Controller Design for Attitude and Position Control of Quadrotor

by

Muhammad Numan-MT133038

A thesis submitted to the
Department of Electrical Engineering
in partial fulfillment of the requirements for the degree of
Master of Science in Electrical Engineering

Faculty of Engineering
Capital University of Science and Technology
Islamabad
June, 2017

CERTIFICATE

This is to certify that Mr. Muhammad Numan has incorporated all observations, suggestions and comments made by the external evaluators as well as the internal examiners and thesis supervisor. The title of his Thesis is: Controller Design for Attitude and Position Control of Quadrotor.

.

Forwarded for necessary action

Dr. Aamer Iqbal Bhatti

(Thesis Supervisor)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

x	Linear position in X direction
y	Linear position in Y direction
z	Linear position in Z direction
ϕ	Roll angle
θ	Pitch angle
Ψ	Yaw angle
U	Linear velocity in x direction in body frame
V	Linear velocity in y direction in body frame
W	Linear velocity in z direction in body frame
P	Roll rate
Q	Pitch rate
R	Yaw rate
m	Mass of Quadrotor
l	Length of arm of Quadrotor
Cr	Throttle to RPM conversion coefficient
Ct	Thrust Coefficient
Cq	Torque Coefficient
Jr	Inertia of each rotor
w_i	Angular speed of ith rotor
Th_i	Throttle command for ith rotor
ϕ_c	Roll correction
θ_c	Pitch correction
Ψ_c	Yaw correction
z_c	Altitude correction
Φ_d	Desired roll

θ_d	Desired pitch
Ψ_d	Desired yaw
z_d	Desired altitude
e_ϕ	Error in roll angle
e_θ	Error in pitch angle
e_Ψ	Error in yaw angle
τ_ϕ	Roll control torque
τ_θ	Pitch control torque
τ_Ψ	Yaw control torque
$\hat{\boldsymbol{e}}_3$	vector in body *Z* axis
$\hat{\mathbf{e}}_z$	Unit vector in the inertial *Z* axis
f_t	Total thrust produced by rotors
g	Gravitational acceleration
$\boldsymbol{f_w}=[f_{wx}\ f_{wy}\ f_{wz}]^{\mathrm{T}}$	Forces produced by winds
$\mathbf{f_B} = [f_x\ f_y\ f_z]^{\mathbf{T}}$	Total force acting on the quadrotor body
$\mathbf{m_B} = [m_x\ m_y\ m_z]^{\mathrm{T}}$	Torque acting on the quadrotor body
$\mathbf{g_p}$	Gyroscopic moment
ϕ_R	Reference roll angle
θ_R	Reference pitch angle
U_{ft}	Control input for altitude
U_ϕ	Control input for roll angle
U_θ	Control input for pitch angle
U_ψ	Control input for yaw angle

ACKNOWLEDGMENT

First and foremost, I would like to thank ALLAH PAK, who gave me the courage, guidance and atmosphere to complete my postgraduate studies in CUST, Islamabad, Pakistan. The perseverance and determination granted by ALLAH PAK helped me to bear the hard times to produce this thesis. All respects to NAABI E PAK MUHAMMAD (P.B.U.H), who is the last messenger, whose life is a perfect model for the whole humanity.

I am really thankful on the efforts of my loving parents, family and friends who kept me motivated, guided and focused throughout my Master's. Their help in various regards contributed in keeping my moral high.

Particularly, I would like to say thanks to my supervisor, Dr. Aamer Iqbal Bhatti with whom it has been an honor to work. He allowed me tremendous freedom in choosing my area of research and has given me outstanding guidance all along the way. His "outside of the box" thinking has taught me to do the same and for that I am grateful to them.

DECLARATION

It is declared that this is an original piece of my own work, except where otherwise acknowledged in text and references. This work has not been submitted in any form for another degree or diploma at any other university or institution for tertiary education and shall not be submitted by me in future for obtaining any degree from this or any other University or Institution.

Muhammad Numan

MT133038

May 2017

ABSTRACT

This thesis discusses the development of detailed mathematical model for specific type of UAV, which has Vertical Takeoff and Landing (VTOL) ability, known as quadcopter. Mathematical model of quadrotor in state space form is derived; it utilizes Newton and Euler equations for three dimensional motions. This mathematical model is nonlinear and accurate enough including the aerodynamic effects and rotor dynamics. Quadrotor dynamics can be divided into two subsystems; translational subsystem and rotational subsystem. Translational subsystem is an under actuated system as it depends on roll, pitch, yaw angles and the translational state variables. The rotational subsystem is fully-actuated and only depends on the rotational states. Then development of two control approaches to control the attitude and position of the quadrotor in space is discussed. In first approach, Four PID controllers are designed to control the roll angle, pitch angle, yaw angle and altitude respectively. Strategy to implement PID controllers on nonlinear quadrotor model is discussed along with simulation results. In second approach, Backstepping controllers are designed to control the roll angle, pitch angle, yaw angle, altitude and positions. Simulation results after implementation Backstepping controllers on MATLAB are presented.

Chapter 1

INTRODUCTION

Quadrotor, a type of UAV, is an aircraft with four motors mounted at its four ends and electronic board held in the center. The quadrotor attains the desired attitude and position by varying the speed of motors. Several forces and moments act on the quadrotor such as gravity, yawing moment, gyroscopic effect, thrust due to rotational motion of the rotors and pitching and rolling moments caused due to different thrust of the rotors.

Advanced research in area of control theory, embedded systems and sensors have made Unmanned Aerial Vehicles (UAVs) very efficient. The computational and cost efficiency of UAVs have made them very suitable for secret operations. However some constraints exist due to their size limitation, such as small sensors are used which have more noise in operation as compared to large size components. However there is a benefit of less effect of environmental disturbance due to small size as compared to large size. Besides this, these UAVs are risk free in terms of life as there is no risk of loss of life. These UAVs can be sent to dangerous areas for spy purpose. These UAVs can be used for weather forecasting, surveillance and coverage of events as well. Due to their importance many control systems have been developed for UAVs.

Stabilization and robustness are the main control objectives which should be kept in mind while designing control for UAVs. A lot of aerodynamics forces exists in atmosphere as a disturbances which results in noisy data from sensors. Highly efficient embedded systems are designed to make the flight of UAVs stable and robust. Due to ever growing need of UAVs, Very sophisticated and efficient control methods are required to ensure the stability and robustness of the flight mission. Initially linear control strategies were applied for the control of UAVs. However optimized mathematical modeling and highly faster computational embedded systems can improve the performance. Non linear control methods improve the performance of the control system and guarantee robustness as well.

1.1 Literature Review

Quadrotor control design and its modeling is a well known field of research and is used in many civil and military applications. There is an extensive literature review of quadcopter's control design and modeling. Modeling of UAVs (Unmanned Aerial Vehicles) should be simple and precise for optimized performance. There are many classical linear approaches used for control of quadrotors namely, PID [12][15][13][16][14][17][19][20], Linear quadratic regulator [15][24][17][18][1][19][20] for optimal control. Besides, many advanced control approaches are also used such as H-infinity control design, adaptive approach [27], non linear feedback linearization [25][21], Backstepping and adaptive backstepping [19][25][22][20], sliding mode control, etc. The main control objective of all control methods is to stabilize and make robustness in the orientation and position of quadrotor.

Sensors played main role in control design. As they provide feedback data to the systems and upon that data control error is generated and ultimately control effort is provided to the actuators of the plant. So highly précised and sophisticated sensors are used in control design. Many sensors like Inertial Measuring Unit (IMU) are used for calculating accelerations and angular position, Infrared and SONAR sensors are used for measuring distance from target, highly efficient cameras, GPS for locating and tracking etc.

Various control approaches have been developed for quadrotors for regulation problem as well as for tracking problems. The control approach that guarantees the states of Quadrotor converges to some set reference point or states. Many work [9],[10],[11],[26] have showed that the control of Quadrotor is possible by using different linear control methods by linearization of non-linear model about its stable point. But non-linear models presents better control as they are considering more general form of actual system. Having uncertainties, disturbances and no approximation is involved. Among these nonlinear methods sliding mode control [9] [10], backstepping [4],[7], and feedback linearization [6] have been showed to be very effective approaches for the control of quadrotors. Especially feedback linearization provides very good results for control of qudrotors. Recent research [6] on feedback linearization presents Quadrotor dynamics into two loops: the inner loop contains the attitude and height of quadrotor, the outer loop contains the position of quadrotor. Another approach of feedback linearization is to linearize the model. All control methodologies require comprehensive knowledge of system

model and model parameters. Errors in the system parameters can badly affect the performance of the controller. Un modeled parameters such as weight can cause a large disturbance in stabilization during flight. For such needs adaptive techniques are available which can estimate the un modeled parameters, can autocorrect parameters on changing or variation and also tackle external disturbing forces. Adaptive methods like Model Reference Adaptive Control (MRAC) discussed in [8]. Hanng etal [2] proposed adaptive backstepping and this approach was further used by Zeng etal [5] to include inertia parameter in the adaptation law. All indirect techniques correct parameter errors base on expected value and actual output of the system, but are not able to correct the system parameters, as done by direct adaptive technique. Direct Adaptive Technique was first introduced by Craig et al [3].

1.2 History

Etienne Oehmichen did the pioneering work in the design of Quad rotor. He designed multi copter with four rotors and eight propellers, which is a single handed engine. He had used a frame of steel tube. At the end of the four arms there are two-bladed rotors. Spinning motion of five rotors in horizontal plane is responsible for literal motion. One propeller is located at the front for steering. Remaining propellers are used for forward direction. This multi copter posses a significant stability and controllability, and have made many test flights in the mid 1920s.

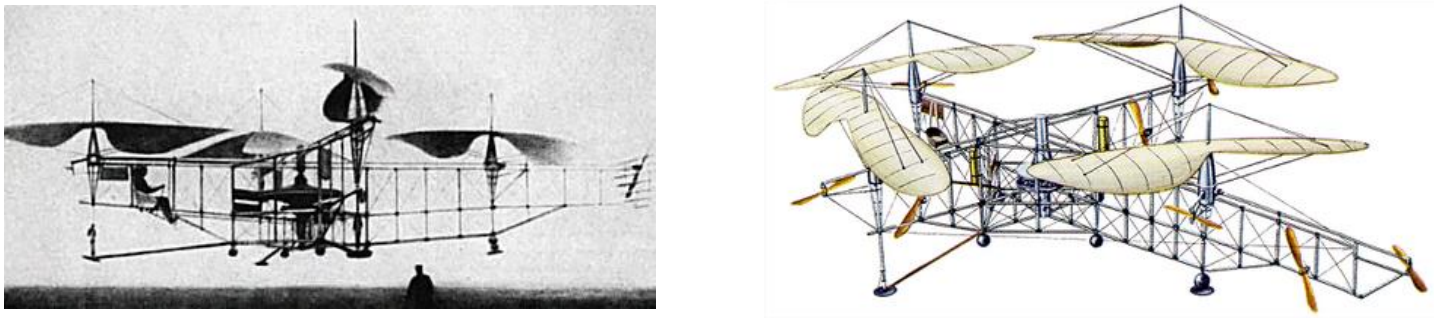


Figure 1.1: Oehmichen Quadrotor [28]

After that Dr. George de Bothezat and Ivan Jerome developed an X- shaped frame with six bladed rotors [28]. Two propellers were used for thrust and yaw control by varying pitch. The first flight of this quadcopter was taken in October 1922. The height attained by this Quadcopter

was 5m. Although there were some problem regarding power, and complexity in the structure and reliability of the quadrotor.

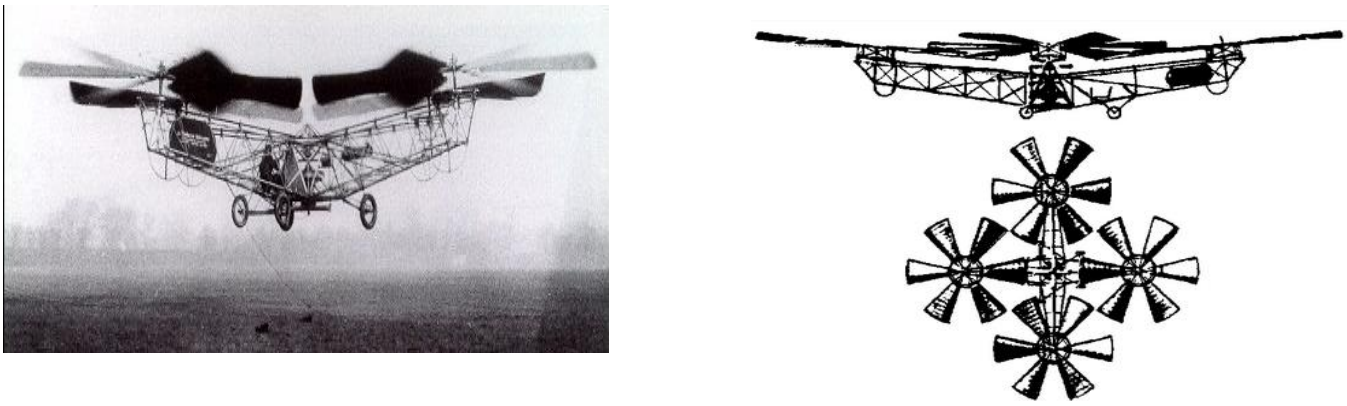


Figure 1.2: de Bothezat Quadcopter [28]

Convertawings model A Quadcopter was presented in 1956 which was a prototype for heavy military and civil multicopters [28]. There were two engines installed for driving four rotors, and wings were used for lift purpose in forward flight mode. There was no need for rotor at tail and whole control of Quadrotor was obtained by changing the thrust between rotors which is caused by varying the speed of rotors. Many successful flights were taken in the middle of 1950. This was the best design of actual Quadrotor and it was also famous for the design of first Quadrotor helicopter for demonstrating successful flights. Due to lack of interest from commercial and military, this project ended.

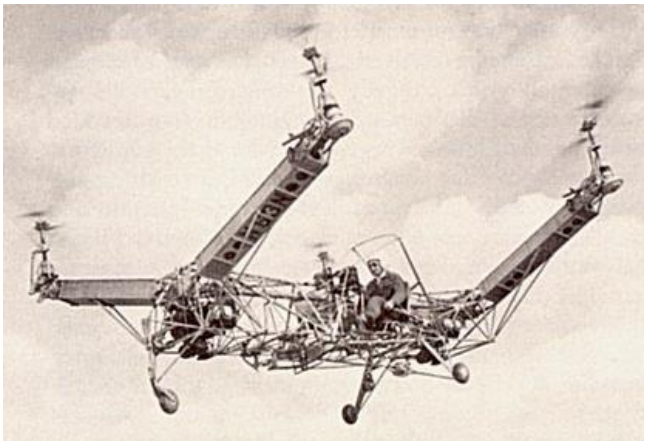


Figure 1.3: Model A Quadrotor [28]

1.3 Objective and Motivation

In this thesis we will focus on mathematical modeling and control of quadrotor. There are many advantages of quadrotor on other type of UAVs. The research in controller design of quadrotor is still having challenges because quadrotor is multivariable and highly nonlinear system. There are six degree of freedom (6DOF) and only four actuators so it is an underactuated system [33]. Underactuated systems are type of system that have more number of outputs than inputs. It is difficult to control quadrotor because there exist nonlinear coupling between actuators and degree of freedom.

1.4 Thesis Structure

This thesis contains the following contents in sequence:

Chapter 2: This chapter includes the derivation of mathematical model of quadrotor in state space form, utilizes Newton and Euler equations for three dimensional motions. The main objective of this chapter is to provide a model that is accurate enough to represent the dynamics of quadrotor for simulation and control purposes, also gives the good understanding of the dynamics of the quadrotor.

Chapter 3: In this chapter the control strategy to design PID controllers for nonlinear mathematical model of quadrotor is discussed. Four PID controllers are designed to control roll, pitch, yaw and altitude separately. Simulation results after PID controller implementation are also presented in this chapter.

Chapter 4**:** In this chapter we will derive the alternative form of state space model. Design of backstepping controller for nonlinear underactuated quadrotor system is discussed. Simulation results after implementation of backstepping controllers for the altitude, latitude and position control of quadrotor are also shown in this chapter.

Chapter 5: Conclusion based on the simulation results and and future work recommendation is presented in this chapter.

Chapter 2

MATHEMATICAL MODEL

Quadrotor is an aircraft with four motors mounted at its four ends and electronic board held in the center. Reference coordinates of the quadrotor describing the structure and position are discussed followed by the mathematical model. The quadrotor uses two reference systems; one fixed and one movable. Newton's first law is considered applicable in the fixed coordinate system, thus it is also known as inertial system. The movable reference system is placed at the centre of axis of the quadrotor as illustrated in Figure 2.1.

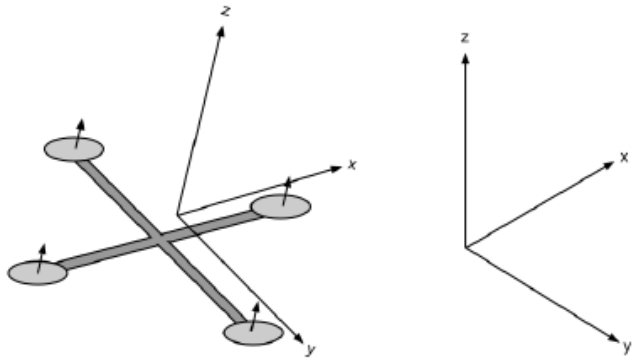


Figure 2.1: Movable and Fixed Reference System

The quadrotor attains the desired attitude and position by varying the speed of motors. Several forces and moments act on the quadrotor such as gravity, yawing moment, gyroscopic effect, thrust due to rotational motion of the rotors and pitching and rolling moments caused due to different thrust of the rotors. The gyroscopic effect is noticed in the lightweight construction quadrotor only. The yawing moment due to unbalanced rotational speeds of the rotors is minimized when two rotors rotate opposite to each other.

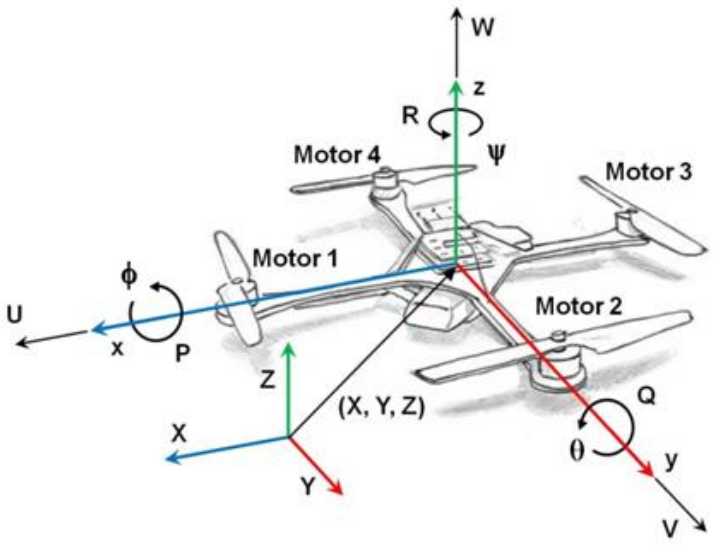


Figure 2.2: Quadrotor Rotation Conventions

Thus, the propellers are classified as (i) front and rear propellers, which rotate in the anticlockwise direction, labeled as 1 and 3 in Figure 2.2, and (ii) right and left propellers, which rotate in the clockwise direction, labeled as 2 and 4 in Figure 2.2.

Two types of motion of aircraft are noted, i.e. barycenter translation and rotation around the barycenter. The motion of an aircraft in space requires six degrees of freedom (6DoF). There are three translations along the three axes and three rotations around the three axes. The 6DoF motions are controlled by the rotational speeds of the motors. The different types of motion include forward, backward, lateral, vertical, roll, pitch and yaw motions. A reactive torque produced by the rotor induces the yaw motion. The size of the reactive torque is relative to the rotor speed. The reactive torques balances each other and quadrotor does not rotate when the speed of the rotors is the same. The quadrotor rotates when the speed of rotors is different. The speed of the rotors increase and decrease synchronously for vertical motion. The quadrotor has four inputs and six outputs, thus it is considered and under actuated nonlinear complex system. Some assumptions are made in the quadrotor model for control such as rigidity of the quadrotor body, symmetry in the structure of the quadrotor body and the ground effect is ignored.

The four basic types of motion of the quadrotor by the rotation speed of each propeller as illustrated in Figures 2.3 to 2.6.

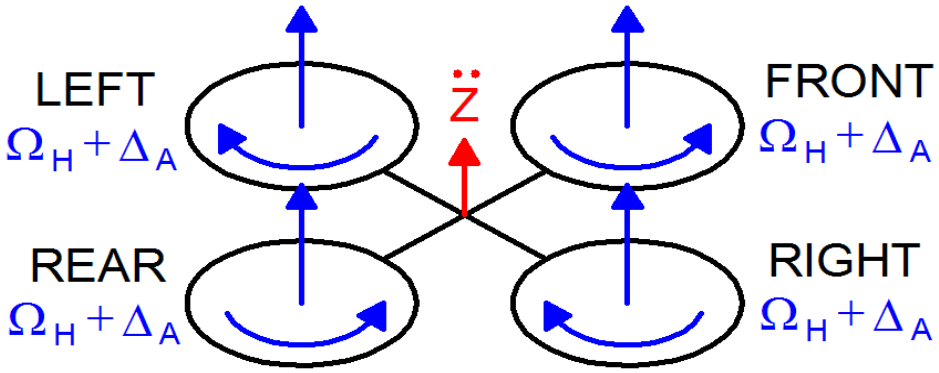


Figure 2.3: Throttle Motion[34]

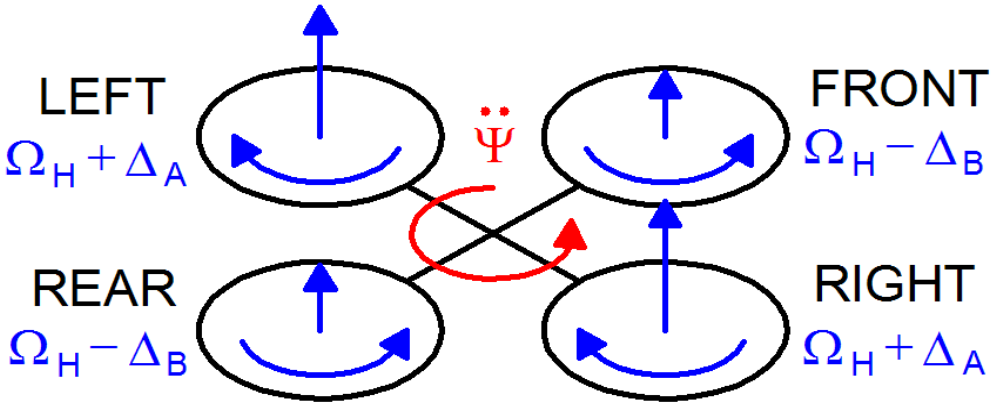


Figure 2.4: Yaw Motion[34]

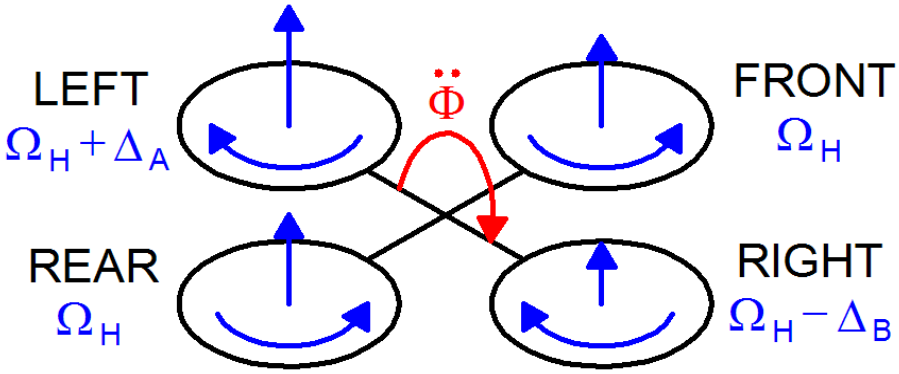


Figure 2.5: Roll Motion[34]

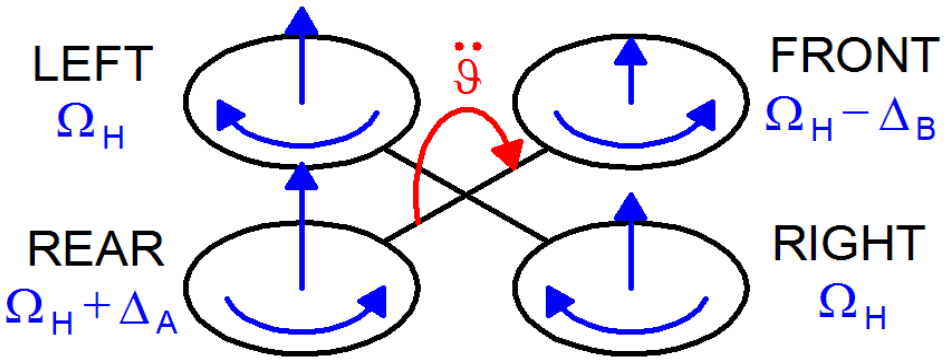


Figure 2.6: Pitch Motion[34]

2.1 Euler Angles

The orientation of rigid body is described by three Euler angles in the 3D Euclidean space. There are several ways; we are using ZYX Euler angles [30]. They are used to transform the coordinates in reference frame to the coordinates in another reference frame and they are used to define the orientation of reference frame relative to another reference frame. Euler angles are mostly denoted by $\emptyset \in [-\pi, \pi]$, $\theta \in \left[\frac{\pi}{2}, \frac{\pi}{2}\right]$, $\psi \in [-\pi, \pi]$. There are three sequences of rotations that are rotations about the X, Y and Z axis, any orientation can be accomplish by combining three elemental rotations. Following rotation matrices are used to define this combination [27].

$$R_x(\emptyset) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C(\emptyset) & -S_n(\emptyset) \\ 0 & S_n(\emptyset) & C(\emptyset) \end{bmatrix} \tag{2.1}$$

$$R_y(\theta) = \begin{bmatrix} C(\theta) & 0 & S_n(\theta) \\ 0 & 1 & 0 \\ -S_n(\theta) & 0 & C(\theta) \end{bmatrix} \tag{2.2}$$

$$R_z(\psi) = \begin{bmatrix} C(\psi) & -S_n(\psi) & 0 \\ S_n(\psi) & C(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.3}$$

$S_n(\emptyset) = Sin(\emptyset)$, $C(\emptyset) = Cos(\emptyset)$, $C(\theta) = Cos(\theta)$, $S_n(\theta) = Sin(\theta)$, $C(\psi) = Cos(\psi)$, and $S_n(\psi) = Sin(\psi)$

The coordinates of inertial frame and the coordinate of the body frame are related by the rotation matrix $R_{zyx}(\emptyset, \theta, \psi) \in SO(3)$. Following matrix is used for transformation from body coordinate system to the inertial reference system.

$$R_{zyx}(\emptyset, \theta, \psi) = R_x(\emptyset).R_y(\theta).R_z(\psi)$$

$$= \begin{bmatrix} C(\theta)C(\psi) & S_n(\emptyset)S_n(\theta)C(\psi) - C(\emptyset)S_n(\psi) & C(\emptyset)S_n(\theta)C(\psi) + S_n(\emptyset)S_n(\psi) \\ C(\theta)C(\psi) & S_n(\emptyset)S_n(\theta)S_n(\psi) + C(\emptyset)C(\psi) & C(\emptyset)S_n(\theta)S_n(\psi) - S_n(\emptyset)C(\psi) \\ -S_n(\theta) & S_n(\emptyset)C(\theta) & C(\emptyset)C(\theta) \end{bmatrix} \quad (2.4)$$

2.2 Mathematical Model

Utilizing Newton and Euler equations for three dimensional motions of rigid body, we are providing mathematical model of the quadrotor. The main objective of this section is to provide a model that is accurate enough to represent the dynamics of quadrotor for simulation and control purposes, this section also gives the good understanding of the dynamics of the quadrotor. Suppose $[x \ y \ z \ \emptyset \ \theta \ \psi]^T$ is the vector containing the positions both linear and angular in the earth reference frame and $[U \ V \ W \ P \ Q \ R]^T$ is vector that hold the velocities both linear and angular in the body movable frame. Earth fixed frame and the body movable frame are related by following relations:

$$\boldsymbol{v} = \boldsymbol{R}.\boldsymbol{v_B} \quad (2.5)$$

$$\boldsymbol{w} = \boldsymbol{T}.\boldsymbol{w_B} \quad (2.6)$$

Whereas:

$$\boldsymbol{v} = [\dot{x} \ \dot{y} \ \dot{z}]^T \in R^3$$

$$\boldsymbol{w} = [\dot{\emptyset} \ \dot{\theta} \ \dot{\Psi}]^T \in R^3$$

$$\boldsymbol{v_B} = [U\ V\ \ W]^T\ \in R^3$$

$$\boldsymbol{w_B} = [P\ Q\ R]^T\ \in R^3$$

Transformation matrix for angular transformation is given by [31]

$$\text{T}=\begin{bmatrix} 1 & S_n(\emptyset)t(\theta) & c(\emptyset)t(\theta) \\ 0 & c(\emptyset) & -S_n(\emptyset) \\ 0 & \frac{S_n(\emptyset)}{c(\theta)} & \frac{c(\emptyset)}{c(\theta)} \end{bmatrix} \tag{2.7}$$

Where: $t(\theta) = \tan(\theta)$

2.2.1 Kinematic Model

By solving (2.5) and (2.6) kinematic model of the quadrotor is derived

$$\dot{x} = W[S_n(\emptyset)S_n(\psi) + c(\phi)c(\psi)S_n(\theta)] - V[c(\phi)S_n(\psi) - c(\psi)S_n(\phi)S_n(\theta)] + U[c(\psi)c(\theta)]$$

$$\dot{y} = V[c(\phi)c(\psi) + S_n(\phi)S_n(\psi)S_n(\theta)] - W[c(\psi)S_n(\phi) - c(\phi)S_n(\psi)S_n(\theta)] + U[c(\theta)S_n(\psi)]$$

$$\dot{z} = W[c(\phi)c(\theta)] - U[S_n(\theta)] + V[c(\theta)S_n(\phi)]$$

$$\dot{\phi} = P + R[c(\phi)t(\theta)] + Q[S_n(\phi)t(\theta)]$$

$$\dot{\theta} = Q[c(\phi)] - R[S_n(\phi)]$$

$$\dot{\psi} = R\frac{c(\phi)}{c(\theta)} + Q\frac{S_n(\phi)}{c(\theta)} \tag{2.8}$$

2.2.2 Dynamic Model

According to Newton's law total force acting on the body of quadrotor is given by:

$$\text{m}(\boldsymbol{\omega_B} \wedge \mathbf{v_B} + \mathbf{v}_\text{B}) = \mathbf{f_B} \tag{2.9}$$

m = Mass of quadrotor

$\mathbf{f_B} = \left[f_x\ f_y\ f_z\right]^\mathbf{T}\ \in\ \mathbb{R}^{3\times3}$ Total force acting on the quadrotor body

$\wedge$ = Cross product

So by solving (2.9)

$$f_x = m(\dot{U} + QW - RV)$$

$$f_y = m(\dot{V} - PW + RU)$$

$$f_z = m(\dot{W} + PV - QU) \tag{2.10}$$

Similarly according to Euler's equation total torque acting on quadrotor body is given by:

$$\mathbf{I}.\dot{\boldsymbol{\omega}}_{\boldsymbol{B}} + \boldsymbol{\omega}_{\mathbf{B}} \wedge (\mathbf{I}.\boldsymbol{\omega}_{\mathbf{B}}) = \mathbf{m}_{\mathbf{B}} \tag{2.11}$$

I = Diagonal inertia matrix

$$\mathrm{I} = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \in \mathbb{R}^{3\times3}$$

Inertia matrix is diagonal because of symmetry of the system.

$\mathbf{m}_{\mathbf{B}} = \begin{bmatrix} m_x & m_y & m_z \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{3\times3}$ Total torque acting on the quadrotor body

I Matrix is diagonal because of symmetry of the system

So by solving (2.11)

$$m_x = \dot{P}I_x - QRI_y + QRI_z$$

$$m_y = \dot{Q}I_y + PRI_x - PRI_z$$

$$m_z = \dot{R}I_z - PQI_x + PQI_y \tag{2.12}$$

These equations are valid as long as the origin of the axes of body frame is coincide with the centre of quadrotor.

2.2.3 External Forces and Moments

External forces acting on the quadrotor in body frame are given by:

$$\boldsymbol{f}_{\boldsymbol{B}} = mg\mathbf{R}^{T}.\hat{\mathbf{e}}_z - f_t\hat{\boldsymbol{e}}_3 + \boldsymbol{f}_{\boldsymbol{w}} \tag{2.13}$$

$\boldsymbol{f_w}=\left[f_{wx}\ \ f_{wy}\ \ f_{wz}\right]^{\mathrm{T}} \in \mathbb{R}^{3\times3}$ Forces produced by winds

$\hat{\boldsymbol{e}}_3$ = Unit vector in body Z axis

$\hat{\mathbf{e}}_z$ = Unit vector in the inertial Z axis

f_t = Total thrust produced by rotors

g = Gravitational acceleration

By solving (2.13) and (2.10):

$$-mg[S_n(\theta)] + f_{wx} = m\left(\dot{U} + QW - RV\right)$$

$$mg[c(\theta)S_n(\phi)] + f_{wy} = m\left(\dot{V} - PW + RU\right)$$

$$mg[c(\theta)S_n(\phi)] + f_{wz} - f_t = m\left(\dot{W} + PV - QU\right) \tag{2.14}$$

Similarly external moments acting on the quadrotor in the body frame are given by:

$$\mathbf{m_B} = \boldsymbol{\tau_B} - \mathbf{g_p} + \boldsymbol{\tau_w} \tag{2.15}$$

$\boldsymbol{\tau_B} = \left[\tau_\phi\ \ \tau_\theta\ \ \tau_\psi\right]^T \in \mathbb{R}^{3\times3}$ Control torques produced by speed difference of rotors

$\boldsymbol{\tau_w} = \left[\tau_{wx}\ \ \tau_{wy}\ \ \tau_{wz}\right]^T \in \mathbb{R}^{3\times3}$ Torques caused by wind on quadrotor body

$\mathbf{g_p}$ = Gyroscopic moment generated due to the combined rotation of quadrotor body and four rotors

$$\mathbf{g_p} = \sum_{i=1}^{4} J_p(\boldsymbol{w}_B \wedge \hat{\mathbf{e}}_3)(-1)^{i+1}\mathrm{w}_i \tag{2.16}$$

w_i = Angular speed of rotor No. i^{th}

J_p= Inertia of each rotor

By solving (2.15) and (2.12):

$$\tau_\theta + \tau_{wx} = \dot{P}I_x - QRI_y + QRI_z$$

$$\tau_\phi + \tau_{wy} = \dot{Q}I_y + PRI_x - PRI_z$$

$$\tau_\psi + \tau_{wz} = \dot{R}I_z - PQI_x + PQI_y \quad (2.17)$$

2.2.4 Actuator Equations

Thrust is caused by rotation of rotors. Thrust is main force behind all the quadrotor movements. Thrust and torques are considered proportional to square of the speed of rotors [32]. Thrust caused by single motor is given by:

$$T = C_T \rho A_r r^2 \varpi^2 \quad (2.18)$$

C_T = Thrust coefficient of the rotor

ρ = Air density

A_r= Cross section area of propeller

r = Radius of propeller

ϖ = Angular velocity of rotor

By approximations, thrust coefficient relation is given by:

$$T = c_T \varpi^2 \quad (2.19)$$

Torque causes the system to move about the Z-axis. Torque created by motor is given by:

$$q = c_q \varpi^2 \quad (2.20)$$

q = Torque generated by the rotor

c_q = Torque coefficient of the rotor

There are four rotors; hence there are four inputs to the system. There are separate control inputs for vertical motion, pitch motion, roll motion and yaw motion given by following relations:

$$f_t = c_T(\mathrm{w}_1^2 + \mathrm{w}_2^2 + \mathrm{w}_3^2 + \mathrm{w}_4^2)$$

$$\tau_\theta = c_T l(\mathrm{w}_3^2 - \mathrm{w}_1^2)$$

$$\tau_\phi = c_T l(\mathrm{w}_4^2 - \mathrm{w}_2^2)$$

$$\tau_\psi = c_q(\mathrm{w}_2^2 + \mathrm{w}_4^2 - \mathrm{w}_1^2 - \mathrm{w}_3^2) \tag{2.21}$$

By substituting (2.18) in (2.14) and (2.17), we get the following dynamic model of quadrotor

$$-\mathrm{mg}[S_n(\theta)] + f_{wx} = m\big(\dot{U} + QW - RV\big)$$

$$mg[c(\theta)S_n(\phi)] + f_{wy} = m\big(\dot{V} - PW + RU\big)$$

$$mg[c(\theta)c(\phi)] + f_{wz} - c_T(\mathrm{w}_1^2 + \mathrm{w}_2^2 + \mathrm{w}_3^2 + \mathrm{w}_4^2) = m\big(\dot{W} + PV - QU\big)$$

$$c_T l(\mathrm{w}_4^2 - \mathrm{w}_2^2) + \tau_{wx} = \dot{P}I_x - QRI_y + QRI_z$$

$$c_T l(\mathrm{w}_3^2 - \mathrm{w}_1^2) + \tau_{wy} = \dot{Q}I_y + PRI_x - PRI_z$$

$$c_q(\mathrm{w}_2^2 + \mathrm{w}_4^2 - \mathrm{w}_1^2 + \mathrm{w}_3^2) + \tau_{wz} = \dot{R}I_z - PQI_x + PQI_y \tag{2.22}$$

2.3 State Space Model

Consider the following state's vector:

$$\boldsymbol{X} = [\emptyset \quad \theta \quad \psi \quad P \quad Q \quad R \quad U \quad V \quad W \quad x \quad y \quad z]^{\mathrm{T}} \in \mathbb{R}^{12} \tag{2.23}$$

We can rewrite the kinematic equations (2.8) and dynamic equations (2.22) in state space form as given:

$$\dot{\emptyset} = P + R[c(\phi)t(\theta)] + Q[S_n(\phi)t(\theta)]$$

$$\dot{\theta} = Q[c(\phi)] - R[S_n(\phi)]$$

$$\dot{\psi} = R\frac{c(\phi)}{c(\theta)} + Q\frac{S_n(\phi)}{c(\theta)}$$

$$\dot{P} = \frac{I_y - I_z}{I_x}RQ + \frac{\tau_\phi + \tau_{wx}}{I_x}$$

$$\dot{Q} = \frac{I_z - I_x}{I_y} PR + \frac{\tau_\theta + \tau_{wy}}{I_y}$$

$$\dot{R} = \frac{I_x - I_y}{I_z} PQ + \frac{\tau_\psi + \tau_{wz}}{I_z}$$

$$\dot{U} = RV - QW - g[S_n(\theta)] + \frac{f_{wx}}{m}$$

$$\dot{V} = PW - RU + g[S_n(\phi)c(\theta)] + \frac{f_{wy}}{m}$$

$$\dot{W} = QU - PV + g[c(\theta)c(\phi)] + \frac{f_{wx} - f_t}{m}$$

$$\dot{x} = W[S_n(\phi)S_n(\psi) + c(\phi)c(\psi)S_n(\theta)] - V[c(\phi)S_n(\psi) - c(\psi)S_n(\phi)S_n(\theta)] + U[c(\psi)c(\theta)]$$

$$\dot{y} = V[c(\phi)c(\psi) + S_n(\phi)S_n(\psi)S_n(\theta)] - W[c(\psi)S_n(\phi) - c(\phi)S_n(\psi)S_n(\theta)] + U[c(\theta)S_n(\psi)]$$

$$\dot{z} = W[c(\phi)c(\theta)] - U[S_n(\theta)] + V[c(\theta)S_n(\phi)] \quad (2.24)$$

$$\dot{X} = f(X, U)$$

X= State vector

U=Input vector

State vector:

$$X = [\emptyset \quad \theta \quad \psi \quad P \quad Q \quad R \quad U \quad V \quad W \quad x \quad y \quad z]^{\mathrm{T}} \in \mathbb{R}^{12} \quad (2.25)$$

Control inputs can be written as:

$$U = [\mathrm{w}_1 \quad \mathrm{w}_2 \quad \mathrm{w}_3 \quad \mathrm{w}_4]$$

Chapter 3

PID CONTROL STRATEGY

Mathematical model developed shows that this is an under actuated system and controller design for these type of system is not an easy task. In this thesis, we have applied two control techniques; PID control and Backstepping control for control of this under actuated quadrotor system. PID is applied on the non linear system (2.24). The brief strategy to apply PID controller will be discussed here and the strategy to implement backstepping control will be discussed later in next chapter.

3.1 Strategy to implement PID controller

Here we have coupled equations. Extracting transfer functions for roll, pitch, yaw and altitude from equation set (2.24) requires a lot of approximations and also it will result in equations that will not represent the true behavior of quadrotor. So we decided to apply PID controller directly on non linear plant model. Four PID controllers will be designed to control the roll, pitch, yaw and altitude separately.

To implement PID controller, we divided the system in four blocks as shown in Figure 3.1:

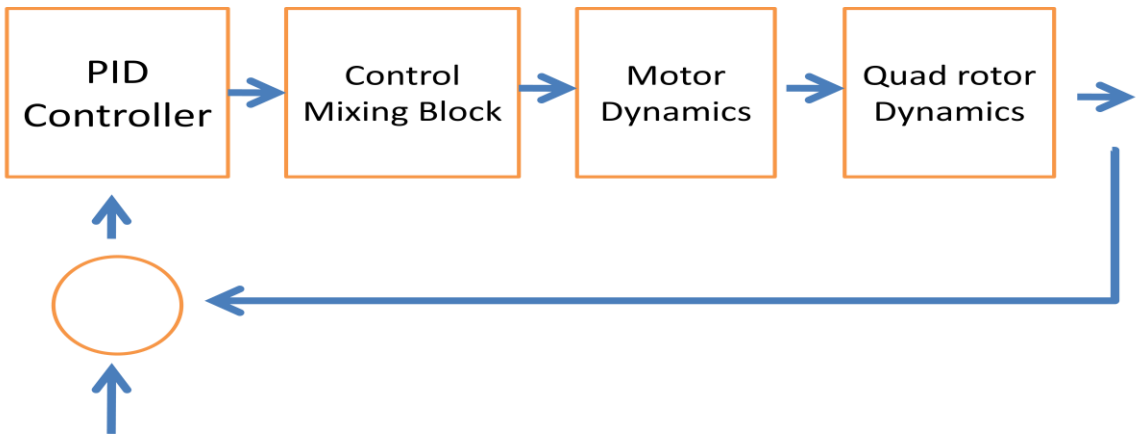


Figure 3.1: PID Control Strategy Block Diagram

3.2 Quadrotor Dynamics Block

This block represents the equation set (2.24). Four motor speeds: w_1, w_2, w_3 and w_4 in revolution per minutes (RPMs) are inputs to this block and output from this block are all the twelve states of the system: $[\emptyset \quad \theta \quad \psi \quad P \quad Q \quad R \quad U \quad V \quad W \quad x \quad y \quad z]^{\mathrm{T}}$ as shown in Figure 3.2. Roll, pitch, yaw and altitude states are also feedback to PID controller block to find the error from desired roll, pitch, yaw and altitude respectively.

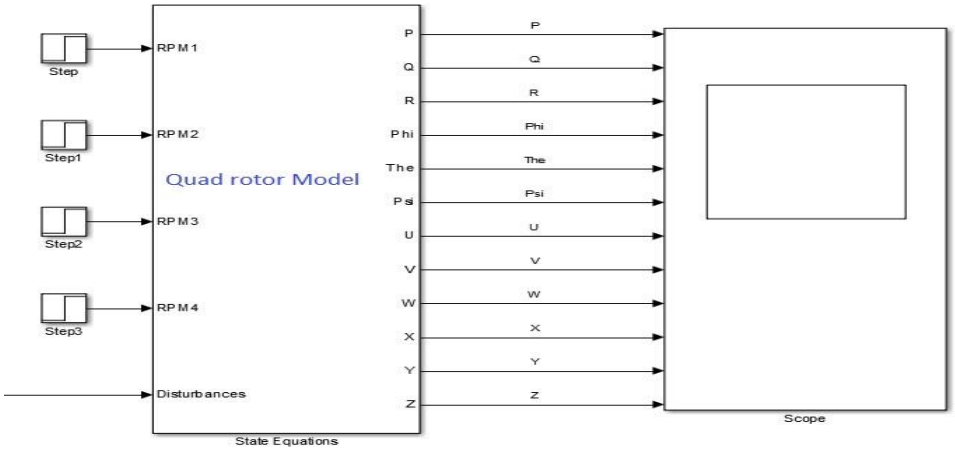


Figure 3.2: Quadrotor Dynamics Block

Open loop step responses for different motor speeds are shown in Figure 3.3 and Figure 3.4.

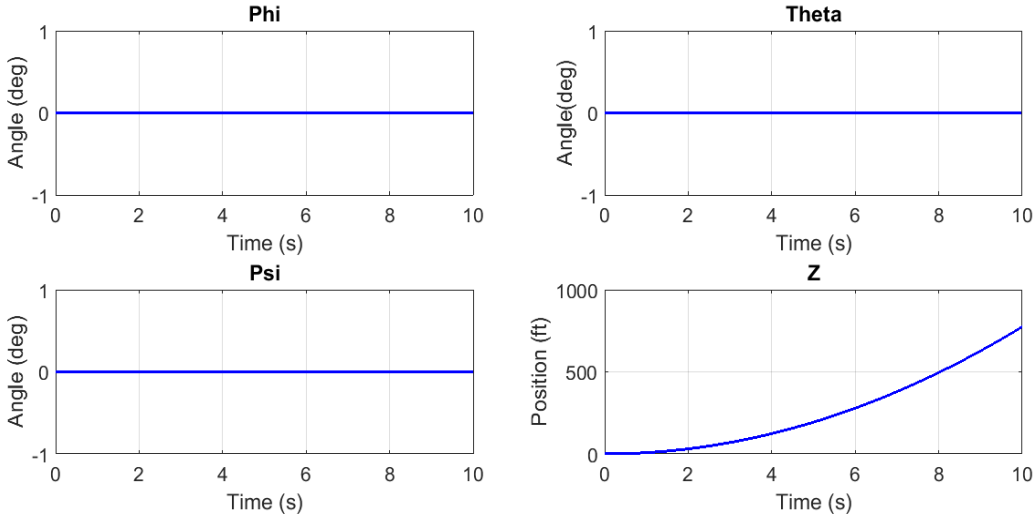


Figure 3.3: Output of Dynamics Block for $w_1 = w_2 = w_3 = w_4$

Figure 3.3 shows output of quadrotor dynamics block when speed of all four motors is 5000 RPMs; This shows that altitude of quadrotor increases when all motors have same speeds. Similarly Figure 3.4 shows output of quadrotor dynamics block when $w_1 = w_2 = w_4 = 5000$ RPMs and $w_3 = 4950$ RPMs .

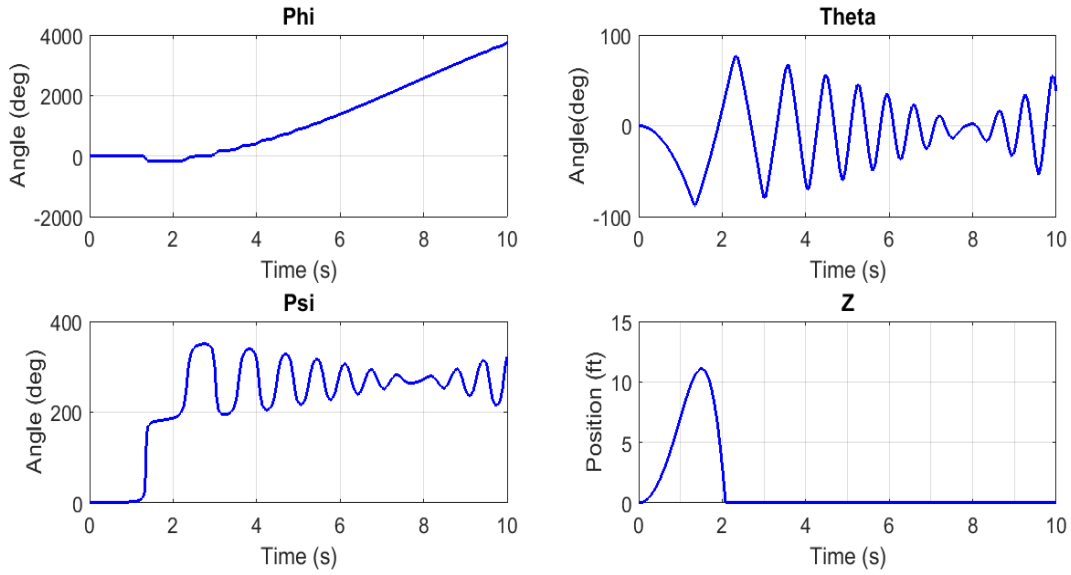


Figure 3.4: Output of Dynamics Block for $w_1 = w_2 = w_4$ and $w_3 < w_1$

3.3 Motor Dynamics Block

The quadrotor's rotors are driven by DC motors. Motor dynamics block represent the mathematical model of DC motor.

$$L\frac{di}{dt} = U - R_{mot}i - k_e w$$

$$J_m\frac{dw}{dt} = \tau_m - \tau_d \tag{3.1}$$

L= Inductance of motor coil

i= Current through armature

R_{mot}= Armature resistance

w= Motor speed in RPMs

J_m= Inertia of motor

τ_m= Motor torque

τ_d= Load torque

We are using motor with low inductance, so the second order mathematical model of DC motor may be given by:

$$J_m \frac{dw}{dt} = -\frac{k_m^2}{R_{mot}} w + \frac{k_m}{R_{mot}} U - \tau_d \tag{3.2}$$

Equation (3.2) may be rewritten by introducing gearbox and propeller model as given:

$$\frac{dw}{dt} = -\frac{1}{\tau} w - \frac{d}{\eta r^3 J_t} w^2 + \frac{1}{k_m \tau} U$$

$$\frac{1}{\tau} = \frac{k_m^2}{R J_t} \tag{3.3}$$

Equation set (3.3) can be linearized on an equilibrium point w_0:

$$\dot{w} = -Aw + BU + C \tag{3.4}$$

Where

$A = \frac{1}{\tau} + \frac{2dw_0}{\eta r^3 J_t}$, B=$\frac{1}{k_m \tau}$, C=$\frac{dw_0{}^2}{\eta r^3 J_t}$

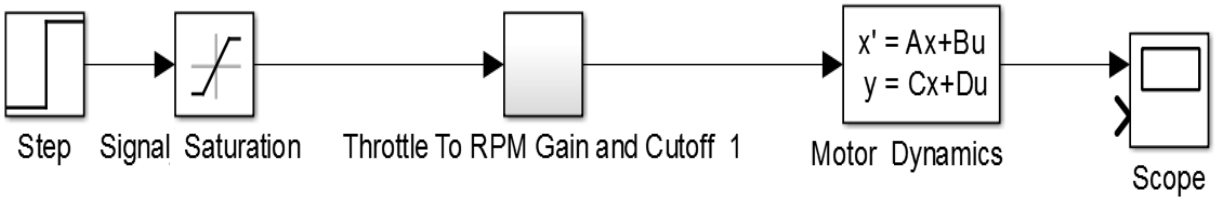


Figure 3.5: Motor Dynamics Block of Motor 1

Input to motor dynamics block is throttle command Th_i and output of motor dynamics block is w_i which is RPM for motor at any given instant of time. Motor dynamics block limits the input command between 0% and 100% throttle, simulates the cutoff behavior at low throttle and applies the given relationship:

$$w = (Throttle\ \%)C_R + b \tag{3.5}$$

Where:

w=expected steady state motor RPM

C_R=Throttle to RPM conversion coefficient

b= y-intercept of linear regression relationship

Input and corresponding output of motor dynamics block is shown in Figure 3.6

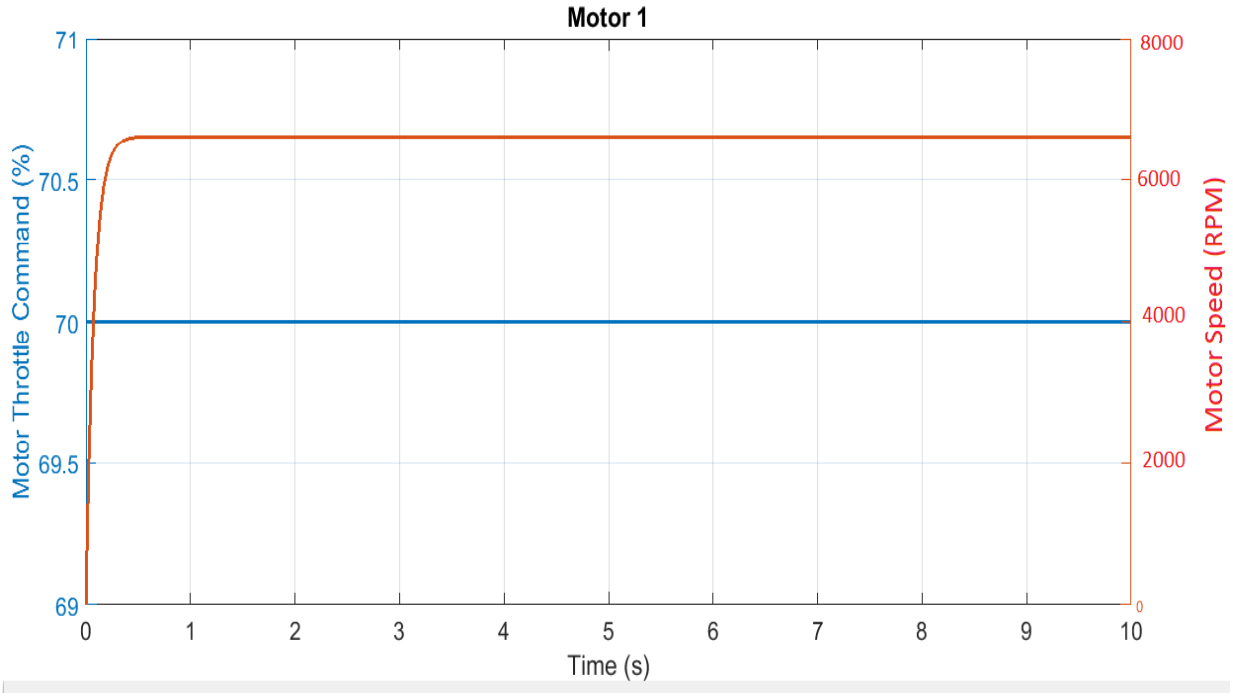


Figure 3.6: Motor Speed and Throttle Command

We have four motors , So this block receives four throttle signals from commands mixing block $[Th_1\ Th_2\ Th_3\ Th_4]^T$ and produces four motor speeds $[w_1\ w_2\ w_3\ w_4]^T$ correspondingly as desired by throttle signals.

3.4 Commands Mixing Block

This block get the correction commands for the roll, pitch, yaw and altitude and mix these commands by considering each correction command be sent to the correct motor.

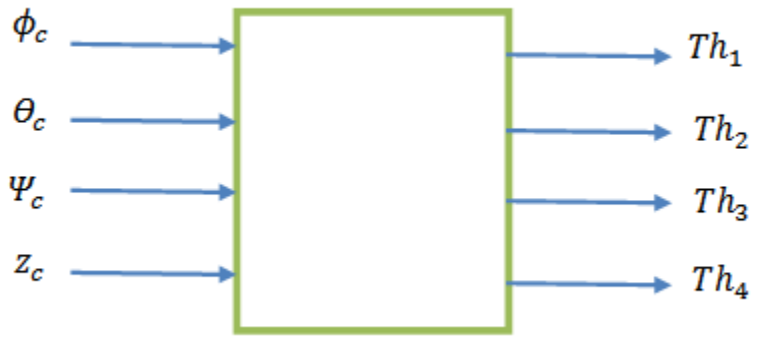


Figure 3.7: Commands Mixing Block

Let ϕ_c, θ_c, Ψ_c and z_c are roll correction command, pitch correction command, yaw correction command and altitude correction command respectively. Then throttle commands to motors are given by following relations:

$$Th_1 = z_c - \Psi_c - \theta_c$$

$$Th_3 = z_c - \Psi_c + \theta_c$$

$$Th_2 = z_c + \Psi_c + \phi_c$$

$$Th_4 = z_c + \Psi_c - \phi_c \tag{3.6}$$

If correction is required in altitude then throttles for all motors are increased or decreased by same amount which in turn increases or decreases the speeds of motors. As a result the altitude of quadrotor increases or decreases as desired. The roll angle can be increased by increasing the speed of motor two and decreasing the speed of motor four. For this purpose we have to increase the throttle two (Th_2) and decrease the throttle four (Th_4) by some amount as correction provided by our PID controller. Similarly change in Th_1 and Th_3 results in change of pitch angle. Motor one and motor three are rotating in clockwise direction so we may call it clockwise pair of motors, while motor two and motor four are rotating in anticlockwise direction so it's the

anticlockwise pair of motors. If increase in yaw angle is required then we have to increase the speed of anticlockwise pair and decrease the speed of clockwise pair for this purpose yaw angle correction is added into throttle command of motor two and motor four while yaw angle correction is subtracted from throttle command of motor one and motor three.

3.5 PID Controller Block

Current altitude and latitudes (Roll angle, pitch angle and yaw angle) of the quadrotor are feedback to the PID Controller block. There are four subtractors that calculate errors for correction in roll, correction in pitch, correction in yaw and correction in altitude by subtracting current altitude and latitudes from the desired altitude and latitudes. These errors are provided to PID controllers. Controllers operate on these errors and try to minimize them. There are four separate PID controllers for roll correction, pitch correction, yaw correction and altitude correction respectively.

PID can be tuned using several methods like Ziegler Nichols Method and standard formulas for first and second order systems, etc. we can design the controller in different ways by depending on situations. For example, we can design PD controller for systems having large overshoot response. Similarly PI controller can be designed for such systems having large steady state error. Similarly PID can be designed for such systems having damping response as well as steady state error. PID can be designed for linear as well as for non linear systems. But it is not recommended for large order system. For such systems we have to use some advance control design techniques such as sliding mode control, H-Infinity Controller, Adaptive Controllers, etc.

3.5.1 PID for Roll Angle Correction

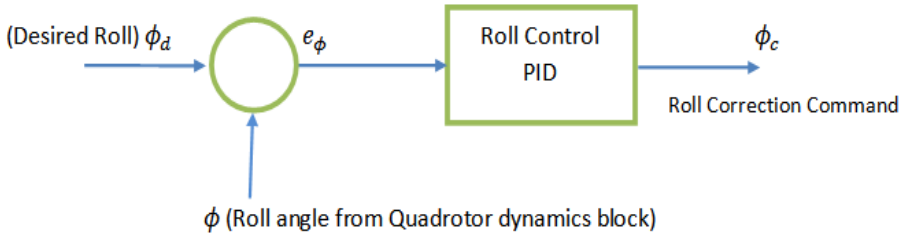


Figure 3.8: Roll Control PID Block

$$e_\phi = \phi_d - \phi$$

$$\phi_c = e_\phi K_p + K_i \int e_\phi \, dt - K_d P$$

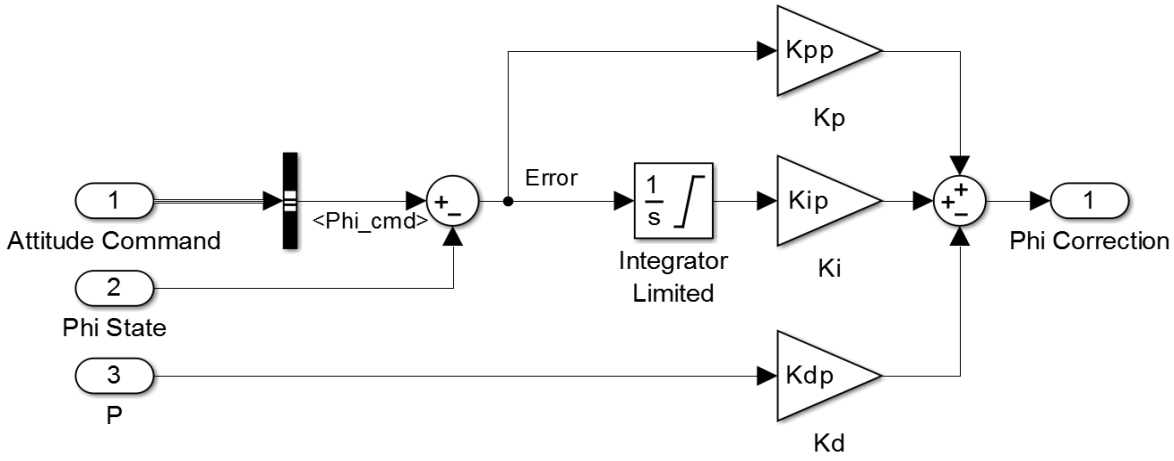


Figure 3.9: Simulink PID for Roll Correction

Error in roll angle e_ϕ is calculated by taking difference between desired roll angle ϕ_d and current roll angle ϕ. Then PID is applied to minimize error as shown in Figure 3.9. K_p is set to decrease the rise time but it cause overshoot in the system. So, K_d is set to reduce overshoot. If there exist some steady state error then K_i will be used for reducing the steady state error.

3.5.2 PID for Pitch Angle Correction

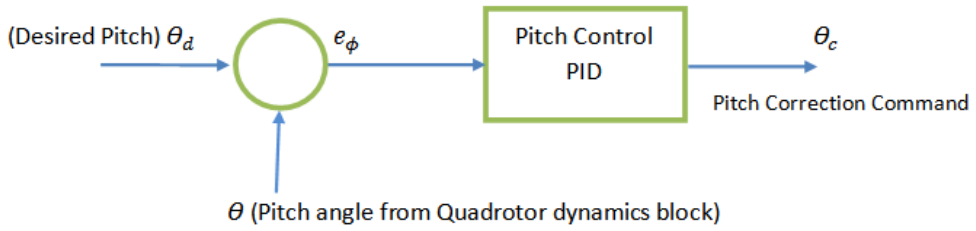


Figure 3.10: Pitch Control PID Block

$$e_\theta = \theta_d - \theta$$

$$\theta_c = e_\theta K_p + K_i \int e_\theta \, dt - K_d Q$$

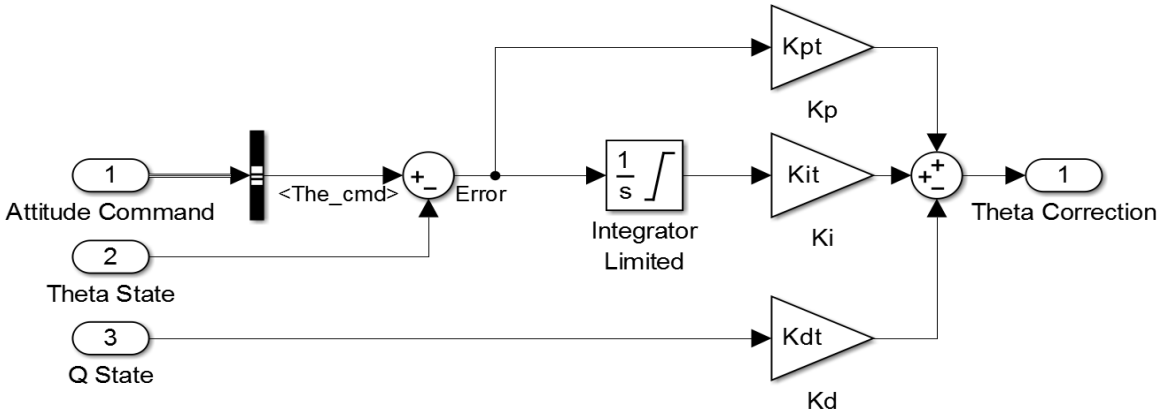


Figure 3.11: Simulink PID for Pitch Correction

Error in pitch angle e_θ is calculated by taking difference between desired pitch angle θ_d and current roll pitch angle θ. Then PID is applied as shown in Figure 3.11. Instead of taking derivative of errors in roll angle, error in pitch angle and error in yaw angle, I directly take the roll angle rate, pitch angle rate and yaw angle rate from the quadrotor dynamic block.

3.5.3 PID for Yaw Angle Correction

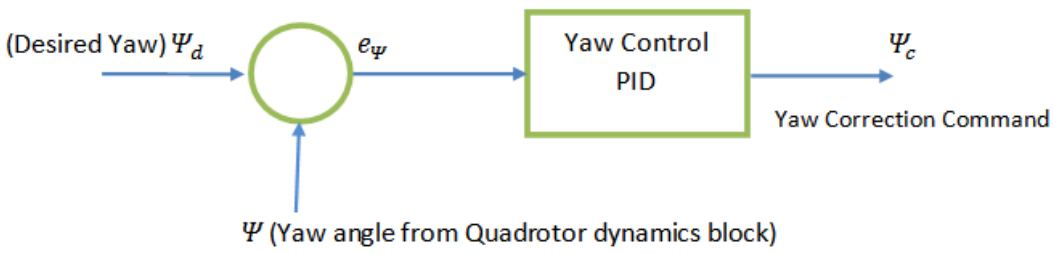


Figure 3.12: Yaw Control PID Block

$$e_\Psi = \Psi_d - \Psi$$

$$\Psi_c = e_\Psi K_p + K_i \int e_\Psi \, dt - K_d R$$

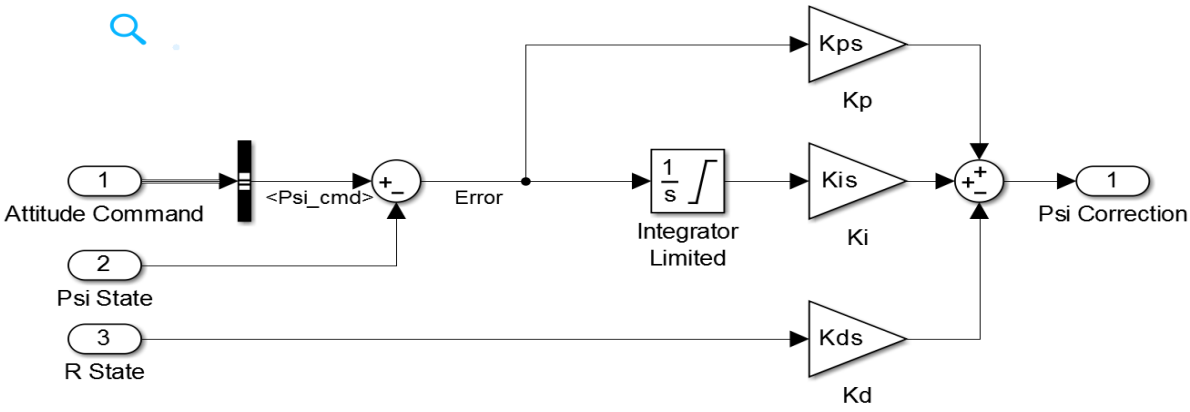


Figure 3.13: Simulink PID for Yaw Correction

Error in yaw angle e_Ψ is calculated by taking difference between desired yaw angle Θ_Ψ and current yaw angle Ψ. Then PID is applied as shown in Figure 3.13. Integrator limit is applied to prevent the output from saturation because it can damage the actuator of the system and also have a bad impact on system energy as it causes instability in the system.

3.5.4 PID for Altitude Correction

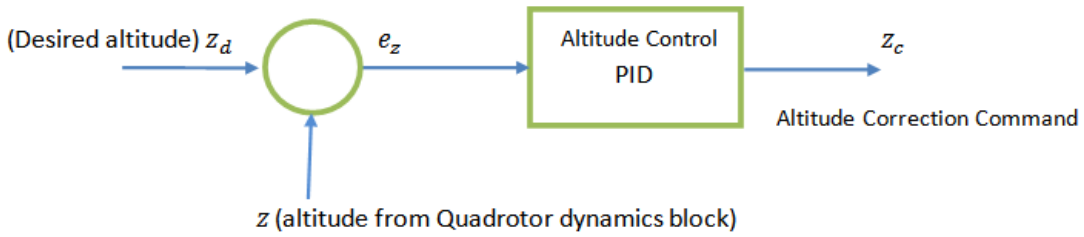


Figure 3.14: Altitude Control PID Block

$$e_z = z_d - z$$

$$z_c = e_z K_p + K_i \int e_z \, dt - K_d \frac{d}{dt} e_z + Gravity\ offset$$

$$Gravity\ offset = \frac{\sqrt{\frac{mg}{4}} - b}{cr}$$

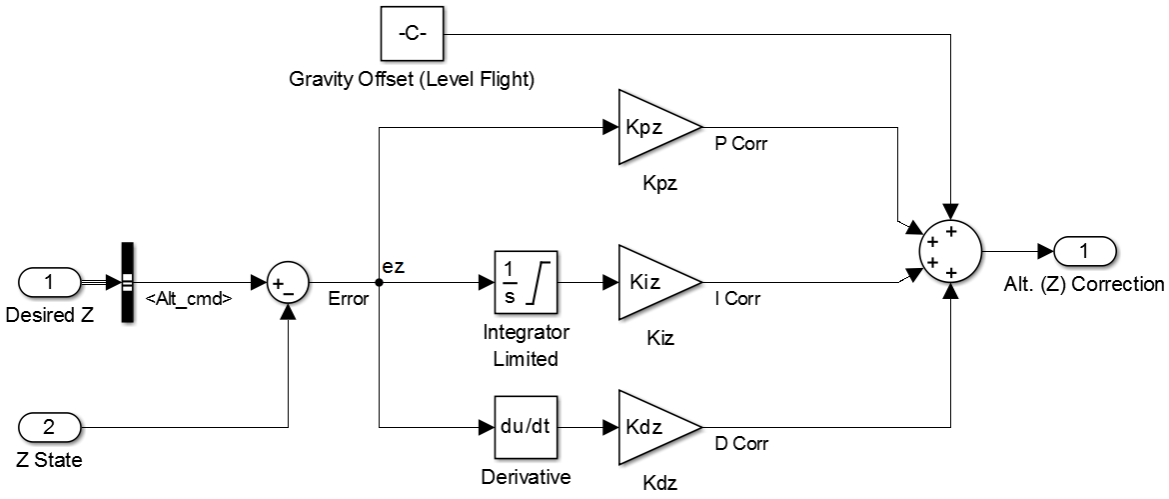


Figure 3.15: Simulink PID for Altitude Correction

3.6 PID Simulation Results

3.6.1 Closed Loop Step Response for Roll Angle

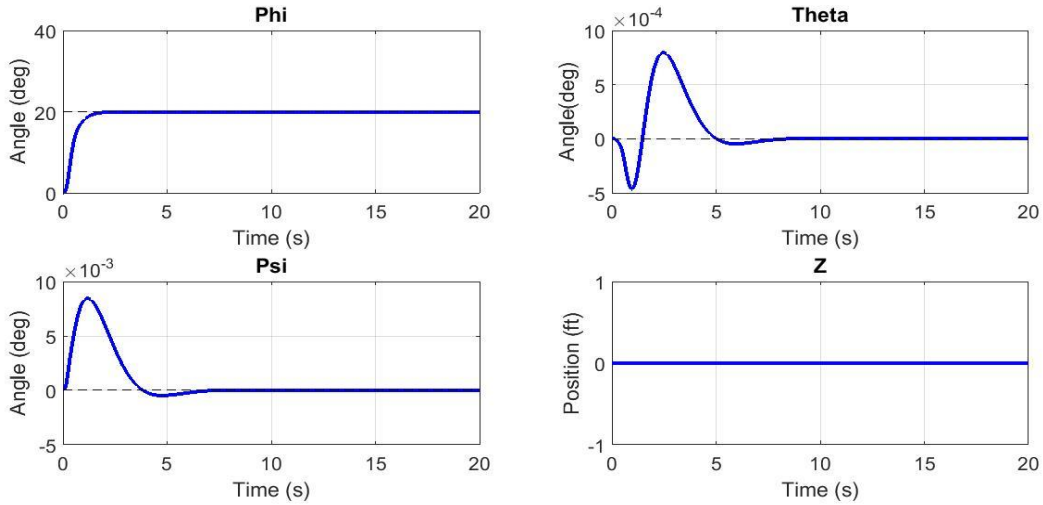


Figure 3.17: Roll Angle Closed Loop Step Response

Characteristics	Values
Rise Time	0.8297 Sec
Settling Time	1.6416 Sec
Overshoot	0 %
Peak	20 degree
Peak Time	14.5 Sec

Table 3.1: Roll Angle Closed Loop Characteristics

Controller Gain	Values
K_p	4
K_i	0
K_d	2

Table 3.2: Roll Angle PID Controller Gains

3.6.2 Closed Loop Step Response for Pitch Angle

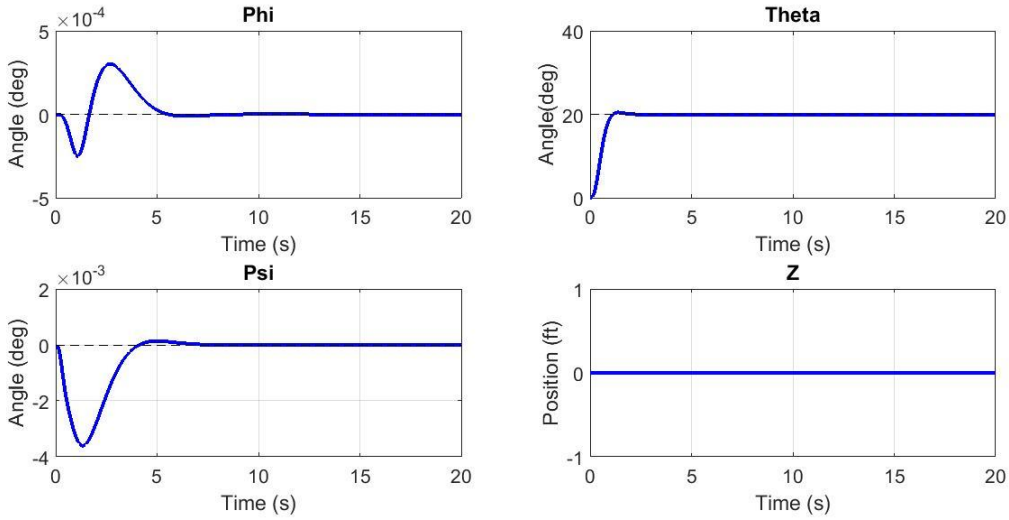


Figure 3.17: Pitch Angle Closed Loop Step Response

Characteristics	Values
Rise Time	0.6362 Sec
Settling Time	1.5003 Sec
Overshoot	2.194 %
Peak	20.4388 degree
Peak Time	1.36 Sec

Table 3.3: Pitch Angle Closed Loop Characteristics

Controller Gain	Values
K_p	2
K_i	0
K_d	1

Table 3.4: Pitch Angle PID Controller Gains

3.6.3 Closed Loop Step Response for Yaw Angle

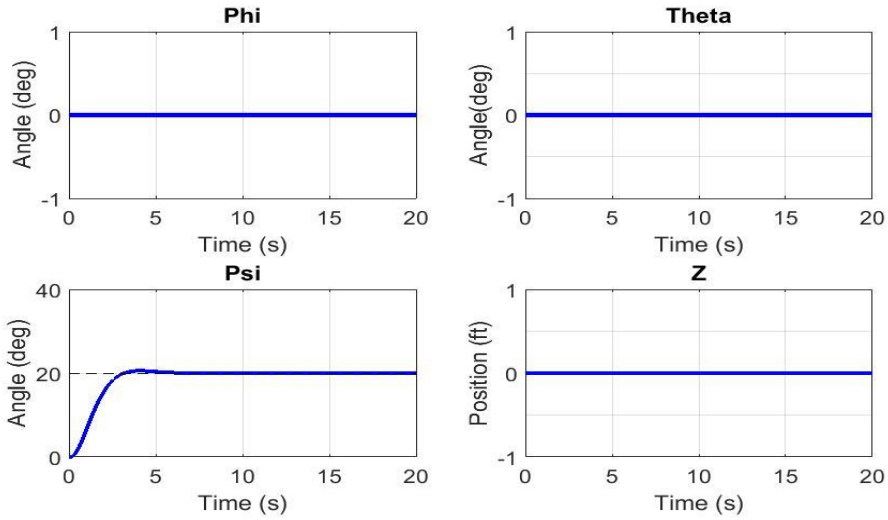


Figure 3.18: Yaw Angle Closed Loop Step Response

Characteristics	Values
Rise Time	1.9118 Sec
Settling Time	4.8940 Sec
Overshoot	2.8245 %
Peak	20.5649 degree
Peak Time	4.0600 Sec

Table 3.5: Yaw Angle Closed Loop Characteristics

Controller Gain	Values
K_p	3
K_i	0
K_d	4

Table 3.6: Yaw Angle PID Controller Gains

3.6.4 Closed Loop Step Response for Altitude

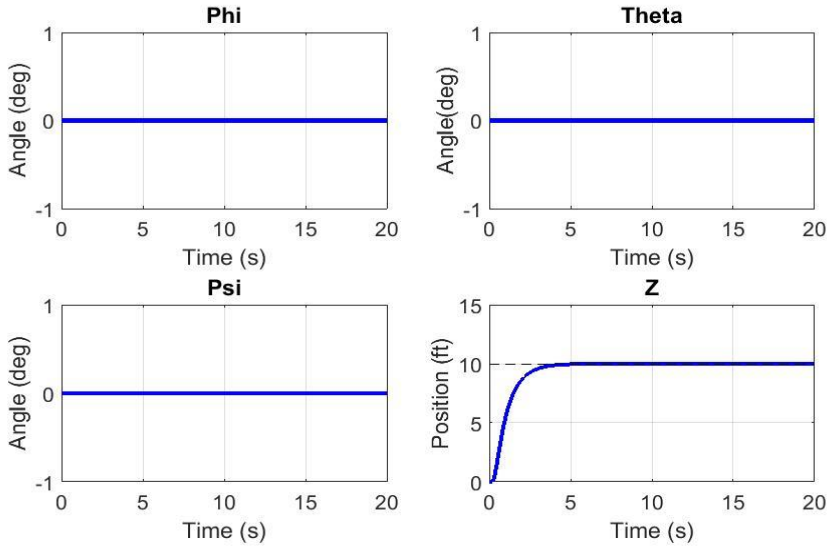


Figure 3.19: Altitude Closed Loop Step Response

Characteristics	Values
Rise Time	1.8516 Sec
Settling Time	3.6043 Sec
Overshoot	0 %
Peak	10 feet
Peak Time	20 Sec

Table 3.7: Altitude Closed Loop Characteristics

Controller Gain	Values
K_p	10
K_i	0
K_d	11

Table 3.8: Altitude PID Controller Gains

3.6.5 Closed Loop Step Response

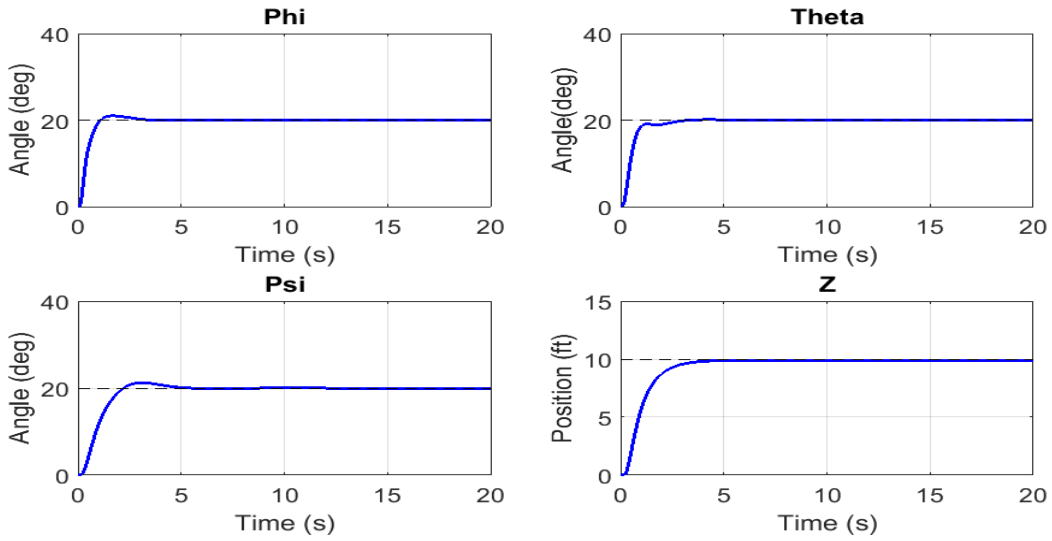


Figure 3.20: Closed Loop Step Response

Characteristics	Roll Values	Pitch Values	Yaw Values	Altitude Values
Rise Time	0.6469 Sec	0.6865 Sec	1.37 Sec	1.739 Sec
Settling Time	2.7579 Sec	2.66 Sec	4.502 Sec	3.45 Sec
Overshoot	4.93 %	0.5927 %	6.153 %	3.1607e-06 %
Peak	20.9 Degree	20.1 Degree	21.23 Degree	9.9108 Degree
Peak Time	1.6 Sec	4.24 Sec	3.1 Sec	14.26 Sec

Table 3.9: Closed Loop Characteristics

Chapter 4

BACKSTEPPING STRATEGY

In this chapter we will derive the alternative form of mathematical model (2.24). And implement backstepping controller for the altitude, latitude and position control of quadrotor.

We can write from Newton's law:

$$\boldsymbol{m\dot{v} = R.f_B = mg\hat{e}_z - f_t R.\hat{e}_3} \tag{4.1}$$

Transformation matrix between body angular velocities $[P\ Q\ R]$ and orientation angles $[\dot{\phi}\ \dot{\theta}\ \dot{\Psi}]$ can be considered unity matrix for small angles of movement [6]. So we can write $[P\ Q\ R]$ $=[\dot{\phi}\ \dot{\theta}\ \dot{\Psi}]$. Now the dynamic model in the inertial frame is:

$$\ddot{x} = -\frac{f_t}{m}[s(\phi)s(\psi) + s(\theta)c(\phi)c(\psi)]$$

$$\ddot{y} = -\frac{f_t}{m}[c(\phi)s(\psi)s(\theta) - c(\psi)s(\phi)]$$

$$\ddot{z} = g - \frac{f_t}{m}[c(\phi)c(\theta)]$$

$$\ddot{\phi} = \frac{I_y - I_z}{I_x}\dot{\theta}\dot{\psi} + \frac{\tau_\phi}{I_x}$$

$$\ddot{\theta} = \frac{I_z - I_x}{I_y}\dot{\phi}\dot{\psi} + \frac{\tau_\theta}{I_y}$$

$$\ddot{\psi} = \frac{I_x - I_y}{I_z}\dot{\phi}\dot{\theta} + \frac{\tau_\psi}{I_z} \tag{4.2}$$

4.1 State Space Model

Writing the acquired mathematical model into a state space form simplifies the implementation of backstepping control technique**.**

$$\dot{X} = f(X, U)$$

X= State vector

U=Input vector

State vector:

$$X = [\emptyset \quad \dot{\emptyset} \quad \theta \quad \dot{\theta} \quad \Psi \quad \dot{\Psi} \quad \dot{z} \quad z \quad \dot{x} \quad x \quad \dot{y} \quad y]^T \tag{4.3}$$

State vector can be written as:

$$X = [x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7 \quad x_8 \quad x_9 \quad x_{10} \quad x_{11} \quad x_{12}] \tag{4.4}$$

Control inputs can be written as:

$$U = [U_{ft} \quad U_\phi \quad U_\theta \quad U_\psi] \tag{4.5}$$

From equation set (2.21) we can write control inputs

U_{ft}= $c_T(\mathrm{w}_1^2 + \mathrm{w}_2^2 + \mathrm{w}_3^2 + \mathrm{w}_4^2) = f_t$

$U_\phi = c_T(\mathrm{w}_4^2 - \mathrm{w}_2^2) = \frac{\tau_\phi}{l}$

$U_\theta = c_T(\mathrm{w}_3^2 - \mathrm{w}_1^2) = \frac{\tau_\theta}{l}$

$$U_\psi = c_q(\mathrm{w}_2^2 + \mathrm{w}_4^2 - \mathrm{w}_1^2 - \mathrm{w}_3^2) = \tau_\psi \tag{4.6}$$

From (4.2), (4.4) and (4.6) we obtain the following state space model:

$$\dot{X} = f(X, U)$$

$\dot{x_1} = x_2$

$\dot{x_2} = \frac{I_y - I_z}{I_x} x_4 x_6 + \frac{l U_\phi}{I_x}$

$\dot{x_3} = x_4$

$\dot{x_4} = \frac{I_z - I_x}{I_y} x_2 x_6 + \frac{l U_\theta}{I_y}$

$\dot{x_5} = x_6$

$$\dot{x_6} = \frac{I_x - I_y}{I_z} x_2 x_4 + \frac{U_\psi}{I_z}$$

$$\dot{x_7} = x_8$$

$$\dot{x_8} = g - \frac{U_{ft}}{m}[c(x_1)c(x_3)]$$

$$\dot{x_9} = x_{10}$$

$$\dot{x_{10}} = -\frac{U_{ft}}{m}[s(x_1)s(x_5) + s(x_3)c(x_1)c(x_5)]$$

$$\dot{x_{11}} = x_{12}$$

$$\dot{x_{12}} = -\frac{U_{ft}}{m}[c(x_1)s(x_5)s(x_3) - c(x_5)s(x_1)] \quad (4.7)$$

Let

$c_1 = \frac{I_y - I_z}{I_x}$, $d_1 = \frac{l}{I_x}$,

$c_2 = \frac{J_r}{I_x}$, $d_2 = \frac{l}{I_y}$

$c_3 = \frac{I_z - I_x}{I_y}$, $d_3 = \frac{l}{I_z}$

$c_4 = \frac{J_r}{I_y}$, $c_5 = \frac{I_x - I_y}{I_z}$

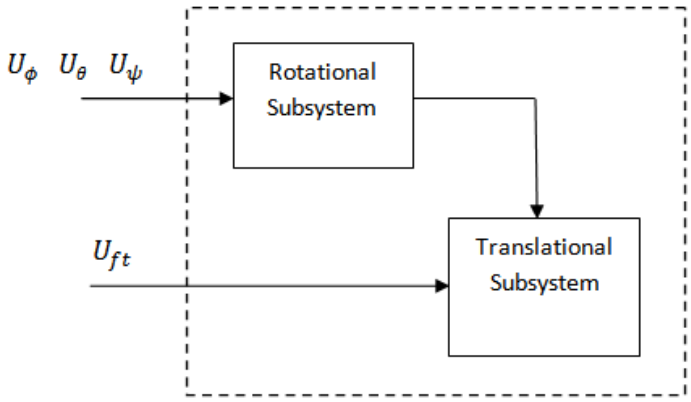


Figure 4.1: Quadrotor Dynamics Block Diagram

From state space (4.7) we can divide quadrotor dynamics into two subsystems; i.e. translational subsystem and rotational subsystem as shown in Figure 4.1. It is clear from state space (4.7) that translational subsystem is an under actuated system as it depends on roll, pitch, yaw angles and the translational state variables. The rotational subsystem is fully-actuated and only depends on the rotational states.

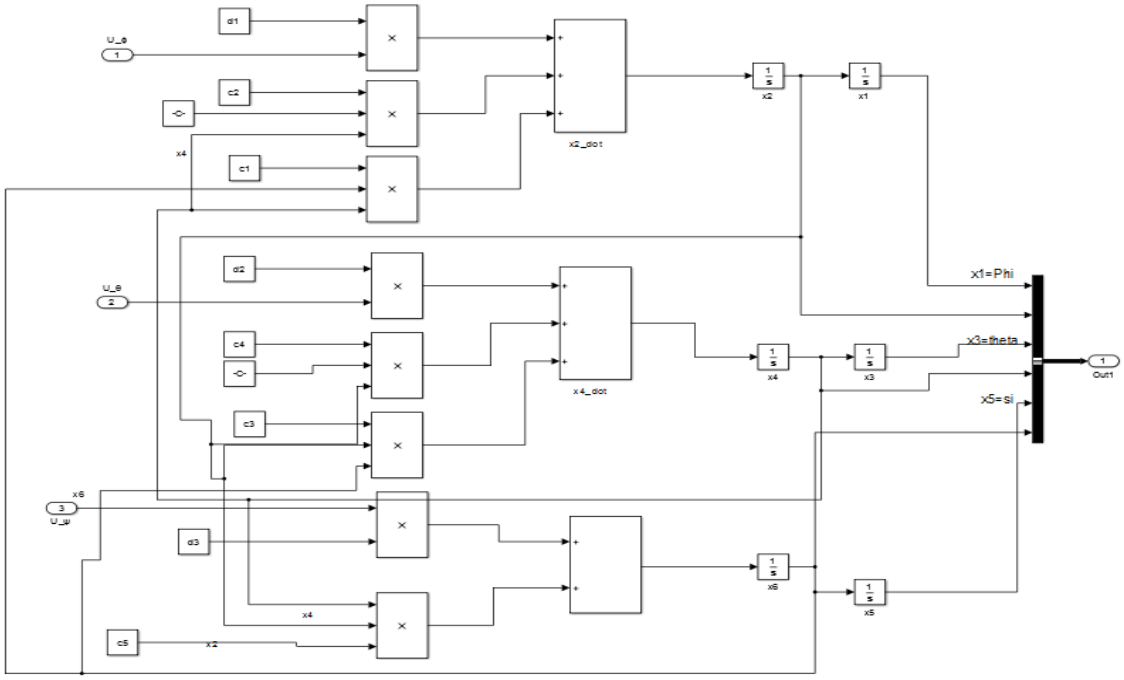


Figure 4.2: Rotational Subsystem Block Simulink

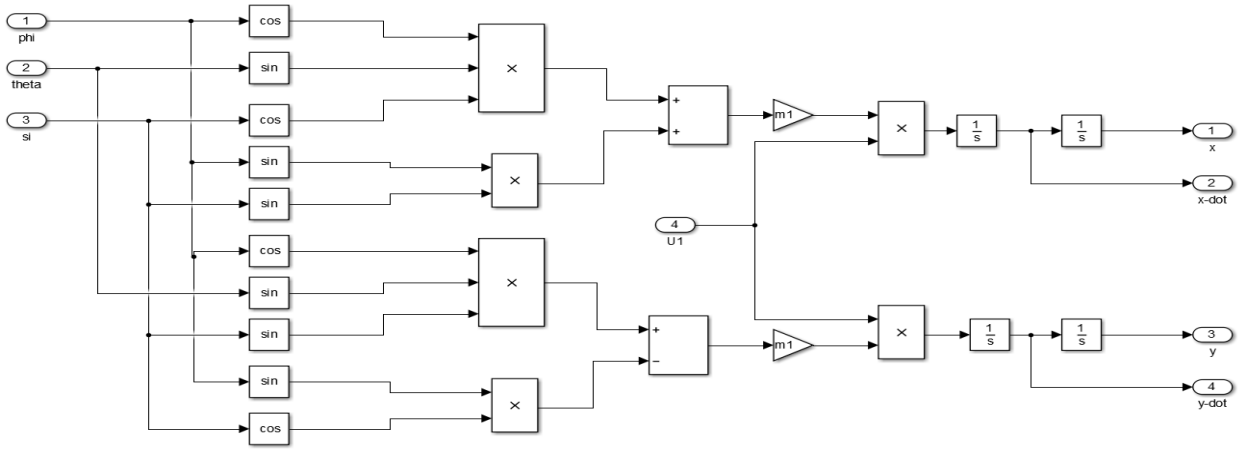


Figure 4.3: Translational Subsystem Block Simulink

4.2 Introduction to Backstepping

Backstepping is such a control technique in which control laws are designed using recursive control algorithm for certain states [29].These certain states are named as virtual control for the system model. Many control algorithms linearize the nonlinear models e.g feedback linearization while backstepping does not approximate the nonlinear terms in the system model. Aproximation of nonlinear terms leads to such a design which is not so much sophisticated and robust. While keeping nonlinear terms can contribute to the stability of the system.

4.3 Roll Controller

In state space model (4.7) first two equations represent the roll angle and roll angle rate.

$$\dot{x_1} = x_2 \tag{4.8}$$

$$\dot{x_2} = c_1 x_4 x_6 - \Omega_r c_2 x_4 + d_1 U_\phi \tag{4.9}$$

Term $\Omega_r c_2 x_4$ represent gyroscopic precession [20]. Gyroscopic precession phenomenon occurs when the axis of rotation of rotating body is shifted. Gyroscopic forces on body depend on inertia of motors (J_r). Rolling rate (P), Pitch rate (Q) and speed of each motor(w_i).

In roll angle subsystem only last state is a function of control input U_ϕ hence it is in strict feedback form. Positive definite Lyapunov function:

$$v(e_1) = \frac{1}{2} e_1^2 \tag{4.10}$$

e_1 is the error that is difference between desired roll angle and actual roll angle

$$e_1 = x_1 d - x_1 \tag{4.11}$$

Time derivative of Lyapunov function is derived to be:

$$\dot{v}(e_1) = e_1 \dot{e}_1 \tag{4.12}$$

$$\dot{v}(e_1) = e_1 (\dot{x}_1 d - \dot{x}_1) \tag{4.13}$$

From equation (4.8) we can rewrite this as:

$$\dot{v}(e_1) = e_1(\dot{x}_1 d - x_2) \tag{4.14}$$

Krasovskii-LaSalle principle says that, if the time derivative of a positive definite Lyapunov function is negative semi-definite then the system is guaranteed to be a stable system [29]. Stabilization of e_1 is obtained by virtual control inputx_2.

$$x_2 = \dot{x}_{1d} + k_1 e_1 \ (k_1 > 0) \tag{4.15}$$

e_2 is the error variable that is difference between the state x_2 and its desire value.

$$e_2 = x_2 - \dot{x}_{1d} - k_1 e_1 \tag{4.16}$$

Augmented Lyapunov function of coordinate (e_1, e_2) can be written by:

$$v(e_1, e_2) = \frac{1}{2}(e_1^2 + e_2^2) \tag{4.17}$$

By writing Lyapunov function time derivative in the new coordinate (e_1, e_2):

$$\dot{v}(e_1, e_2) = e_1\dot{e}_1 + e_2\dot{e}_2 \tag{4.18}$$

$$\dot{v}(e_1, e_2) = e_1(\dot{x}_1 d - x_2) + e_2(\dot{x_2} - \ddot{x}_{1d} - k_1\dot{e_1})$$

$$\dot{v}(e_1, e_2) = e_1(\dot{x}_1 d - x_2) + e_2\left[c_1 x_4 x_6 + \ \mathrm{d}_1 U_\phi - \ \Omega_\mathrm{r} c_2 x_4 - \ddot{x}_{1d} - k_1(\dot{x}_{1d} - \dot{x}_1)\right] \tag{4.19}$$

$$\because e_2 + k_1 e_1 = \dot{x}_1 - \dot{x}_1 d$$

So (4.19) can be written as:

$$\dot{v}(e_1, e_2) = e_1(-e_2 - k_1 e_1) + e_2\left(c_1 x_4 x_6 + \ \mathrm{d}_1 U_\phi - \ \Omega_\mathrm{r} c_2 x_4\right) - e_2(\ddot{x}_{1d} - k_1(\dot{x}_{1d} - \dot{x}_1))$$

$$\dot{v}(e_1, e_2) = e_2\left(c_1 x_4 x_6 + \ \mathrm{d}_1 U_\phi - \ \Omega_\mathrm{r} c_2 x_4\right) - e_2\left(\ddot{x}_{1d} - k_1(e_2 + k_1 e_1)\right) - e_1 e_2 - k_1 e_1^2) \tag{4.20}$$

The control input U_ϕ is extracted from (4.20) $\left(\ddot{x}_{1,2,3d} = 0\right)$ satisfying $\dot{v}(e_1, e_2) < 0$

$$U_\phi = \frac{1}{d_1}[e_1 - x_4 x_6 c_1 - x_4 c_2 \Omega_r - c_1(e_2 + k_1 e_1) - k_2 e_2] \quad (4.21)$$

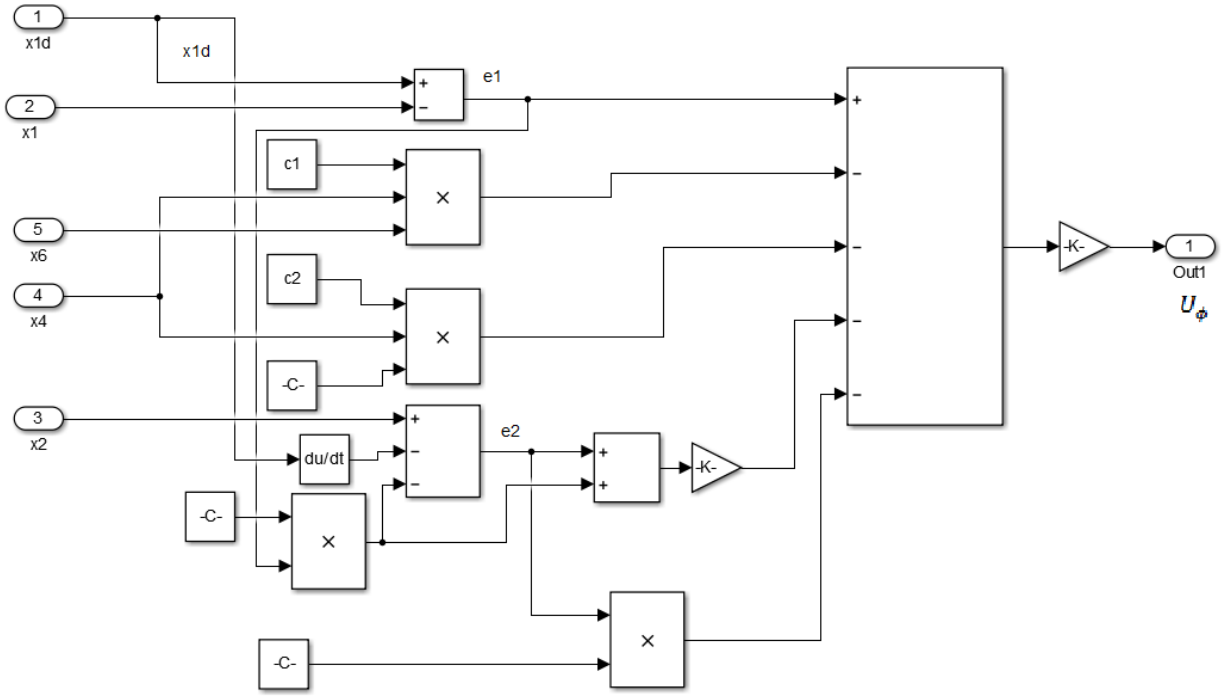


Figure 4.3: Roll Controller Block Simulink

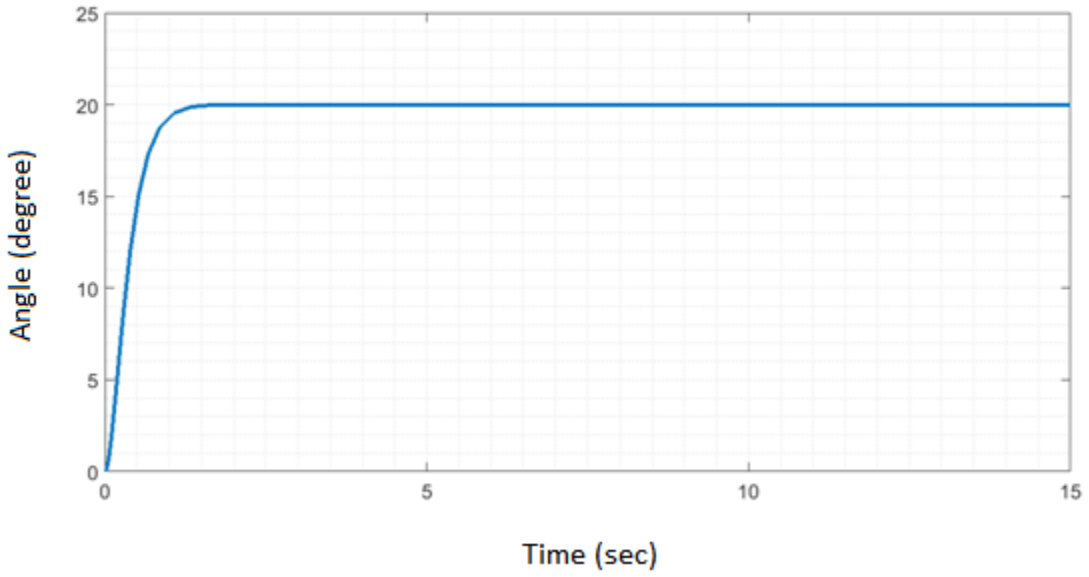


Figure 4.4: Closed Loop Step Response Roll Angle

Characteristics	Values
Rise Time	0.6548 Sec
Settling Time	1.1206 Sec
Overshoot	4.4960e-06 %
Peak	20 Degree
Peak Time	3.44 Sec

Table 4.1: Roll Angle Closed Loop Characteristics

Gains	Values
k_1	5
k_2	5

Table 4.2: Roll Angle Backstepping Controller Gains

4.4 Pitch Controller

In state space model (4.7) following two equations represent the pitch angle and pitch angle rate.

$$\dot{x_3} = x_4 \tag{4.22}$$

$$\dot{x_4} = \mathrm{c}_3 x_2 x_6 + \Omega_\mathrm{r} c_4 x_2 + \mathrm{d}_2 U_\theta \tag{4.23}$$

Term $\Omega_\mathrm{r} c_4 x_2$ represent gyroscopic precession [20]. In pitch angle subsystem only last state is a function of control input U_θ hence it is in strict feedback form. Positive definite Lyapunov function:

$$v(e_3) = \frac{1}{2} e_3^2 \tag{4.24}$$

e_3 is the error that is difference between desired pitch angle and actual pitch angle

$$e_3 = x_3 d - x_3 \tag{4.25}$$

Time derivative of Lyapunov function is derived to be:

$$\dot{v}(e_3) = e_3\dot{e}_3 \tag{4.26}$$

$$\dot{v}(e_3) = e_3(\dot{x}_3 d - \dot{x}_3) \tag{4.27}$$

From equation (4.22) I can rewrite this as:

$$\dot{v}(e_3) = e_3(\dot{x}_3 d - x_4) \tag{4.28}$$

Krasovskii-LaSalle principle says that, if the time derivative of a positive definite Lyapunov function is negative semi-definite then system is guaranteed to be a stable system [29]. Stabilization of e_3 is obtain by virtual control inputx_4.

$$x_4 = \dot{x}_{3d} + k_3 e_3 \ (k_3 > 0) \tag{4.29}$$

e_4 is the error variable that is difference between the state x_4 and its desire value.

$$e_4 = x_4 - \dot{x}_{3d} - k_3 e_3 \tag{4.30}$$

Augmented Lyapunov function of coordinate (e_3, e_4) can be written by:

$$v(e_3, e_4) = \frac{1}{2}(e_3^2 + e_4^2) \tag{4.31}$$

By writing Lyapunov function time derivative in the new coordinate (e_3, e_4):

$$\dot{v}(e_3, e_4) = e_3\dot{e}_3 + e_4\dot{e}_4 \tag{4.32}$$

$$\dot{v}(e_3, e_4) = e_3(\dot{x}_3 d - x_4) + e_4(\dot{x_4} - \ddot{x}_{3d} - k_3\dot{e_3}) \tag{4.33}$$

$$\dot{v}(e_3, e_4) = e_3(\dot{x}_3 d - x_4) + e_4[c_3 x_2 x_6 + \Omega_r c_4 x_2 + d_2 U_\theta - \ddot{x}_{3d} - k_3(\dot{x}_{3d} - \dot{x}_3)] \tag{4.34}$$

$$\because e_4 + k_3 e_3 = \dot{x}_3 - \dot{x}_3 d$$

So (4.34) can be written as:

$$\dot{v}(e_3, e_4) = e_3(-e_4 - k_3 e_3) + e_4(c_3 x_2 x_6 + \Omega_r c_4 x_2 + d_2 U_\theta) - e_4(\ddot{x}_{3d} - k_3(\dot{x}_{3d} - \dot{x}_3))$$

$$\dot{v}(e_3, e_4) = e_4(c_3 x_2 x_6 + \Omega_r c_4 x_2 + d_2 U_\theta) - e_4\big(\ddot{x}_{3d} - k_3(-e_4 - k_3 e_3)\big) - e_3 e_4 - k_3 e_3^2) \tag{4.35}$$

The control input U_θ is extracted from (4.35) $\left(\ddot{x}_{1,2,3d} = 0\right)$ satisfying $\dot{v}(e_1, e_2) < 0$

$$U_\theta = \frac{1}{\mathrm{d}_2}[e_3 - x_2 x_6 c_3 - x_2 c_2 \Omega_\mathrm{r} - c_3(e_4 + k_3 e_3) - k_4 e_4] \tag{4.36}$$

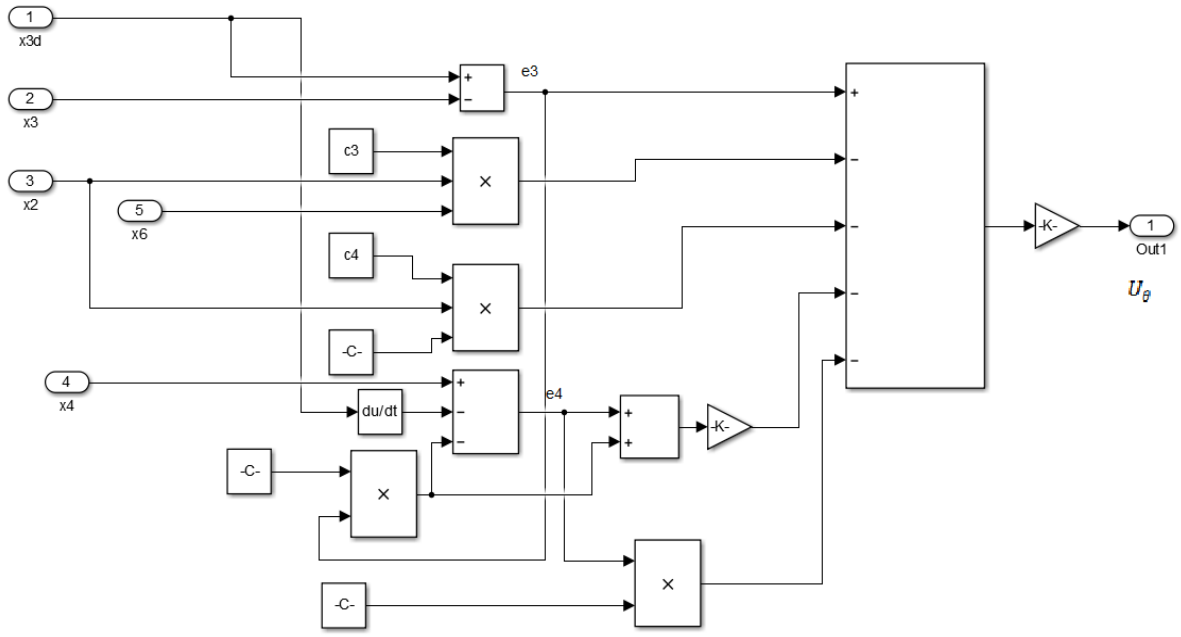


Figure 4.5: Pitch Controller Block Simulink

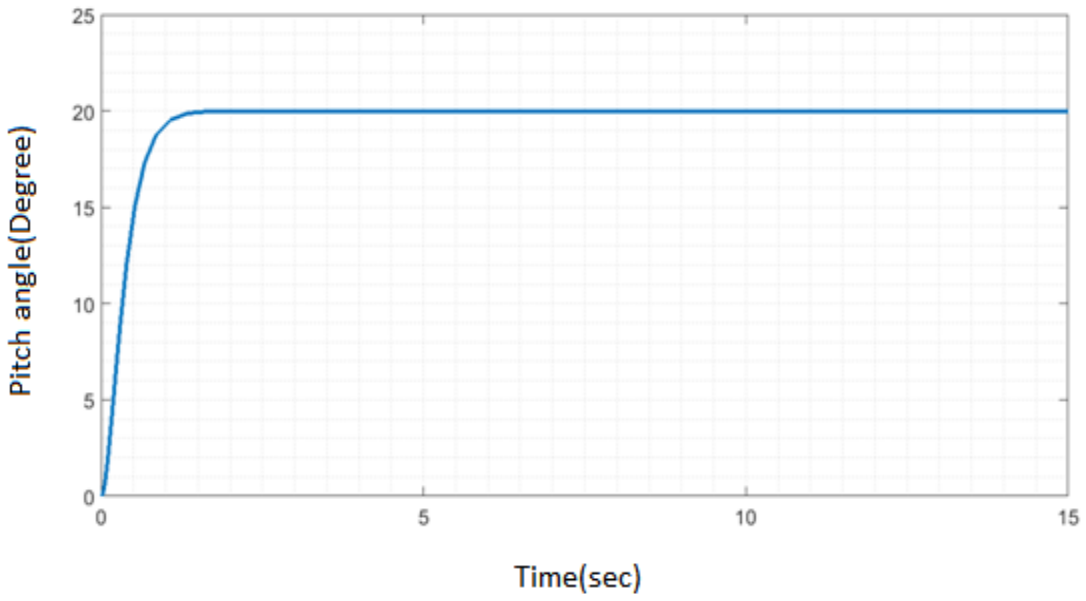


Figure 4.6: Closed Loop Step Response Pitch Angle

Characteristics	Values
Rise Time	0.6548 Sec
Settling Time	1.1206 Sec
Overshoot	4.4960e-06 %
Peak	20 Degree
Peak Time	3.44 Sec

Table 4.3: Pitch Angle Closed Loop Characteristics

Gains	Values
k_3	5
k_4	5

Table 4.4: Pitch Angle Backstepping Controller Gains

4.5 Yaw Controller

Using the same steps as I derived U_ϕ and U_θ . Yaw controller U_ψ is derived:

$$U_\psi = \frac{1}{\mathrm{d}_3}[e_5 - x_2 x_4 c_5 - c_5(e_6 + k_5 e_5) - k_6 e_6] \tag{4.37}$$

Where:

$$e_5 = x_5 d - x_5 \tag{4.38}$$

$$e_6 = x_6 - \dot{x}_{5d} - k_5 e_5 \tag{4.39}$$

k_5 and k_6 are positive constants.

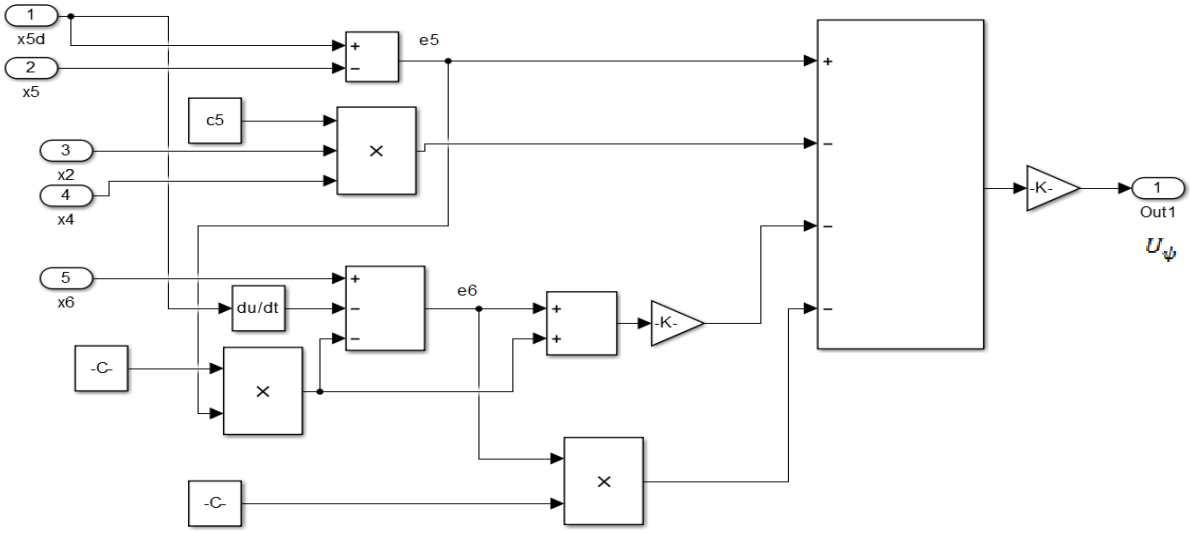


Figure 4.7: Yaw Controller Block Simulink

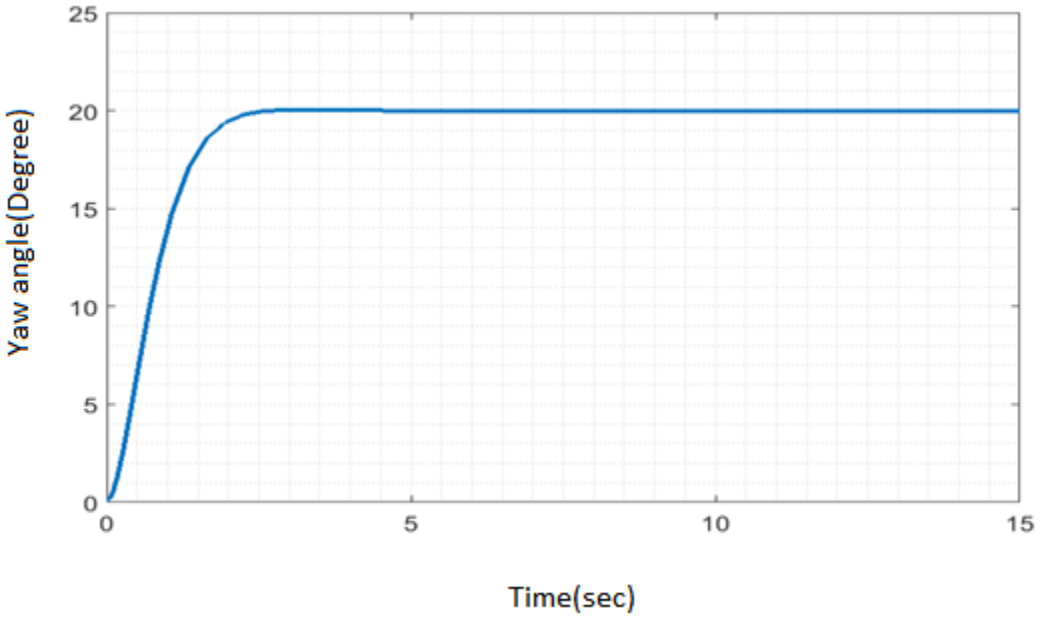


Figure 4.8: Closed Loop Step Response Yaw Angle

Characteristics	Values

Rise Time	1.3022 Sec
Settling Time	2.1033 Sec
Overshoot	0.1857 %
Peak	20.0371 Degree
Peak Time	3.145 Sec

Table 4.5: Yaw Angle Closed Loop Characteristics

Gains	Values
k_5	2
k_6	2

Table 4.6: Yaw Angle Backstepping Controller Gains

4.6 Altitude Controller

In state space model (4.7) following two equations represent the altitude and altitude rate.

$$\dot{x_7} = x_8 \tag{4.40}$$

$$\dot{x_8} = g - \frac{U_{ft}}{m}[c(x_1)c(x_3)] \tag{4.41}$$

In altitude subsystem only last state is a function of control input U_θ hence it is in strict feedback form. Positive definite Lyapunov function:

$$v(e_7) = \frac{1}{2}e_7^2 \tag{4.42}$$

e_7 is the error that is difference between desired altitude and actual altitude:

$$e_7 = x_7 d - x_7 \tag{4.43}$$

Time derivative of Lyapunov function is derived to be:

$$\dot{v}(e_7) = e_7\dot{e}_7 \tag{4.44}$$

$$\dot{v}(e_7) = e_7(\dot{x}_7 d - \dot{x}_7) \tag{4.45}$$

From equation (4.40) I can rewrite this as:

$$\dot{v}(e_7) = e_7(\dot{x}_7 d - x_8) \tag{4.46}$$

Krasovskii-LaSalle principle says that, if the time derivative of a positive definite Lyapunov function is negative semi-definite then system is guaranteed to be a stable system [29]. Stabilization of e_7 is obtain by virtual control inputx_8.

$$x_8 = \dot{x}_{7d} + k_7 e_7 \ (k_7 > 0) \tag{4.47}$$

e_8 is the error variable that is difference between the state x_8 and its desire value.

$$e_8 = x_8 - \dot{x}_{7d} - k_7 e_7 \tag{4.48}$$

Augmented Lyapunov function of coordinate (e_7, e_8) can be written by:

$$v(e_7, e_8) = \frac{1}{2}(e_7^2 + e_8^2) \tag{4.49}$$

By writing Lyapunov function time derivative in the new coordinate (e_7, e_8):

$$\dot{v}(e_7, e_8) = e_7\dot{e}_7 + e_8\dot{e}_8 \tag{4.50}$$

$$\dot{v}(e_7, e_8) = e_7(\dot{x}_7 d - x_8) + e_8(\dot{x_8} - \ddot{x}_{7d} - k_7\dot{e_7}) \tag{4.51}$$

$$\dot{v}(e_7, e_8) = e_7(\dot{x}_7 d - x_8) + e_8\left[g - \frac{U_{ft}}{m}[c(x_1)c(x_3)] - \ddot{x}_{7d} - k_7(\dot{x}_{7d} - \dot{x}_7)\right] \tag{4.52}$$

$$\because e_8 + k_7 e_7 = \dot{x}_7 - \dot{x}_7 d$$

So (4.52) can be written as:

$$\dot{v}(e_7, e_8) = e_7(-e_8 - k_7 e_7) + e_8\left(\frac{U_{ft}}{m}[c(x_1)c(x_3)]\right) - e_8(\ddot{x}_{7d} - k_7(\dot{x}_{7d} - \dot{x}_7))$$

$$\dot{v}(e_7, e_8) = e_8\left(\frac{U_{ft}}{m}[c(x_1)c(x_3)]\right) - e_8\big(\ddot{x}_{7d} - k_7(-e_8 - k_7 e_7)\big) - e_7 e_8 - k_7 e_7^2) \tag{4.53}$$

The control input U_{ft} is extracted from (4.53) $(\ddot{x}_{7d} = 0)$ satisfying $\dot{v}(e_7, e_8) < 0$

$$U_{ft} = \frac{m}{c(x_1)c(x_3)}[e_7 + g - k_7(e_8 + k_7 e_7) - k_8 e_8] \tag{4.54}$$

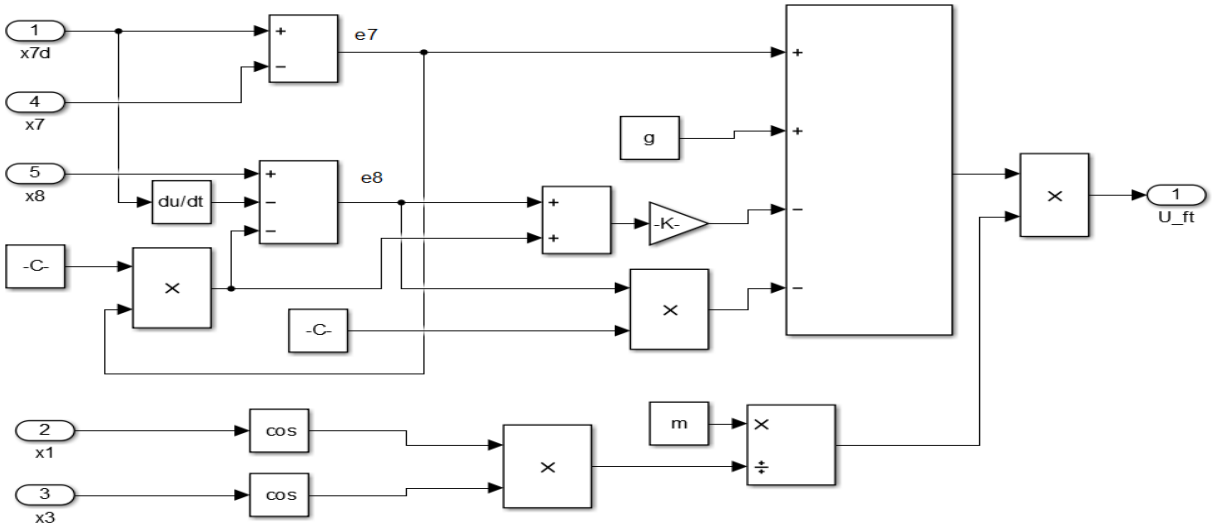


Figure 4.9: Altitude Controller Block Simulink

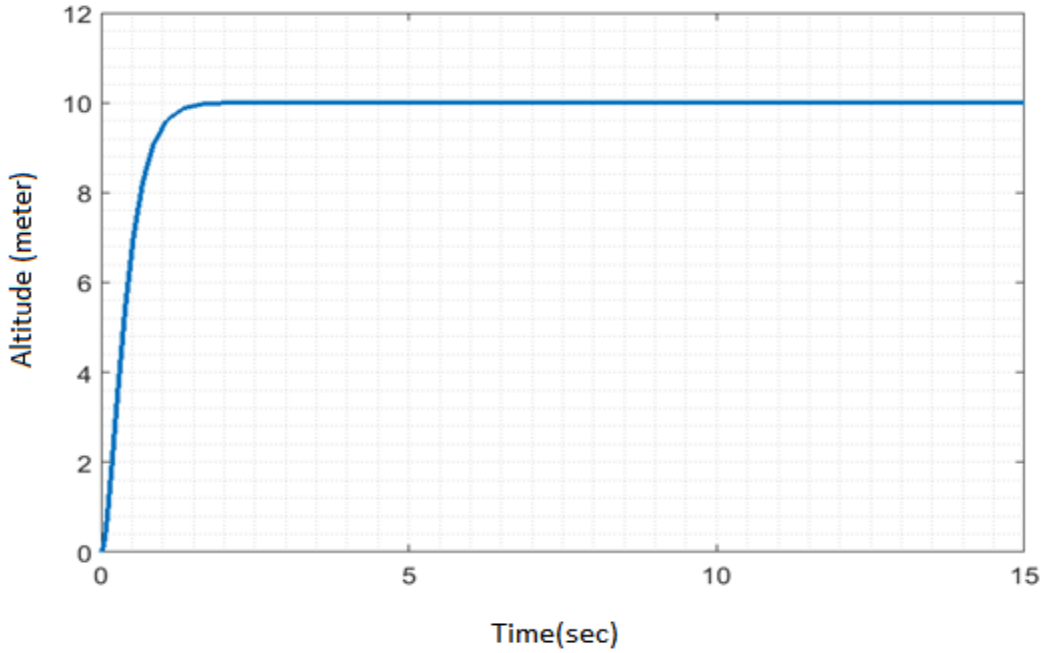


Figure 4.10: Closed Loop Step Response Altitude

Characteristics	Values
Rise Time	0.7195 Sec
Settling Time	1.2633 Sec
Overshoot	4.5488e-06 %
Peak	10.0 meter
Peak Time	3.745 Sec

Table 4.7: Altitude Control Closed Loop Characteristics

Gains	Values
k_7	5
k_8	4

Table 4.8: Altitude Backstepping Controller Gains

4.7 Position Controller

x and y positions of quadrotor are not directly controlled by one of the four control laws $[U_{ft} \;\; U_\phi \;\; U_\theta \;\; U_\psi]$. Position control is performed by rolling and pitching the quadrotor. So the position controller will generate reference roll $\phi_R$and reference pitch θ_R for the roll controller and pitch controller respectively. Then roll controller and pitch controller will produce U_ϕ and U_θ to move the quadrotor to desired position as shown in Figure 4.11.

In state space model (4.7) following equations represent x , y positions and their rates:

$$\dot{x_9} = x_{10} \tag{4.55}$$

$$\dot{x_{10}} = -\frac{U_{ft}}{m} U_x \tag{4.56}$$

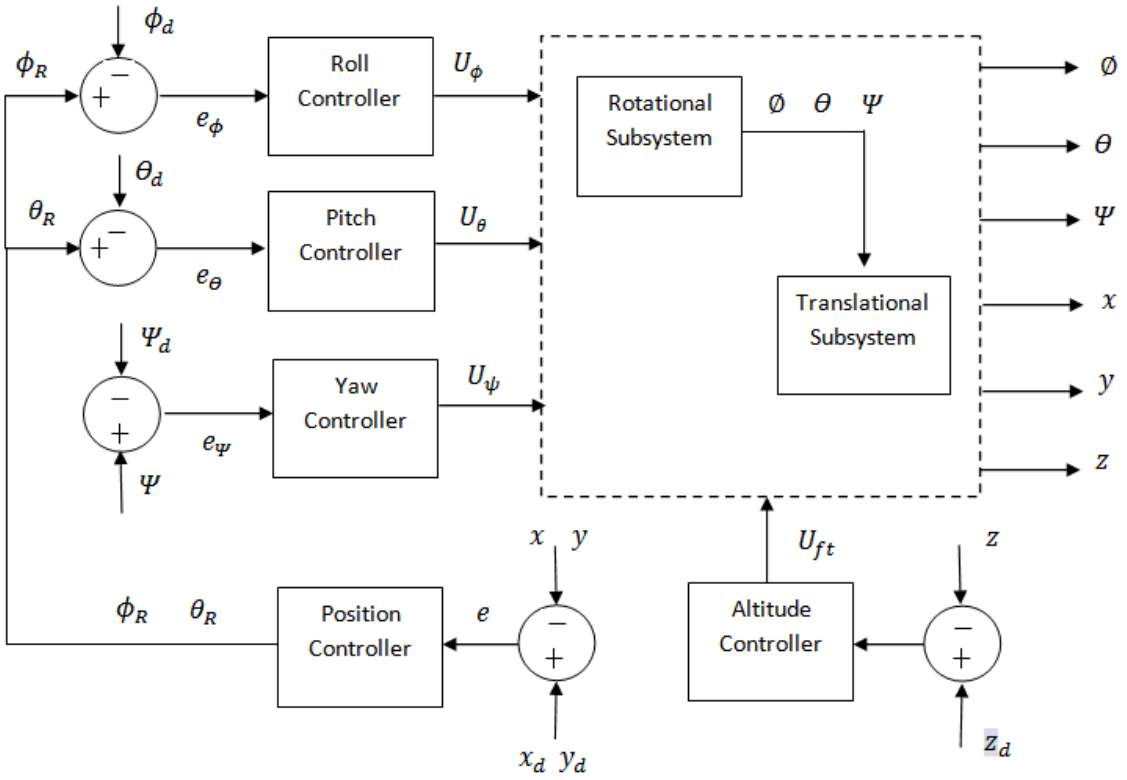
$$\dot{x_{11}} = x_{12} \tag{4.57}$$

$$\dot{x_{12}} = -\frac{U_{ft}}{m} U_y \tag{4.58}$$

Whereas:

$$U_x = s(x_1)s(x_5) + s(x_3)c(x_1)c(x_5) \tag{4.59}$$

$$U_y = c(x_1)s(x_5)s(x_3) - c(x_5)s(x_1) \tag{4.60}$$



.Figure 4.11: Complete System Block Diagram

Using the same steps as we derived U_ϕ, U_θ, U_ψ and U_{ft}, x position controller U_x and y position controller U_y can be derived :

$$U_x = -\frac{m}{U_{ft}}[e_9 - k_9(e_{10} + k_9 e_9) - k_{10} e_{10}] \tag{4.61}$$

Whereas:

$$e_9 = x_9 d - x_9 \tag{4.62}$$

$$e_{10} = x_{10} - \dot{x}_{9d} - k_9 e_9 \tag{4.63}$$

$$U_y = -\frac{m}{U_{ft}}[e_{11} - k_{11}(e_{12} + k_{11}e_{11}) - k_{12}e_{12}] \tag{4.64}$$

Whereas:

$$e_{11} = x_{11}d - x_{11} \tag{4.65}$$

$$e_{12} = x_{12} - \dot{x}_{11d} - k_{11}e_{11} \tag{4.66}$$

Reference roll angle ϕ_R $(x_1)$and reference pitch angle $\theta_R(x_3)$ can be derived as given below:

Let:

$$r_\phi = e_9 - k_9(e_{10} + k_9e_9) - k_{10}e_{10} \tag{4.67}$$

$$r_\theta = e_{11} - k_{11}(e_{12} + k_{11}e_{11}) - k_{12}e_{12} \tag{4.68}$$

Therefore equation (4.61) and (4.64) can be written as:

$$U_x = -\frac{m}{U_{ft}}r_\phi \tag{4.69}$$

$$U_y = -\frac{m}{U_{ft}}r_\theta \tag{4.70}$$

Equation (4.69) and (4.70) can be written as:

$$U_x\frac{U_{ft}}{m} + r_\phi = 0 \tag{4.71}$$

$$U_y\frac{U_{ft}}{m} + r_\theta = 0 \tag{4.72}$$

Putting values of U_x and U_y from equation (4.59) and (4.60)

$$[s(x_1)s(x_5) + s(x_3)c(x_1)c(x_5)]\frac{U_{ft}}{m} + r_\phi = 0 \tag{4.73}$$

$$[c(x_1)s(x_5)s(x_3) - c(x_5)s(x_1)]\frac{U_{ft}}{m} + r_\theta = 0 \tag{4.74}$$

Multiplying r_ϕ with (4.74) and r_θ with (4.73):

$$r_\theta[s(x_1)s(x_5) + s(x_3)c(x_1)c(x_5)]\frac{U_{ft}}{m} + r_\phi r_\theta = 0 \tag{4.75}$$

$$r_\phi\ [c(x_1)s(x_5)s(x_3) - c(x_5)s(x_1)]\frac{U_{ft}}{m} + r_\phi r_\theta = 0 \tag{4.76}$$

Subtracting (4.76) from (4.75) and simplifying:

$$s(x_1)[U_y s(x_5) + U_x c(x_5)] = c(x_1)\ s(x_3)[U_x s(x_5) - U_y c(x_5)] \tag{4.77}$$

Multiplying $c(x_5)$ with (5.59) and $s(x_5)$ with (5.60):

$$U_x c(x_5)\ = s(x_1)s(x_5)c(x_5)\ + s(x_3)c(x_1)(c(x_5))^2 \tag{4.78}$$

$$U_y s(x_5)\ = c(x_1)(s(x_5))^2 s(x_3) - c(x_5)s(x_1)s(x_5) \tag{4.79}$$

Adding (4.78) and (4.79):

$$U_x c(x_5) + U_y s(x_5) = s(x_3)c(x_1)[\big(c(x_5)\big)^2 + (s(x_5))^2] \tag{4.80}$$

$$s(x_3)c(x_1) = U_x c(x_5) + U_y s(x_5) \tag{4.81}$$

$$x_3 = s^{-1}[\frac{U_x c(x_5) + U_y s(x_5)}{c(x_1)}]$$

Put (4.81) into (4.77):

$$x_1 = s^{-1}[U_x s(x_5) - U_y c(x_5)] \tag{4.82}$$

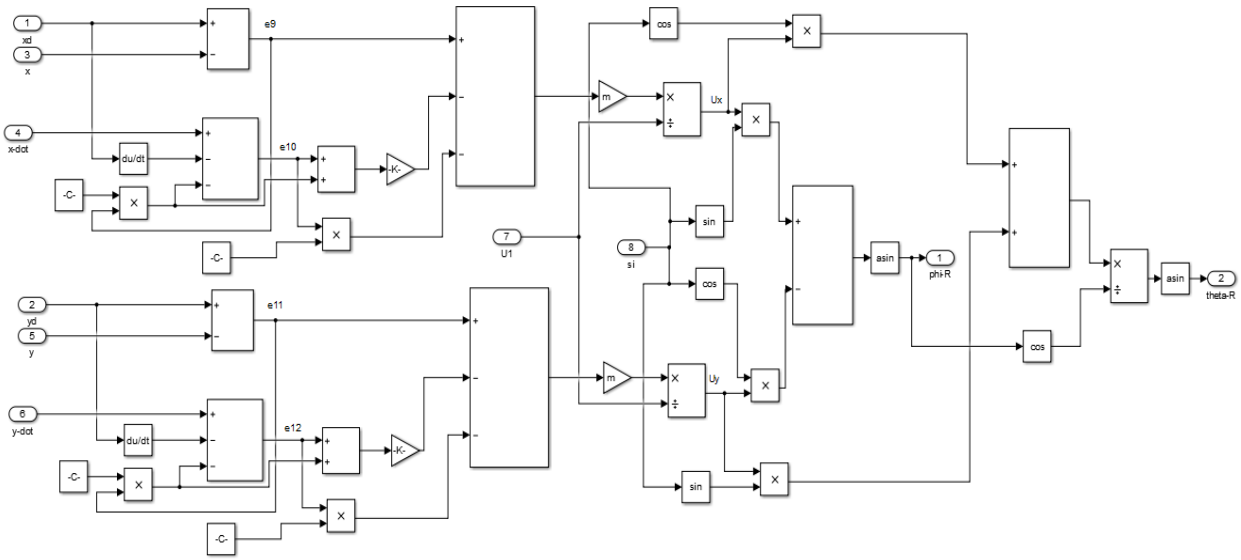


Figure 4.12: Position Controller Block Simulink

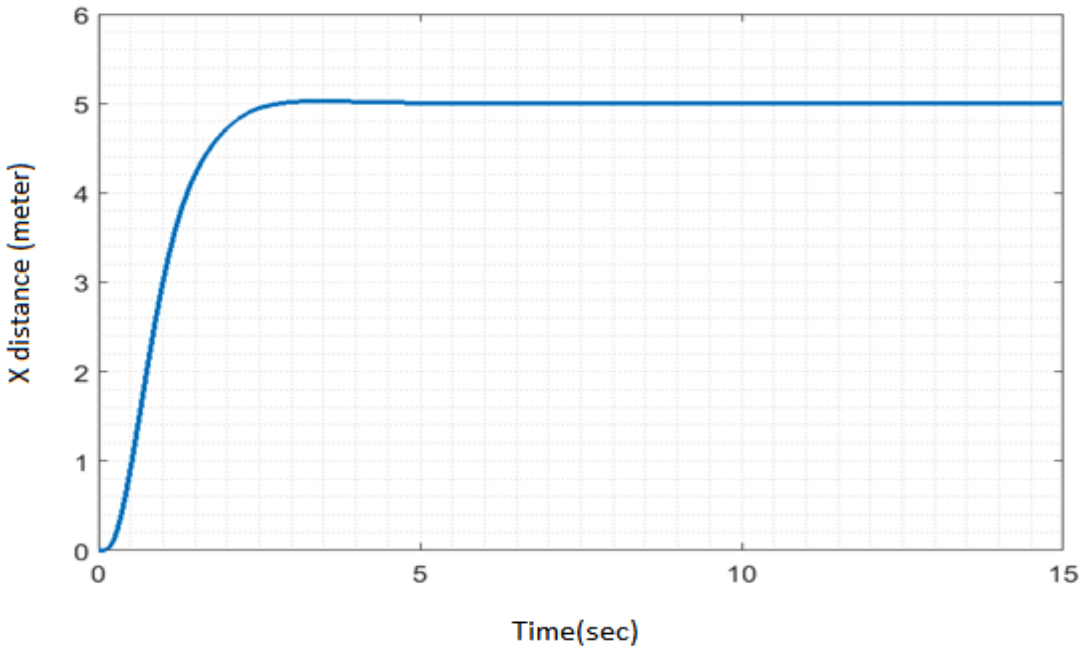


Figure 4.13: Closed Loop Step Response X Position

Characteristics	Values
Rise Time	1.3502 Sec
Settling Time	2.340 Sec

Overshoot	0.5090 %
Peak	5.025 meter
Peak Time	3.337 Sec

Table 4.9: X Position Closed Loop Characteristics

Gains	Values
k_9	2
k_{10}	2

Table 4.10: X Position Backstepping Controller Gains

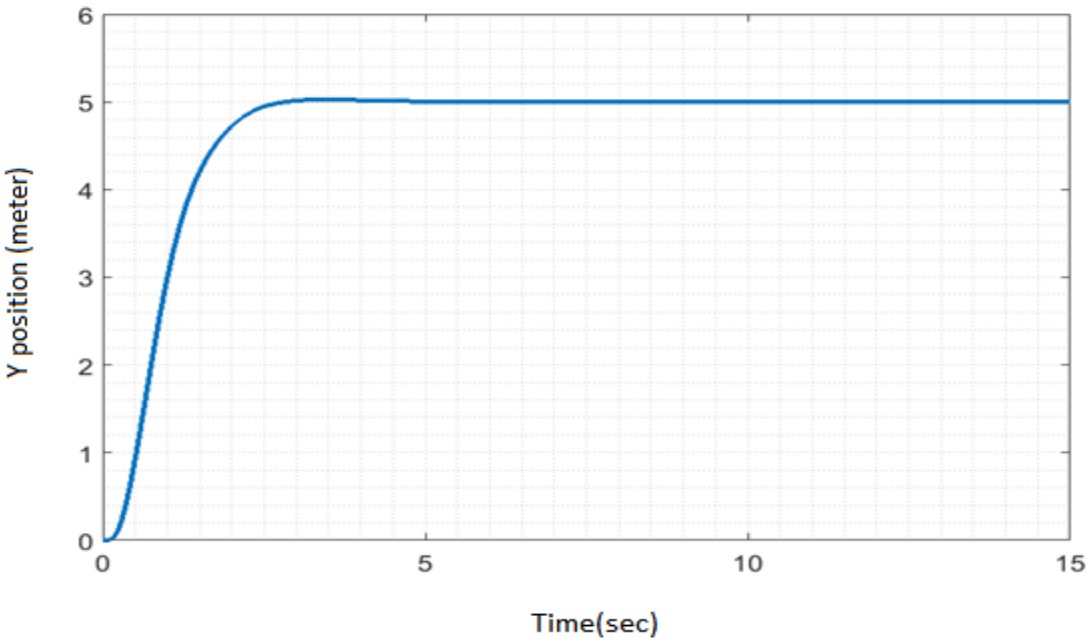


Figure 4.14: Closed Loop Step Response Y Position

Characteristics	Values
Rise Time	1.398 Sec
Settling Time	2.455 Sec

Overshoot	0.347 %
Peak	5.0173 meter
Peak Time	3.570 Sec

Table 4.11: Y Position Closed Loop Characteristics

Gains	Values
k_{11}	2
k_{12}	2

Table 4.12: Y Position Backstepping Controller Gains

4.8 Wind Shear Effect

Applying vertical wind shear of 6m/s at the height 6m:

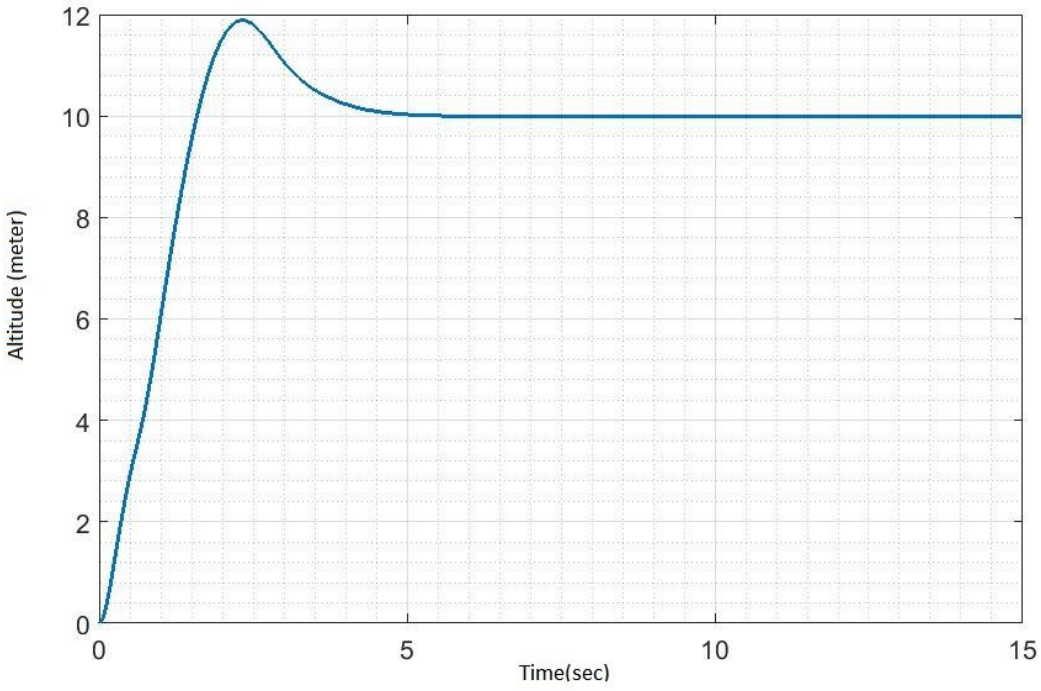


Figure 4.15: Closed Loop Step Response Altitude with Wind Shear

Characteristics	Values
Rise Time	1.1946 Sec
Settling Time	4.0894 Sec
Overshoot	18.9368 %
Peak	11.8937 meter
Peak Time	2.3541Sec

Table 4.13: Altitude Control Closed Loop Characteristics with Wind Shear

Chapter 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

Mathematical model for specific type of UAV, which has Vertical Takeoff and Landing (VTOL) ability, known as quadcopter has been discussed in this work. Mathematical model of quadrotor in state space form is derived, utilizing Newton and Euler equations for three dimensional motions. This mathematical model is nonlinear and accurate enough including the aerodynamic effects and rotor dynamics. Development of two control approaches to control the attitude and position of the quadrotor in space are discussed. In first approach, Four PID controllers are designed to control the roll angle, pitch angle, yaw angle and altitude respectively. Strategy to implement PID controllers on nonlinear quadrotor model is discussed along with simulation results. Simulation results shows that designed PID controllers give better performance in terms of rise time, settling time and overshoot. Controller gains need not to be changed over all attitude angles. In second approach, Backstepping controllers are designed to control the roll angle, pitch angle, yaw angle, altitude and positions. Simulation results shows that Backstepping Controllers give better results as compared to PID controllers in term of rise time, settling time and overshoot. Backstepping controller can also reject the disturbance of wind shear effect up to 6m/s. PID is easy to implement and less computationally intensive ,because it is function of errors and derivative of errors only while backstepping is computationally intensive.

5.2 Future Work

Following areas could be focus for future work

- Sensor modeling and effect of noise on sensors need to be considered.
- Our system show huge degradation when wind shear effect is more than 10m/s so robustification in backstepping controller designed need to be considered.
- Uncertainties in system parameters are required to be considered.
- Implementation of purposed PID controllers and backstepping controllers on real quadcopter hardware.

REFERENCES

[1] L. Minh and C. Ha, "Modeling and control of quadrotor mav using vision-based measurement," in Strategic Technology (IFOST), 2010 International Forum on. IEEE, 2010, pp. 70–75.

[2] M.Huang, B.Xian, C.Diao, K.Yang, and Y.Feng. Adaptive tracking control of underactuated quadrotor unmanned aerial vehicles via backstepping. In American Control Conference (ACC), 2010, pages 2076 –2081, 30 2010-july 2 2010

[3] J.J. Craig, P. Hsu, and S.S. Sastry. Adaptive control of mechanical manipulators. The International Journal of Robotics Research, 6(2):16, 1987.

[4] T. Madani and A. Benallegue. Backstepping Control for a Quadrotor Helicopter. In Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on, pages 3255 –3260, 2006.

[5] W. Zeng, B. Xian, C. Diao, Q. Yin, H. Li, and Y. Yang. Nonlinear adaptive regulation control of a quadrotor unmanned aerial vehicle. In Control Applications (CCA), 2011 IEEE International Conference on, pages 133–138, 2011.

[6] A. Das, K. Subbarao, and F. Lewis. Dynamic inversion with zero-dynamics stabilisation for quadrotor control. Control Theory & Applications, IET, 3(3):303–314, 2009.

[7] S. Bouabdallah and R. Siegwart. Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. In International Conference on Robotics and Automation 2005, pages 2247 – 2252, april 2005.

[8] B. Whitehead and S. Bieniawski. Model Reference Adaptive Control of a Quadrotor UAV. In Guidance Navigation and Control Conference 2010, Toronto, Ontario, Canada, 2010. AIAA.

[9] R.V. Jategaonkar. Flight vehicle system identification: a time domain methodology. American Institute of Aeronautics and Astronautics, 2006.

[10] I.D. Cowling, O.A. Yakimenko, J.F. Whidborne, and A.K. Cooke. A prototype of an autonomous controller for a quadrotor uav. In European Control Conference, pages 1–8, 2007.

[11] S. Bouabdallah, A. Noth, and R. Siegwart. PID vs LQ control techniques applied to an indoor micro quadrotor. In International Conference on Intelligent Robots and Systems 2004, volume 3, pages 2451 – 2456 vol.3, 2004.

[12] T. Bresciani, "Modeling, identification and control of a quadrotor helicopter," Master's thesis, Lund University, 2008.

[13] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin, "Quadrotor helicopter flight dynamics and control: Theory and experiment," in Proc. of the AIAA Guidance, Navigation, and Control Conference, 2007, pp. 1–20.

[14] B. Erginer and E. Altug, "Modeling and pd control of a quadrotor vtol vehicle," in Intelligent Vehicles Symposium, 2007 IEEE. IEEE, 2007, pp. 894–899.

[15] M. D. L. C. de Oliveira, "Modeling, identification and control of a quadrotor aircraft," Master's thesis, Czech Technical University in Prague, 2011.

[16] R. Goela, S. Shahb, N. Guptac, and N. Ananthkrishnanc, "Modeling, simulation and flight testing of an autonomous quadrotor," Proceedings of ICEAE, 2009.

[17] C. Balas, "Modelling and linear control of a quadrotor," Cranfield Unicersity, MSc Thesis, vol. 2007, 2006.

[18] I. Cowling, J. Whidborne, and A. Cooke, "Optimal trajectory planning and lqr control for a quadrotor uav," in Proceedings of the International Conference Control–2006, Glasgow, Scotland, vol. 30, 2006.

[19] Y. Al-Younes, M. Al-Jarrah, and A. Jhemi, "Linear vs. nonlinear control techniques for a quadrotor vehicle," in Mechatronics and its Applications (ISMA), 2010 7th International Symposium on. IEEE, 2010, pp. 1–10.

[20] S. Bouabdallah, "design and control of quadrotors with application to autonomous flying," Ph.D. dissertation, Federal Polytechnic School of Lausanne, 2007.

[21] A. Mokhtari, N. M'Sirdi, K. Meghriche, and A. Belaidi, "Feedback linearization and linear observer for a quadrotor unmanned aerial vehicle," Advanced Robotics, vol. 20, no. 1, pp. 71–91, 2006.

[22] T. Madani and A. Benallegue, "Control of a quadrotor mini-helicopter via full state backstepping technique," in Decision and Control, 2006 45th IEEE Conference on. IEEE, 2006, pp. 1515–1520.

[23] M. Hehn and R. DAndrea, "Quadrocopter trajectory generation and control," in Proceedings of the IFAC world congress, 2011.

[24] T. Buchholz, D. Gretarsson, and E. Hendricks, "Construction of a four rotor helicopter control system," Master's thesis, Technical University of Denmark, 2009.

.[25] E. Altug, J. Ostrowski, and R. Mahony, "Control of a quadrotor helicopter using visual feedback," in Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on, vol. 1. IEEE, 2002, pp. 72–77.

[26] P. Pounds, R. Mahony, and P. Corke. Modelling and control of a quadrotor robot. In Australasian conference on robotics and automation 2006, Auckland, NZ, 2006.

[27] D. Mellinger, Q. Lindsey, M. Shomin, and V. Kumar, "Design, modeling, estimation and control for aerial grasping and manipulation," in Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on. IEEE, 2011, pp. 2668–2673.

[28] http://krossblade.com/history-of-quadcopters-and-multirotors/

[29] Miroslav Krstic, Petar V Kokotovic, and Ioannis Kanellakopoulos. Nonlinear and adaptive control design. John Wiley & Sons, Inc., 1995.

[30] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo. Robotics. McGraw-Hill.

[31] D. Lee, T. Burg, D. Dawson, D. Shu, B. Xian, and E. Tatlicioglu, "Robust tracking control of an underactuated quadrotor aerial-robot based on a parametric uncertain model", in Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on, 2009, pp. 3187–3192.

[32] T. Bresciani, “Modelling, Identification and Control of a Quadrotor Helicopter”, Master’s thesis, Lund University, Sweden, 2008.

[33] Hongning Hou, Jian Zhuang, Hu Xia, Guanwei Wang, and Dehong Yu. A simple controller of minisize quad-rotor vehicle. In Mechatronics and Automation (ICMA), 2010 International Conference on, pages 1701{1706, 2010. doi:10.1109/ICMA.2010.5588802.

[34] Quadrotor Stability using PID, JULKIFLI BIN AWANG BESAR, MS thesis, Universiti Tun Hussein Onn Malaysia.