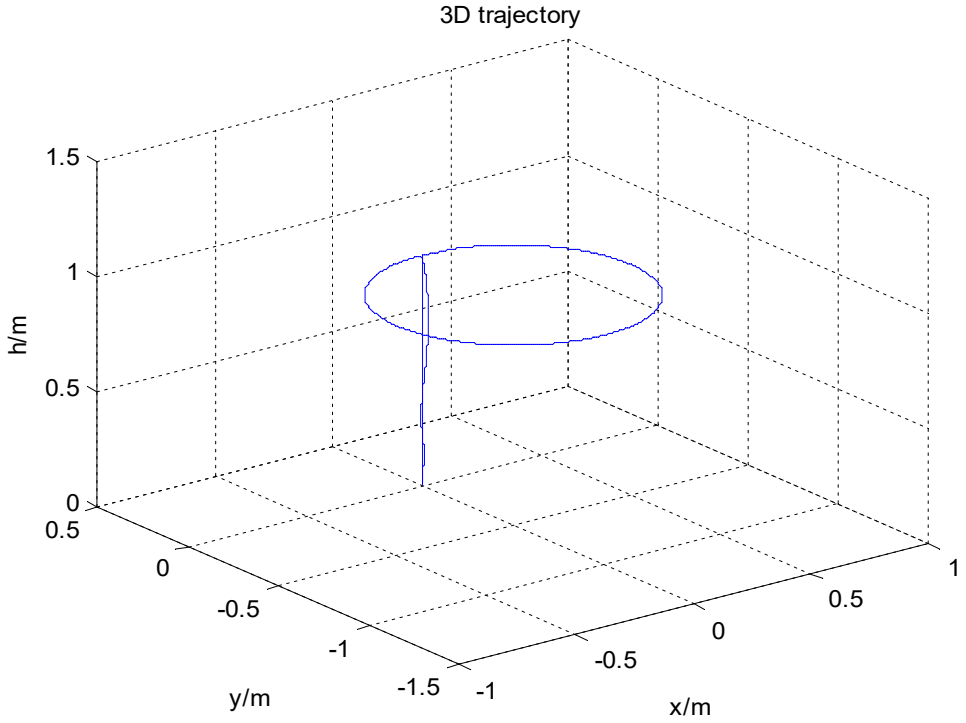
that the backstepping controller was capable of keeping the desired height and trajectory. It could also be obviously seen from Figure 7.12 that total thrust abrupt change of Qball-X4 when payload drops.



(a) The altitude of Qball-X4



(b) The X-Y position of Qball-X4



(c) The 3D trajectory of Qball-X4

Figure 7.10 The flying path of Qball-X4 for payload dropping with backstepping controller

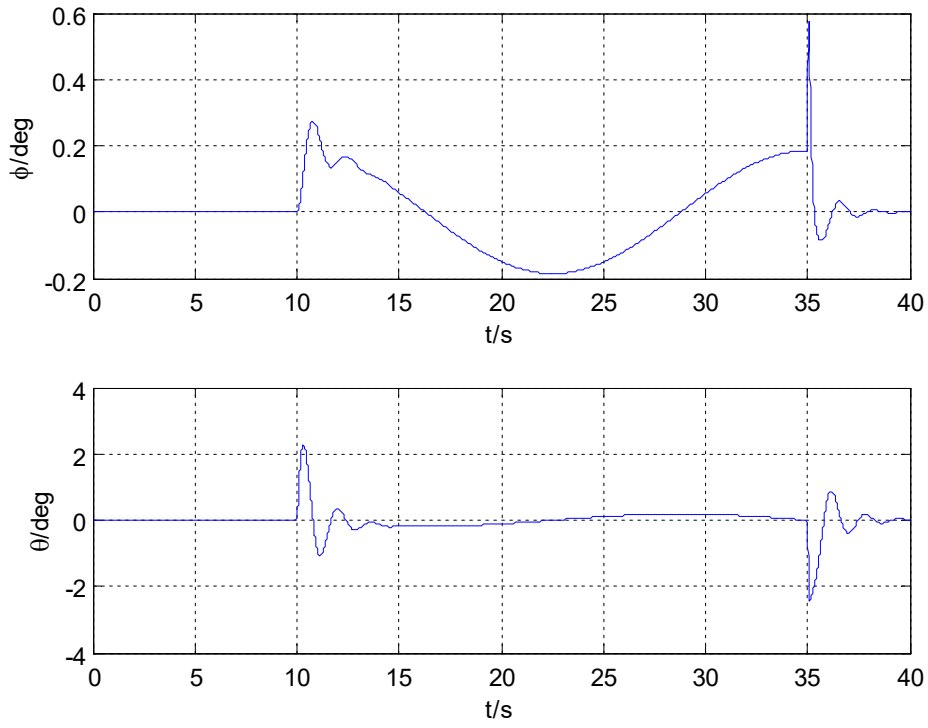


Figure 7.11 The flying attitude of Qball-X4 for payload dropping with backstepping controller

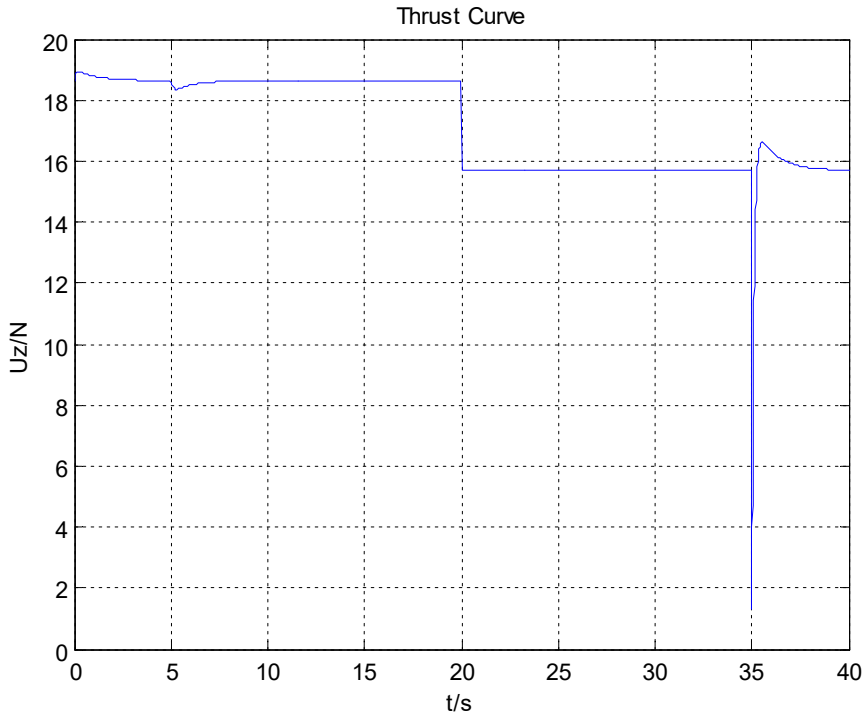


Figure 7.12 The total thrust of Qball-X4 with backstepping controller

7.3.2 Flight Test Experimental Results Analysis

In this section, the above discussed backstepping control techniques is implemented real- time on the Qball-X4 quadrotor helicopter for performance analysis with payload dropping application. The flight test was also carried out in the NAVL. As indicated in Chapter 3, the Qball-X4 quadrotor helicopter available at the NAVL works based on indoor environment.

Similarly, before the flight test, the control platform was analyzed and adjusted based on the simulation platform. In consideration of the test safety and stability of the platform, certain control parameters have been altered from the ones implemented in the simulation as in the experiment in Chapter 5.

The desired altitude was changed into 0.7 m and the rotating diameter remains 1m. Flight time is changed to 100 seconds. When the Qball-X4 helicopter is hovering on its rotating trajectory, the payload with 300g is dropping at the 50 second.

The output result of the altitude position of the Qball-X4 helicopter payload dropping control with backstepping controller is shown in Figure 7.13.

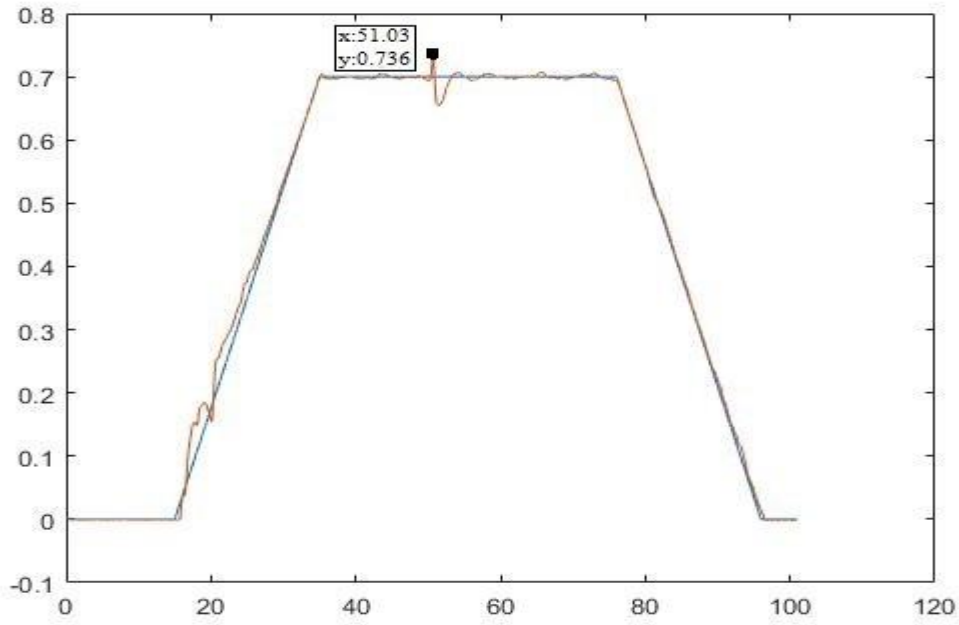


Figure 7.13 Payload dropping experiment with backstepping controller altitude output result

It could be seen from Figure 7.13 that the Qball-X4 quadrotor with backstepping control method comes with a small overshoot and an overall satisfying performance. With backstepping control method implied, the control performance of quadrotor helicopter and the ability of interference suppression have been successfully achieved. Compared with the results in previous chapters, it could be seen that with backstepping control method, the overshoot is smaller, and the adjustment time is shorter than those of a Qball-X4 with GS-PID controller. The overshoot of Qball-X4 helicopter at the payload dropping is about 6.5% and the adjustment time is about 5 seconds with the backstepping controller. With the comparison of these control methods, the backstepping controller is the best controller for the control of quadrotor helicopter.

7.4 Summary

In order to obtain a better stability and performance of the quadrotor under payload drop scenario, the backstepping controller is used to design the height and position controller of Qball-X4. In this chapter, the proposed controller, backstepping controller presented in Chapter 6 was implemented in the Qball-X4 quadrotor helicopter. The path tracking control simulation and payload dropping simulation of Qball-X4 helicopter with backstepping control were presented.

Compared with the GS-PID controller, the superiority of backstepping controller is proved by the simulation experimental result and flight test experimental output results.

As stated previously, the focus of this work is to maintain the height of quadrotor helicopter for payload dropping applications with objective to reduce the overshoot of a quadrotor helicopter at the moment of payload dropping. Among all the controllers studied in this thesis, the backstepping controller is suggested as the best choice for the scenario of payload drop in view of its good performance.

Chapter 8

Conclusions and Future Works

8.1 Conclusions

Two useful control techniques (GS-PID, and backstepping controller) were investigated and applied to an unmanned quadrotor helicopter in this thesis for a practical and important scenario of using a quadrotor helicopter for trajectory tracking with a payload dropping. The research status and key technologies of the quadrotor UAV are firstly reviewed. The system characteristics of the quadrotor with multivariable, nonlinear, underactuated and strongly symmetric systems are analyzed. The two control methods were successfully implemented on the Qball-X4 quadrotor helicopter at the NAVL of Concordia University, and were analyzed and verified by simulation experiments and flight tests.

Using MATLAB software to build the quadrotor helicopter simulation platform, the flight simulation experiments of the quadrotor UAV is carried out in this thesis. Instead of single PID, the GS-PID was carried out in dealing with the abrupt mass variation over the period of flight. The GS-PID control method reduces time and the amplitude of the UAV to recover to the steady state. The simulation results show that the GS-PID control structure for the quadrotor UAV could increases the performance of the quadrotor UAV in tracking the desired trajectory and increases reliability of the UAV. Based on the proposed pre-tuned PID controller design, advantages, for example, robust properties, easy design and implementation of PID controller can be employed and the drawback of lack of tuning of PID controller gains is avoided. The simulation results also verify the effectiveness of the proposed backstepping controller. It is found that the backstepping controller is able to control the quadrotor in the presence of payload dropping with a better performance than the GS-PID controller. The backstepping controller reduces time and the

amplitude of the quadrotor UAV to recover to the steady state.

The simulation experiments and flight tests of payload dropping was carried out based on the Qball-X4 flying platform in the NAVL. Compared all the controllers in the tests, it could be found out that all the controllers (PID, GS-PID, backstepping) could maintain the desired height after the transient period induced by payload dropping. However, according to the flight test experiments taken, backstepping controller could noticeably improve system's reaction and overall performance. Among all the controllers, the backstepping controller is suggested as the best choice for the scenario of payload dropping in view of its simplicity and good performance.

8.2 Future Works

It must be pointed out that it is difficult to accurately model the quadrotor unmanned aerial vehicle under complicated circumstances. The control design of this thesis only investigated the problem of aircraft control when there is a payload dropping in certain smooth flight trajectory using GS-PID controller and backstepping controller. As it has mentioned, the system should maintain acceptable performance with partial actuator faults using fault tolerant control strategies for the control of UAV. Therefore, in the future, with adopting the backstepping control approach and utilizing the quadrotor mathematical model, a fault tolerant control law could be further designed and evaluated by single, triple or even quadruple partial actuator faults.

References

[1] Y. M. Zhang, A. Chamseddine, C. A. Rabbath, B. W. Gordon, C.-Y. Su, S. Rakheja, C. Fulford, J. Apkarian, and P. Gosselin, Development of advanced FDD and FTC techniques with application to an unmanned quadrotor helicopter testbed, J. Franklin Inst. 350(9) (2013) 2396–2422.

[2] F. Lin, K. Z. Y. Ang, F. Wang, B. M. Chen, T. H. Lee, B. Yang, M. Dong, X. Dong, J. Cui, S. K. Phang, B. Wang, D. Luo, K. Peng, G. Cai, S. Zhao, M. Yin, and K. Li, Development of an unmanned coaxial rotorcraft for the DARPA UAV Forge challenge, Unmanned Syst. 1(2) (2013) 211–245.

[3] J. Keller, D. Thakur, V. Dobrokhodov, K. Jones, M. Pivtoraiko, J. Gallier, I. Kaminer, and V. Kumar, A computationally efficient approach to trajectory management for coordinated aerial surveillance, Unmanned Syst. 1(1) (2013) 59–74.

[4] P. C. Niedfeldt, B. T. Carroll, J. A. Howard, R. W. Beard, B. S. Morse, and S. Pledgie, Enhanced UAS surveillance using a video utility metric, Unmanned Syst. 1(2) (2013) 277–296.

[5] C. C. Haddal and J. Gertler, Homeland security: Unmanned aerial vehicles and border surveillance, CRS Report for Congress, Congressional Research Service (2010).

[6] J. Hu, J. Xu, and L. Xie, Cooperative search and exploration in robotic networks, Unmanned Syst. 1(1) (2013) 121–142.

[7] M. K. Habib and Y. Baudoin, Robot-assisted risky intervention, search, rescue and environmental surveillance, Int. J. Adv. Robot. Syst. 7(1) (2010) 1–8.

[8] T. Skrzypietz, Unmanned aircraft systems for civilian missions, BIGS Policy Paper No. 1 (2012).

[9] R. W. Beard, T. W. McLain, D. B. Nelson, D. Kingston, and D. Johanson, Decentralized cooperative aerial surveillance using fixed-wing miniature UAVs, Proc. IEEE 94(7) (2006) 1306–1324.

[10] D. W. Casbeer, D. B. Kingston, R. W. Beard, and T. W. McLain, Cooperative forest fire surveillance using a team of small unmanned air vehicles, Int. J. Syst. Sci. 37(6) (2006) 351–360.

[11] H. Dieter, Z. Werner, S. Gunter, and S. Peter, Monitoring of gas pipelines: A civil UAV application, Aircraft Eng. Aerosp. Technol. 77(5) (2005) 352–360.

[12] S. Magazine (2007), Available at http://findarticles.com/p/ articles/mi m0OXU/is 5 62/ai n27236124/ (Accessed on 3 November 2012)

[13] B. Erginer and E. Altug, Modeling and PD control of a quadrotor VTOL vehicle, IEEE Intelligent Vehicles Symp. (2007), pp. 894–899.

[14] A. Wahyudie, T. B. Susilo, and H. Noura, Robust PID controller for quad-rotors, J. Unmanned Syst. Technol. 1(1) (2013) 14–19.

[15] K. Oner, E. Cetinsoy, M. Unel, M. Aksit, I. Kandemir and K. Gulez, Dynamic model and control of a new quadrotor UAV with tilt-wing mechanism, World Academy of Science, Engineering and Technology (2008).

[16] M. O. Efe, Robust low altitude behavior control of a quadrotor rotorcraft through sliding modes, Mediterranean Conf. on Control and Automation (2007).

[17] H. Bouadi, M. Bouchoucha, and M. Tadjine, Sliding mode control based on backstepping approach for an UAV type-quadrotor, World Academy of Science, Engineering and Technology (2007).

[18] M. Abdolhosseini, Y. M. Zhang and C. A. Rabbath, An efficient model predictive control scheme for an unmanned quadrotor helicopter, J. Int. Robot. Syst. 70(1) (2013) 27–38.

[19] T. Madani and A. Benallegue, Backstepping control for a quadrotor helicopter. Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, 9-15 October 2006, 3255-3260.

[20] De Bothezat, Helicopter: development history, photos, technical data. http://www.aviastar.org/ helicopters_eng/bothezat.php

[21] P. McKerrow, Modelling the draganflyer four-rotor helicopter. In Proceedings of the 2004 IEEE International Conference Robotics and Automation, 2004.

[22] R. K. Arning and S. Sassen. Flight control of micro aerial vehicles. In AIAA Guidance, Navigation, and Control Conference and Exhibit, 2004

[23] P. Pounds, J. Gresham, R. Mahony, J. Robert, and P. Corke. Towards dynamically favourable quad-rotor aerial robots. Proceedings of the Australasian Conference on Robotics and Automation, Canberra, ACT, Australia, 2004.

[24] P. Pounds, R. Mahony and P. Corke, Modelling and control of a quad-rotor, Proceedings of the Australasian Conference on Robotics and Automation, Auckland, New Zealand., December 2006

[25] E. Altu, J. P. Ostrowski, and R. Mahony, Control of a quadrotor helicopter using visual feedback, Proceedings of the 2002 IEEE International Conference on Robotics and Automation, Washington, D.C, May 2002, pp 72-77

[26] E. Altu, J. P. Ostrowski, and C. Taylor, Quadrotor control using dual visual feedback. Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Taipei, Taiwan, September 2003, pp. 4294-4299.

[27] Matthew G. Earl and Raffaello D'Andrea. Real-time attitude estimation technique applied to a quadrotor helicopter. Proceedings of the 43rd IEEE Conference on Decision and Control, 2004.

[28] E. B. Nice, Design of a Quadrotor Hovering Vehicle, Master Thesis, Cornell University, May, 2004

[29] S. Bouabdallah, A. Noth, and R. Siegwan. PID vs. LQ control techniques applied to an indoor micro quadrotor. Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004.

[30] G. Hoffmann, D. G. Rajnarayan, S. L. Waslander, D. Dostal, J. S. Jang, and C. J. Tomlin. The Standard Testbed of Autonomous Rotorcraft for Multi Agent Control (STARMAC). Proceedings of the 23rd Digital Avionics Systems Conference, 2004.

[31] M. Chen and M. Huzmezan, A combined MBPC/ 2 DOF H∞ controller for a quadrotor UAV, 2003 AIAA Guidance, Navigation, and Control Conference

[32] H. K. Khallil, Nonlinear Systems (3rd Edition), Prentice Hall, 2003

[33] J. D. Nicoud and J. C. Zufferey, Toward indoor flying robots, IEEE Conf． on Robots and Systems，2002: 787-792．

[34] S. Bouabdallah and R. Siegwart. Towards intelligent miniature flying robots． Proceeding of Field and Service Robotics, Australia, Port Douglas, 2005: 429-440．

[35] R. Olfati-Saber, Nonliner control of underactuated mechanical system with application to robotics and aerospace vehicles, MIT, 2001.

[36] F. Bin, S. Jia, and Y. Yao, An LQR controller for quadrotor: design and experiment, The 31st Youth Academic Annual Conference of Chinese Association of Automation, Wuhan, 2016: 81-86.

[37] L. Besnard, Y. B. Shtessel, and B. Landrum, Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer, Journal of the Franklin Institute, 2012, 349(2):658-684.

[38] A. Das, F. Lewis, and K. Subbarao, Backstepping approach for controlling a quadrotor using Lagrange form dynamics, Journal of Intelligent and Robotic System: Theory and Application, 2009, 56(1-2):127-151

[39] B. T. Whitehead, and S. R. Bieniawski, Model reference adaptive control of a quadrotor UAV, AIAA Guidance Navigation and Control Conference. Toronto, 2010:1-13.

[40] H. H. Wang, Y. M. Zhang, Y. M. Yi, et al, Nonlinear Tracking control method applied to Qball-4X quadrotor UAV against actuator faults, Proceedings of the 28th Chinese Control and Decision Conference. Yinchuan, 2016: 3478-3483.

[41] B. Yu, Y. M. Zhang, I. Minchala, et al. Fault-tolerant control with linear quadratic and model predictive control techniques against actuator faults in a quadrotor UAV, 2nd International Conference on Control and Fault-Tolerant System, Nice, 2013: 661-666.

[42] Z. X. Liu, C. Yuan, Y. M. Zhang, et al, A learning-based fault tolerant tracking control of an unmanned quadrotor helicopter. Journal of Intelligent and Robotic Systems, 2016, 84(1-4): 145-162.

[43] H. L. Yang, B. Jiang, and K. Zhang, Direct self-repairing control of the quadrotor helicopter based on adaptive sliding mode control technique, Proceedings of 2014 IEEE Chinese Guidance Navigation and Control Conference, 2014: 1403-1408.

[44] Z. H. Cen, H. Noura, T. Susilo, et al, Engineering implementation on fault diagnosis for quadrotors based on nonlinear observer, The 25th Chinese control and Decision Conference, Guiyang, 2013: 2971-2975.

[45] A. Drak, H. Noura, and M. Hejase, Sensor fault diagnostic and fault-tolerant control for the altitude control of a quadrotor UAV, 2015 IEEE 8th GCC Conference and Exhibition. Muscat, 2015: 1-5.

[46] Swiss Meteomatics new patent [EB/OL]. (2017-03-01) /[2017-04-03]. http://www.81uav.cn/UAV-news/201703/01/23200.html.

[47] I. Sadeghzadeh, A. Mehta, A. Chamseddine, et al. Active fault tolerant control of a quadrotor UAV based on gain-scheduled PID control, the 25th IEEE Canadian Conference on Electrical and Computer Engineering, Montreal, 2012: 1-4.

[48] J. H. Lee, Model predictive control: Review of the three decades of development, Int. J. Control, Autom. Syst. 9(3) (2011) 415–424.

[49] V. Santibanez and R. Kelly, A class of nonlinear PID global regulators of robot manipulators, in Proc. IEEE Int. Conf. Robotics and Automation (1998), pp. 3601–3608.

[50] D. Sun, S. Hu, X. Shao, and C. Liu, Global stability of a saturated nonlinear PID controller for robot manipulators, IEEE Trans. Control Syst. Technol. 17 (2009) 892–899.

[51] A. Loria, E. Lefeber, and H. Nijmeijer, Global asymptotic stability of robot manipulators with linear PID and PI 2D Control, SACTA 3(2) (2000) 138–149.

[52] A. Yarza, V. Santibanez, and J. Moreno-Valenzuela, Global asymptotic stability of the classical PID controller by considering saturation effects in industrial robots, Int. J. Adv. Robot. Syst. 8(4) (2011) 34–42.

[53] P. Rocco, Stability of PID control for industrial robot arms, IEEE Trans. Robot. Autom. 12(4) (1996) 606–614.

[54] M. Lazar, W. P. M. H. Heemels, A. Bemporad, and S. Weiland, Discrete time non-smooth nonlinear MPC: Stability and robustness, in Assessment and Future Directions of Nonlinear Model Predictive Control, eds. R. Findeisen, et al, Lecture Notes in Control and Information Sciences, Vol. 358 (Springer, 2007), pp. 93–103.

[55] E. Siva, P. Goulart, J. Maciejowski, and N. Kantas, Stability of model predictive control using Markov chain Monte Carlo optimisation, in Proc. European Control Conf. (2009).

[56] D. Mayne, Nonlinear and Adaptive Control Design [Book Review]. IEEE Transactions on Automatic Control, 2002, 41(12):1849.

[57] R. Hull, D. Schumacher, and Z.Qu, Design and evaluation of robust nonlinear missile autopilots from a performance perspective, Proceedings of the IEEE American Control Conference, 1995: 189-193.

[58] M. Steinberg and A. Page, Nonlinear adaptive flight control with a backstepping design approach, 1998.

[59] M. L. Steinberg. Comparison of intelligent, adaptive, and nonlinear flight control laws. Journal of Guidance Control & Dynamics, 2001, 24(4): 12.

[60] M. Krstic, I. Kanellakopoulos, and P. V. Kokotovic, Nonlinear design of adaptive controllers for linear systems. IEEE Transactions on Automatic Control, 2002, 39(4): 738-752.

[61] T. Lee and Y. Kim, Nonlinear adaptive flight control using backstepping and neural networks controller. Journal of Guidance Control & Dynamics, 2012, 24(4): 675-682.

[62] S. John, Artificial intelligent-based feedforward optimized PID wheel slip controller. AFRICON, 12 September 2013, Pointe-Aux-Piments, 1-6.

[63] K. U. Lee, H. S. Kim, J. B. Park, and Y. H. Choi, Hovering control of a quadrotor. The 12th International Conference on Control, Automation and Systems (ICCAS), 17-21 October 2012, 162-167.

[64] J. Li and Y. Li, Dynamic analysis and PID control for a quadrotor. International Conference on Mechatronics and Automation (ICMA), 7-10 August 2011, 573-578.

[65] T. Madani and A. Benallegue, Backstepping control for a quadrotor helicopter. Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, 9-15 October 2006, 3255-3260.

[66] X. Huo, M. Huo, and H. R. Karimi, Attitude stabilization control of a quadrotor UAV by using backstepping approach. Mathematical Problems in Engineering, 2014, 1-9.

[67] Z. Fang and W. Gao, Adaptive integral backstepping control of a micro-quadrotor. Proceedings of the 2nd International Conference on Intelligent Control and Information Processing (ICICIP), Harbin, 25-28 July 2011, 910-915.

[68] J. M. Maciejowski, Predictive Control with Constraints, Prentice Hall, Pearson Education Limited, Harlow, UK, 2002.

[69] M.Araki,"PID_Control"(http://www.eolss.net/ebooks/Sample%20Chapters/C18/E6-43-03-03.pdf)

[70] S. Bennett, A History of Control Engineering 1930-1955. IET 1993. pp. 48

[71] S. Bouabdallah and R. Siegwart. Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2247-2252.

[72] A. Nagaty, S. Saeedi, C. Thibault, M. Seto, and H. Li, Control and navigation framework for quadrotor helicopters. Journal of Intelligent and Robotic Systems, 70(1-4):1-12, 2013.

[73] Quaser, Quanser Qball-X4, User Manual, 2010.