Minimum Jerk Trajectory Tracking and Control of Quadcopter Using Adaptive Sliding Mode Controller

Lidiya Girma Mekonen

A Thesis Submitted to

The Department of Electrical Power and Control Engineering

School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of Master's in Electrical Power and Control Engineering (Control)

Office of Graduate Studies

Adama Science and Technology University

September 2021

Adama, Ethiopia

Minimum Jerk Trajectory Tracking and Control of Quadrotor Using Adaptive Sliding Mode Controller

Lidiya Girma Mekonen

Advisor: Dr. Shubhashish Bhakta

A Thesis Submitted to

The Department of Electrical Power and Control Engineering School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of Master's in Electrical Power and Control Engineering (Control)

Office of Graduate Studies
Adama Science and Technology University

September 2021

Adama, Ethiopia

DECLARATION

I hereby declare that this Master Thesis entitled “Minimum Jerk Trajectory Tracking and Control of Quadcopter Using Adaptive Sliding Mode Controller” is my original work. That is, it has not been submitted for the award of any academic degree, diploma or certificate in any other university. All sources of materials that are used for this thesis have been duly acknowledged through citation.

Name of the Student	Signature	Date

RECOMMENDATION

We, the advisors of this thesis, hereby certify that we have read the revised version of the thesis entitled “Minimum Jerk Trajectory Tracking and Control of Quadcopter Using Adaptive Sliding Mode Controller” prepared under our guidance by Lidiya Girma Mekonen submitted in partial fulfilment of the requirements for the degree of Mater’s of Science in Control Engineering. Therefore, we recommend the submission of the revised version of the thesis to the department following the applicable procedures.

______________________	______________________	____________
Major Advisor	Signature	Date

______________________	______________________	____________
Co-advisor	Signature	Date

APPROVAL SHEET

We, the advisors of the thesis entitled "Minimum Jerk Trajectory Tracking and Control of Quadcopter Using Adaptive Sliding Mode Controller" and developed by Lidiya Girma Mekonen, hereby certify that the recommendation and suggestions made by the board of examiners are appropriately incorporated into the final version of the thesis.

Major Advisor	Signature	Date
Co-advisor	Signature	Date

We, the undersigned, members of the Board of Examiners of the thesis by Lidiya Girma Mekonen have read and evaluated the thesis entitled "Minimum Jerk Trajectory Tracking and Control of Quadcopter Using Adaptive Sliding Mode Controller" and examined the candidate during the open defence. This is, therefore, to certify that the thesis is accepted for partial fulfilment of the requirement of the degree of Master of Science in Control Engineering.

Chairperson	Signature	Date
Internal Examiner	Signature	Date
Eternal Examiner	Signature	Date

Finally, approval and acceptance of the thesis is contingent upon submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and School Graduate Committee (SGC).

Department Head	Signature	Date
School Dean	Signature	Date
Office of Postgraduate Studies, Dean	Signature	Date

ACKNOWLEDGEMENTS

First and foremost, I would like to thank Almighty God for giving me the strength, opportunity and making it possible to undertake and complete this thesis. I would also like to thank Adama Science and Technology University, the School of Electrical Engineering and Computing along the Power and Control engineering chair for the opportunity they have given me to attend this Master's program.

It gives me great pleasure to express my deepest gratitude to my advisor and mentor **Dr Shubhashish Bhakta**, for his heartfelt support and guidance at all times. He has given me invaluable guidance, inspiration and suggestions throughout doing this thesis, and I am so thankful.

Besides my advisor, my thanks are extended to **Mr. Minyamir Gelaw** and **Mr. Daniel Abebe** for their continuous support, motivation and immense knowledge. My acknowledgement would be incomplete without thanking the biggest source of my strength, my families and friends for all the love, support and encouragement. It would have been impossible without having you around.

Thank you.

Lidiya Girma Mekonen

TABLE OF CONTENTS

DECLARATION....................i
RECOMMENDATION....................ii
APPROVAL SHEET.................... iii
ACKNOWLEDGEMENTS iv
TABLE OF CONTENTS v
LIST OF FIGURES....................ix
LIST OF TABLES xi
LIST OF ABBREVIATIONS xii
LIST OF SYMBOLS.................... xiii
ABSTRACT xv
CHAPTER ONE....................1
1. INTRODUCTION....................1
1.1. Background....................1
1.2. Statement of the Problem 3
1.3. Objectives of the Study 4
1.3.1. General Objective....................4
1.3.2. Specific Objectives....................4
1.4. Scope of the Study....................4
1.5. Limitation of the Study....................4
1.6. Motivation of the Study....................4
1.7. Significance of the Study....................5
1.8. Thesis Organization....................5
CHAPTER TWO....................6
2. LITERATURE REVIEW....................6
2.1. Chapter Overview....................6
2.2. Unmanned Aerial Vehicles (UAVs)....................6
2.3. History of UAVs....................6
2.4. Applications of UAVs 7
2.5. Classification of UAVs....................7
2.6. Quadcopters....................8
2.6.1. The Quadcopter Concept....................8

2.6.2. Advantages and Disadvantages of Quadcopters 10
2.7. Related Works 11
CHAPTER THREE 14
3. MATERIALS AND METHODS 14
3.1. Chapter Overview 14
3.2. Materials 14
3.3. Methods 14
3.4. Block Diagram of the Proposed ASMC for Quadcopter System 15
3.5. Mathematical Modeling of Quadcopter 16
3.5.1. Six-Degrees of Freedom (6-DOF) 16
3.5.2. Quadcopter System Descriptions 16
3.5.2.1. 3.4.2.1 Control Inputs 17
3.6. Reference Frames and Transformation Matrices 18
3.6.1. Transformation Matrices for Translational Velocities 20
3.6.2. Transformation Matrices for Angular Velocities 20
3.7. Nonlinear Dynamic Model 21
3.7.1. Forces and Moments Acting on the Quadcopter 22
3.7.2. Rotational Equations of Motion 24
3.7.3. Translational Equations of Motion 25
3.7.4. State Space Equation for Quadcopter Dynamics 27
3.8. Equilibrium Points and Stability Analysis 29
3.8.1. Review on Equilibrium Points 29
3.8.2. Equilibrium Points for Φ 29
3.8.3. Equilibrium Points for Θ 30
3.8.4. Equilibrium Points for Ψ 30
3.8.5. Equilibrium Points for Z 30
3.8.6. Equilibrium Points for X 31
3.8.7. Equilibrium Points for Y 31
3.9. Checking Local Stability of Equilibrium Points 32
3.9.1. Review on Stability of Nonlinear Systems 32
3.9.2. Local Stability Analysis using Lyapunov Direct Method 32
3.9.2.1. Local Stability Analysis for Φ 33
3.9.2.2. Local Stability Analysis for Θ 33
3.9.2.3. Local Stability Analysis for Ψ 33
3.9.2.4. Local Stability Analysis for Z 33
3.9.2.5. Local Stability Analysis for X 33
3.9.2.6. Local Stability Analysis for Y 34

3.10. Open Loop Analysis 34
3.10.1. Open-loop model simulation 34
CHAPTER FOUR 39
4. CONTROLLER DESIGN 39
4.1. Trajectory Generation Using Minimum Jerk Trajectory 39
4.1.1. Minimum Jerk Trajectory for X Position 40
4.2. Sliding Mode Controller 42
4.2.1. Sliding Mode Control Law 43
4.3. Adaptive Controller 44
4.4. Adaptive Sliding Mode Controller 45
4.5. Application of Sliding Mode Controller for Quadcopter 47
4.5.1. Attitude Control Law 48
4.5.1.1. Roll Control 48
4.5.1.2. Pitch Control 49
4.5.2. Heading Control 49
4.5.2.1. Yaw Control 49
4.5.3. Altitude Control Law 50
4.5.3.1. Z Control 50
4.5.4. Position Control Law 50
4.5.4.1. X Control Law 50
4.5.4.2. Y Control Law 51
4.6. Application of Adaptive Sliding Mode Controller for Quadcopter 51
4.6.1. Attitude Control Law 52
4.6.1.1. Roll Control 52
4.6.1.2. Pitch Control 54
4.6.2. Heading Control Law 55
4.6.2.1. Yaw Control 55
4.6.3. Altitude Control Law 57
4.6.3.1. Z Control 57
4.7. Rotor Speed Calculation from Control Signal 59
CHAPTER FIVE 60
5. RESULT AND ANALYSIS 60
5.1. Proposed Control System Result 60
5.1.1. Trajectory Tracking Performance of the Control System at *m*=0.5kg 61
5.2. Control Signals for Tracking Using ASMC 69
5.3. Control signals for Tracking Using SMC 69

5.4. Simulation Result of the ASMC with Parameter Uncertainty and External Random Disturbance 71
5.4.1. Trajectory Tracking Performance of ASMC with Parameter Uncertainty 72
5.4.2. Trajectory Tracking Performance of the Control System with External Disturbances 76
5.5. Sliding Mode Controller results 78
5.5.1. Trajectory tracking performance of SMC with parametric uncertainty 78
6. CONCLUSION AND RECOMMENDATION 82
6.1. Conclusion 82
6.2. Recommendation 82
Reference 84
Reference Links 88
Appendices 89
A MATLAB script for ASMC design 89
B MATLAB/SIMULINK Block Diagrams 89
C 3D Animation Using VR Sink 90

LIST OF FIGURES

Figure 1.1 Quadcopter system 1
Figure 2.1 Lawrence and Sperry UAV 6
Figure 2.2 Quadcopter 8
Figure 2.3 Quadcopter configuration: a. Plus and b. cross 8
Figure 2.4 Euler angles for a quadcopter UAV 9
Figure 2.5 Generated motion of the quadcopter 10
Figure 3.1 Block diagram of the research methodology 15
Figure 3.2 Block diagram of the proposed quadcopter control system. 16
Figure 3.3 Force and moments of quadcopter 18
Figure 3.4 Reference frames for the quadcopter system 19
Figure 3.5 Response of nonlinear translational position to throttle input at
$\hat{\Omega}_1 = \hat{\Omega}_2 = \hat{\Omega}_3 = \hat{\Omega}_4 = 203\ (rad/\sec)$ 34
Figure 3.6 Response of nonlinear orientation to throttle input at
$\hat{\Omega}_1 = \hat{\Omega}_2 = \hat{\Omega}_3 = \hat{\Omega}_4 = 203\ (rad/\sec)$ 35
Figure 3.7 Response of nonlinear translational position to roll input at
$\hat{\Omega}_1 = \hat{\Omega}_3 = 203,\ \hat{\Omega}_2 = 202.9,\ \hat{\Omega}_4 = 203.1\ (rad/\sec)$ 35
Figure 3.8 Response of nonlinear orientation to roll input at
$\hat{\Omega}_1 = \hat{\Omega}_3 = 203,\ \hat{\Omega}_2 = 202.9,\ \hat{\Omega}_4 = 203.1\ (rad/\sec)$ 36
Figure 3.9 Response of nonlinear translational position to pitch input at
$\hat{\Omega}_1 = 202.9,\ \hat{\Omega}_2 = \hat{\Omega}_4 = 203,\ \hat{\Omega}_3 = 203.1\ (rad/\sec)$ 36
Figure 3.10 Response of nonlinear angular orientation to pitch input at
$\hat{\Omega}_1 = 202.9,\ \hat{\Omega}_2 = \hat{\Omega}_4 = 203,\ \hat{\Omega}_3 = 203.1\ (rad/\sec)$ 37
.Figure 3.11 translational position to yaw input at $\hat{\Omega}_1 = \hat{\Omega}_3 = 203,\ \hat{\Omega}_2 = \hat{\Omega}_4 = 204(rad/\sec)$
.... 37
Figure 3.12 Angular orientation to yaw input at $\hat{\Omega}_1 = \hat{\Omega}_3 = 203,\ \hat{\Omega}_2 = \hat{\Omega}_4 = 204(rad/\sec)$
.... 38
Figure 4.1 Block Diagram of sliding mode controller for quadcopter 44
Figure 4.2 Adaptive control 45
Figure 4.3 Block diagram of Adaptive sliding mode controller for quadcopter 52
Figure 5.1 Altitude *Z* reference trajectory tracking controller performance using ASMC . 61
Figure 5.2 Altitude reference tracking error 62
Figure 5.3 Position *X* reference trajectory tracking controller performance using ASMC . 62
Figure 5.4 Position X reference tracking error 63
Figure 5.5 Position *Y* reference trajectory tracking controller performance using ASMC . 63
Figure 5.6 Position *Y* reference tracking error 64

Figure 5.7 Roll angle Φ reference trajectory tracking controller performance using ASMC 64
Figure 5.8 Roll angle *Φ* reference tracking error 65
Figure 5.9 pitch angle *Θ* reference trajectory tracking controller performance using ASMC 66
Figure 5.10 Pitch reference tracking error..66
Figure 5.11 Yaw angle *Ψ* reference trajectory tracking controller performance using ASMC...67
Figure 5.12 3D reference trajectory tracking controller performance using ASMC..........68
Figure 5.13 Control signals of trajectory tracking controller using ASMC 69
Figure 5.14 Control signals of trajectory tracking controller using ASMC 70
Figure 5.15 Control signals of trajectory tracking controller using ASMC 70
Figure 5.16 Control signals of trajectory tracking controller using ASMC 71
Figure 5.17 Altitude *Z* reference trajectory tracking controller performance for different values of *m* using ASMC..73
Figure 5.18 Altitude Z reference trajectory tracking error for different values of *m* using ASMC...73
Figure 5.19 Position *X* reference trajectory tracking controller performance for different values of m using ASMC 74
Figure 5.20 Position *X* reference trajectory tracking error for different values of m using ASMC...74
Figure 5.21 Position *Y* reference trajectory tracking controller performance for different values of m using ASMC 75
Figure 5.22 Position *X* reference trajectory tracking error for different values of m using ASMC...75
Figure 5.23 Altitude *Z* trajectory tracking when there is wind gust disturbance 76
Figure 5.24 Position *X* trajectory tracking when there is wind gust disturbance 77
Figure 5.25 Position *Y* trajectory tracking when there is wind gust disturbance 77
Figure 5.26 *3D* trajectory tracking when there is wind gust disturbance 78
Figure 5.27 Altitude *Z* reference trajectory tracking controller performance for different values of *m* using SMC...79
Figure 5.28 Altitude Z reference trajectory tracking error using SMC...............................79
Figure 5.29 Position *X* reference trajectory tracking controller performance for different values of m using SMC 80
Figure 5.30 Position *X* reference trajectory tracking error using SMC...............................80
Figure 5.31 Position *Y* reference trajectory tracking controller performance for different values of m using SMC 81
Figure 5.32 Position *X* reference trajectory tracking error using ASM...............................81

LIST OF TABLES

Table 4.1 Initial and final position, velocity and acceleration..41
Table 5.1 Physical parameter for quadcopter 60
Table 5.2 Range of external wind disturbance given to the system 72

LIST OF ABBREVIATIONS

3D	Three Dimensional
ASMC	Adaptive Sliding Mode Controller
ASTU	Adama Science and Technology University
CSMC	Cascade Sliding Mode Control
DC	Direct Motor
DOF	Degrees of Freedom
FOPID	Fractional Order Proportional Integral Derivative
IAE	Integral of Absolute Magnitude of the Error
LPD	Locally Positive Definite
LQR	Linear Quadratic Regulator
MATLAB	Matrix Laboratory
MIMO	Multiple-Input and Multiple-Output
PI	Proportional-Integral
PID	Proportional-Integral-Derivative
SMC	Sliding Mode Control
STI	Space Technology Institute
UAV	Unmanned Aerial Vehicle
USA	United States of America
VTOL	Vertical Take-Off and Landing

LIST OF SYMBOLS

C_D	Aerodynamic drag coefficient
d	Aerodynamic drag contribution
C_τ	Aerodynamic thrust coefficient
b	Aerodynamic thrust contribution
R_r	Angular transformation matrix from f_E to f_B
$\hat{\Omega}_i$	Angular velocity of i_{th} rotor
$\bar{\omega}$	Angular velocity vector of quadrotor
p, q, r	Body angular velocities
f_B	Body reference frame
u,v,w	Body velocities
$\tilde{U}_1,\tilde{U}_2,\tilde{U}_3,\tilde{U}_4$	Control inputs
K_D	Derivative gain
f_E	Earth reference frame
Φ, Θ, Ψ	Euler angles
$\dot{\Phi}, \dot{\Theta}, \dot{\Psi}$	Euler angles rates
Θ	Euler pitch angle
Φ	Euler roll angle
Ψ	Euler yaw angle
F_i	Force generated by i_{th} rotor
g	Gravitational acceleration
$F_{g,B}$	Gravitational force expressed in f_B
$F_{g,E}$	Gravitational force expressed in f_E
I	Inertia matrix
K_I	Integral gain
x_i	i_{th} system state
l	Length of the moment arm
m	Mass of quadcopter
r	Motor gear box reduction ratio

$\bar{\eta}$	Orientation Vector of Quadrotor in f_E
$\bar{\xi}$	Position Vector of Quadrotor in f_E
I_x, I_y, I_z	Products of inertia
K_P	Proportional gain
$\hat{\Omega}$	Relative speed of the propellers
I_r	Rotor inertia
F_{ext}	Sum of external forces acting on the quadcopter
M_{ext}	Sum of external moments acting on the quadcopter
T_i	Torque generated by i_{th} rotor
$F_{prop,B}$	Total force generated by the propeller expressed in f_B
e_i	Tracking error for i_{th} state
R	Transformation matrix
R_{EB}	Transformation matrix from f_B to f_E
R_{BE}	Transformation matrix from f_E to f_B
$\tilde{V}_B$	Translational velocity vector of quadrotor expressed in f_B

ABSTRACT

In recent years, Unmanned Aerial Vehicles (UAVs) have become increasingly popular research topics due to their wide applications and advantages. In addition, it is a good platform for control systems research, as it is a highly non-linear and underactuated system. The quadcopter is a four-rotor UAV which is a multivariable, nonlinear, unstable and under-actuated system. This research work addresses a minimum jerk trajectory generation and an Adaptive Sliding Mode Controller (ASMC), which is used for the attitude, position, altitude, and heading control of the quadcopter UAV with parameter uncertainty and external disturbance. Continuous adaptive dynamic vectors are used to minimize chattering problems in Sliding Mode Control (SMC) and Lyapunov stability analysis is used to achieve asymptotic stability. To verify the robustness of the proposed control method, the controller was implemented in the quadcopter using MATLAB / Simulink for simulation. The MATLAB/Simulink simulation shows that the proposed controller adapts to the unknown mass of quadcopter between 0.1kg - 6kg and it performs effectively at the reference of altitude, position, attitude, and heading.

Keywords: *Quadcopter, UAVs, Minimum jerk, Trajectory tracking, ASMC, SMC, Quadcopter model, Lyapunov stability, Parameter uncertainties, MATLAB/Simulink*

CHAPTER ONE

1. INTRODUCTION

1.1. Background

An unmanned aerial vehicle (UAV) is an aircraft that can fly autonomously according to pre-programmed mission profiles and well-defined flight plans or be piloted remotely without human pilots on board. Within the starting, they were solely used for military purposes. One of the primary applications of these vehicles was aerial photography. Within 1883, Douglas Archibald, an English man, provided one of the world's first reconnaissance UAVs. But until World War, I UAVs were not recognized. Since then, they are being broadly utilized in military missions such as airfield base security, surveillance of enemy activity, end of unexploded bombs, airfield damage assessment, etc [1].

The quadcopter is a vertical takeoff and landing (VTOL) aerial vehicle which has four rotors. By controlling the angular velocity of each rotor that contains a motor and a propeller group, the quadcopter changes its attitude and altitude [1]. The shape of rotor arms in quadcopters is symmetric and combined with a central board, where all the equipment and useful stuff are located. This symmetry within the shape is useful to operate the quadcopters in vertical movement and hovering capabilities, and the control of the vehicle in any direction.

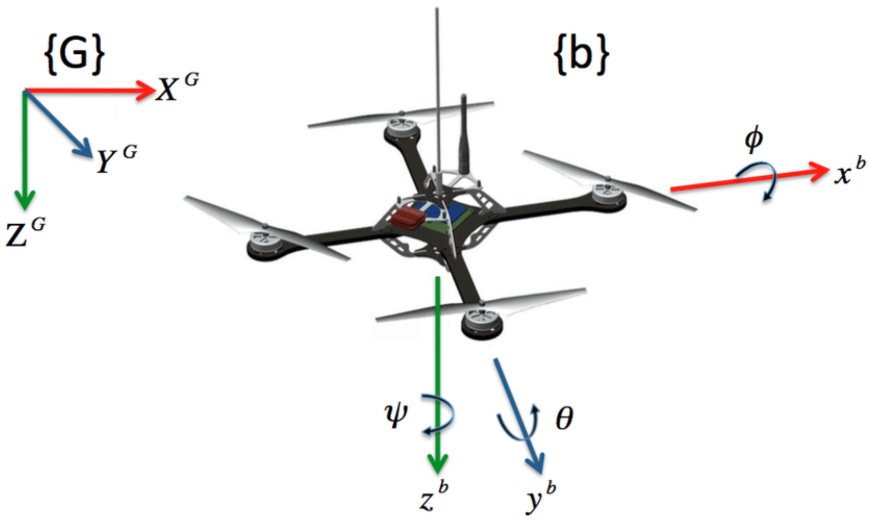


Figure 1.1 Quadcopter system [L1]

In the last few years, with the advancement of equipment stages such as microcontrollers, engines, and sensors, quadcopter frameworks have picked up great commercial potential and became worldwide accessible. Although the first quadrotors were manned, nowadays most of these robots are unmanned and utilized with straightforward controller units. This fast advancement lies within the points of interest of these flying robots in comparison to ordinary helicopters and aircraft. These are [3];

- To be able to move with high maneuverability in every direction,
- Small sizes, simple mechanical components, and reduced aerodynamic influences,
- Vehicle control simplicity and,
- Inexpensive.

Due to these factors, the motivation in quadcopter systems has developed rapidly, and nowadays quadcopters have entered our lives within various usage areas. These robots were initially used for simple photography and video shooting. Nowadays, they are used for aerial delivery, communication, mapping, surveillance, search and rescue, border control, mine detection, etc. There are lots of commercial products available on the market such as the products of Chinese and French companies DJI and Parrot [3].

Either the Newton-Euler (which is vectorial) or the Euler-Lagrange (which is scalar in nature) formulations can be used to drive the modelling of quadcopters dynamics. The control inputs describing its motion are essentially due to the coupling of generated thrust and torque as the rotational speed of its respective propeller is varied. System identification, as well as the torque-voltage relationship, are some of the advanced methods employed to obtain the motor dynamics. Currently, the ASTU Space Technology Institute (STI) centre of excellence has drone researches and the members are developing quadcopters structural drones. The main challenge and complicated system in any UAV are designing of a suitable control system. Altitude, attitude, and position controls of quadcopters have been studied extensively using classical and modern control schemes [4]. The system is highly nonlinear, underactuated, coupled dynamics, and affected by disturbances. As a result, controlling the motion and orientation is difficult using conventional control systems like Proportional Integral Derivative (PID) controllers. Advanced controllers like sliding mode controllers (SMC) are effective in case of disturbance rejection capability and mitigating the nonlinearity effects. However, it has limitations due to the chattering effect.

In this research work, an adaptive sliding mode controller is implemented to control the altitude, attitude, position, and heading of a quadcopter system. A self-tuning adaptive controller is used to adapt the system parameters. A self-tuning adaptive controller is obtained by coupling the controller with an online parameter estimator. At each time instant, the estimator sends to the controller a set of estimated plant parameters, which is computed based on the past plant input and output. The corresponding controller parameters are found, and then a control input is computed based on the controller parameters. This control input causes a new plant output to be generated, and the whole cycle of parameter and input updates is repeated. The proposed system is implemented in the MATLAB software tool.

1.2. Statement of the Problem

The modelling and control of any UAVs are complex. Quadcopters are one of the types of multirotor UAVs. It is a highly nonlinear, multivariable, and underactuated system. Underactuated systems are those having fewer control inputs compared to the system's degrees of freedom. They are very difficult to control due to the nonlinear coupling between the actuators, degrees of freedom. Although the most common flight control algorithms are linear like PID flight controllers, these controllers can only perform when the quadcopter is flying around specific flights. They suffer from a huge performance degradation whenever the quadrotor leaves the nominal conditions or performs aggressive manoeuvres under different disturbance conditions [3].

In this research work, a sliding mode controller is proposed. SMC has good performance in the case of trajectory tracking and disturbance rejection capabilities. However, SMC has a drawback due to its internal chattering problems. This may lead to having different voltage and current distributions to the motors. So, the torques in each rotor will be different and generate vibration on the quadcopter system. To mitigate this problem an adaptive sliding mode controller based on exponential reaching law for controlling the altitude, position, attitude, and heading of quadcopter UAV is proposed. The system performance is checked using MATLAB software tool analysis.

1.3. Objectives of the Study

1.3.1. General Objective

The general objective of this research work is to design an adaptive sliding mode controller using a PID sliding surface for the altitude, position, attitude, and heading control of a quadcopter system.

1.3.2. Specific Objectives

The specific objectives are:

- Designing a suitable minimum jerk reference trajectory model.
- Designing an auto-tuning-based adaptive sliding mode controller for a quadcopter system.
- Analyzing the performance and robustness of adaptive sliding mode controller using simulation.

1.4. Scope of the Study

The scope of this research work is deriving a mathematical model of quadcopter dynamics and designing an ASMC for controlling the altitude, position, attitude, and heading of a quadcopter using MATLAB/Simulink.

1.5. Limitation of the Study

The actual hardware implementation has many electronic devices to be integrated and can't be easily found in Ethiopia. Due to the complexity, time, and cost incurred in making the actual prototype implementations, this research work is limited to only the simulation of the quadcopter system.

1.6. Motivation of the Study

Advancement and introduction of impressive technology in quadcopter drone has developed new fields of applications for it. This kind of vehicle can maneuvers in small spaces and is used to perform a large set of tasks. Today drones are being used in several areas for various purposes. Thus, the design of control algorithms takes an important role in designing these advanced technology systems.

1.7. Significance of the Study

Controlling the UAV's position, altitude and orientation is a very interesting research field in control system areas. Currently, these types of researches are implemented in many countries due to multipurpose application areas and the advantages of the multirotor UAVs. Especially, for Adama Science and Technology University (ASTU), Space Technology Institute (STI) Center of Excellence is very important because the research and development teams are trying to design and develop their quadcopter drones for the Ethiopian market.

1.8. Thesis Organization

The thesis is organized as follows:

Chapter 1: introduces the historical background of UAVs. It also includes a statement of the problem, the detailed objective of the thesis, scope, limitation, motivation, and significance of the thesis.

Chapter 2: deals with the literature works previously done in the area of quadcopter control.

Chapter 3: derive the nonlinear dynamic model of quadcopter based on Euler Newton formalism. Then based on the mathematical model, it computes equilibrium points, determines the local stability of equilibrium points, and simulates the open-loop model of the quadcopter system.

Chapter 4: design the control strategies, SMC and ASMC for the nonlinear dynamic model of the quadcopter system.

Chapter 5: provided simulation results and detailed analysis of the results.

Chapter 6: In the last chapter, a conclusion and recommendation for the future researcher are provided.

CHAPTER TWO

2. LITERATURE REVIEW

2.1. Chapter Overview

In this chapter, a brief description, history, types, and uses of the UAV quadcopter is presented. Then the concept and architecture of the quadcopter-type UAVs are discussed. Finally, closely related works along with their merits and demerits are reviewed and discussed.

2.2. Unmanned Aerial Vehicles (UAVs)

UAVs are small aircraft that are designed or modified for operation without a human pilot and operated by electronic input activated by a flight controller. They can be remotely operated by humans or they can be autonomous. Self-driving vehicles are controlled by onboard computers, which can be pre-programmed to perform specific tasks or a wide range of tasks. Aircraft UAVs are mainly used for military applications, but recently they have begun to be used in civilian applications [5].

2.3. History of UAVs

The UAVs were originally manufactured in 1916 by Lawrence and Sperry (USA). They called it an aerial torpedo and, shown in Figure 2.1. They were able to fly it for 30 miles. Lawrence and Sperry reportedly use a gyroscope to balance the body [3].

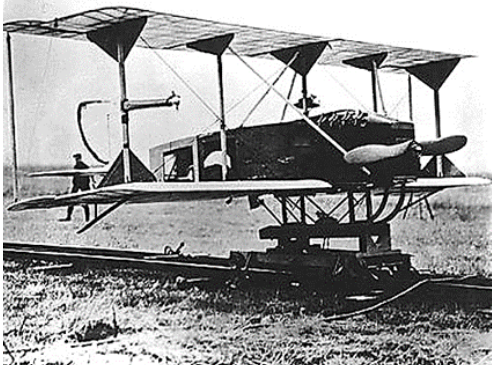


Figure 2.1 Lawrence and Sperry UAV [3]

2.4. Applications of UAVs

In addition to military applications, UAVs can also be used for many civilian or commercial applications that are too boring, dirty, or dangerous for manned aircraft. These uses include but are not limited to [3]:

- Earth Science: UAVs observations can be used in conjunction with satellite observations.
- Wildfire suppression: UAVs equipped with infrared sensors are sent flying over fire-prone forests to detect it in time and send the exact location of the fire to the ground station before it spreads.
- Law enforcement: UAVs are used as a cost-effective alternative to traditional manned police helicopters.
- Border surveillance; UAVs are used to patrol the border to prevent illegal immigration, drug, and gun smuggling.
- Research: UAVs are also used in research conducted by universities to test certain theories. In addition, environmental research institutions use drones equipped with appropriate sensors to monitor certain environmental phenomena, such as pollution in large cities.
- Industrial applications: UAVs are used in various industrial applications, such as pipeline inspection or surveillance and surveillance of nuclear plants.
- Agriculture development: UAVs also have agricultural uses, such as crop spraying.

2.5. Classification of UAVs

There are different ways to classify UAVs, either according to their range of action, aerodynamic configuration, size, and payload and according to their levels of autonomy based on their [6];

- Range of actions with maximum altitude and endurance,
- Size and payload configuration classifications.
- Aerodynamic configurations,
  - Fixed-wing UAVs.
  - Blimps UAVs.
  - Flapping-wing UAVs.

- Rotary-wing UAVs: single-rotor, coaxial, quadcopter, and multi-rotor UAVs.

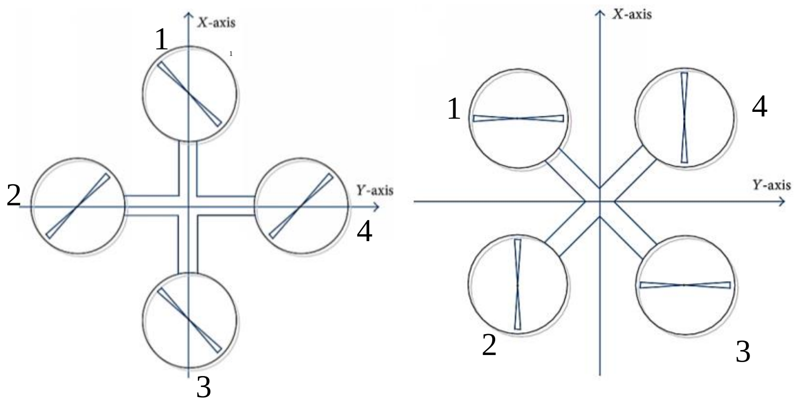
2.6. Quadcopters

Quadcopters are rotary-wing UAVs whose concept of aircraft was developed a long time ago. According to reports, the Breguet Richet quadrotor manufactured in 1907 has already flown. Since its configuration will be discussed later, the quadrotor is considered a rotary-wing UAV [3] [6].

Figure 2.2 Quadcopter [L2]

2.6.1. The Quadcopter Concept

The quadcopter consists of four rotors, each of which is installed at one end of the cross-shaped frame, as shown in Figures 2.3 a and b. Each rotor consists of a propeller mounted on a separately powered DC motor. Propellers 1 and 3 rotate in the same direction, while propellers 2 and 4 rotate in opposite directions, thus balancing the total torque of the system and counteracting the gyro and aerodynamic torque in hovering flight [7].



a. Plus b. cross

Figure 2.3 Quadcopter configuration: a. Plus and b. cross [8].

The quadcopter is a 6 DOF object, so it uses 6 variables to represent its position in space $(X,\ Y,\ Z,\ \Phi,\ \Theta,\ and\ \Psi)$. X, Y, and Z represent the distance of the quadrotor's center of mass from the earth's fixed inertial system along the X, Y, and Z axes, respectively. Φ, Θ and Ψ are the three Euler angles that represent the direction of the quadcopter. Φ is called the roll angle, it is the angle concerning the X-axis, Θ is the pitch angle concerning the Y-axis, and Ψ is the yaw angle for the Z-axis. Figure 2.4 clearly explains the Euler angles. The roll and pitch angles are usually called the quadcopter attitude, and the yaw angle is called the quadcopter heading. For linear motion, the distance from the ground is called altitude, and the X and Y positions in space are usually called quadrotor positions [7].

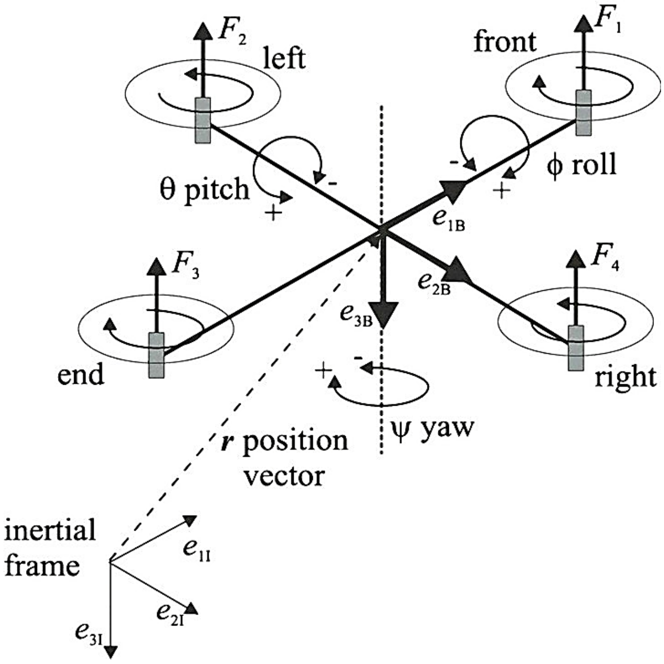


Figure 2.4 Euler angles for a quadcopter UAV [L3].

To produce vertical upward movement, the speeds of all four propellers are increased at the same time, while the speed is reduced to produce vertical downward movement. To produce a roll rotation along the X-axis, the speed of the second and fourth propellers needs to be changed, and for the pitch rotation along the Y-axis, the speed of the first and third propellers needs to be changed. One problem with the quadcopter configuration is that to produce yaw rotation, there must be a difference in the reverse torque produced by each pair of propellers.

For example, for a positive yaw rotation, it is necessary to increase the speed of two rotors rotating clockwise, while the speed of two rotors rotating counter-clockwise should be reduced [3]. Figure 2.5 shows how to produce different motions, note that thicker arrows indicate higher helix speeds.

2.6.2. Advantages and Disadvantages of Quadcopters

Compared with helicopters, some of the advantages of quadcopter helicopters are that the rotor mechanics is simplified because it relies on four fixed-pitch rotors instead of the helicopter's variable-pitch rotors, which makes manufacturing and maintenance easier. In addition, due to the symmetry of the setting, the gyro effect is reduced, making it easier to control.

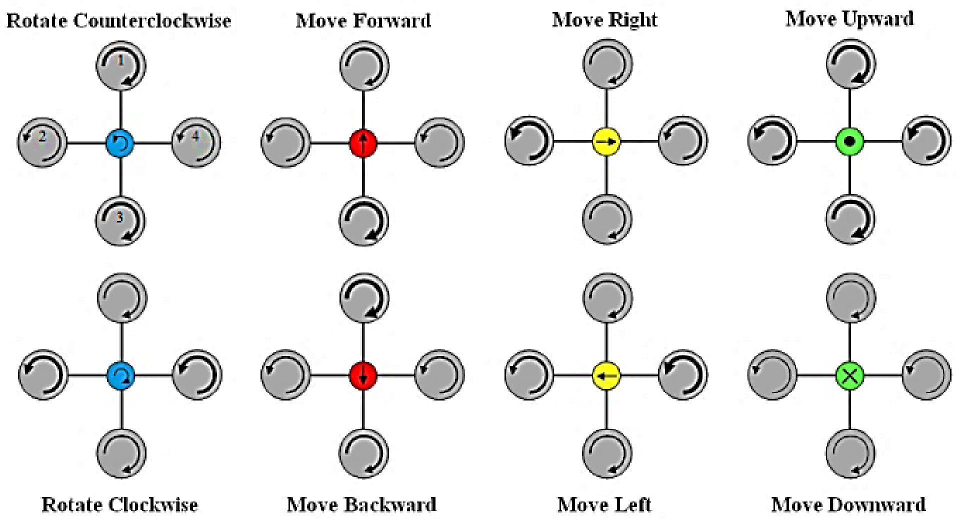


Figure 2.5 Generated motion of the quadcopter [9].

Four propellers that provide four thrust powers are not only a propeller, as well as a propeller, but from the centre of gravity, the presence of four propellers moved more than the helicopter can be stable. More advantages are better manoeuvres due to vertical takeoff and landing force, and the absence of queues, which are smaller sizes, these skills are useful for monitoring the small area and the exploration of buildings.

It has a high payload capacity because four motors have high payload capacity and, therefore, greater impulse. On the other hand, four separate propellers are present, so four propellers consume a lot of energy. In addition, the facts of the four separate propellers are again greater than some of their counterparts, and heavier [3].

2.7. Related Works

Several research papers are reported on the quadcopter altitude, position, attitude, and heading control systems by using different tuning and optimization algorithms. In this section, the closely related works are reviewed based on the trajectory generation, the chattering effect, disturbance rejection, parameter uncertainty, and controller performance, and so on.

The modelling and robust integral sliding mode control of quadrotor UAV have been developed in [10]. The result shows that integral SMC has a better performance parameter result than the convectional SMC in terms of settling time and overshoot. However, since the plant is usually affected by external disturbances and parameter uncertainty. Even though the controller minimizes the effect of uncertainty, the authors didn’t discuss the controller performance in case of when the external disturbance is applied.

A comparative analysis of Adaptive PID, SMC, PID control system approaches for quadrotor UAV attitude and position, stabilization has been formulated in [11]. The authors implemented the proposed system based on external disturbances like the aerodynamic effect. As a result, an adaptive PID controller has a better performance based on the performance comparison criteria of fitness function IAE than others. The drawback of this research study is that the performance of the proposed Adaptive PID controller was not checked in case of when there is a parameter uncertainty like variable mass.

Similarly, LQR based control of the quadcopter dynamics has been proposed in [12]. The simulation has been carried out using setpoint tracking capability. It has good performance parameter results without considering the effects of external disturbance and internal parameter uncertainty. However, the result can be further improved by using adaptive control techniques like overshoot problems.

A comparative analysis between the PID and FOPID controllers for attitude, altitude, and position control of a quadrotor UAV has been analyzed in [13]. The result shows that the FOPID controller has a better performance parameter fitness value (IAE) than the conventional PID controller. The system has been performed without the effect of external and internal disturbances. However, the result can be further improved using additional online tuning techniques of the proposed controller along with disturbance rejecting capabilities.

Trajectory tracking controller design for a quadrotor aircraft based on fuzzy sliding-mode control is proposed in [14]. The simulation was carried out using set-point tracking and disturbance rejection capabilities. Both the external disturbance and internal parameter uncertainty (mass) were considered. Its result shows that when the mass is increased slightly the controlled variables can't track their reference trajectories. Initially, the system has high overshoot and this may lead to additional stability problems.

Attitude control of a cascade sliding mode control (CSMC) for quadrotor dynamics has been carried out in [15]. The proposed control system was implemented for the roll, pitch, and yaw control by considering external disturbances to the system. As a result, the CSMC has a good result in the case of applying constant mass.

A Neural network-based adaptive sliding mode control design for position and attitude control of a quadrotor UAV has been proposed in [16]. The simulation has been carried out by considering both in the case of underactuated and full actuated system types. A neural network is applied for adapting the parameters of the adaptation law. However, using online tuning like neural network creates some computation time in the simulation and it may lead to high transient time. The result of this study also shows that the settling time has almost 10 sec out of 80 sec of the simulation time. As compared to the previously reviewed literature, this work has a high settling time.

In general, from the above literature reviewed in this section, the performances of altitude, position, attitude, and heading controls of a quadcopter using different controller techniques are reviewed and discussed. Usually controlling the positions (x and y) of the quadcopter is not common. Most of the works that are done before are based on attitude (pitch, roll)and heading (yaw) control. Hence, UAVs like quadcopter systems are highly nonlinear and affected by external disturbance and internal parameter variations. Most of the literature was considered only trajectory tracking and external disturbance effects and some of the literature only considered the internal parameter uncertainty effects.

To check and validate the performance of the controllers there must be consideration of these effects. Nonlinear control systems like SMC have high performance in case of disturbance rejection capabilities and set point tracking problems. However, the conventional SMC has its chattering effect and it may lead to the vibration in the quadcopter system. Eliminating the chattering effect is another problem in the control system of the quadcopter.

Therefore, in this research study, an adaptive sliding mode controller for controlling the altitude, position, attitude, and heading of the quadcopter UAV is proposed and implemented. The proposed control system performance is checked based on the minimum jerk trajectory and both disturbance effects (external disturbance and internal parameter uncertainty).

CHAPTER THREE

3. MATERIALS AND METHODS

3.1. Chapter Overview

This chapter deals with the methods used to complete this research work and the derivation of kinematics and dynamic mathematical modelling of the quadcopter system based on Euler-Newton formalism.

3.2. Materials

In this research work, software such as MATLAB/2019b, Microsoft office 2019, Microsoft Visio 2016, and Math Type 7.4.8.0 equation is used. MATLAB is the computing software that consists of technical toolboxes and Simulink. Microsoft office is used for editing the thesis documentation. Microsoft Visio 2016 is drawing software used to draw a variety of diagrams. Math Type 7.4.8.0 version is used for writing mathematical formulas and equations in Microsoft offices word and PowerPoint presentation tools.

3.3. Methods

The method used for this research study to accomplish the required task is shown in Figure 3.1. The closely related papers and literature are reviewed and data are collected. The collected data is analyzed mathematically and graphically for developing system modelling. The dynamics modelling of the quadcopter is analyzed. Then the minimum jerk trajectory is designed. The sliding mode control is designed with a PID sliding surface. An adaptive controller is designed for the SMC and then the proposed modelling and simulation result of the system is analyzed. The block diagram in Figure 3.1 summarizes the method followed throughout the research work.

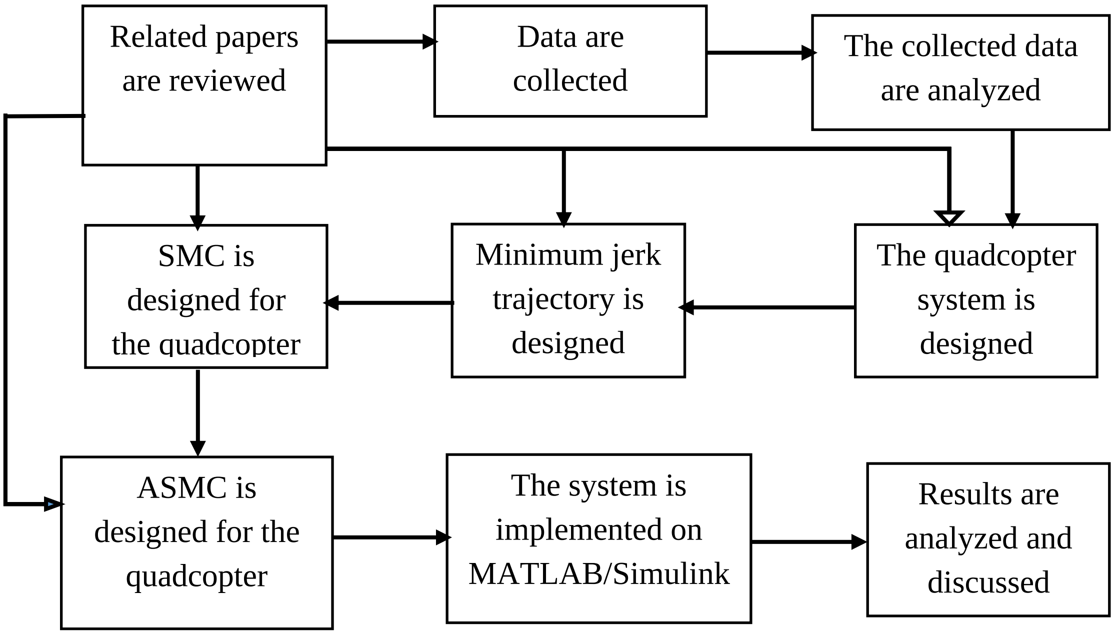


Figure 3.1 Block diagram of the research methodology

3.4. Block Diagram of the Proposed ASMC for Quadcopter System

The full control scheme of the system is proposed as a multiloop consisting of an inner loop that controls attitude and heading and an outer loop to control the position of the quadcopter system. The objective of the research study is to design a robust ASMC for altitude, position, attitude, and heading dynamics that guarantees a fast response, rapid adaptation, and good tracking with parameter uncertainty and external disturbance. The block diagram of the proposed ASMC for the quadcopter system is shown in Figure 3.2. The desired reference trajectory is given to the six states of the quadcopter and then to the ASMC. And finally outputs of the controller is given to the quadcopter system and the actual values of the states are obtained.

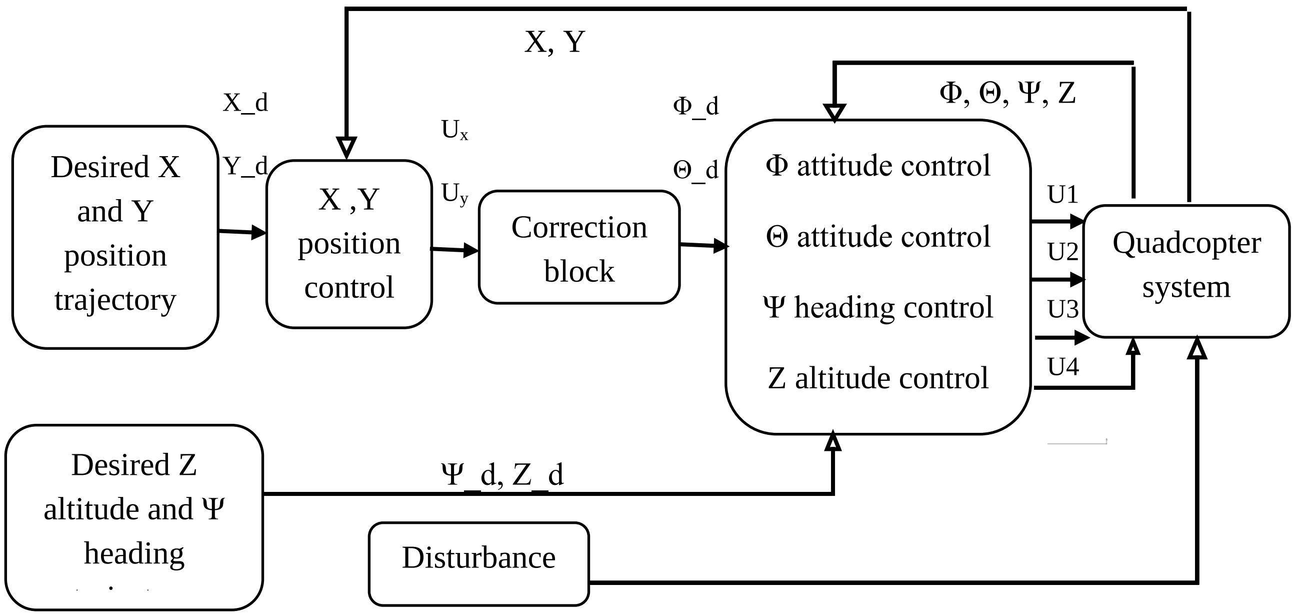


Figure 3.2 Block diagram of the proposed quadcopter control system.

3.5. Mathematical Modeling of Quadcopter

3.5.1. Six-Degrees of Freedom (6-DOF)

To express the motion of the quadcopter in terms of equations it is important to have the idea of 6-degrees of freedom (6-DOF). The 6-DOF concept gives the position and the orientation of the quadcopter in three dimensions (3-D). The quadcopter dynamic equations represent the change in position and orientation over time. To achieve the control of the quadcopter system in space six coordinates are required, three for describing the position and three for describing the orientation in space [17].

3.5.2. Quadcopter System Descriptions

The following assumptions are considered for the derivation of kinematics and dynamics models of the quadcopter system [18].

- The quadcopter's structure is rigid and symmetrical.
- The quadcopter's centre of gravity coincides with the origin of the body-fixed frame.
- The Propellers are rigid.
- Thrust and drag forces are proportional to the square of the propeller's speed.

3.5.2.1. 3.4.2.1 Control Inputs

There are four identical propeller and DC motor pairs that can generate lift forces to generate the motion of the quadcopter. Two opposite side motors rotate clockwise, while the other pair turn counter-clockwise. There are two possible control orientations for the quadcopter system. One is the plus configuration and the other one is the cross configuration as shown in Figure 3.4.

There are four rotors of the quadcopter that means four control inputs can be defined.

- $\tilde{U}_1$: throttle input, which is the total force generated by the four propellers.
- $\tilde{U}_2$: roll input, which is the moment difference around x-axis due to left-side and right-side propellers.
- $\tilde{U}_3$: pitch input, which is the moment difference around y-axis due to front and rear propellers.
- $\tilde{U}_4$: yaw input, which is the net torque exerted on the system around z-axis by the 4 propellers..
- According to the above descriptions, the four inputs are expressed with the following equations [19]:

$$\begin{aligned} \tilde{U}_1 &= F_1 + F_2 + F_3 + F_4 \\ \tilde{U}_2 &= l\left(F_1 - F_2 - F_3 + F_4\right) \\ \tilde{U}_3 &= l\left(F_1 + F_2 - F_3 - F_4\right) \\ \tilde{U}_4 &= -M_1 + M_2 - M_3 + M_4 \end{aligned} \tag{3.1}$$

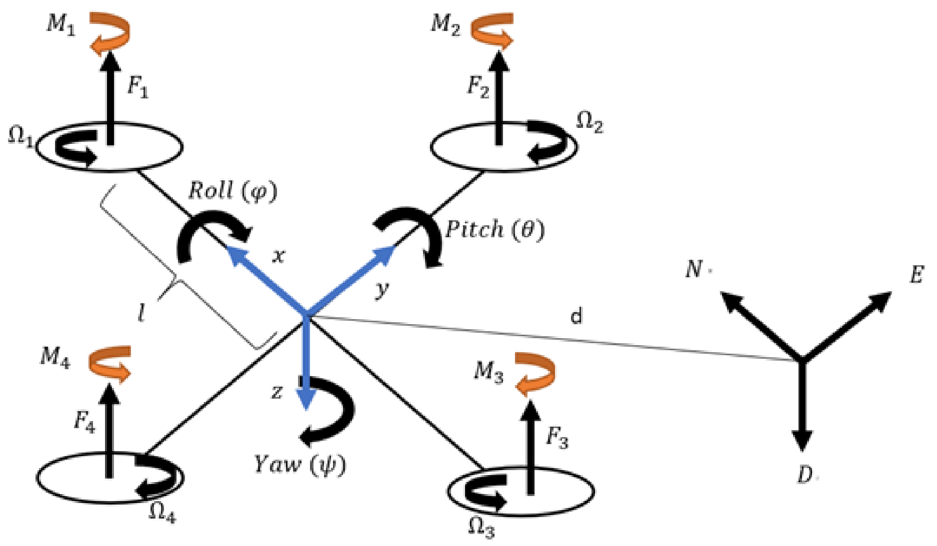


Figure 3.3 Force and moments of quadcopter [L2].

Since quadcopter has a four-rotor, that produces thrust forces, which are proportional to the square of rotor angular speed. From Figure 3.4 above M_1 and M_3 rotating in counterclockwise and M_2 and M_4 rotating in a clockwise direction.

3.6. Reference Frames and Transformation Matrices

A reference frame is a set of points in space for which the distance between any two points are fixed at all times. There are two coordinate frames to be defined for the modelling of the quadcopter system. The first one is an inertial frame so that the dynamics model can be derived from Newton's laws. This frame can be selected as the Earth reference frame which is fixed at the ground. It has axes notated with N, E, and D symbols which indicates that the axes extend in North, East, and Downwards directions. The quadcopter flies over this fixed inertial reference frame, and its orientation can be expressed using Euler angles (Φ, Θ, Ψ) relative to this inertial frame. The second frame is the body-fixed reference frame which, is attached to the centre of gravity of the quadcopter [19].

The transformation can be derived by rotating the body frame around the *Z*-axis of the earth frame by the yaw angle Ψ, which is then followed by rotating around the *Y*-axis by the pitch angle Θ and finally by rotating around the *X*-axis by the roll angle Φ, where all angles are bounded; to avoid the system singularities it is sufficient to assume:

$$\frac{-\pi}{2} < \Phi < \frac{\pi}{2},\ \frac{-\pi}{2} < \Theta < \frac{\pi}{2},\ -\pi < \Psi < \pi \tag{3.2}$$

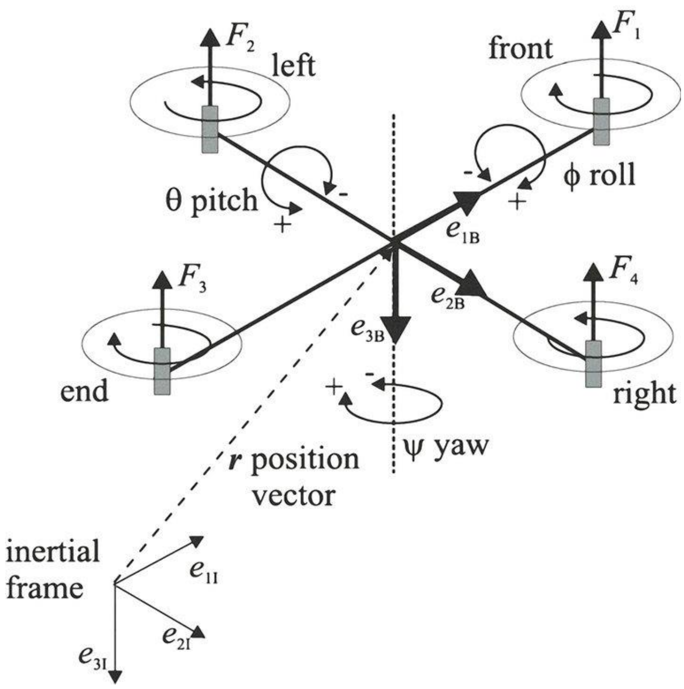


Figure 3.4 Reference frames for the quadcopter system [L3]

The position and orientation of the quadcopter can be defined relative to the ground with the following notations [18].

$$\begin{aligned} \bar{\xi} &= \begin{bmatrix} X & Y & Z \end{bmatrix}^T \\ \bar{\eta} &= \begin{bmatrix} \Phi & \Theta & \Psi \end{bmatrix}^T \end{aligned} \tag{3.3}$$

In Equation (3.3), $\bar{\xi}$ is expressed in Earth fixed frame (f_E) and represents the absolute position of the quadcopter centre of gravity relative to the Earth reference frame. And $\bar{\eta}$ represents the orientation of the quadcopter relative to the Earth frame as expressed in (f_E) [20].

The velocities (w.r.t f_E) in the body-fixed frame, are expressed in Equation (3.4) below [18].

$$\begin{aligned} \tilde{v}_B &= \begin{bmatrix} u & v & w \end{bmatrix}^T \\ \tilde{\omega} &= \begin{bmatrix} p & q & r \end{bmatrix}^T \end{aligned} \tag{3.4}$$

3.6.1. Transformation Matrices for Translational Velocities

The translational velocity of the body-fixed frame can be expressed in terms of the translational velocity of the Earth frame by making three successive rotations.

The transformation matrix R_{BE} that transforms Earth frame (f_E) to body-fixed frame (f_B) can be written as [20]:

$$R_{BE} = R_\Phi R_\Theta R_\Psi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\Phi) & s(\Phi) \\ 0 & -s(\Phi) & c(\Phi) \end{bmatrix} \begin{bmatrix} c(\Theta) & 0 & -s(\Theta) \\ 0 & 1 & 0 \\ s(\Theta) & 0 & c(\Theta) \end{bmatrix} \begin{bmatrix} c(\Psi) & s(\Psi) & 0 \\ -s(\Psi) & c(\Psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.5}$$

For simplicity, cosine and sine functions are denoted with c and s symbols in long expressions. By simplifying Euquation (3.5), the final form of the transformation matrix R_{BE} is obtained as follows:

$$R_{BE} = \begin{bmatrix} c(\Phi)c(\Psi) & c(\Theta)s(\Psi) & -s\Theta \\ c(\Psi)s(\Theta)s(\Phi) - c(\Phi)s(\Psi) & c(\Phi)c(\Psi) + s(\Theta)s(\Phi)s(\Psi) & c(\Theta)s(\Phi) \\ s(\Phi)s(\Psi) + c(\Phi)c(\Psi)s(\Theta) & c(\Phi)s(\Theta)s(\Psi) - c(\Psi)s(\Phi) & c(\Theta)c(\Phi) \end{bmatrix} \tag{3.6}$$

The translational velocity of the quadcopter, as expressed in the Earth frame f_E, can be represented with the following equation:

$$\tilde{v}_B = R_{BE}\tilde{v}_E = R_{BE}\dot{\tilde{\xi}} \tag{3.7}$$

The inverse relation is shown in Equation (3.8).

$$\dot{\tilde{\xi}} = \tilde{v}_E = R_{EB}\tilde{v}_B \tag{3.8}$$

From Equations (3.7) and (3.8), the following result is concluded:

$$R_{EB} = R_{BE}{}^{-1} \tag{3.9}$$

3.6.2. Transformation Matrices for Angular Velocities

Using the following relation, body angular velocities $\begin{bmatrix} p & q & r \end{bmatrix}$ can be related with Euler rates $\begin{bmatrix} \dot{\Phi} & \dot{\Theta} & \dot{\Psi} \end{bmatrix}$ [21]:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = R_\Phi R_\Theta R_{\dot\Psi} \begin{bmatrix} 0 \\ 0 \\ \dot\Psi \end{bmatrix} + R_\Phi R_{\dot\Theta} \begin{bmatrix} 0 \\ \dot\Theta \\ 0 \end{bmatrix} + R_{\dot\Phi} \begin{bmatrix} \dot\Phi \\ 0 \\ 0 \end{bmatrix} \tag{3.10}$$

R_Φ, R_Θ and R_Ψ are already defined in Equation (3.5). Besides, $R_{\dot\Phi}, R_{\dot\Theta}$ and $R_{\dot\Psi}$ can be accepted as a unit matrix I since $\dot\Phi, \dot\Theta$ and $\dot\Psi$ is small. By substituting them into Equation (3.10), the following expression is obtained [21]:

$$\begin{aligned} \begin{bmatrix} p \\ q \\ r \end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\Phi) & \sin(\Phi) \\ 0 & -\sin(\Phi) & \cos(\Phi) \end{bmatrix} \begin{bmatrix} \cos(\Theta) & 0 & -\sin(\Theta) \\ 0 & 1 & 0 \\ \sin(\Theta) & 0 & \cos(\Theta) \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot\Psi \end{bmatrix} \\ &+ \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\Phi) & \sin(\Phi) \\ 0 & -\sin(\Phi) & \cos(\Phi) \end{bmatrix} \begin{bmatrix} 0 \\ \dot\Theta \\ 0 \end{bmatrix} + \begin{bmatrix} \dot\Phi \\ 0 \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} -\dot\Phi - \sin(\Theta)\dot\Psi \\ \cos(\Phi)\dot\Theta + \sin(\Phi)\cos(\Theta)\dot\Psi \\ -\sin(\Phi)\dot\Theta + \cos(\Phi)\cos(\Theta)\dot\Psi \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & -\sin(\Theta) \\ 0 & \cos(\Phi) & \sin(\Phi)\cos(\Theta) \\ 0 & -\sin(\Phi) & \cos(\Phi)\cos(\Theta) \end{bmatrix} \begin{bmatrix} \dot\Phi \\ \dot\Theta \\ \dot\Psi \end{bmatrix} \end{aligned} \tag{3.11}$$

By calling the angular transformation matrix with R_r notation, the relation of angular velocities can be expressed as follows [21]:

$$\tilde\omega = R_r \dot{\tilde\eta} \tag{3.12}$$

where,

$$R_r = \begin{bmatrix} 1 & 0 & -\sin(\Theta) \\ 0 & \cos(\Phi) & \sin(\Phi)\cos(\Theta) \\ 0 & -\sin(\Phi) & \cos(\Phi)\cos(\Theta) \end{bmatrix} \tag{3.13}$$

3.7. Nonlinear Dynamic Model

The combined rotational and translational dynamics of a rigid body is described using Newton's equations of motion. These equations can be investigated under two subsystems: the translational subsystem (altitude z, x, and y) and the rotational subsystem (roll, pitch, and yaw). Following these, dynamics of the rigid body are expressed under the following generalized form [22]:

$$\mathrm{I}\dot{\tilde{\omega}} + \tilde{\omega} \times \mathrm{I}\tilde{\omega} = \sum M_{ext} \tag{3.14}$$

$$m\dot{\tilde{v}}_B + \tilde{\omega} \times m\tilde{v}_B = \sum F_{ext} \tag{3.15}$$

3.7.1. Forces and Moments Acting on the Quadcopter

Aerodynamic forces and moments are produced and exerted on the system as the propellers rotate. The force and moment of each rotor are shown in Figure 3.5. By specifying each rotor with "*i*" subscript, force and moments of the i^{th} rotor are expressed as follows [23]:

$$\begin{aligned} F_i &= b\omega_i^2 \\ M_i &= d\omega_i^2 \end{aligned} \tag{3.16}$$

where *b* and *d* terms represent the aerodynamic thrust and drag contributions. These factors depend on rotor blade radius and area, air density, speed of the rotor and experimental aerodynamics coefficients C_τ and C_D. It can be assumed that air density is constant as the quadcopter's altitude variation is somehow limited.

The control inputs are defined as $\tilde{U}_1, \tilde{U}_2, \tilde{U}_3$ and $\tilde{U}_4$ should have to be put into vector form to be able to use in the equations of motion.

The total force generated by the propellers can be written in a frame f_B as [23]:

$$F_{prop,B} = \begin{bmatrix} 0 \\ 0 \\ -\tilde{U}_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -b\left(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2\right) \end{bmatrix} \tag{3.17}$$

The moments acting on the quadcopter system $M_{prop,B}$ can be written as the following equation [23]:

$$M_{prop,B} = \begin{bmatrix} \tilde{U}_2 \\ \tilde{U}_3 \\ \tilde{U}_4 \end{bmatrix} = \begin{bmatrix} F_1 l - F_2 l - F_3 l + F_4 l \\ F_1 l + F_2 l - F_3 l - F_4 l \\ -M_1 + M_2 - M_3 + M_4 \end{bmatrix} = \begin{bmatrix} bl\left(\omega_1^2 - \omega_2^2 - \omega_3^2 + \omega_4^2\right) \\ bl\left(\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2\right) \\ d\left(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2\right) \end{bmatrix} \tag{3.18}$$

There is a contribution of the gravitational force in addition to propeller force and moments. The effect of gravitational force can be expressed in Earth frame f_E as follows:

$$F_{g,E} = m\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3.19}$$

Using the rotation matrix R_{BE} found in Equation (3.6), the gravitational force can also be expressed in body-fixed frame f_B as follows.

$$F_{g,B} = mR_{BE}\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3.20}$$

Another moment contributor is the gyroscopic effects produced by the propellers. The following equation can be written to describe the gyroscopic moment in f_B [24]:

$$M_{g,B} = -\tilde{\omega} \times \begin{bmatrix} 0 \\ 0 \\ \mathrm{I}_r\hat{\Omega} \end{bmatrix} \tag{3.21}$$

In Equation (3.21), $\hat{\Omega}$ defines the propeller's relative speed and given as follows:

$$\hat{\Omega} = \hat{\Omega}_1 - \hat{\Omega}_2 + \hat{\Omega}_3 - \hat{\Omega}_4 \tag{3.22}$$

Additionally, there is also aerodynamic drag force which is neglected for this research work. Now, Equations (3.1), (3.18), (3.19), (3.20), and (3.21) can be combined to obtain $\sum F_{ext}$ and $\sum M_{ext}$,

$$\sum F_{ext} = \begin{bmatrix} 0 \\ 0 \\ -\tilde{U}_1 \end{bmatrix} + mR_{BE}\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3.23}$$

$$\sum M_{ext} = \begin{bmatrix} \tilde{U}_2 \\ \tilde{U}_3 \\ \tilde{U}_4 \end{bmatrix} - \tilde{\omega} \times \begin{bmatrix} 0 \\ 0 \\ \mathrm{I}_r \end{bmatrix}\hat{\Omega} \tag{3.24}$$

3.7.2. Rotational Equations of Motion

The rotational equations of motion can be derived using Equation(3.14). The left-hand side of the equation $\tilde{\omega}$ is already defined in Equation (3.4). $\dot{\tilde{\omega}}$ Can also be expressed as follows:

$$\dot{\tilde{\omega}} = \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} \tag{3.25}$$

Since the derivation has been made in the body-fixed frame, a time-independent inertia matrix is obtained. Finally, the quadcopter's inertia matrix can be expressed as:

$$\mathrm{I} = \begin{bmatrix} \mathrm{I}_x & 0 & 0 \\ 0 & \mathrm{I}_y & 0 \\ 0 & 0 & \mathrm{I}_z \end{bmatrix} \tag{3.26}$$

In Equation (3.26), $\mathrm{I}_x, \mathrm{I}_y$ and I_z represent the moments of inertia around principal axes. The rotational equations of motion can be written with the previously introduced vectors as:

$$\begin{bmatrix} \mathrm{I}_x & 0 & 0 \\ 0 & \mathrm{I}_y & 0 \\ 0 & 0 & \mathrm{I}_z \end{bmatrix} \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} \mathrm{I}_x & 0 & 0 \\ 0 & \mathrm{I}_y & 0 \\ 0 & 0 & \mathrm{I}_z \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \tilde{U}_2 \\ \tilde{U}_3 \\ \tilde{U}_4 \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ \mathrm{I}_r \hat{\Omega} \end{bmatrix} \tag{3.27}$$

By rearranging Equation (3.27), the final form of the rotational equations of motion are obtained as follows:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \left(\mathrm{I}_y - \mathrm{I}_z\right) qr / \mathrm{I}_x \\ \left(\mathrm{I}_z - \mathrm{I}_x\right) pr / \mathrm{I}_y \\ \left(\mathrm{I}_x - \mathrm{I}_y\right) pq / \mathrm{I}_z \end{bmatrix} + \begin{bmatrix} \tilde{U}_2 / \mathrm{I}_x \\ \tilde{U}_3 / \mathrm{I}_y \\ \tilde{U}_4 / \mathrm{I}_z \end{bmatrix} + \begin{bmatrix} -\mathrm{I}_r q \hat{\Omega} / \mathrm{I}_x \\ \mathrm{I}_r p \hat{\Omega} / \mathrm{I}_y \\ 0 \end{bmatrix} \tag{3.28}$$

To obtain Euler angles, the body rates should be used with the transformation matrix R_r that is found in Equation (3.13):

$$\begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\Phi)\tan(\Theta) & \cos(\Phi)\tan(\Theta) \\ 0 & \cos(\Phi) & -\sin(\Phi) \\ 0 & \sin(\Phi)/\cos(\Theta) & \cos(\Phi)/\cos(\Theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{3.29}$$

Then the simplified rotational equations of motion can be expressed as follows:

$$\begin{bmatrix} \ddot{\Phi} \\ \ddot{\Theta} \\ \ddot{\Psi} \end{bmatrix} = \begin{bmatrix} (\mathrm{I}_y - \mathrm{I}_z)\dot{\Theta}\dot{\Psi}/\mathrm{I}_x \\ (\mathrm{I}_z - \mathrm{I}_x)\dot{\Phi}\dot{\Psi}/\mathrm{I}_y \\ (\mathrm{I}_x - \mathrm{I}_y)\dot{\Phi}\dot{\Theta}/\mathrm{I}_z \end{bmatrix} + \begin{bmatrix} \tilde{U}_2/\mathrm{I}_x \\ \tilde{U}_3/\mathrm{I}_y \\ \tilde{U}_4/\mathrm{I}_z \end{bmatrix} + \begin{bmatrix} -\mathrm{I}_r\dot{\Theta}\hat{\Omega}/\mathrm{I}_x \\ \mathrm{I}_r\dot{\Phi}\hat{\Omega}/\mathrm{I}_y \\ 0 \end{bmatrix} \tag{3.30}$$

3.7.3. Translational Equations of Motion

From Equation (3.15), $\sum F_{ext}$ has already been derived. The only unknown term is $\dot{\tilde{v}}_B$ that can be found by differentiating Equation (3.7) concerning time.

$$\ddot{\tilde{\xi}} = R_{EB}\tilde{v}_B + R_{EB}\dot{\tilde{v}}_B \tag{3.31}$$

Since the transformation matrix R_{EB} is orthogonal, the properties of orthogonality can be used, so that $\dot{R}_{EB}$ can be expressed as the dot product of R_{EB} the skew-symmetric matrix of ω. Then, $\dot{R}_{EB}$ becomes [22]:

$$\dot{R}_{EB} = R_{EB}\tilde{\tilde{\omega}} \tag{3.32}$$

It is also known that the dot product of any vector with a skew-symmetric matrix $\tilde{\tilde{\omega}}$ can be expressed as a cross product of $\tilde{\omega}$ with that vector [22].

$$\tilde{\tilde{\omega}}S = \tilde{\omega} \times S \tag{3.33}$$

By substituting Equations (3.32) and (3.33) into Equation (3.31), the following expression is obtained:

$$\ddot{\tilde{\eta}} = R_{EB}\left(\tilde{\omega} \times \tilde{v}_B\right) + R_{EB}\dot{\tilde{v}}_B \tag{3.34}$$

Finally, $\dot{V}_B$ term can be as follows:

$$\dot{V}_B = R_{EB}^{-1}\ddot{\vartheta} - \omega \times V_B \tag{3.35}$$

Now, by substituting Equations (3.35) and (3.23) into Equation (3.15), translational equations of motion are obtained as:

$$\begin{bmatrix} 0 \\ 0 \\ -\tilde{U}_1 \end{bmatrix} + mR_{BE}\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = m\left(R_{EB}^{-1}\ddot{\eta} - \tilde{\omega}\times\tilde{v}_B\right) + \tilde{\omega}\times m\tilde{v}_B \tag{3.36}$$

By multiplying both sides of the Equation (3.36) with R_{EB} and simply dividing both sides with constant m , the following simplified form is obtained:

$$R_{EB}\begin{bmatrix} 0 \\ 0 \\ -\tilde{U}_1/m \end{bmatrix} + R_{EB}R_{BE}\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = R_{EB}R_{EB}{}^{-1}\ddot{\eta} \tag{3.37}$$

In Equation (3.37), $R_{EB}R_{EB}{}^{-1}$ and $R_{EB}R_{BE}$ terms are simplifying to yield an identity matrix $I_{3\times3}$. Thus, equation can be rewritten as follows:

$$\ddot{\eta} = R_{EB}\begin{bmatrix} 0 \\ 0 \\ -\tilde{U}_1/m \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3.38}$$

At the last step, the final version of translational equations of motion can be obtained by putting $\left[\ddot{X},\ddot{Y},\ddot{Z}\right]$into $\ddot{\xi}$ as follows:

$$\begin{bmatrix} \ddot{\mathrm{X}} \\ \ddot{Y} \\ \ddot{\mathrm{Z}} \end{bmatrix} = R_{EB}\begin{bmatrix} 0 \\ 0 \\ -\tilde{U}_1/m \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3.39}$$

Finally, after substituting R_{EB} in to Equation(3.39) the final equation becomes:

$$\begin{aligned} \ddot{\mathrm{X}} &= -\frac{\tilde{U}_1}{m}\left(\cos(\Phi)\sin(\Theta)\cos(\Psi)+\sin(\Phi)\sin(\Psi)\right) \\ \ddot{Y} &= -\frac{\tilde{U}_1}{m}\left(\cos(\Phi)\sin(\Theta)\sin(\Psi)-\sin(\Phi)\cos(\Psi)\right) \\ \ddot{Z} &= g-\frac{\tilde{U}_1}{m}\left(\cos(\Phi)\cos(\Theta)\right) \end{aligned} \tag{3.40}$$

3.7.4. State Space Equation for Quadcopter Dynamics

The obtained rotational and translational equations of motion can also be represented in a state-space form. There are 12 states for a quadcopter system, and the state vector can be defined as;

$$x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & x_{11} & x_{12} \end{bmatrix}^T \tag{3.41}$$

Euler angles and their rates are used in the rotational equations of motion and the orientation of the quadcopter is directly defined with *X*, *Y* and *Z* for translational dynamics. Thus, the states can be written as:

$$X = \begin{bmatrix} \Phi & \dot{\Phi} & \Theta & \dot{\Theta} & \Psi & \dot{\Psi} & Z & \dot{Z} & X & \dot{X} & Y & \dot{Y} \end{bmatrix}^T \tag{3.42}$$

In order to form the state space model of the system the control input vector should also be defined. Therefore control vector $\tilde{U}$ can be represented as:

$$\tilde{U} = \begin{bmatrix} \tilde{U}_1 & \tilde{U}_2 & \tilde{U}_3 & \tilde{U}_4 \end{bmatrix}^T \tag{3.43}$$

By obtaining the control input expressions from Equations (3.17) and (3.18), the control vector U can be written as follows:

$$\begin{bmatrix} \tilde{U}_1 \\ \tilde{U}_2 \\ \tilde{U}_3 \\ \tilde{U}_4 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ bl & -bl & -bl & bl \\ bl & bl & -bl & -bl \\ -d & d & -d & d \end{bmatrix} \begin{bmatrix} \hat{\Omega}_1^2 \\ \hat{\Omega}_2^2 \\ \hat{\Omega}_3^2 \\ \hat{\Omega}_4^2 \end{bmatrix} \tag{3.44}$$

Control inputs have been defined in terms of the rotor speeds and the inverse relationship can be written as follows:

$$\begin{bmatrix} \hat{\Omega}_1^2 \\ \hat{\Omega}_2^2 \\ \hat{\Omega}_3^2 \\ \hat{\Omega}_4^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{4b} & \frac{1}{4bl} & \frac{1}{4bl} & -\frac{1}{4d} \\ \frac{1}{4b} & -\frac{1}{4bl} & \frac{1}{4bl} & \frac{1}{4d} \\ \frac{1}{4b} & -\frac{1}{4bl} & -\frac{1}{4bl} & -\frac{1}{4d} \\ \frac{1}{4b} & \frac{1}{4bl} & -\frac{1}{4bl} & \frac{1}{4d} \end{bmatrix} \begin{bmatrix} \tilde{U}_1 \\ \tilde{U}_2 \\ \tilde{U}_3 \\ \tilde{U}_4 \end{bmatrix} \tag{3.45}$$

From each row of Equation (3.30), separate equations can be written down as follows:

$$\ddot{\Phi} = \frac{\mathrm{I}_y - \mathrm{I}_z}{\mathrm{I}_x}\dot{\Theta}\dot{\Psi} + \frac{\tilde{U}_2}{\mathrm{I}_x} - \frac{\mathrm{I}_r}{\mathrm{I}_x}\dot{\Theta}\hat{\Omega}$$
$$\ddot{\Theta} = \frac{\mathrm{I}_z - \mathrm{I}_x}{\mathrm{I}_y}\dot{\Phi}\dot{\Psi} + \frac{\tilde{U}_3}{\mathrm{I}_y} + \frac{\mathrm{I}_r}{\mathrm{I}_y}\dot{\Phi}\hat{\Omega} \tag{3.46}$$
$$\ddot{\Psi} = \frac{\mathrm{I}_x - \mathrm{I}_y}{\mathrm{I}_z}\dot{\Phi}\dot{\Theta} + \frac{\tilde{U}_4}{\mathrm{I}_z}$$

Equation (3.46) can be further simplified by defining new variables as follows:

$$\ddot{\Phi} = a_1 x_4 x_6 + b_1 \tilde{U}_2 - a_2 x_4 \hat{\Omega}$$
$$\ddot{\Theta} = a_3 x_2 x_6 + b_2 \tilde{U}_3 + a_4 x_2 \hat{\Omega} \tag{3.47}$$
$$\ddot{\Psi} = a_5 x_2 x_4 + b_3 \tilde{U}_4$$

Where,

$$a_1 = \frac{\mathrm{I}_y - \mathrm{I}_z}{\mathrm{I}_x}, a_2 = \frac{\mathrm{I}_r}{\mathrm{I}_x}, a_3 = \frac{\mathrm{I}_z - \mathrm{I}_x}{\mathrm{I}_y}, a_4 = \frac{\mathrm{I}_r}{\mathrm{I}_y}, a_5 = \frac{\mathrm{I}_x - \mathrm{I}_y}{\mathrm{I}_z}$$
$$b_1 = \frac{1}{\mathrm{I}_x}, b_2 = \frac{1}{\mathrm{I}_y}, b_3 = \frac{1}{\mathrm{I}_z}$$

For the translational dynamics by defining new variables, Equation (3.40) can be further simplified as follows:

$$\ddot{\mathrm{X}} = -\frac{\tilde{U}_1}{m}\left(\cos(x_1)\sin(x_3)\cos(x_5) + \sin(x_1)\sin(x_5)\right)$$
$$\ddot{Y} = -\frac{\tilde{U}_1}{m}\left(\cos(x_1)\sin(x_5)\sin(x_3) - \sin(x_1)\cos(x_5)\right) \tag{3.48}$$
$$\ddot{\mathrm{Z}} = g - \frac{\tilde{U}_1}{m}\left(\cos(x_1)\cos(x_3)\right)$$

As a remark for Equation (3.49), u_x and u_y terms can also be defined to yield the following:

$$\tilde{U}_x = \cos(\Phi)\sin(\Theta)\cos(\Psi) + \sin(\Phi)\sin(\Psi)$$
$$\tilde{U}_y = \cos(\Phi)\sin(\Theta)\sin(\Psi) - \sin(\Phi)\cos(\Psi) \tag{3.49}$$

The final complete state-space representation of the quadcopter system can be represented as follows:

$$
\dot{x} = f\left(x,\tilde{U}\right) = \begin{cases}
\dot{x}_1 = x_2 \\
\dot{x}_2 = a_1 x_4 x_6 + b_1 \tilde{U}_2 - a_2 x_4 \hat{\Omega} \\
\dot{x}_3 = x_4 \\
\dot{x}_4 = a_3 x_2 x_6 + b_2 \tilde{U}_3 + a_4 x_2 \hat{\Omega} \\
\dot{x}_5 = x_6 \\
\dot{x}_6 = a_5 x_2 x_4 + b_3 \tilde{U}_4 \\
\dot{x}_7 = x_8 \\
\dot{x}_8 = g - \dfrac{\tilde{U}_1}{m}\left(\cos\left(x_1\right)\cos\left(x_3\right)\right) \\
\dot{x}_9 = x_{10} \\
\dot{x}_{10} = -\dfrac{\tilde{U}_1}{m}\left(\cos\left(x_1\right)\sin\left(x_3\right)\cos\left(x_5\right) + \sin\left(x_1\right)\sin\left(x_5\right)\right) \\
\dot{x}_{11} = x_{12} \\
\dot{x}_{12} = -\dfrac{\tilde{U}_1}{m}\left(\cos\left(x_1\right)\sin\left(x_5\right)\sin\left(x_3\right) - \sin\left(x_1\right)\cos\left(x_5\right)\right)
\end{cases} \tag{3.50}
$$

3.8. Equilibrium Points and Stability Analysis

3.8.1. Review on Equilibrium Points

- Consider a nonlinear state-space model given by

$$\dot{x}\left(t\right) = f\left[t, x\left(t\right)\right], t \geq 0$$

  - A vector xo is called an equilibrium if

$$f\left(t, x_0\right) = 0, \forall t \geq 0$$

  - Equilibrium points are solutions to the differential equation.
  - Each solution point determines an equilibrium point [25].

3.8.2. Equilibrium Points for Φ

To find the equilibrium point for Φ,

$$
\begin{cases}
\dot{x}_1 = 0 \\
\dot{x}_2 = 0
\end{cases} \tag{3.51}
$$

$x_2 = 0$ is obtained from the first equation and $\tilde{U}_2 = \frac{-x_4}{b_1}\left(a_1\hat{\Omega} - a_2 x_6\right)$ is obtained from the second equation and then, the condition to be fulfilled for the equilibrium point to exist in the Φ direction is obtained. x_1 can be any value. The equilibrium point for Φ direction is $\left(x_1, 0\right)$

3.8.3. Equilibrium Points for Θ

To find the equilibrium point for Θ,

$$\begin{cases} \dot{x}_3 = 0 \\ \dot{x}_4 = 0 \end{cases} \tag{3.52}$$

$x_4 = 0$ is obtained from the first equation, and $\tilde{U}_3 = \frac{-x_6}{b_2}\left(a_3 x_2 + a_4\hat{\Omega}\right)$ is obtained from the second equation, the condition to be fulfilled for the equilibrium point to exist in Θ direction. x_3 Can be of any value. The equilibrium point for Θ direction is $\left(x_3, 0\right)$.

3.8.4. Equilibrium Points for Ψ

To find the equilibrium point for Ψ,

$$\begin{cases} \dot{x}_5 = 0 \\ \dot{x}_6 = 0 \end{cases} \tag{3.53}$$

$x_6 = 0$ is obtained from the first equation, and $\tilde{U}_4 = \frac{-a_5}{b_3} x_2 x_4$ is obtained from the second equation, the condition to be fulfilled for the equilibrium point to exist in Ψ direction. x_5 Can be of any value. The equilibrium point for Ψ direction is $\left(x_5, 0\right)$.

3.8.5. Equilibrium Points for Z

To find the equilibrium point for Z,

$$\begin{cases} \dot{x}_7 = 0 \\ \dot{x}_8 = 0 \end{cases} \tag{3.54}$$

$x_8 = 0$ is obtained from the first equation, and $\left(c(x_1)c(x_3)\right)\frac{\widehat{U}_1}{m} - g = 0$ is obtained from the second equation, and for this condition to be fulfilled for any value of u_1, the term $\left(c(x_1)c(x_3)\right)\frac{\tilde{U}_1}{m}$ must be g. Evaluating $\left(c(x_1)c(x_3)\right)\frac{\tilde{U}_1}{m} = g$, the result $\left(c(x_1)c(x_3)\right) = \frac{mg}{\tilde{U}_1}$, the condition to be fulfilled for the equilibrium point to have existed in Z direction. x_7 can be any value. The equilibrium points for Z direction are $(x_7, 0)$.

3.8.6. Equilibrium Points for *X*

To find the equilibrium point for *X*,

$$\begin{cases} \dot{x}_9 = 0 \\ \dot{x}_{10} = 0 \end{cases} \tag{3.55}$$

$x_{10} = 0$ is obtained from the first equation and $\left(c(x_1)s(x_3)c(x_5) + s(x_5)s(x_1)\right)\frac{\tilde{U}_1}{m} = 0$ is obtained from the second equation, for this condition to be fulfilled for any value of u_1, the term $\left(c(x_1)s(x_3)c(x_5) + s(x_5)s(x_1)\right)$ must be 0. Then the result is obtained to be $x_3 = \sin^{-1}\left(-\tan(x_5)\tan(x_1)\right)$, the condition to be fulfilled for the equilibrium point to have existed in *X* direction and x_9 can be any value. The equilibrium points for *X* direction are $(x_9, 0)$.

3.8.7. Equilibrium Points for *Y*

To find the equilibrium point for *Y*,

$$\begin{cases} \dot{x}_{11} = 0 \\ \dot{x}_{12} = 0 \end{cases} \tag{3.56}$$

$x_{11} = 0$ is obtained from the first equation and $\left(c(x_1)s(x_3)s(x_5) - s(x_1)c(x_5)\right)\frac{\tilde{U}_1}{m} = 0$ is obtained from the second equation, for this condition to be fulfilled for any value of U_1, the term $\left(c(x_1)s(x_3)s(x_5) - s(x_1)c(x_5)\right)$ must be 0.

Evaluating $\left(c(x_1)s(x_3)s(x_5)-s(x_1)c(x_5)\right)=0$, the result obtained is $x_5=\tan^{-1}\left(\sec(x_1)\right)$, the condition to be fulfilled for the equilibrium point to have existed in the Y direction. x_{11} can be any value. The equilibrium points for the Y direction are $(x_{11},0)$.

3.9. Checking Local Stability of Equilibrium Points

3.9.1. Review on Stability of Nonlinear Systems

- Stability in nonlinear systems is defined based on the equilibrium points.
- An equilibrium point is stable if a solution starting from an initial state inside a ball of a small radius should remain inside a small radius for all time.
- An equilibrium point is unstable if it is not stable, this can be in any one of the following ways:
  - The system trajectory goes to infinity starting from an initial condition.
  - The solution trajectory goes far from the equilibrium point.
  - The system trajectory forms a Limit cycle [25].

3.9.2. Local Stability Analysis using Lyapunov Direct Method

In the process of checking local stability using the Lyapunov direct method, there are two main steps.

Step 1**:** choosing positive definite Lyapunov candidate function.

Definition 1: - Let V (Lyapunov function) be a continuous map from R^n to R. We call $V(x)$ a locally positive definite (LPD) function around $X=x_0$ (equilibrium point) if

1. $V(x_0)=0$
2. $V(x)>0, 0<\|x\|<r,$ for some r [25].

Step 2: checking Lyapunov function derivative along state trajectories around an equilibrium point.

Theorem 3.1: If there exists a Lyapunov function of a system $V(x)$, then $X = x_0$ point in the sense of Lyapunov, if in addition $\dot{V}(x) < 0, 0 < \|x\| < r$, for some *r*. Then $X = x_0$ is an asymptotically stable equilibrium point [25].

3.9.2.1. Local Stability Analysis for Φ

Choose Lyapunov function $V = \frac{1}{2}x_1^2 + \frac{1}{2}x_2^2$ (Positive-definite function), Then computing derivative of Lyapunov function $\dot{V} = x_1\dot{x}_1 + x_2\dot{x}_2$, evaluating $\dot{V}$ around an equilibrium point $\dot{V} > 0$, so all equilibrium points in Φ direction exhibit unstable behaviour.

3.9.2.2. Local Stability Analysis for Θ

Choose Lyapunov function $V = \frac{1}{2}x_3^2 + \frac{1}{2}x_4^2$ (Positive-definite function), Then computing derivative of Lyapunov function $\dot{V} = x_3\dot{x}_3 + x_4\dot{x}_4$, evaluating $\dot{V}$ around an equilibrium point $\dot{V} > 0$, so all equilibrium points in Θ direction exhibit unstable behaviour.

3.9.2.3. Local Stability Analysis for Ψ

Choose Lyapunov function $V = \frac{1}{2}x_5^2 + \frac{1}{2}x_6^2$ (Positive-definite function), Then computing derivative of Lyapunov function $\dot{V} = x_5\dot{x}_5 + x_6\dot{x}_6$, evaluating $\dot{V}$ around an equilibrium point $\dot{V} > 0$, so all equilibrium points in Ψ direction exhibit unstable behaviour.

3.9.2.4. Local Stability Analysis for Z

Choose Lyapunov function $V = \frac{1}{2}x_7^2 + \frac{1}{2}x_8^2$ (Positive-definite function), Then computing derivative of Lyapunov function $\dot{V} = x_7\dot{x}_7 + x_8\dot{x}_8$, evaluating $\dot{V}$ around an equilibrium point $\dot{V} > 0$, so all equilibrium points in Z direction exhibit unstable attitude.

3.9.2.5. Local Stability Analysis for X

Choose Lyapunov function

$$V = \frac{1}{2}x_9^2 + \frac{1}{2}x_{10}^2$$

(Positive-definite function), Then computing the derivative of the Lyapunov function $\dot{V} = x_9\dot{x}_9 + x_{10}\dot{x}_{10}$, evaluating $\dot{V}$ around an equilibrium point $\dot{V} > 0$, so all equilibrium points in X direction exhibit unstable attitude.

3.9.2.6. Local Stability Analysis for Y

Choose Lyapunov function $V = \frac{1}{2}x_{11}^2 + \frac{1}{2}x_{12}^2$ (Positive-definite function), Then computing derivative of Lyapunov function $\dot{V} = x_{11}\dot{x}_{11} + x_{12}\dot{x}_{12}$, evaluating $\dot{V}$ around an equilibrium point $\dot{V} > 0$, so all equilibrium points in Y direction exhibit unstable state.

3.10. Open Loop Analysis

3.10.1. Open-loop model simulation

The open-loop dynamic demonstrates the effect of input on the output state by using rotor speed for calculating input $\tilde{U}_1, \tilde{U}_2, \tilde{U}_3$ and $\tilde{U}_4$ or throttle input, pitch input, roll input and yaw input respectively.

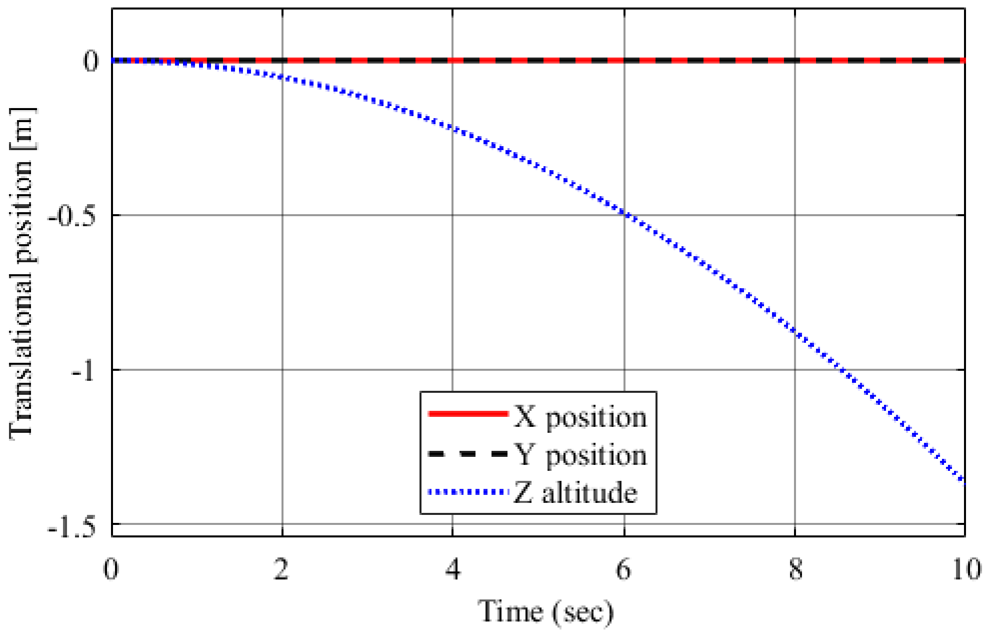


Figure 3.5 Response of nonlinear translational position to throttle input at

$$\hat{\Omega}_1 = \hat{\Omega}_2 = \hat{\Omega}_3 = \hat{\Omega}_4 = 203 \;\; (rad/\sec)$$

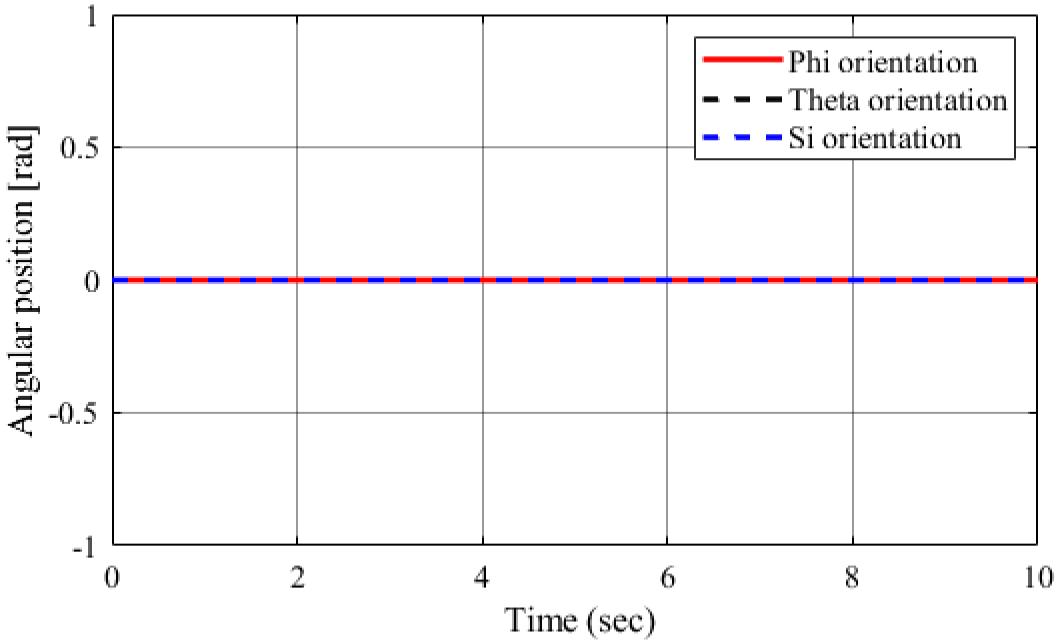


Figure 3.6 Response of nonlinear orientation to throttle input at

$$\hat{\Omega}_1 = \hat{\Omega}_2 = \hat{\Omega}_3 = \hat{\Omega}_4 = 203 \ (rad/\mathrm{sec})$$

The open-loop response result of Figures 3.5 and 3.6 shows when all rotor speed is the same only input $\tilde{U}_1$ exist and affect the altitude state Z-dynamic which means the movement of the quadcopter is vertical along the Z-axis.

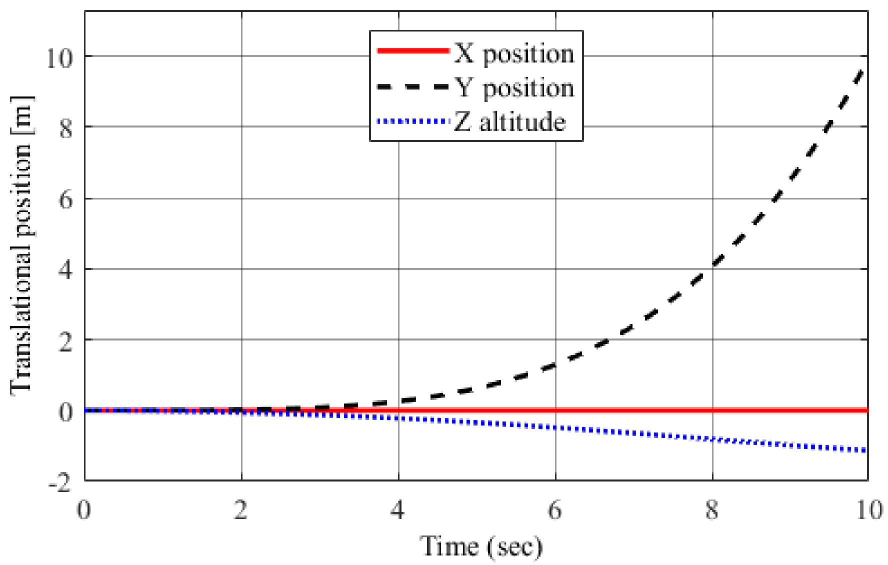


Figure 3.7 Response of nonlinear translational position to roll input at

$$\hat{\Omega}_1 = \hat{\Omega}_3 = 203, \ \hat{\Omega}_2 = 202.9, \ \hat{\Omega}_4 = 203.1 \ (rad/\mathrm{sec})$$

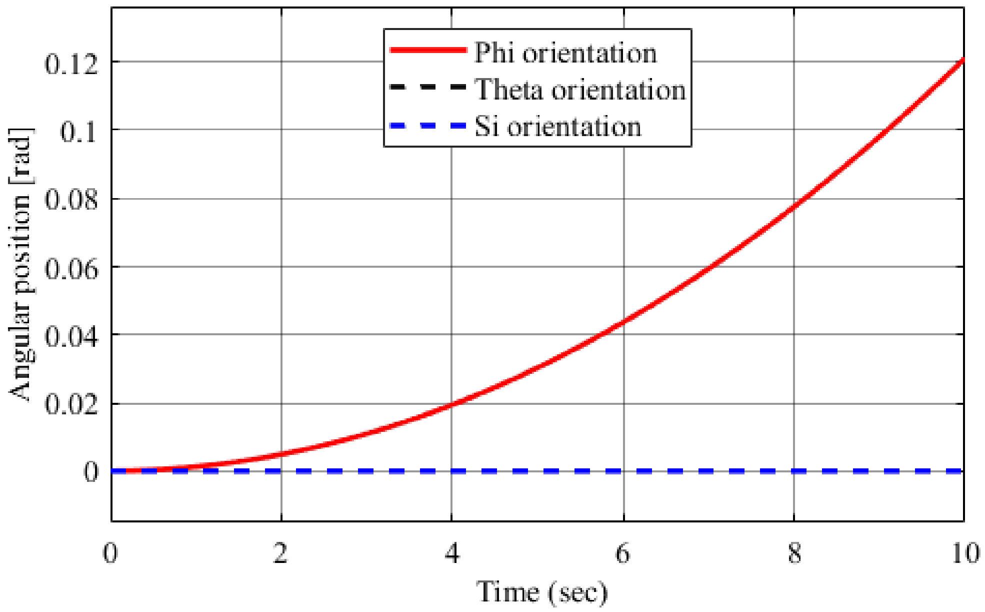


Figure 3.8 Response of nonlinear orientation to roll input at

$$\hat{\Omega}_1 = \hat{\Omega}_3 = 203,\ \hat{\Omega}_2 = 202.9,\ \hat{\Omega}_4 = 203.1\ (rad/\sec)$$

The simulation result of Figures 3.7 and 3.8 shows when the two rotor speeds vary ($\hat{\Omega}_2$ and $\hat{\Omega}_4$), it affects translational dynamics state output (*Z, Y*) and rotational dynamics (Φ). The input $\tilde{U}_2$ affects the theta dynamics or state which is called roll dynamics.

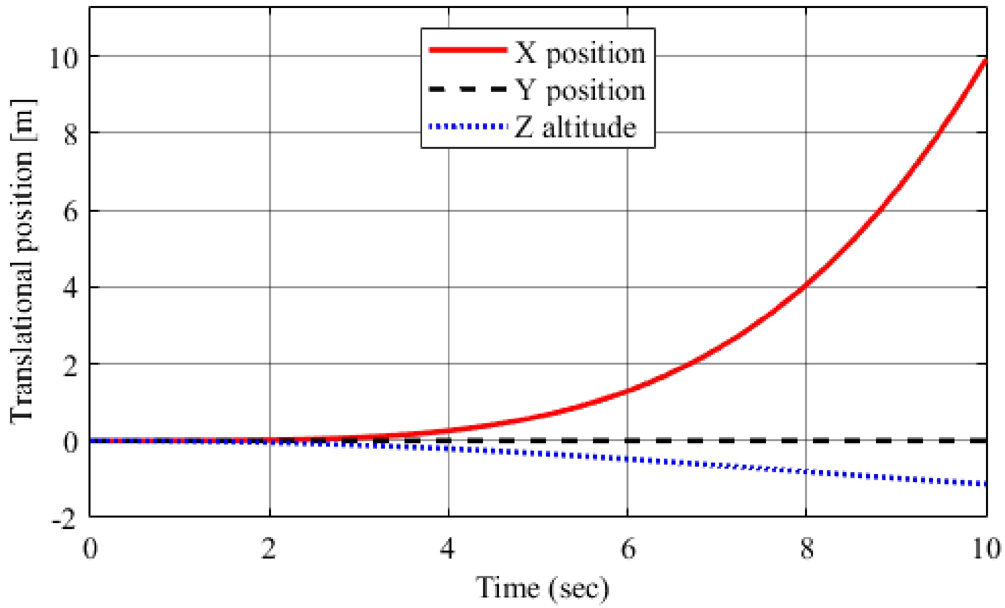


Figure 3.9 Response of nonlinear translational position to pitch input at

$$\hat{\Omega}_1 = 202.9,\ \ \hat{\Omega}_2 = \hat{\Omega}_4 = 203,\ \hat{\Omega}_3 = 203.1\ (rad/\sec)$$

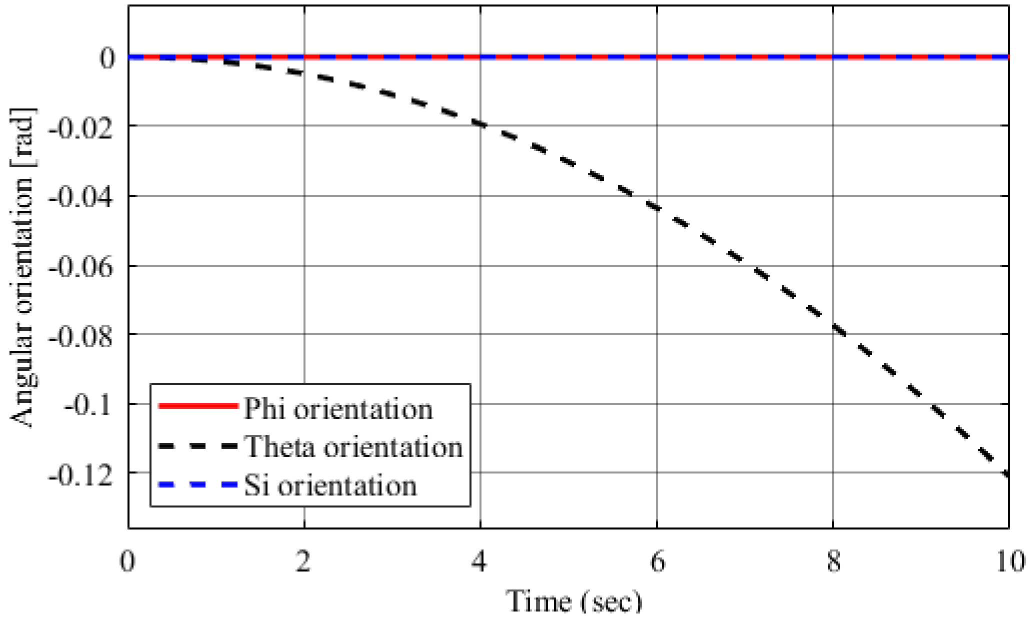


Figure 3.10 Response of nonlinear angular orientation to pitch input at

$$\hat{\Omega}_1 = 202.9, \ \ \hat{\Omega}_2 = \hat{\Omega}_4 = 203, \ \hat{\Omega}_3 = 203.1 \ (rad/\sec)$$

The simulation result of Figures 3.9 and 3.10 shows when the two rotor speeds vary ($\hat{\Omega}_1$ and $\hat{\Omega}_2$), it affects translational dynamics state output (*X*, *Y* and *Z*) and rotational dynamics (Θ). The input $\tilde{U}_3$ affects the theta dynamics or state which is called roll dynamics.

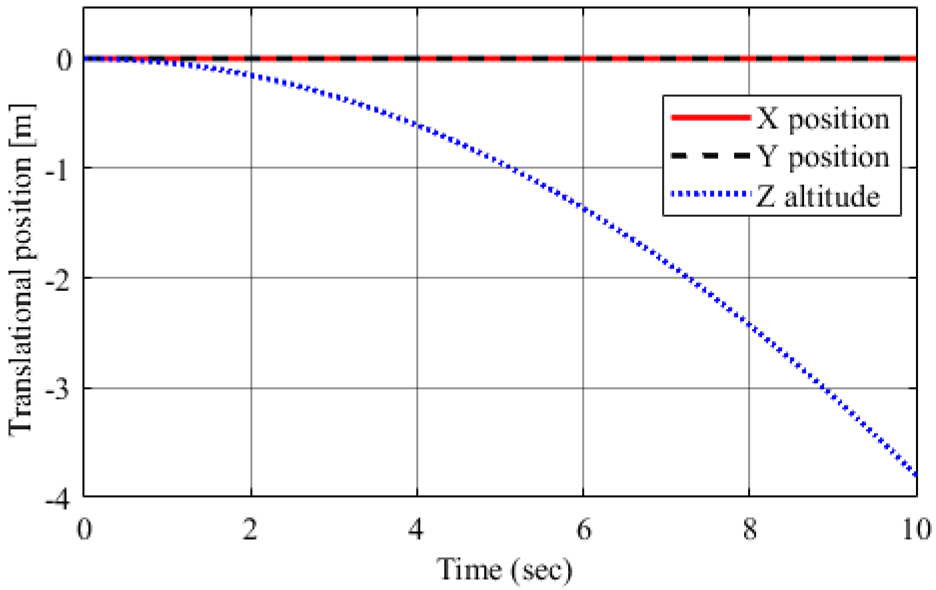


Figure 3.11 translational position to yaw input at $\hat{\Omega}_1 = \hat{\Omega}_3 = 203, \ \ \hat{\Omega}_2 = \hat{\Omega}_4 = 204(rad/\sec)$

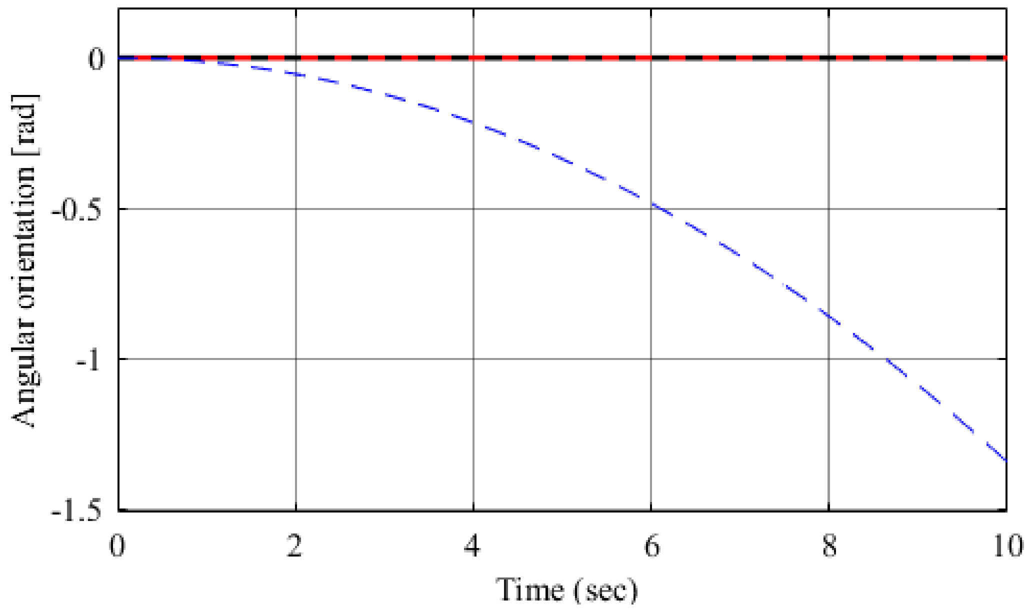


Figure 3.12 Angular orientation to yaw input at $\hat{\Omega}_1 = \hat{\Omega}_3 = 203,\ \hat{\Omega}_2 = \hat{\Omega}_4 = 204\left(rad/\sec\right)$

The simulation result of Figures 3.11 and 3.12 shows that the yaw input affects the Ψ and Z dynamics. This implies $\tilde{U}_4$ affects the angular position psi state called yaw dynamics.

CHAPTER FOUR

4. CONTROLLER DESIGN

4.1. Trajectory Generation Using Minimum Jerk Trajectory

When something needs to be smoothly moved from one starting position to another starting position, the path generator is necessary for the control system. There are different types of trajectories generated using different methods. Minimum jerk trajectory generation is one way of generating a path to be trackd. Jerk is the time derivative of acceleration, which is an important factor in suppressing vibration and achieving high accuracy.

If the position of the system is specified by the variable $x(t)$, then the Jerk of the system is [26]:

$$\text{Jerk } \dddot{x}(t) = \frac{d^3x(t)}{dt^3} \tag{4.1}$$

The trajectory of the minimum jerk is based on the minimization of the sum of the squares of the jerk of the movement along its trajectory or from one point to another. The purpose of the minimum time path planning is to plan the path with the shortest execution time based on a given point when the boundary constraints are met.

The minimum jerk trajectory is used in this research work to generate a minimum trajectory in the X, Y and Z directions to allow the quadcopter to move from one waypoints to another with minimized vibration [27]. The application of the quadcopter for this research work is to perform any targeted task that may require the quadcopter to move from some starting point to the endpoint by calculating its trajectory through a given different waypoints. Therefore, the designed quadcopter for;

- First responder: can be sent into buildings as a first responder to look for biochemical and gaseous leaks.
- Constructions: can carrying and assemble construction materials.
- Transportation: can carry cargo and home delivery services.
- Search and rescue: can be sent into collapsed buildings to assist the damage of disaster and can be sent into the reactive building to map radiation levels and so on.

4.1.1. Minimum Jerk Trajectory for X Position

Now, let's design a minimum jerk trajectory $x(t)$ such that it has a starting position $x(0)=a$ and its end position $x(T)=b$.

The functional is the integral of the square of the jerk as shown below [L3];

$$\begin{aligned} x^*(t) &= \arg\min_{x(t)} \int_0^T L(\dddot{x}, \ddot{x}, \dot{x}, x, t)\,dt \\ &= \arg\min_{x(t)} \int_0^T \dddot{x}^2 dt \end{aligned} \tag{4.2}$$

The aim is to find the best of *x* that minimizes this function. By using the Euler-Lagrange equation;

$$\frac{\partial L}{\partial x} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) + \frac{d^2}{dt^2}\left(\frac{\partial L}{\partial \ddot{x}}\right) - \frac{d^3}{dt^3}\left(\frac{\partial L}{\partial \dddot{x}}\right) = 0 \tag{4.3}$$

$\frac{\partial L}{\partial x}, \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right), \frac{d^2}{dt^2}\left(\frac{\partial L}{\partial \ddot{x}}\right)$, are becomes zero as they don't explicitly depend on $\dddot{x}$. What remains will be simplified into six order differential equations.

$$(x)^6 = 0 \tag{4.4}$$

Which then can be solved analytically and the result would be the fifth order of polynomials in time.

$$x = c_5 t^5 + c_4 t^4 + c_3 t^3 + c_2 t^2 + c_1 t^1 + c_0 \tag{4.5}$$

This fifth-order polynomial has to be solved to find the six coefficients. To solve this additional boundaries needed to be specified.

Table 4.1 Initial and final position, velocity and acceleration

Time	Position	Velocity	Acceleration
t=0	a	0	0
t=T	b	0	0

By assuming the initial position $x(0)=a$, the final position $x(T)=b$, the initial velocity $\dot{x}(0)=0$, the final velocity $\dot{x}(T)=0$, the initial acceleration $\ddot{x}(0)=0$ and the final acceleration $x(T)=0$ let's write each of these boundary conditions in terms of six unknown coefficients and six known boundaries exclusively.

$$
\begin{aligned}
x(t=0) &= c_5 t_0{}^5 + c_4 t_0{}^4 + c_3 t_0{}^3 + c_2 t_0{}^2 + c_1 t_0{}^1 + c_0 t_0{}^0 \\
x(t=T) &= c_5 T_T{}^5 + c_4 T_T{}^4 + c_3 T_T{}^3 + c_2 T_T{}^2 + c_1 T_T{}^1 + c_0 t_T{}^0 \\
\dot{x}(t=0) &= 5c_5 t_0{}^4 + 4c_4 t_0{}^3 + 3c_3 t_0{}^2 + 2c_2 t_0{}^1 + c_1 t_0{}^0 + 0 \\
\dot{x}(t=T) &= 5c_5 T_T{}^4 + 4c_4 T_T{}^3 + 3c_3 T_T{}^2 + 2c_2 T_T{}^1 + c_1 T_T{}^0 + 0 \\
\ddot{x}(t=0) &= 20c_5 t_0{}^3 + 12c_4 t_0{}^2 + 6c_3 t_0{}^1 + 2c_2 t_0{}^0 + 0 + 0 \\
\ddot{x}(t=T) &= 20c_5 T_T{}^3 + 12c_4 T_T{}^2 + 6c_3 T_T{}^1 + 2c_2 T_T{}^0 + 0 + 0
\end{aligned}
\tag{4.6}
$$

Then in matrix form;

$$
\begin{bmatrix} a \\ b \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
=
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 1 \\
T^5 & T^4 & T^3 & T^2 & T & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
5T^4 & 4T^3 & 3T^2 & 2T & 1 & 0 \\
0 & 0 & 0 & 2 & 0 & 0 \\
20T^3 & 12T^2 & 6T & 2 & 0 & 0
\end{bmatrix}
\begin{bmatrix} c_5 \\ c_4 \\ c_3 \\ c_2 \\ c_1 \\ c_0 \end{bmatrix}
\tag{4.7}
$$

Let,
$$
A=\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 1 \\
T^5 & T^4 & T^3 & T^2 & T & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
5T^4 & 4T^3 & 3T^2 & 2T & 1 & 0 \\
0 & 0 & 0 & 2 & 0 & 0 \\
20T^3 & 12T^2 & 6T & 2 & 0 & 0
\end{bmatrix},\quad
B=\begin{bmatrix} a \\ b \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},\quad
C=\begin{bmatrix} c_5 \\ c_4 \\ c_3 \\ c_2 \\ c_1 \\ c_0 \end{bmatrix}
$$

The coefficients can be solved using Equation (4.8).

$$C = A^{-1}B \tag{4.8}$$

The same procedures are used for the other state variables to generate a minimum jerk trajectory.

4.2. Sliding Mode Controller

Sliding Mode Control (SMC) is a reliable controller for handling sudden and large changes in system dynamics. It has emerged as one of the useful tools for designing controllers for systems with uncertainties. The sliding mode uses a discontinuous feedback control law to force the state of the system to a predefined sliding surface, which is called a sliding surface or a switching surface. To achieve the goal of sliding mode control, the design of the system must be able to overcome the uncertainties involved. The design process of the sliding mode controller can be explained as follows considering the uncertain linear time-invariant (LTI) system [28].

$$\dot{x} = A(x) + Bu(t) + D\rho(t,x) \tag{4.9}$$

where $A \in R^{n\times n}$ and $B \in R^{n\times m}$.

Assumption 1: (A, B) pair is controllable.

Assumption 2: $C \in R^{n\times l}$ is known and lies in the range space of the input distribution matrix *B*.

Therefore, it is possible to write $D = BI$ for some $I \in R^{n\times l}$. The function $\rho(t,x)$ is an external disturbance with an upper bound for all *x* and *t*. The uncertainty in the system can be considered as a matched uncertainty as it is acting as the channel of the input distribution matrix. Therefore, the uncertain system can be written as;

$$\dot{x} = A(x) + Bu(t) + BI\rho(t,x) \tag{4.10}$$

The sliding surface can be defined as $P = \{X = R^n : s(t) = 0\}$ $s(t)$ is a linear switching function which can be defined as $s(t) = Gx(t)$

where $G = R^{m\times m}$, is the design matrix.

It is assumed that the square matrix GB is non-singular i.e. $\det(GB) \neq 0$. The time derivative of the switching function is considered to analyze the sliding mode associated with the sliding surface.

$$\dot{s}(t) = G\dot{x}(t) \tag{4.11}$$

Then,

$$\dot{s}(t) = G\left(A(x) + Bu(t) + C\rho(t,x)\right) \tag{4.12}$$

Therefore, for all time $t > t_s$, it is assumed that all the states are forced to reach the sliding surface and an ideal sliding motion can be obtained. For all $t > t_s$

$$s(t) = \dot{s}(t) = 0 \tag{4.13}$$

By equating the time derivative to $0, U_{Eq}(t)$ is obtained. For all $t > t_s$

$$U_{Eq}(t) = -(GB)^{-1}\left(GA(x) + GBI\rho(t,x)\right) \tag{4.13}$$

GB is non-singular. $U_{Eq}(t)$ is an equivalent control with an average value that the control signal should maintain to ensure sliding on the sliding surface [29].

4.2.1. Sliding Mode Control Law

After the sliding surface, the next task is to define a control law such the sliding motion on the surface is guaranteed. A sliding mode controller consists of a linear and a non-linear part.

$$U(t) = U_L(t) + U_N(t) \tag{4.14}$$

The non-linear, $U_N(t)$ part is responsible for generating a sliding mode on the surface and the linear Part, $U_L(t)$ for helping to maintain sliding [29].

Reachability: The input $u(t)$ in sliding mode control is designed in such a way that the reachability condition is satisfied. Reachability condition is the approach of states towards $s(t) = 0$. The following condition is satisfied to reach the sliding mode.

$$\lim_{s(t)\to 0+} \dot{s}(t) < 0 \, and \lim_{s(t)\to 0-} \dot{s}(t) > 0 \tag{4.15}$$

The equation can be explained by the process by which the derivative of $s(t)=0$ is less than 0, $s(t)$ tends to the positive side and when $\dot{s}(t)$ is greater than 0, $s(t)$ tends to the negative side. For a MIMO version of the reachability condition is:

$$s^T(t)\dot{s}(t) < -\eta \|s(t)\| \tag{4.16}$$

This condition shows that the sliding condition is attractive [30].

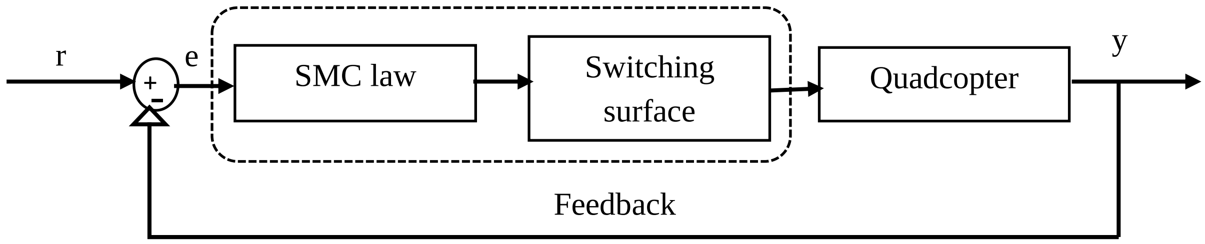


Figure 4.1 Block Diagram of sliding mode controller for quadcopter

4.3. Adaptive Controller

In some control tasks, such as manipulating robots, there are uncertainties in the parameters of the controlled system at the beginning of the control operation. Unless the uncertainty of the parameter is gradually reduced online through estimation or adaptive mechanisms, it may lead to inaccuracy or instability of the control system. The basic goal of adaptive control is to maintain constant system performance in the presence of uncertain or unknown changes in equipment parameters. Since this uncertainty or parameter change occurs in many practical problems, adaptive control is useful in many industrial settings.

The difference between the adaptive controller and the ordinary controller is that the controller parameters are variable and there is a mechanism to adjust these parameters online according to the system signals. There are two main ways to build an adaptive controller. One is the model reference adaptive control method and the other is the auto-tuning method [30].

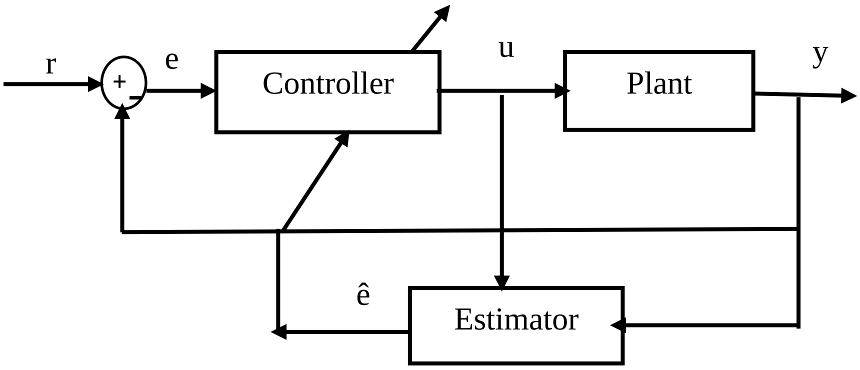


Figure 4.2 Adaptive control

The controller obtained by coupling the controller with an online (recursive) parameter estimator is called a self-tuning controller. Therefore, the self-tuning controller is a controller that recognizes unknown parameters at the same time. At each moment, the estimator sends a set of estimated plant parameters to the controller. It is calculated based on the input u and output y of the upper layer; the computer finds the corresponding controller parameters, and then calculates the control input u based on the controller parameters and measurement signals; this control input u leads to a new plant output. And repeat the entire cycle of input and parameter update. Please note that the controller parameters are calculated based on plant parameter estimates as if they were real plant parameters [31].

4.4. Adaptive Sliding Mode Controller

Consider a nonlinear affine control system,

$$\dot{x} = f\left(x\right) + g\left(x\right)u + \beta \tag{4.17}$$

Where $x \in R^n$ is the state vector, $u \in R^m$ is the control vector of the system, $m = n$, and $\beta \in R^n$, is the unknown vector describing external disturbance or modelling error, and changes very slowly $\dot{\beta} \approx 0$. Assuming that $f\left(x\right)$ and $g\left(x\right)$ is continuously differentiable [32].

Consider the following sliding surface,

$$\sigma = x - x_r \tag{4.18}$$

Where, $x - x_r$ is the error state vector about the desired state vector x_r. The first derivative of the sliding surface can be written as:

$$\dot{\sigma} = \dot{x} - \dot{x}_r = f(x) + g(x)u + \beta - \dot{x}_r \tag{4.19}$$

To let the $\dot{\sigma} = 0$, let us propose the following control input,

$$u = g^{-1}(x)\left(-K\sigma - f(x) - \hat{\beta}\right) \tag{4.20}$$

Where K is a positive definite diagonal matrix and $\hat{\beta}$ is online estimated vector. Substitute the proposed control input vector to the first derivative of the sliding surface, having the following form,

$$\dot{\sigma} = -K\sigma + \beta - \dot{x}_r - \hat{\beta} = -K\sigma + \tilde{\beta} \tag{4.21}$$

Where, $\hat{\beta}$ is the online estimated vector. In the above equations, assume the $\tilde{\beta} = \beta - \dot{x}_r - \hat{\beta}$, then construct the adaptive dynamic vector as follows:

$$\dot{\tilde{\beta}} = \dot{\beta} - \ddot{x}_r - \dot{\hat{\beta}} = -\lambda\sigma \tag{4.22}$$

Where λ is the positive definite diagonal gain matrix. When the external disturbance or un modelling error changes very slowly $\dot{\beta} \approx 0, \ddot{x}_r \approx 0$, we have $\dot{\hat{\beta}} = \lambda\sigma$.

Theorem 4.1: If the appropriate adaptive law Equation (4.22) is used with the recommended control input Equation (4.20), then the sliding surface affine nonlinear system described in Equation (4.22) will converge to zero over time [32].

Proof: Consider the following Lyapunov candidate function

$$V = \frac{1}{2}\sigma^T\sigma + \frac{1}{2}\tilde{\beta}^T\lambda^{-1}\tilde{\beta} \tag{4.23}$$

The first derivative of the Lyapunov candidate function can be represented as

$$\dot{V} = \sigma^T\dot{\sigma} + \tilde{\beta}^T\lambda^{-1}\dot{\tilde{\beta}} \tag{4.24}$$

Expanding the above equation,

$$\begin{aligned} \dot{V} &= \sigma^T \sigma + \tilde{\beta}^T \lambda^{-1} \dot{\tilde{\beta}} \\ &= \sigma^T \left(-a\sigma + \tilde{\beta}\right) + \tilde{\beta}^T \lambda^{-1} \left(-\lambda\sigma\right) \\ &= -\sigma^T a\sigma < 0 \end{aligned} \tag{4.25}$$

So, it can be stated that the sliding surface concerning the error states converges to zero as time increases.

4.5. Application of Sliding Mode Controller for Quadcopter

X and Y states are not controlled directly as they don't have their control inputs like the other state variables because the system is underactuated system. So, a mechanism for controlling this two-position state variable should be derived using an indirect way of controlling them in terms of Φ_d, and Θ_d [33]. With linearization, the trigonometric relation between angles and the translational velocity is lost. Since the design will be based on the control model, this nonlinearity can be introduced into controller with an inverse relation shown in Equation (4.31),

The derivation of the actual control actions (Φ, Θ) from the obtained (α_x, α_x) is performed as follows:

$$\begin{aligned} \Theta_d &= \arcsin\left(\frac{\alpha_x}{g}\right) \\ \Phi_d &= -\arcsin\left(\frac{\alpha_y}{g}\right) \end{aligned} \tag{4.26}$$

A proportional-integral-derivative (PID) sliding surface is developed for determining the control law $\tilde{U}$ for Z, Φ, Θ, and Ψ and a proportional-integral (PI) sliding surface is developed for X, Y.

For Z, Φ, Θ, and Ψ the sliding surface is;

$$\sigma(t) = \mathrm{K}_P e(t) + \mathrm{K}_I \int_0^t e(t)\, dt + \mathrm{K}_D \dot{e}(t) \tag{4.27}$$

And for X, Y the sliding surface is;

$$\sigma(t) = \mathrm{K}_P e(t) + \mathrm{K}_I \int_0^t e(t)dt \tag{4.28}$$

where $\mathrm{K}_P, \mathrm{K}_I, \mathrm{K}_D \in R^+$ are controller gains.

The derivative term of the sliding surfaces is shown as follows:

$$\dot{\sigma}(t) = \mathrm{K}_P \dot{e}(t) + \mathrm{K}_I e(t) + \mathrm{K}_D \ddot{e}(t) \tag{4.29}$$

$$\dot{\sigma}(t) = \mathrm{K}_P e(t) + \mathrm{K}_I e(t)$$

The SMC is designed based on exponential reaching law.

4.5.1. Attitude Control Law

The attitude control law is formulated as follows to control the rotation of the quadcopter.

4.5.1.1. Roll Control

$$\sigma_\Phi = \mathrm{K}_{P\Phi}(\Phi_d - \Phi) + \mathrm{K}_{I\Phi}\int_0^t (\Phi_d - \Phi)\,dt + \mathrm{K}_{D\phi}(\dot{\Phi}_d - \dot{\Phi}) \tag{4.30}$$

$$\dot{\sigma}_\Phi = \mathrm{K}_{P\phi}(\dot{\Phi}_d - \dot{\Phi}) + \mathrm{K}_{I\Phi}(\Phi_d - \Phi) + \mathrm{K}_{D\Phi}(\ddot{\Phi}_d - \ddot{\Phi}) \tag{4.31}$$

$$= -\varepsilon_\Phi \operatorname{sgn} \sigma_\Phi - \eta_\Phi \sigma_\Phi \tag{4.32}$$

$$\dot{\sigma}_\Phi = \mathrm{K}_{P\Phi}(\dot{\Phi}_d - \dot{\Phi}) + \mathrm{K}_{I\Phi}(\Phi_d - \Phi) + \mathrm{K}_{D\Phi}\ddot{\Phi}_d - \mathrm{K}_{D\Phi}\left(\left(\left(\mathrm{I}_y - \mathrm{I}_z\right)\dot{\Psi}\dot{\Theta} + \tilde{U}_2 - \mathrm{I}_r\Theta\hat{\Omega}\right)\big/\mathrm{I}_x\right) \tag{4.33}$$

$$\sigma_\Phi\dot{\sigma}_\Phi = \sigma_\Phi\left(\mathrm{K}_{P\Phi}(\dot{\Phi}_d - \dot{\Phi}) + \mathrm{K}_{I\Phi}(\Phi_d - \Phi) + \mathrm{K}_{D\Phi}\ddot{\Phi} - \mathrm{K}_{D\Phi}\left(\frac{\left(\mathrm{I}_y - \mathrm{I}_z\right)\dot{\Psi}\dot{\Theta} + \tilde{U}_2 - \mathrm{I}_r\dot{\Theta}\hat{\Omega}}{\mathrm{I}_x}\right)\right) \tag{4.34}$$

Select $\tilde{U}_2$ to cancel out some known terms from $\dot{\sigma}_\Phi$.

$$\begin{aligned}\tilde{U}_2 = {} & \frac{\mathrm{I}_x}{\mathrm{K}_{D\Phi}}\mathrm{K}_{P\Phi}(\dot{\Phi}_d - \dot{\Phi}) + \frac{\mathrm{I}_x}{\mathrm{K}_{D\Phi}}\mathrm{K}_{I\Phi}(\Phi_d - \Phi) + \mathrm{I}_x\ddot{\Phi}_d - \dot{\Theta}\dot{\Psi}\left(\mathrm{I}_y - \mathrm{I}_z\right) \\ & + \mathrm{I}_r\hat{\Omega}\dot{\Theta} + \frac{\mathrm{I}_x}{K_{D\Phi}}\varepsilon_\Phi \operatorname{sgn}\sigma_\Phi + \frac{\mathrm{I}_x}{\mathrm{K}_{D\Phi}}\eta_\Phi\sigma_\Phi\end{aligned} \tag{4.35}$$

4.5.1.2. Pitch Control

$$\sigma_\Theta = \mathrm{K}_{P\Theta}\left(\Theta_d - \Theta\right) + \mathrm{K}_{I\Theta}\int_0^t \left(\Theta_d - \Theta\right)dt + \mathrm{K}_{D\Theta}\left(\dot{\Theta}_d - \dot{\Theta}\right) \tag{4.36}$$

$$\dot{\sigma}_\Theta = \mathrm{K}_{P\Theta}\left(\dot{\Theta}_d - \dot{\Theta}\right) + \mathrm{K}_{I\Theta}\left(\Theta_d - \Theta\right) + \mathrm{K}_{D\Theta}\left(\ddot{\Theta}_d - \ddot{\Theta}\right) \tag{4.37}$$

$$= -\varepsilon_\Theta \operatorname{sgn}\sigma_\Theta - \eta_\Theta \sigma_\Theta \tag{4.38}$$

$$\dot{\sigma}_\Theta = \mathrm{K}_{P\Theta}\left(\dot{\Theta}_d - \dot{\Theta}\right) + \mathrm{K}_{I\Theta}\left(\Theta_d - \Theta\right) + \mathrm{K}_{D\Theta}\ddot{\Theta}_d - \mathrm{K}_{D\Theta}\left(\left(\left(\mathrm{I}_z - \mathrm{I}_x\right)\dot{\Phi}\dot{\Psi} + \tilde{U}_3 - \mathrm{I}_r\dot{\Phi}\hat{\Omega}\right)\big/\mathrm{I}_y\right) \tag{4.39}$$

$$\begin{aligned}\sigma_\Theta\dot{\sigma}_\Theta = {} & \sigma_\Theta\left(\mathrm{K}_{P\Theta}\left(\dot{\Theta}_d - \dot{\Theta}\right) + \mathrm{K}_{I\Theta}\left(\Theta_d - \Theta\right) + \mathrm{K}_{D\Theta}\ddot{\Theta}_d\right) \\ & -\sigma_\Theta\left(\mathrm{K}_{D\Theta}\left(\left(\left(\mathrm{I}_z - \mathrm{I}_x\right)\dot{\Phi}\dot{\Psi} + \tilde{U}_3 - \mathrm{I}_r\dot{\Phi}\hat{\Omega}\right)\big/\mathrm{I}_y\right)\right)\end{aligned} \tag{4.40}$$

Select $\tilde{U}_3$ to cancel out some known terms from $\dot{\sigma}_\Theta$.

$$\begin{aligned}\tilde{U}_3 = {} & \frac{\mathrm{I}_y}{\mathrm{K}_{D\Theta}}\mathrm{K}_{P\Theta}\left(\dot{\Theta}_d - \dot{\Theta}\right) + \frac{\mathrm{I}_y}{\mathrm{K}_{D\Theta}}\mathrm{K}_{I\Theta}\left(\Theta_d - \Theta\right) + \mathrm{I}_y\ddot{\Theta}_d - \ddot{\Theta}\dot{\Psi}\left(\mathrm{I}_z - \mathrm{I}_x\right) \\ & -J_r\hat{\Omega}_r\dot{\Phi} + \frac{\mathrm{I}_y}{\mathrm{K}_{D\Theta}}\varepsilon_\Theta \operatorname{sgn}\sigma_\Theta + \frac{I_y}{K_{d\Theta}}\eta_\Theta\sigma_\Theta\end{aligned} \tag{4.41}$$

4.5.2. Heading Control

4.5.2.1. Yaw Control

$$\sigma_\Psi = \mathrm{K}_{P\Psi}\left(\Psi_d - \Psi\right) + \mathrm{K}_{I\Psi}\int_0^t \left(\Psi_d - \Psi\right)dt + \mathrm{K}_{D\Psi}\left(\ddot{\Psi}_d - \dot{\Psi}\right) \tag{4.42}$$

$$\dot{\sigma}_\Psi = \mathrm{K}_{P\Psi}\left(\dot{\Psi}_d - \dot{\Psi}\right) + \mathrm{K}_{I\Psi}\left(\Psi_d - \Psi\right) + \mathrm{K}_{D\Psi}\left(\ddot{\Psi}_d - \ddot{\Psi}\right) \tag{4.43}$$

$$= -\varepsilon_\Psi \operatorname{sgn}\sigma_\Psi - \eta_\Psi \sigma_\Psi \tag{4.44}$$

$$\dot{\sigma}_\Psi = \mathrm{K}_{P\Psi}\left(\dot{\Psi}_d - \dot{\Psi}\right) + \mathrm{K}_{I\Psi}\left(\Psi_d - \Psi\right) + \mathrm{K}_{D\Psi}\ddot{\Psi}_d - \mathrm{K}_{D\Psi}\left(\left(\left(\mathrm{I}_x - \mathrm{I}_y\right)\dot{\Phi}\dot{\Theta} + \tilde{U}_4\right)\big/\mathrm{I}_z\right) \tag{4.45}$$

$$\begin{aligned}\sigma_\Psi\dot{\sigma}_\Psi = {} & \sigma_\Psi\left(\mathrm{K}_{P\Psi}\left(\dot{\Psi}_d - \dot{\Psi}\right) + \mathrm{K}_{I\Psi}\left(\Psi_d - \Psi\right) + \mathrm{K}_{D\Psi}\ddot{\Psi}_d\right)\Psi \\ & -\sigma_\Psi\mathrm{K}_{D\Psi}\left(\left(\left(\mathrm{I}_z - \mathrm{I}_x\right)\dot{\Phi}\dot{\Theta} + \tilde{U}_4\right)\big/\mathrm{I}_z\right)\end{aligned} \tag{4.46}$$

Select $\tilde{U}_4$ to cancel out some known terms from $\dot{\sigma}_\Psi$

$$\tilde{U}_4 = \frac{\mathrm{I}_z}{\mathrm{K}_{D\Psi}} \mathrm{K}_{P\Psi}\left(\dot{\Psi}_d - \dot{\Psi}\right) + \frac{\mathrm{I}_z}{\mathrm{K}_{D\Psi}} \mathrm{K}_{I\Psi}\left(\Psi_d - \Psi\right) + \mathrm{I}_z \ddot{\Psi}_d - \dot{\Phi}\dot{\Theta}\left(\mathrm{I}_x - \mathrm{I}_y\right) + \frac{\mathrm{I}_z}{\mathrm{K}_{D\Psi}} \varepsilon_\Psi \operatorname{sgn} \sigma_\Psi + \frac{\mathrm{I}_z}{\mathrm{K}_{D\Psi}} \varepsilon_\Psi \operatorname{sgn} \sigma_\Psi \tag{4.47}$$

4.5.3. Altitude Control Law

4.5.3.1. Z Control

The motion of the quadcopter in the Z-axis is manipulated by the altitude control law as follows.

$$\sigma_Z = \mathrm{K}_{PZ}\left(Z_d - Z\right) + \mathrm{K}_{IZ}\int_0^t \left(Z_d - Z\right) dt + \mathrm{K}_{DZ}\left(\dot{Z}_d - \dot{Z}\right) \tag{4.48}$$

$$\dot{\sigma}_Z = \mathrm{K}_{PZ}\left(\dot{Z}_d - \dot{Z}\right) + \mathrm{K}_{IZ}\left(Z_d - Z\right) + \mathrm{K}_{DZ}\left(\ddot{Z}_d - \ddot{Z}\right) \tag{4.49}$$

$$\dot{\sigma}_Z = \mathrm{K}_{PZ}\left(\dot{Z}_d - \dot{Z}\right) + \mathrm{K}_{IZ}\left(Z_d - Z\right) + \mathrm{K}_{DZ}\ddot{Z}_d - \mathrm{K}_{DZ}\left(g - \frac{\tilde{U}_1}{m}\left(\cos(\phi)\cos(\theta)\right)\right) \tag{4.50}$$

$$= -\varepsilon_Z \operatorname{sgn} \sigma_Z - \eta_Z \sigma_Z \tag{4.51}$$

$$\sigma_Z \dot{\sigma}_Z = \sigma_Z\left(\mathrm{K}_{PZ}\left(\dot{Z}_d - \dot{Z}\right) + \mathrm{K}_{IZ}\left(Z_d - Z\right) + \mathrm{K}_{DZ}\ddot{Z}_d - \mathrm{K}_{DZ}\left(g - \frac{\tilde{U}_1}{m}\left(\cos(\phi)\cos(\theta)\right)\right)\right) \tag{4.52}$$

Select $\tilde{U}_1$ to cancel out some known terms from $\dot{s}_Z$

$$\tilde{U}_1 = \frac{mg}{\cos(\phi)\cos(\theta)} - \frac{m\mathrm{K}_{PZ}\left(\dot{Z}_d - \dot{Z}\right)}{\mathrm{K}_{DZ}\cos(\phi)\cos(\theta)} - \frac{m\mathrm{K}_{IZ}\left(Z_d - Z\right)}{\mathrm{K}_{DZ}\cos(\phi)\cos(\theta)} - \frac{m\ddot{Z}_d}{\cos(\phi)\cos(\theta)} - \frac{m\varepsilon_Z \operatorname{sgn} \sigma_Z}{\mathrm{K}_{DZ}\cos(\phi)\cos(\theta)} - \frac{m\eta_Z \sigma_Z}{\mathrm{K}_{DZ}\cos(\phi)\cos(\theta)} \tag{4.53}$$

4.5.4. Position Control Law

4.5.4.1. X Control Law

The motion of the quadcopter in the X-axis is manipulated by the position control law as follows.

$$\sigma_X = K_{PX}\left(X_d - X\right) + K_{IX}\int_0^t \left(X_d - X\right)dt \tag{4.54}$$

$$\dot{\sigma}_X = K_{PX}\left(\dot{X}_d - \dot{X}\right) + K_{IX}\left(X_d - X\right) \tag{4.55}$$

4.5.4.2. Y Control Law

The motion of the quadcopter in the Y-axis is manipulated by the position control law as follows.

$$\sigma_Y = K_{PY}\left(Y_d - Y\right) + K_{IY}\int_0^t \left(Y_d - Y\right)dt \tag{4.56}$$

$$\dot{\sigma}_Y = K_{PY}\left(\dot{Y}_d - \dot{Y}\right) + K_{IY}\left(Y_d - Y\right) \tag{4.57}$$

4.6. Application of Adaptive Sliding Mode Controller for Quadcopter

The ASMC technique used for this research work which is shown in Figure 4.3 can be described as follows;

In order to control the system a desired trajectory which is used as a reference for the system to be controlled should be designed for all six state variables of the system. Then, the desired values are given to the designed ASMC and then the output of the controller is given to the quadcopter system as inputs. Finally, the six output variables can follow the desired trajectory which is designed before the controller. Therefore, the controller can control the system.

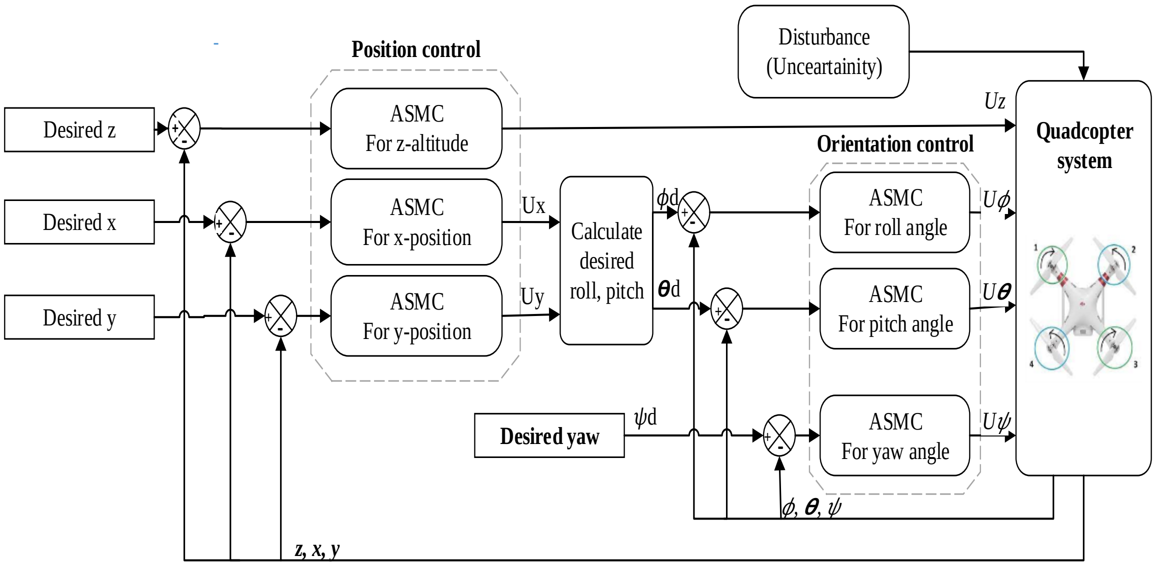


Figure 4.3 Block diagram of Adaptive sliding mode controller for quadcopter

4.6.1. Attitude Control Law

4.6.1.1. Roll Control

From the dynamic model of quadcopter system of Equation (3.46) ;

$$\ddot{\Phi} = \frac{\mathrm{I}_y - \mathrm{I}_z}{\mathrm{I}_x}\dot{\Theta}\dot{\Psi} + \frac{\tilde{U}_2}{\mathrm{I}_x} - \frac{\mathrm{I}_r}{\mathrm{I}_x}\dot{\Theta}\hat{\Omega} + \beta$$

And from Equation (4.30);

$$\sigma_\Phi = \mathrm{K}_{P\Phi} e_\Phi(t) + \mathrm{K}_{I\Phi}\int_0^t e_\Phi(t)\,dt + \mathrm{K}_{D\Phi}\dot{e}_\Phi(t)$$

Using the adaptive input controller in Equation (4.20);

$$u = g^{-1}(x)\left(-as - f(x) - \hat{\beta}\right)$$

$$\tilde{U}_2 = \mathrm{I}_x\left(-a\sigma_\Phi - \frac{\dot{\Theta}\dot{\Psi}\left(\mathrm{I}_y - \mathrm{I}_z\right) - \mathrm{I}_r\dot{\Theta}\hat{\Omega}}{\mathrm{I}_x} - \hat{\beta}\right) \quad (4.58)$$

Then,

$$\ddot{\Phi} = \frac{I_y - I_z}{I_x}\dot{\Theta}\dot{\Psi} + \frac{I_{xx}\left(-a\sigma_\Phi - \frac{\dot{\Theta}\dot{\Psi}\left(I_y - I_z\right) - I_r\dot{\Theta}\hat{\Omega}}{I_x} - \hat{\beta}\right)}{I_{xx}} - \frac{I_r}{I_x}\dot{\Theta}\hat{\Omega} + \beta$$
$$= -a\sigma_\Phi + \left(\beta - \hat{\beta}\right) \tag{4.59}$$

Let $\left(\beta - \hat{\beta}\right) = \tilde{\beta}$

$$\ddot{\Phi} = -a\sigma_\Phi + \tilde{\beta} \tag{4.60}$$

From Equation(4.60)

$$\tilde{\beta} = \ddot{\Phi} + a\sigma_\Phi \tag{4.61}$$

And its derivative is;

$$\dot{\tilde{\beta}} = \dddot{\Phi} + a\dot{\sigma}_\Phi \tag{4.62}$$

Then finally,

$$\dot{\tilde{\beta}} = -\lambda\sigma_\Phi \tag{4.63}$$

Theorem 4.1: is used to show the asymptotical stability of the sliding surface to show the Lyapunov stability law for the adaptive sliding mode controller obtained.

Consider the following Lyapunov candidate function

$$V = \frac{1}{2}{\sigma_\Phi}^T\sigma_\Phi + \frac{1}{2}\tilde{\beta}^T\lambda^{-1}\tilde{\beta} \tag{4.64}$$

The first derivative of the Lyapunov candidate function can be represented as

$$\dot{V} = {\sigma_\Phi}^T\dot{\sigma}_\Phi + \tilde{\beta}^T\lambda^{-1}\dot{\tilde{\beta}} \tag{4.65}$$

Expanding the above equation,

$$\begin{aligned}\dot{V} &= {\sigma_\Phi}^T\dot{\sigma}_\Phi + \tilde{\beta}^T\lambda^{-1}\dot{\tilde{\beta}} \\ &= {\sigma_\Phi}^T\left(-a\sigma_\Phi + \tilde{\beta}\right) + \tilde{\beta}^T\lambda^{-1}\left(-\lambda\sigma_\Phi\right) \\ &= -{\sigma_\Phi}^T a s_\Phi < 0\end{aligned} \tag{4.66}$$

So it can be stated that the sliding surface concerning the error states converges to zero as time increases.

4.6.1.2. Pitch Control

From the dynamic model of quadcopter system of Equation(3.46) ;

$$\ddot{\Theta} = \frac{\mathrm{I}_z - \mathrm{I}_x}{\mathrm{I}_y}\dot{\Phi}\dot{\Psi} + \frac{\tilde{U}_3}{\mathrm{I}_y} + \frac{\mathrm{I}_r}{\mathrm{I}_y}\dot{\Phi}\hat{\Omega} + \beta$$

And from Equation (4.30);

$$\sigma_\Theta = \mathrm{K}_{P\Theta} e_\Theta(t) + \mathrm{K}_{I\Theta}\int_0^t e_\Theta(t)\,dt + \mathrm{K}_{D\Theta}\dot{e}_\Theta(t)$$

Using the adaptive input controller in Equation (4.20);

$$u = g^{-1}(x)\left(-as - f(x) - \hat{\beta}\right)$$

$$\tilde{U}_3 = \mathrm{I}_y\left(-a\sigma_\Theta - \frac{\dot{\Phi}\dot{\Psi}\left(\mathrm{I}_z - \mathrm{I}_x\right) + \mathrm{I}_r\dot{\Phi}\hat{\Omega}}{\mathrm{I}_y} - \hat{\beta}\right) \tag{4.67}$$

Then,

$$\begin{aligned}\ddot{\Theta} &= \frac{\mathrm{I}_{zz} - \mathrm{I}_{xx}}{\mathrm{I}_{yy}}\dot{\Phi}\dot{\Psi} + \frac{\mathrm{I}_{yy}\left(-a\sigma_\Phi - \frac{\dot{\Theta}\dot{\Psi}\left(\mathrm{I}_z - \mathrm{I}_x\right) - \mathrm{I}_r\dot{\Phi}\hat{\Omega}}{\mathrm{I}_y} - \hat{\beta}\right)}{I_y} - \frac{\mathrm{I}_r}{\mathrm{I}_{yy}}\dot{\Phi}\hat{\Omega} + \beta \\ &= -a\sigma_\Phi + \left(\beta - \hat{\beta}\right)\end{aligned} \tag{4.68}$$

Let $\left(\beta - \hat{\beta}\right) = \tilde{\beta}$

$$\ddot{\Theta} = -as_\Theta + \tilde{\beta} \tag{4.69}$$

From Equation (4.69)

$$\tilde{\beta} = \ddot{\Theta} + a\sigma_\Theta \tag{4.70}$$

And its derivative is;

$$\dot{\tilde{\beta}} = \dddot{\Theta} + a\dot{\sigma}_{\Theta} \tag{4.71}$$

Then finally,

$$\dot{\tilde{\beta}} = -\lambda\sigma_{\Theta} \tag{4.72}$$

Theorem 4.1: is used to show the asymptotical stability of the sliding surface to show the Lyapunov stability law for the adaptive sliding mode controller obtained.

Consider the following Lyapunov candidate function;

$$V = \frac{1}{2}s_{\Theta}{}^{T}s_{\Theta} + \frac{1}{2}\tilde{\Delta}^{T}\lambda^{-1}\tilde{\Delta}$$

The first derivative of the Lyapunov candidate function can be represented as;

$$\dot{V} = \sigma_{\Theta}{}^{T}\dot{\sigma}_{\Theta} + \tilde{\beta}^{T}\lambda^{-1}\dot{\tilde{\beta}} \tag{4.73}$$

Expanding the above equation,

$$\begin{aligned} \dot{V} &= \sigma_{\Theta}{}^{T}\dot{\sigma}_{\Theta} + \tilde{\beta}^{T}\lambda^{-1}\dot{\tilde{\beta}} \\ &= \sigma_{\Theta}{}^{T}\left(-a\sigma_{\Theta} + \tilde{\beta}\right) + \tilde{\beta}^{T}\lambda^{-1}\left(-\lambda\sigma_{\Theta}\right) \\ &= -\sigma_{\Theta}{}^{T}a\sigma_{\Theta} < 0 \end{aligned} \tag{4.74}$$

So it can be stated that the sliding surface concerning the error states converges to zero as time increases.

4.6.2. Heading Control Law

4.6.2.1. Yaw Control

From the dynamic model of quadcopter system of Equation (3.46) ;

$$\ddot{\Psi} = \frac{\mathrm{I}_x - \mathrm{I}_y}{\mathrm{I}_z}\dot{\Phi}\dot{\Theta} + \frac{\tilde{U}_4}{\mathrm{I}_z} + \beta$$

And from Equation (4.30);

$$\sigma_{\Psi} = \mathrm{K}_{P\Psi}e_{\Psi}(t) + \mathrm{K}_{I\Psi}\int_0^t e_{\Psi}(t)\,dt + \mathrm{K}_{D\Psi}\dot{e}_{\Psi}(t)$$

Using the adaptive input controller in Equation (4.20);

$$u = g^{-1}(x)\left(-a\sigma - f(x) - \hat{\beta}\right)$$

$$\tilde{U}_4 = \mathrm{I}_z\left(-a\sigma_\Psi - \frac{\dot{\Phi}\dot{\Psi}\left(\mathrm{I}_x - \mathrm{I}_y\right)}{\mathrm{I}_z} - \hat{\beta}\right) \tag{4.75}$$

Then,

$$\begin{aligned}\ddot{\Psi} &= \frac{\mathrm{I}_x - \mathrm{I}_y}{\mathrm{I}_z}\dot{\Phi}\dot{\Theta} + \frac{\mathrm{I}_y\left(-a\sigma_\Psi - \frac{\dot{\Theta}\dot{\Phi}\left(\mathrm{I}_x - \mathrm{I}_y\right)}{\mathrm{I}_z} - \hat{\beta}\right)}{\mathrm{I}_z} + \beta \\ &= -a\sigma_\Psi + \left(\beta - \hat{\beta}\right)\end{aligned} \tag{4.76}$$

Let $\left(\beta - \hat{\beta}\right) = \tilde{\beta}$

$$\ddot{\Psi} = -a\sigma_\Psi + \beta \tag{4.77}$$

From Equation (4.77);

$$\tilde{\beta} = \ddot{\Psi} + a\sigma_\Psi \tag{4.78}$$

And its derivative is;

$$\dot{\tilde{\beta}} = \dddot{\Psi} + a\dot{\sigma}_\Psi \tag{4.79}$$

Then finally,

$$\dot{\tilde{\beta}} = -\lambda\sigma_\Psi \tag{4.80}$$

Theorem 4.1: is used to show the asymptotical stability of the sliding surface to show the Lyapunov stability law for the adaptive sliding mode controller obtained.

Consider the following Lyapunov candidate function

$$V = \frac{1}{2}\sigma_\Psi{}^T\sigma_\Psi + \frac{1}{2}\tilde{\beta}^T\lambda^{-1}\tilde{\beta} \tag{4.81}$$

The first derivative of the Lyapunov candidate function can be represented as

$$\dot{V} = \sigma_\Psi{}^T \dot{\sigma}_\Psi + \tilde{\beta}^T \lambda^{-1} \dot{\tilde{\beta}} \tag{4.82}$$

Expanding the above equation,

$$\begin{aligned} \dot{V} &= \sigma_\Psi{}^T \dot{\sigma}_\Psi + \tilde{\beta}^T \lambda^{-1} \dot{\tilde{\beta}} \\ &= \sigma_\Psi{}^T \left(-a\sigma_\Psi + \tilde{\beta}\right) + \tilde{\beta}^T \lambda^{-1} \left(-\lambda \sigma_\Psi\right) \\ &= -\sigma_\Psi{}^T a \sigma_\Psi < 0 \end{aligned} \tag{4.83}$$

So it can be stated that the sliding surface concerning the error states converges to zero as time increases.

4.6.3. Altitude Control Law

4.6.3.1. Z Control

From the dynamic model of quadcopter system of Equation (3.40);

$$\ddot{Z} = g - \frac{\tilde{U}_1}{m}\left(\cos\Phi\cos\Theta\right) + \beta$$

And from Equation(4.30);

$$\sigma_Z = \mathrm{K}_{PZ} e_Z(t) + \mathrm{K}_{IZ}\int_0^t e_Z(t)\,dt + \mathrm{K}_{DZ}\dot{e}_Z(t)$$

Using the adaptive input controller in Equation (4.20);

$$u = g^{-1}(x)\left(-as - f(x) - \hat{\beta}\right)$$

$$\tilde{U}_1 = \frac{-m}{\cos\Phi\cos\Theta}\left(-ks_Z - g - \hat{\Delta}\right) \tag{4.84}$$

Then,

$$\begin{aligned} \ddot{Z} &= g - \frac{1}{m}\left(\frac{-m}{\cos\Phi\cos\Theta}\left(-a\sigma_Z - g - \hat{\beta}\right)\right)\cos\Phi\cos\Theta + \beta \\ &= -a\sigma_Z + \left(\beta - \hat{\beta}\right) \end{aligned} \tag{4.85}$$

Let $\left(\beta - \hat{\beta}\right) = \tilde{\beta}$

$$\ddot{Z} = -a\sigma_Z + \tilde{\beta} \tag{4.86}$$

From Equation (4.86)

$$\tilde{\beta} = \ddot{Z} + a\sigma_Z \tag{4.87}$$

And its derivative is;

$$\dot{\tilde{\beta}} = \dddot{Z} + a\dot{\sigma}_Z \tag{4.88}$$

Then finally,

$$\dot{\tilde{\beta}} = -\lambda\sigma_\Psi \tag{4.89}$$

Theorem 4.1: is used to show the asymptotical stability of the sliding surface to show the Lyapunov stability law for the adaptive sliding mode controller obtained.

Consider the following Lyapunov candidate function;

$$V = \frac{1}{2}\sigma_Z{}^T\sigma_Z + \frac{1}{2}\tilde{\beta}^T\lambda^{-1}\tilde{\beta} \tag{4.90}$$

The first derivative of the Lyapunov candidate function can be represented as

$$\dot{V} = \sigma_Z{}^T\dot{\sigma}_Z + \tilde{\beta}^T\lambda^{-1}\dot{\tilde{\beta}} \tag{4.91}$$

Expanding the above equation,

$$\begin{aligned} \dot{V} &= \sigma_Z{}^T\dot{\sigma}_Z + \tilde{\beta}^T\lambda^{-1}\dot{\tilde{\beta}} \\ &= \sigma_Z{}^T\left(-a\sigma_Z + \tilde{\beta}\right) + \tilde{\beta}^T\lambda^{-1}\left(-\lambda\sigma_Z\right) \\ &= -\sigma_Z{}^T a\sigma_Z < 0 \end{aligned} \tag{4.92}$$

So it can be stated that the sliding surface concerning the error states converges to zero as time increases.

4.7. Rotor Speed Calculation from Control Signal

The inputs for the quadcopter plant is as follows from Equation (3.1);

$$\begin{aligned}
\tilde{U}_1 &= F_1 + F_2 + F_3 + F_4 \\
\tilde{U}_2 &= l\left(F_1 - F_2 - F_3 + F_4\right) \\
\tilde{U}_3 &= l\left(F_1 + F_2 - F_3 - F_4\right) \\
\tilde{U}_4 &= -M_1 + M_2 - M_3 + M_4
\end{aligned}$$

The above equation can be put in matrix form as from Equation (3.44);

$$\begin{bmatrix} \tilde{U}_1 \\ \tilde{U}_2 \\ \tilde{U}_3 \\ \tilde{U}_4 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ bl & -bl & -bl & bl \\ bl & bl & -bl & -bl \\ -d & d & -d & d \end{bmatrix} \begin{bmatrix} \hat{\Omega}_1^2 \\ \hat{\Omega}_2^2 \\ \hat{\Omega}_3^2 \\ \hat{\Omega}_4^2 \end{bmatrix}$$

$$\begin{bmatrix} \hat{\Omega}_1^2 \\ \hat{\Omega}_2^2 \\ \hat{\Omega}_3^2 \\ \hat{\Omega}_4^2 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ bl & -bl & -bl & bl \\ bl & bl & -bl & -bl \\ -d & d & -d & d \end{bmatrix}^{-1} \begin{bmatrix} \tilde{U}_1 \\ \tilde{U}_2 \\ \tilde{U}_3 \\ \tilde{U}_4 \end{bmatrix} \tag{4.93}$$

$$\begin{bmatrix} \hat{\Omega}_1^2 \\ \hat{\Omega}_2^2 \\ \hat{\Omega}_3^2 \\ \hat{\Omega}_4^2 \end{bmatrix} = \begin{bmatrix} 2.9842\times10-5 & 2.9842\times10-5 & 2.9842\times10-5 & 2.9842\times10-5 \\ 2.9842\times10-5\times0.5 & -2.9842\times10-5 & -2.9842\times10-5 & 2.9842\times10-5 \\ 2.9842\times10-5 & 2.9842\times10-5 & -2.9842\times10-5 & -2.9842\times10-5 \\ -3.2320\times10-7 & 3.2320\times10-7 & -3.2320\times10-7 & 3.2320\times10-7 \end{bmatrix}^{-1} \begin{bmatrix} \tilde{U}_1 \\ \tilde{U}_2 \\ \tilde{U}_3 \\ \tilde{U}_4 \end{bmatrix} \tag{4.94}$$

Therefore, the speeds of the rotors are calculated from the input signals.

CHAPTER FIVE

5. RESULT AND ANALYSIS

In this Chapter, the validity and stabilization of the proposed adaptive sliding mode controller (ASMC) for the Quadcopter dynamics are simulated using MATLAB/Simulink environment. The minimum jerk trajectory tracking is applied to test the controller performance. For the simulation of the proposed system, the following parameters shown in Table 5.1 are used which is collected from STI of ASTU other related references.

Table 5.1 Physical parameter for quadcopter [32], [34]

Parameters	Value and Unit
Arm length (*l*)	0.5m
Total mass (*m*)	0.1-6kg
Body inertia of *x*-axis (*Ix*)	3.8278×10−3
Body inertia of *y*-axis (*Iy*)	3.8278×10−3
Body inertia of *z*-axis (*Iz*)	7.6566×10−3
Motor inertia (*Ir*)	2.8385×10−5N.m/rad/s2
Lift Coefficient (*b*)	2.9842×10−5N.m/rad/s
Drag Coefficient (*d*)	3.2320×10−7N.m/rad/s
Gravitational acceleration (*g*)	9.8m/s^2

5.1. Proposed Control System Result

In the quadcopter model as described in chapter-3 gyroscopic moment is incorporated on the nonlinear model of the system. The proposed controller minimum jerk tracking performance is performed by giving the desired trajectory to the states of the system. The following reference trajectories are used for the simulation.

$$\begin{cases} a\_x = \begin{bmatrix} 0 & 4 & 2 \end{bmatrix} \\ b\_x = \begin{bmatrix} 4 & 2 & -2 \end{bmatrix} \end{cases}, \quad \begin{cases} a\_y = \begin{bmatrix} 0 & 1 & -3 \end{bmatrix} \\ b\_y = \begin{bmatrix} 1 & -3 & 0 \end{bmatrix} \end{cases}, \text{ and } \begin{cases} a\_z = \begin{bmatrix} 0 & 1 & 2 \end{bmatrix} \\ b\_z = \begin{bmatrix} 1 & 2 & 4 \end{bmatrix} \end{cases}$$

$$\begin{cases} a\_phi = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \\ b\_phi = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \end{cases}, \quad \begin{cases} a\_theta = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \\ b\_theta = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \end{cases}, \text{ and } \begin{cases} a\_si = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \\ b\_si = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \end{cases}$$

Where a is the initial points and b is the final points. The following section shows the performance analysis of the ASMC controller for each of all states independently for the same trajectory reference. To check the performance of the controller a simulation analysis has been done by considering external uncertainty and different masses (0.5kg, 3kg, and 6kg) for the minimum jerk reference trajectory.

5.1.1. Trajectory Tracking Performance of the Control System at *m*=0.5kg

A) Altitude (*Z*) controller tracking performance: The result in Figure 5.1, shows the trajectory tracking performance of the altitude controller. Always the quadcopter starts its flight from ground level, *Z=0*, so altitude is always positive at the start or during takeoff. The following desired minimum jerk trajectory is given to the proposed altitude controller.

$$\begin{cases} a\_z = \begin{bmatrix} 0 & 1 & 2 \end{bmatrix} \\ b\_z = \begin{bmatrix} 1 & 2 & 4 \end{bmatrix} \end{cases}$$

The error between the desired altitude trajectory and actual altitude trajectory which is called tracking error is shown in Figure 5.2. For the first 5 seconds, it is around 0.014, this is due to the adaptation of system parameters. After the time of 5 seconds the error is very small and it is around 0.0001, and then it is reduced to zero, this implies that the actual altitude tracks exactly the reference trajectory.

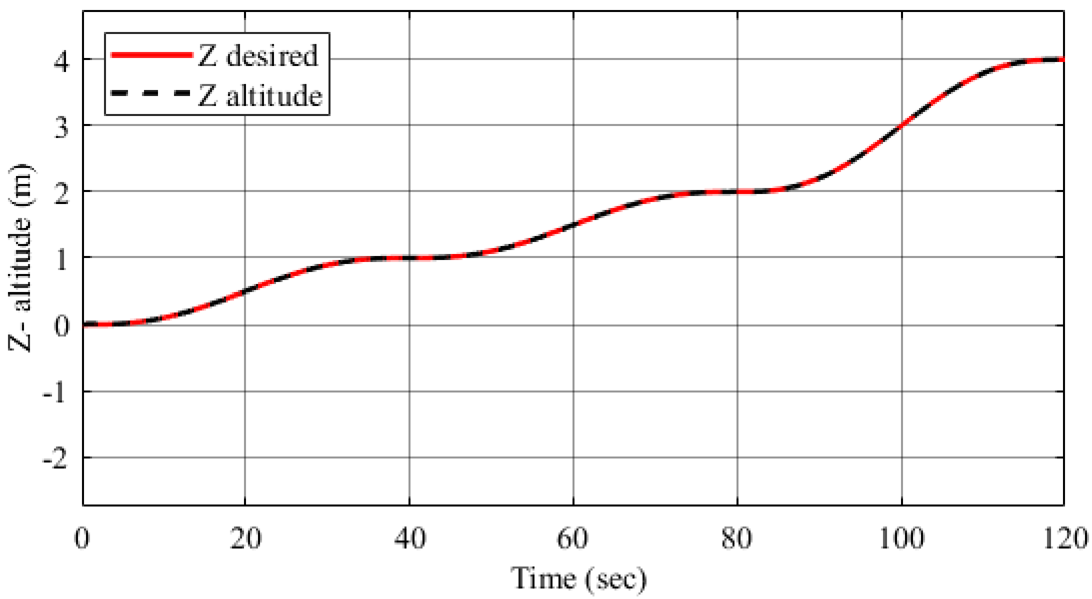


Figure 5.1 Altitude *Z* reference trajectory tracking controller performance using ASMC

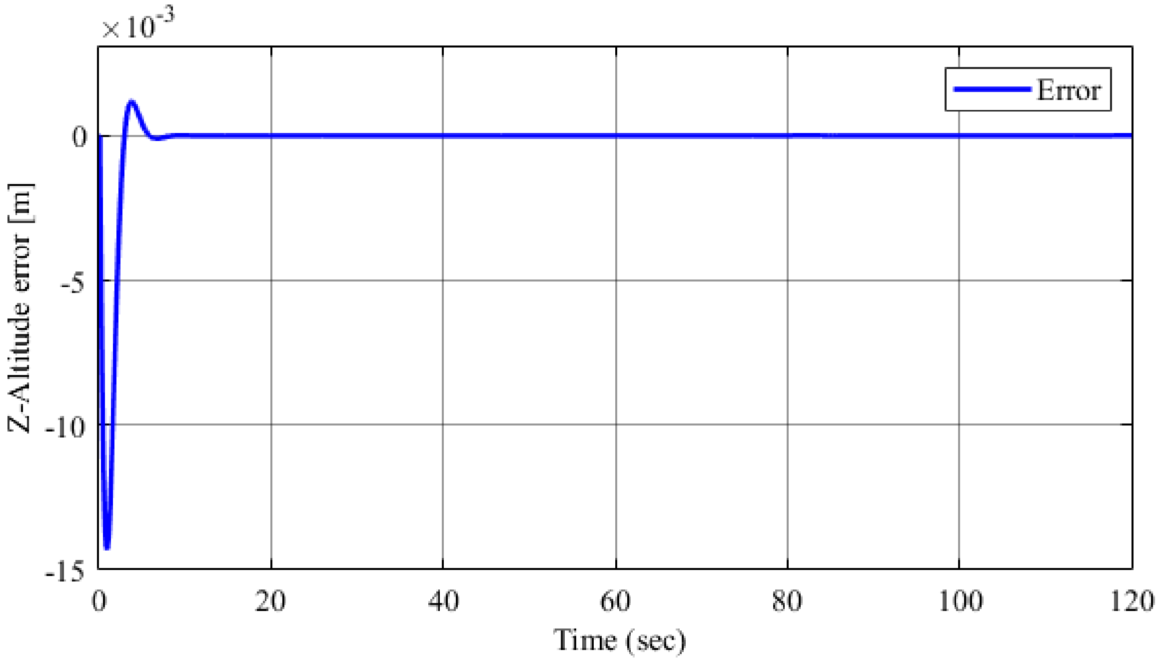


Figure 5.2 Altitude reference tracking error

B) Position (*X*) controller tracking performance: Figure 5.3, shows that the actual *X* position of the quadcopter tracks the desired trajectories starting from the initial up to the final within 120 seconds of simulation time. Then, from Figure 5.4, the error between the reference trajectory and the actual trajectory is shown clearly. The maximum value of the error is 0.00002, which is almost zero. Therefore, the controller acts effectively.

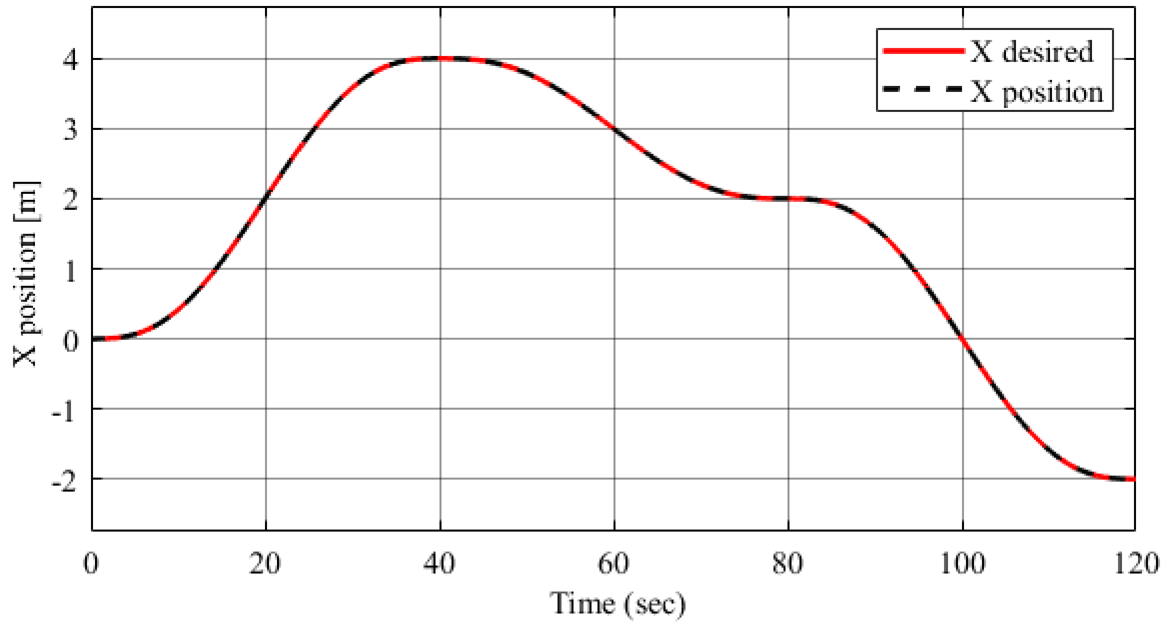


Figure 5.3 Position *X* reference trajectory tracking controller performance using ASMC

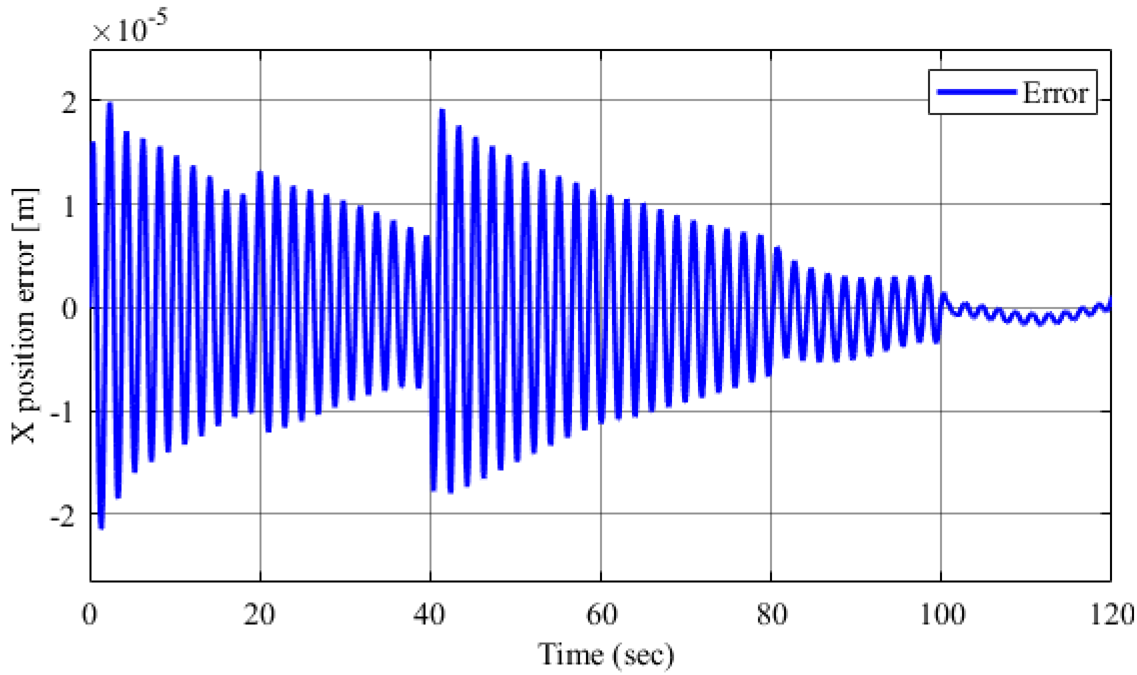


Figure 5.4 Position X reference tracking error

C) Position (*Y*) controller tracking performance: The actual trajectories of the *Y* position track the desired trajectories starting from the initial up to the final within 120 seconds of simulation time as shown in Figure 5.5. From Figure 5.6, the controller acts perfectly because the tracking error remains very near to zero throughout the simulation that is almost 0.000004.

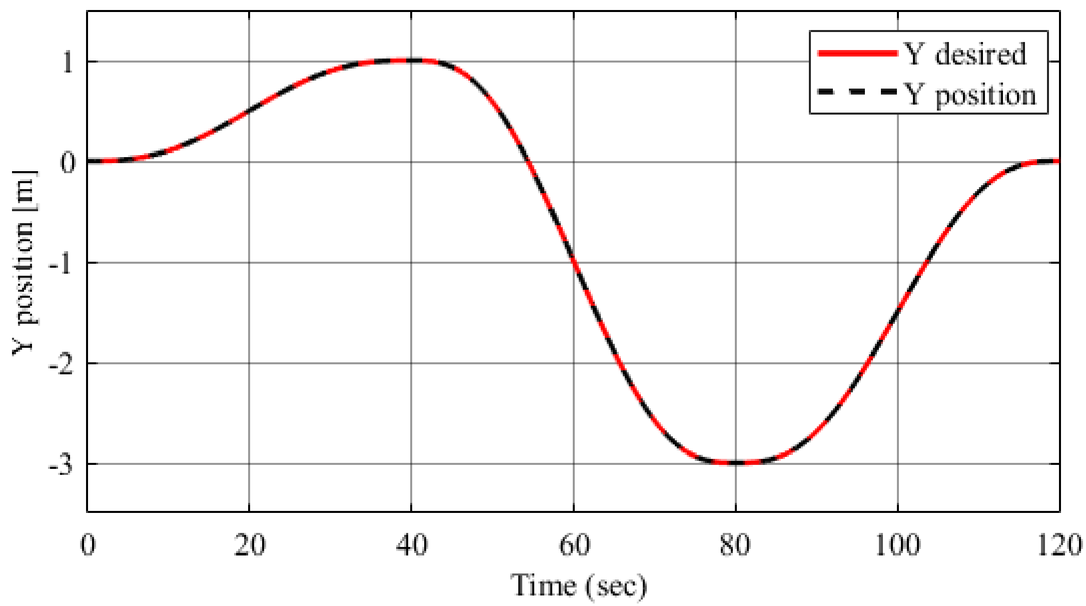


Figure 5.5 Position *Y* reference trajectory tracking controller performance using ASMC

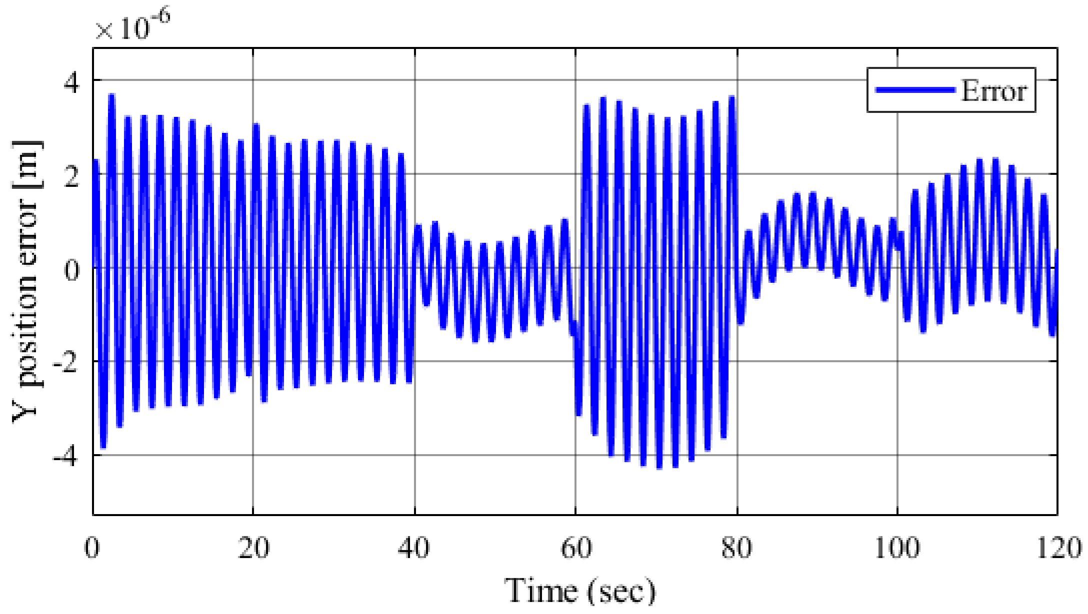


Figure 5.6 Position *Y* reference tracking error

D) Roll angle controller tracking performance (*Φ*): In Figure 5.7, the actual trajectories of *phi (roll)* tracks the desired trajectory starting from the initial up to the final. Even though there are no input trajectories given to this roll controller, it shows a little variation from zero inputs. This is because of the interrelation between *phi* and *Y* positions. As the system is underactuated, there must be some control relations between some of the states.

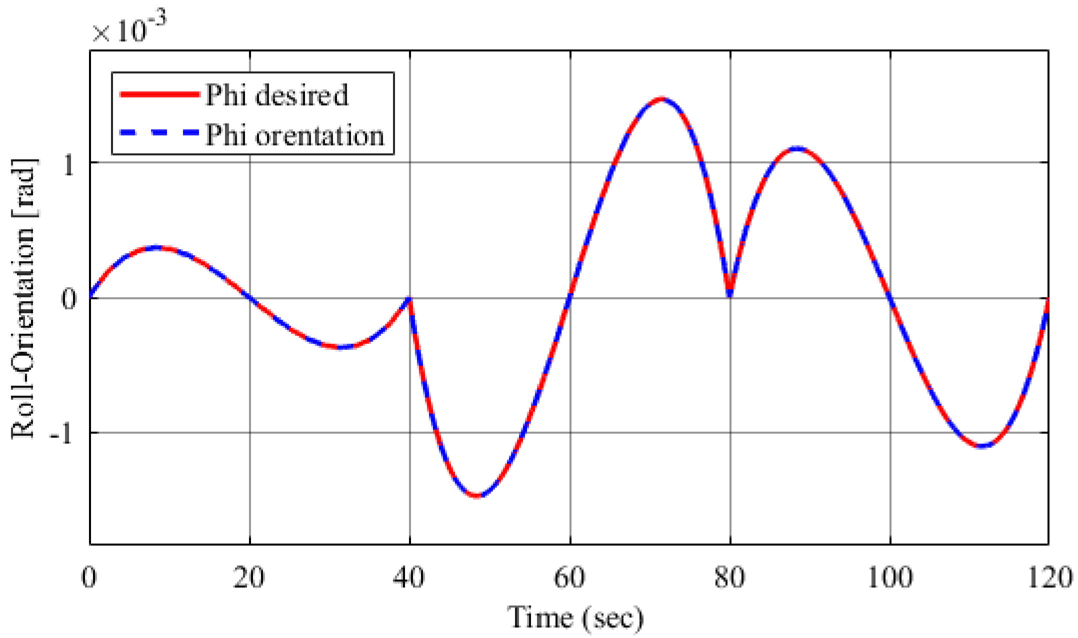


Figure 5.7 Roll angle Φ reference trajectory tracking controller performance using ASMC

Unless only four outputs can be controlled using their inputs. So the generated trajectory which if given to *Y* position creates some disturbance or trajectory to *phi* orientation angle. The value is around 0.0018 and theta follows this trajectory as it is controlled. Additionally, from Figure 5.8, the maximum tracking error is around 10.5×10^{-6} which is almost zero. So, it can be concluded that the controller perfectly controls the roll orientation angle.

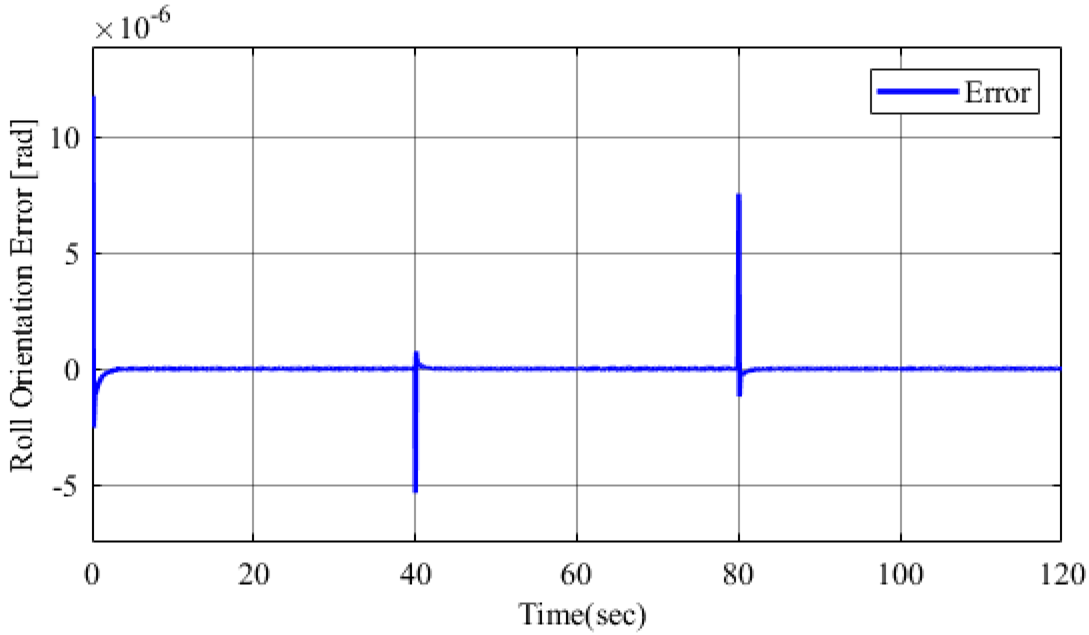


Figure 5.8 Roll angle Φ reference tracking error

E) Pitch controller tracking performance (Θ): As shown in Figure 5.9, the actual trajectories of pitch orientation angle tracks the given desired trajectories starting from the initial up to the final within the given simulation time.

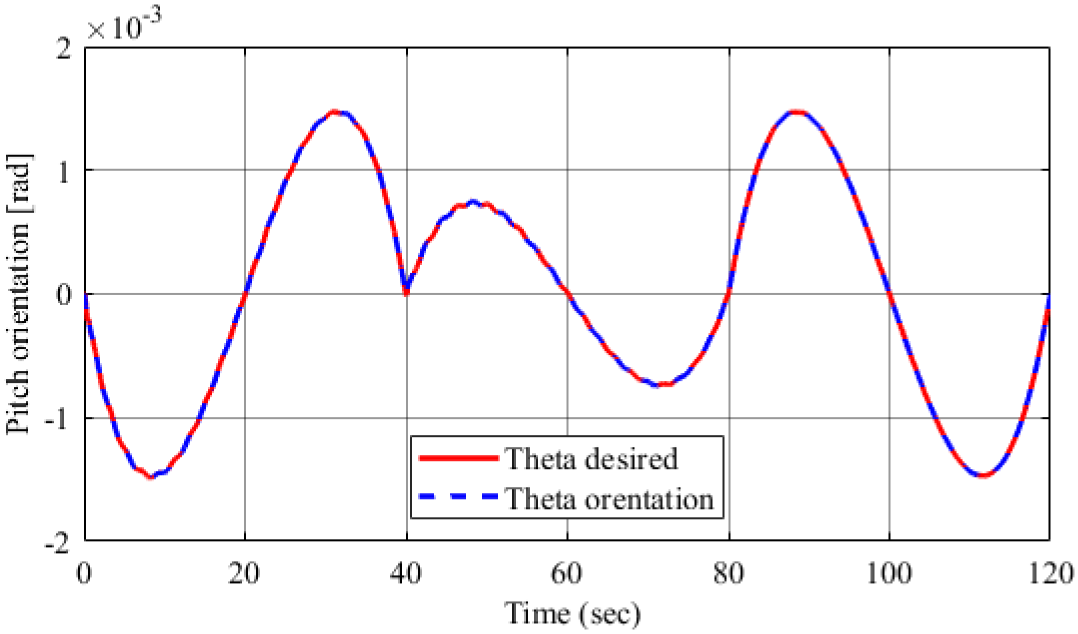


Figure 5.9 pitch angle Θ reference trajectory tracking controller performance using ASMC

As X position and pitch angle is related internally the small pitch reference trajectory is generated. Therefore, the actual pitch orientation angle follows the generated trajectory. Its peak tracking error of pitch orientation angle is -5.5x10^{-5}, which is very small and near to zero as shown in Figure 5.10. So, the controller performs almost exactly on the pitch angle.

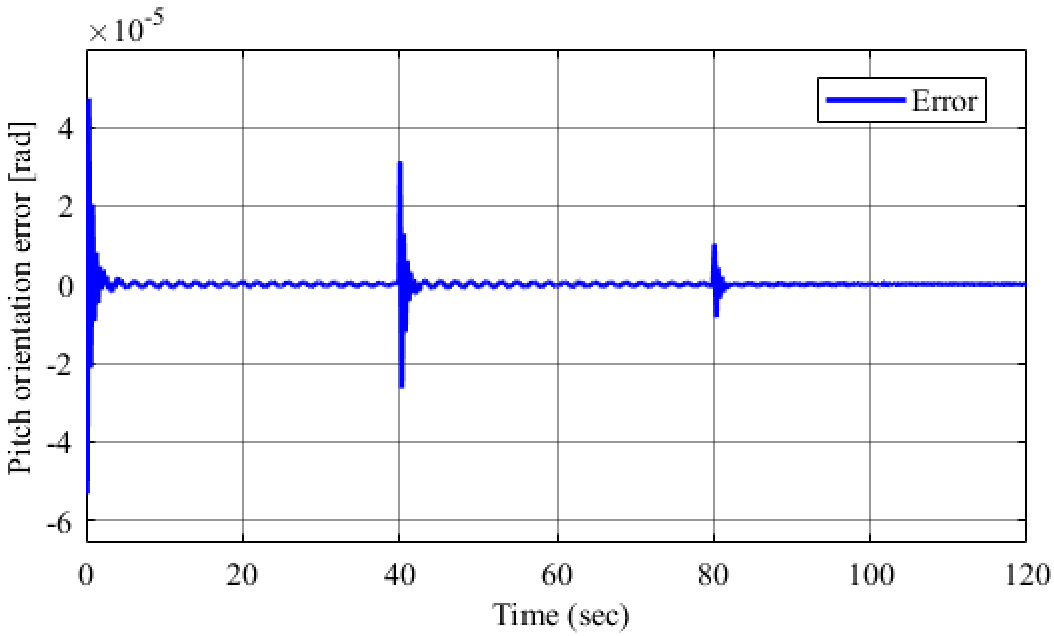


Figure 5.10 Pitch reference tracking error

F) Yaw controller tracking performance (Ψ): In Figure 5.11 the yaw orientation angle tracking is zero as there are no reference trajectories given to it. Even if this research work controls all the six outputs of the quadcopter system the trajectory is only generated for the X, Y position and Z altitude and *yaw* should be zero due to the application area of the quadcopter chosen. So there is no need to show trajectory tracking of the three orientation angles.

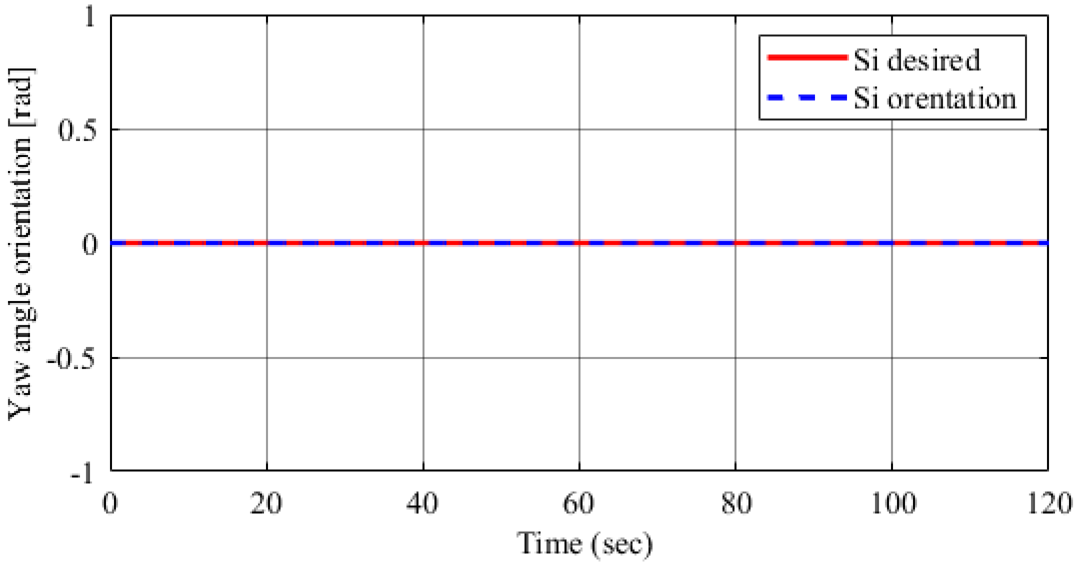
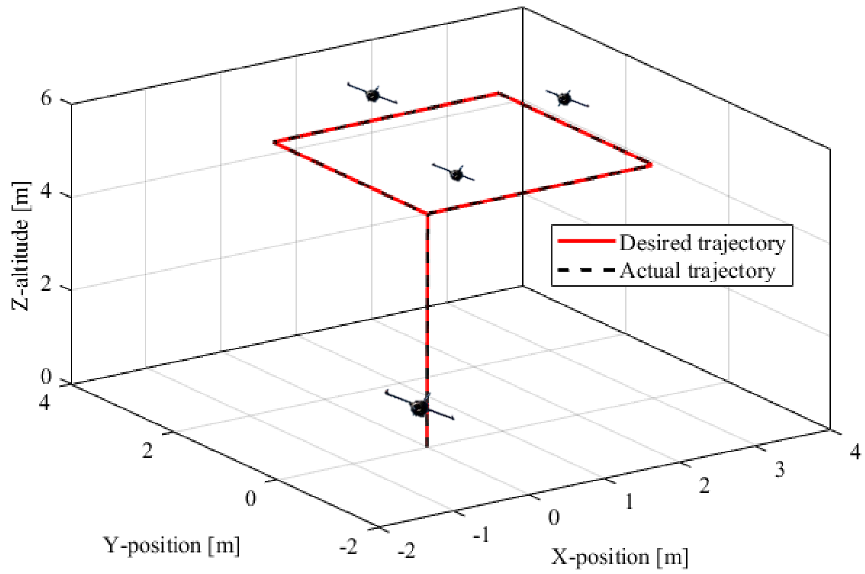
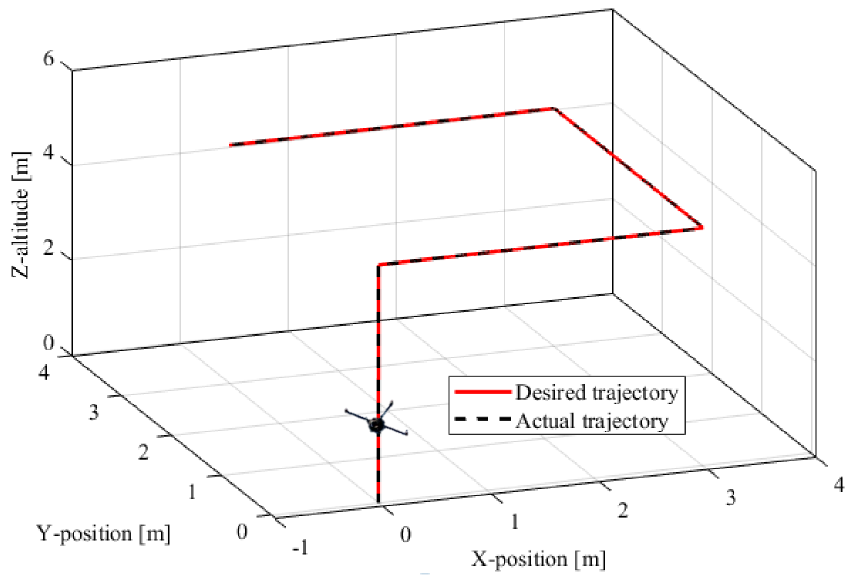


Figure 5.11 Yaw angle Ψ reference trajectory tracking controller performance using ASMC

G) 3D minimum jerk trajectory generated: Figure 5.11 a and b shows the 3D square and one side open square shape reference trajectory tracking of a quadcopter system using ASMC. The flight of the quadcopter by combining the X, Y position, Z altitude, roll, theta, yaw orientation is shown briefly as they track the given generated minimum jerk trajectory starting from the ground throughout its flight. As the quadcopter starts flight from the ground position based on the given reference trajectory it starts from zero position and goes to the required areas to accomplish its tasks by following the given trajectory to the control system.



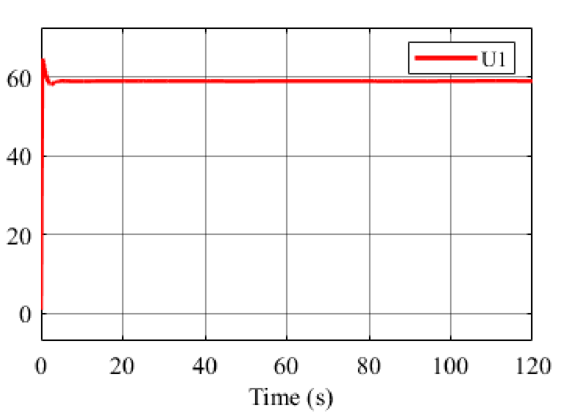
a. 3D square shape trajectory tracking



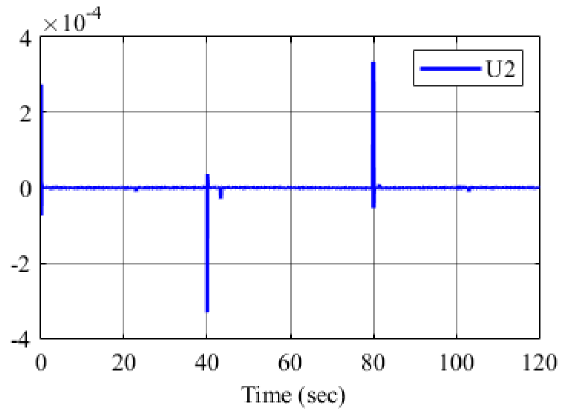
b. 3D one side open square shape trajectory tracking

Figure 5.12 3D reference trajectory tracking controller performance using ASMC

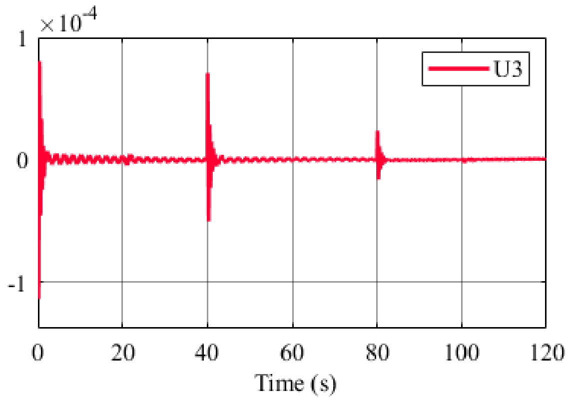
5.2. Control Signals for Tracking Using ASMC



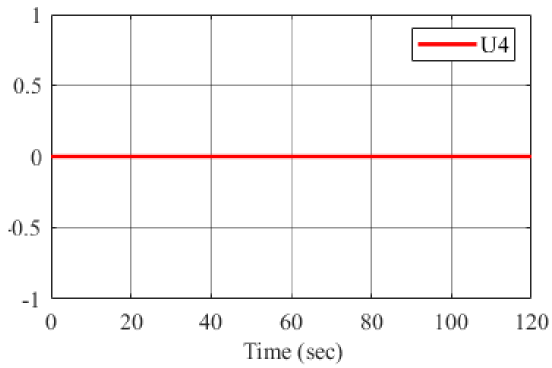
a. Control signal $\tilde{U}_1$



b. Control signal $\tilde{U}_2$



c. Control signal $\tilde{U}_1$



d. Control signal $\tilde{U}_1$

Figure 5.13 Control signals of trajectory tracking controller using ASMC

5.3. Control signals for Tracking Using SMC

The chattering effects of the SMC controller which is reduced using ASMC is shown using the input signals to the system. This chattering is minimized using ASMC.

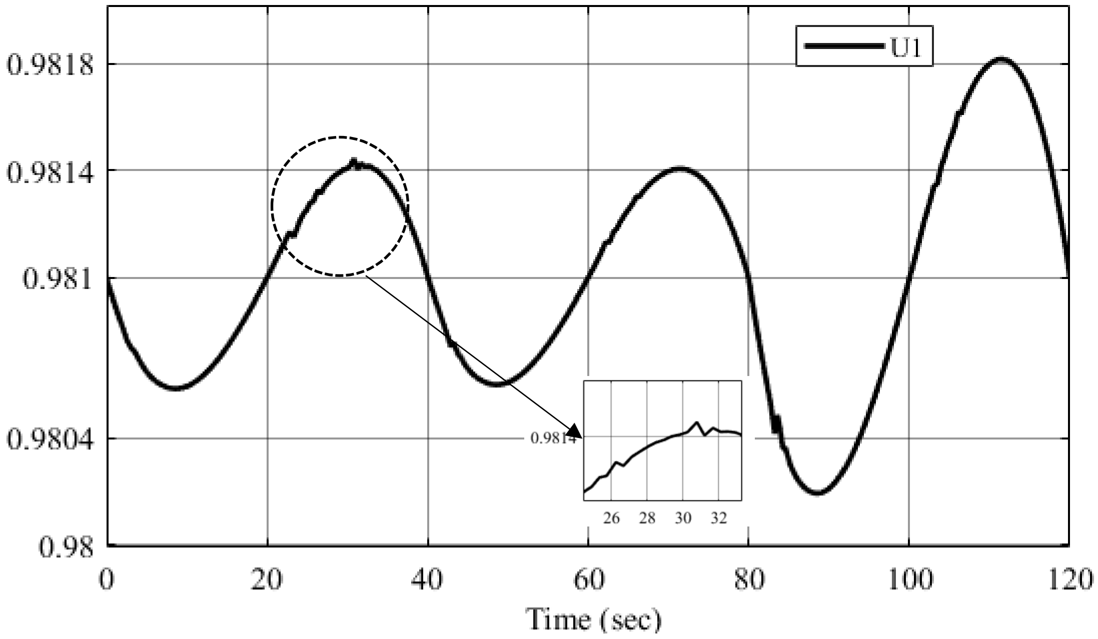


Figure 5.14 Control signals of trajectory tracking controller using ASMC

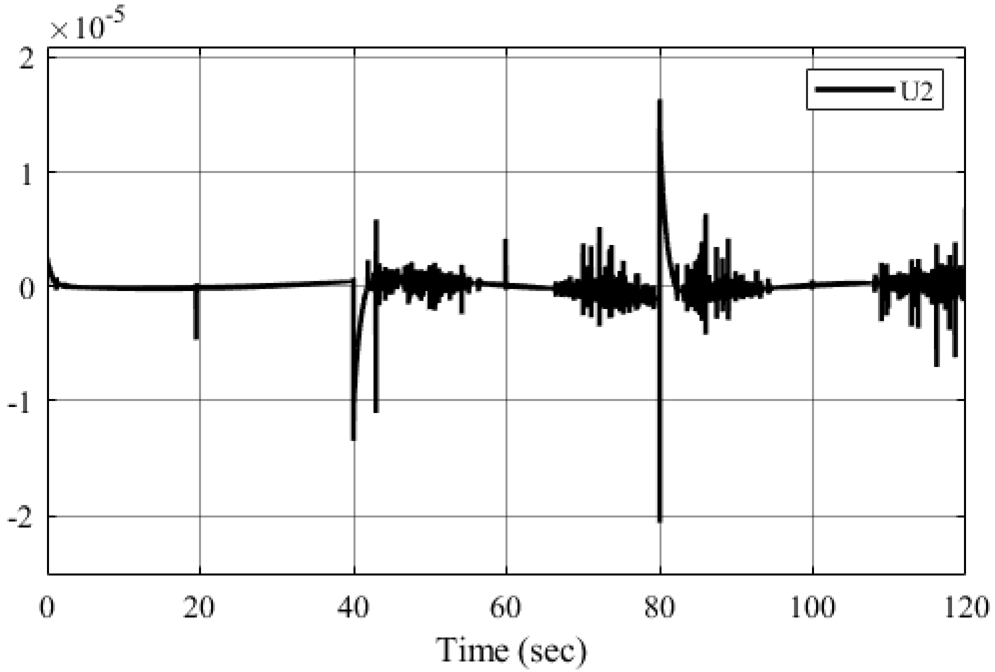


Figure 5.15 Control signals of trajectory tracking controller using ASMC

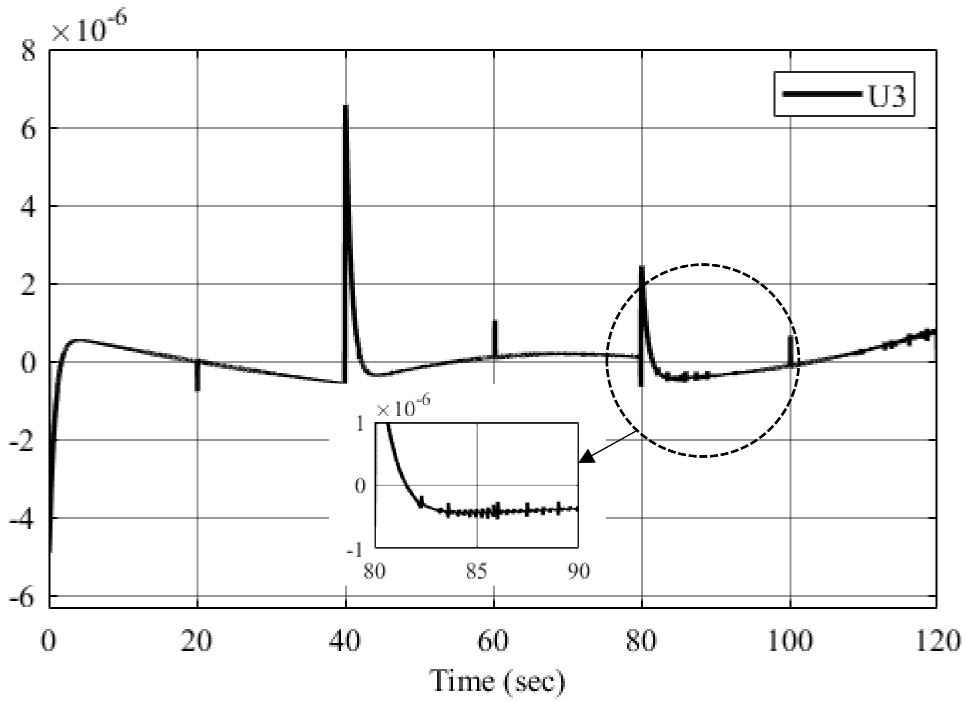


Figure 5.16 Control signals of trajectory tracking controller using ASMC

As shown from the above Figures 5.14, 5.15 and 5.16 there are chattering effects on the control inputs of the system. As there is no input given U_4 there is no chattering on it. Therefore the proposed controller minimizes the chattering effects of SMC by using an adaptive controller and exponential reaching law.

5.4. Simulation Result of the ASMC with Parameter Uncertainty and External Random Disturbance

In this section, the robustness and ability of the controller to track the given minimum jerk reference trajectory in the presence of parameter uncertainties and external disturbances are evaluated and analyzed using MATLAB/Simulink implementation results.

The parametric uncertainties and external disturbances applied to the system are as follows.

1. The parameter uncertainty is represented by 0.1kg-6kg mass variation.
2. Step input disturbance is added to the plant as wind disturbance. By assuming the quadcopter system is under the pressure of constant wind of value given in Table 5.2.

Table 5.2 Range of external wind disturbance given to the system

Dynamic states	Wind disturbance range(constant)
Altitude *Z*	-0.1 to 0.1 Newton
Position *X*	-0.1 to 0.1 Newton
Position *Y*	-0.1 to 0.1 Newton

5.4.1. Trajectory Tracking Performance of ASMC with Parameter Uncertainty

The parametric uncertainty of the quadcopter system is mass variation and it is varied initially not during flight time. This means the mass of the quadcopter is changed before the quadcopter flies and all the signals are arranged for the changed mass of the quadcopter. The controller is designed once and applied to the system. The quadcopter does not change its mass during flights, the mass is changed initially. And this ASMC is used for different masses (0.1kg-6kg) of the quadcopter.

A) Altitude (*Z*) controller tracking performance: As shown in Figure 5.17, when the mass of the quadcopter changes its inertias also changes due to the relation between mass and inertia of a system. As the controller is an adaptive controller the results are expected to overcome and adapt the change in parameter uncertainty and to be stable by following the desired trajectory. So the result of altitude tracking shows how the system stabilized after some overshoot or peak error of 0.23 at a mass of 6kg as shown in Figure 5.18.

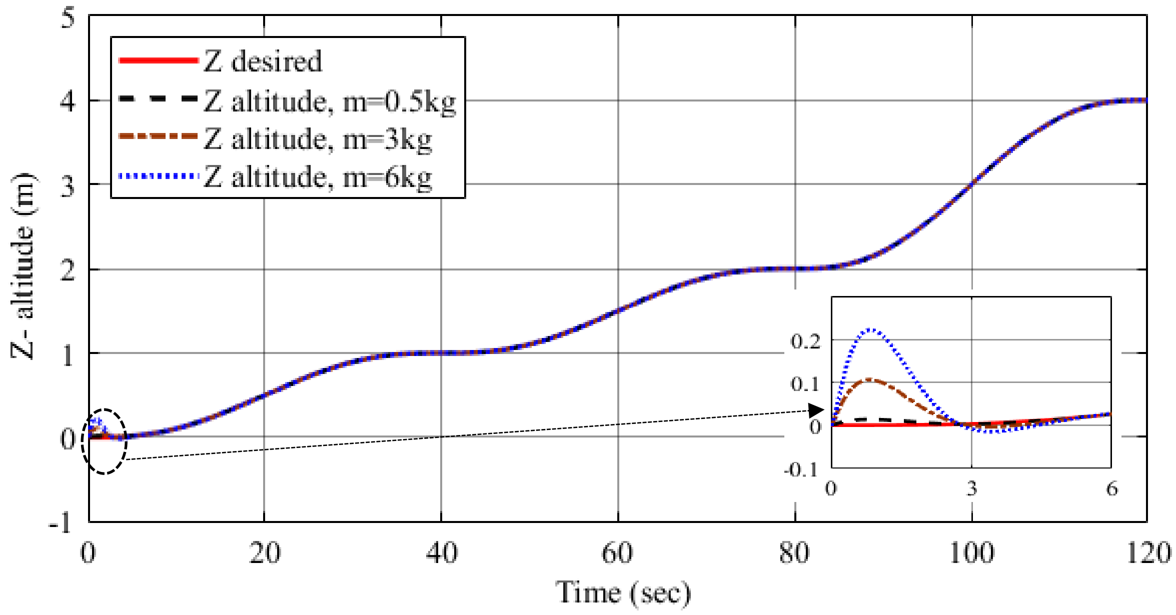


Figure 5.17 Altitude Z reference trajectory tracking controller performance for different values of m using ASMC

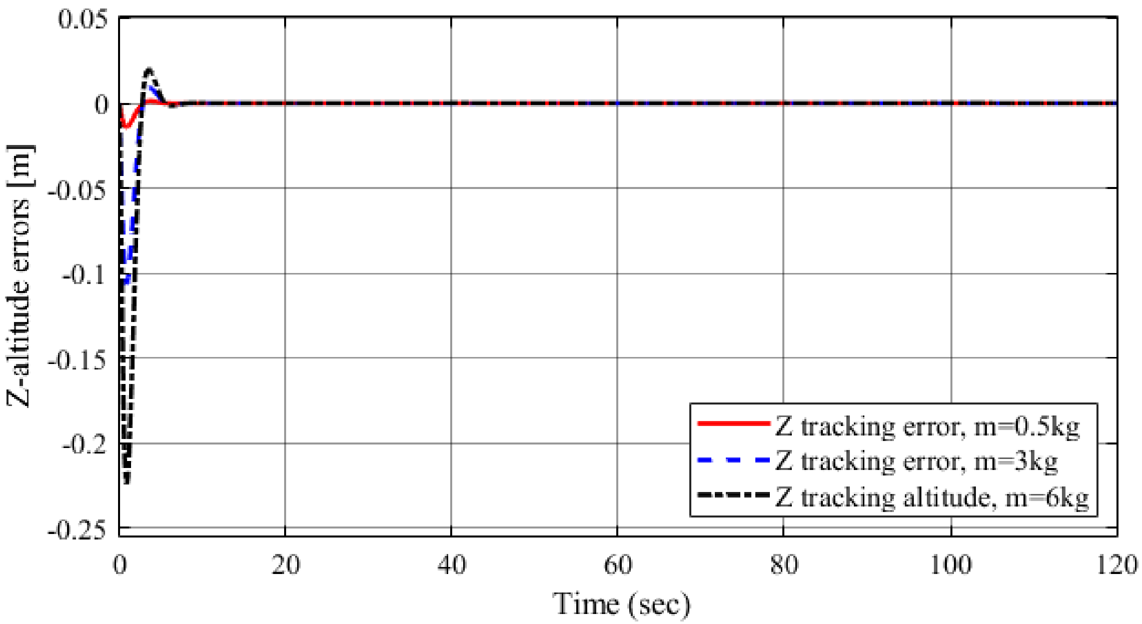


Figure 5.18 Altitude Z reference trajectory tracking error for different values of m using ASMC

B) Position (*X*) controller tracking performance: Figure 5.19 shows how position X tracks the reference trajectory with a mass variation. And as the controller adapts parameter variations the system can track the desired trajectory with a peak error shown in Figure 5.20 which is $4x10^{-5}$ at mass 6kg and it is almost zero error.

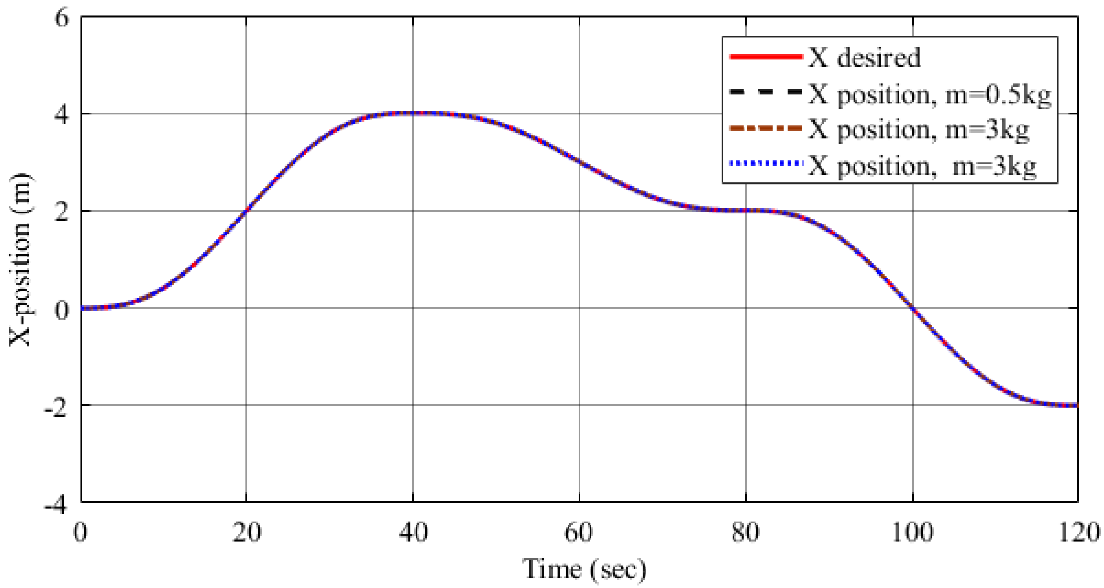


Figure 5.19 Position X reference trajectory tracking controller performance for different values of m using ASMC

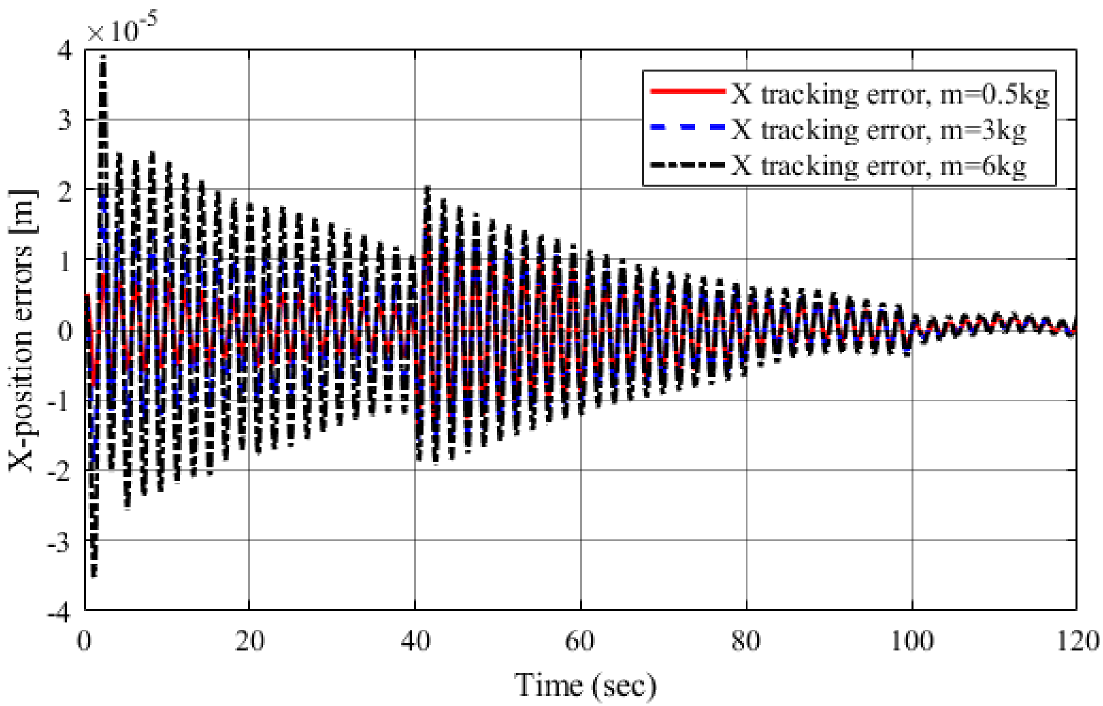


Figure 5.20 Position X reference trajectory tracking error for different values of m using ASMC

C) Position (*Y*) controller tracking performance: Position *Y* tracking the performance of ASMC with a varying mass of the quadcopter system is shown in Figure 5.21 and the result shows that the ASMC performs effectively with a peak error of almost zero which is shown in Figure 5.22.

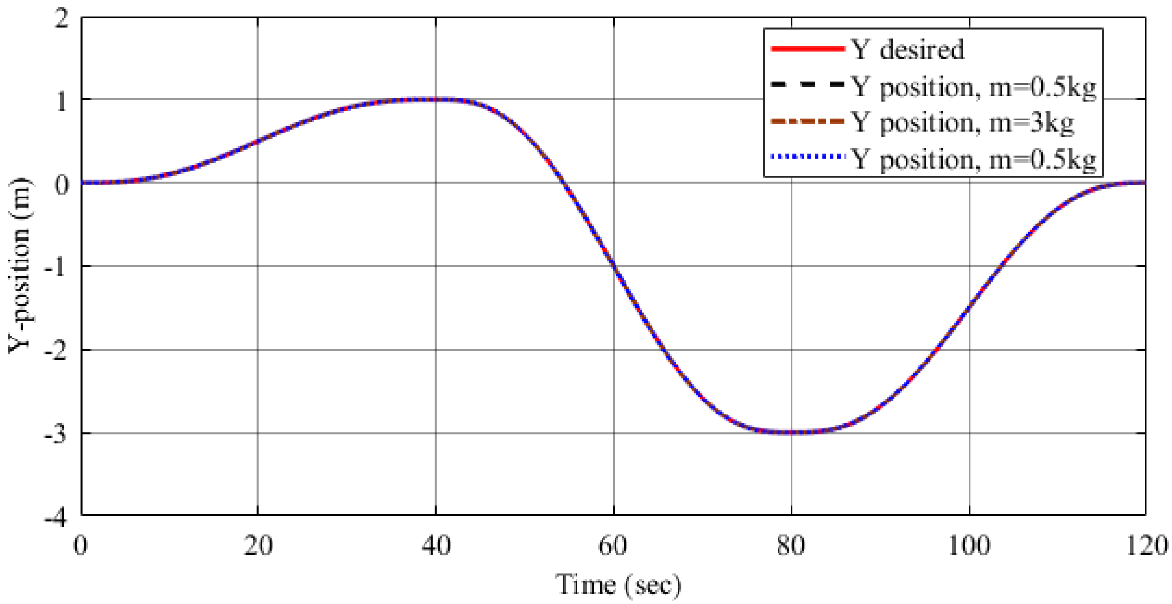


Figure 5.21 Position Y reference trajectory tracking controller performance for different values of m using ASMC

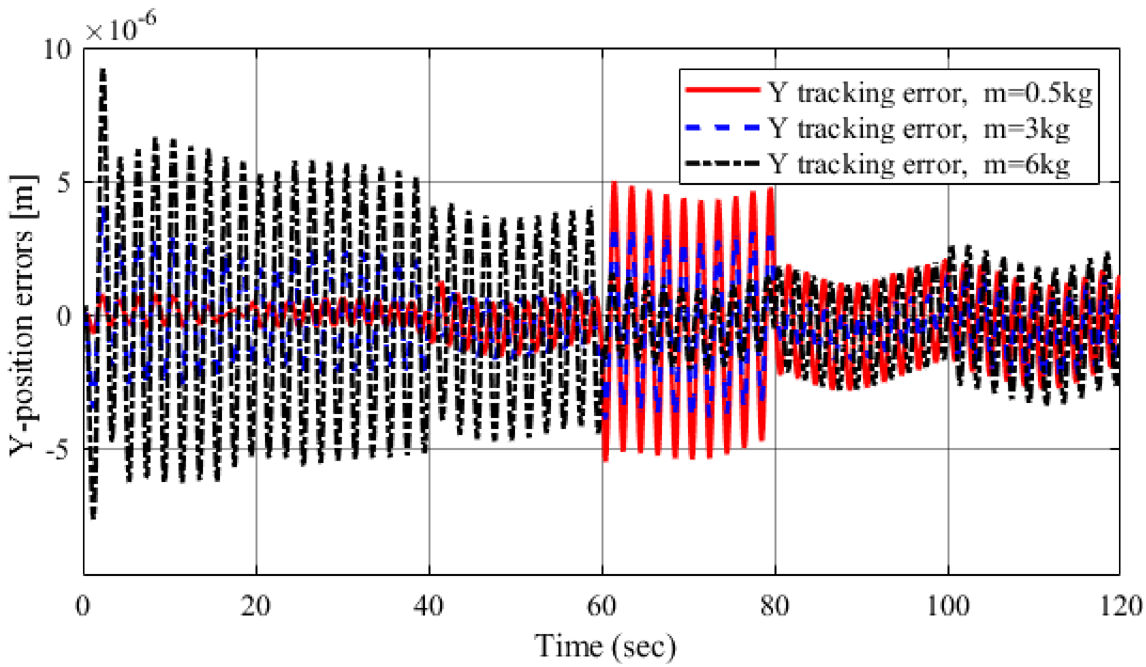


Figure 5.22 Position X reference trajectory tracking error for different values of m using ASMC

As the response of the test results of altitude, Z is shown in Figure 5.17, in presence of parameter uncertainty, a little bit increasing of error is observed (about 0.23 m) at the mass of 6 kg; which is a small change as compared to the nominal one. Despite the increase of the very small error, the performance of the system is still accepted and reasonable. The results X and Y positions are shown in Figures 5.19 and 5.21, which indicates the controller successfully compensates the effects of disturbances maintaining the stability of the system by regaining all positions to the desired trajectory values. This implies that the proposed controller is properly performed and also output results demonstrated that the proposed controllers have good pilot performance.

5.4.2. Trajectory Tracking Performance of the Control System with External Disturbances

A) Altitude (Z) controller tracking performance:

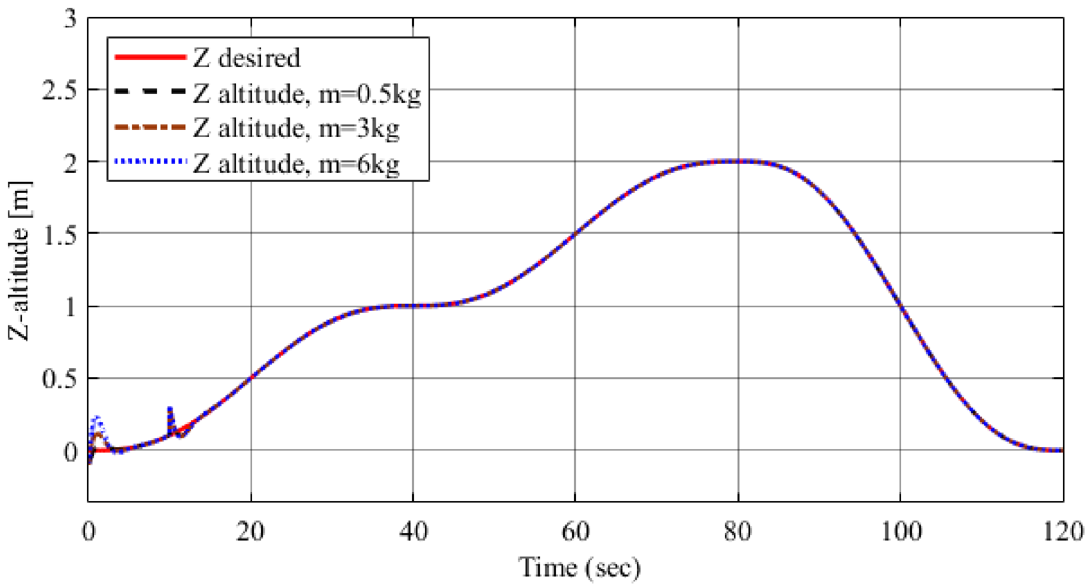


Figure 5.23 Altitude Z trajectory tracking when there is wind gust disturbance

B) Position (X) controller tracking performance:

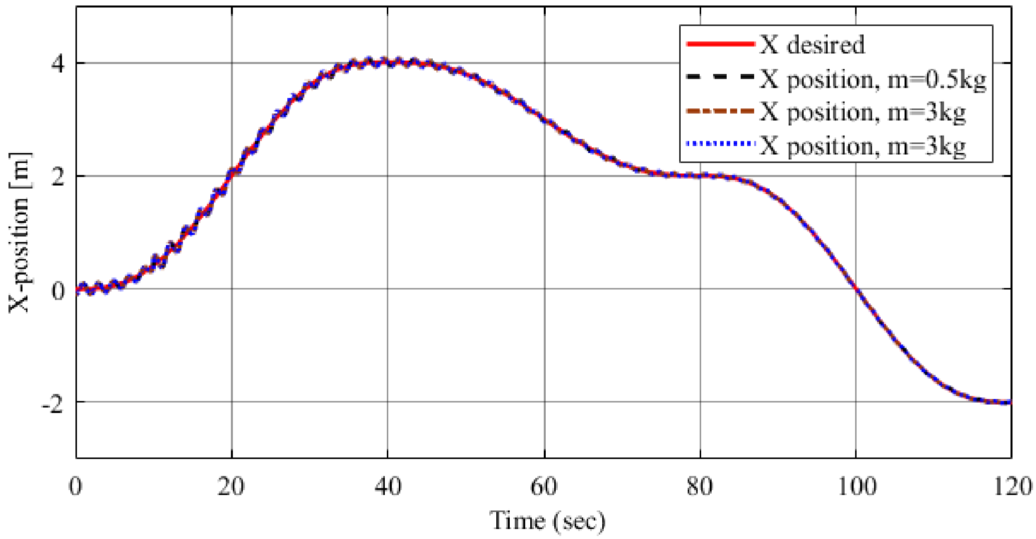


Figure 5.24 Position *X* trajectory tracking when there is wind gust disturbance

B) Position (Y) controller tracking performance:

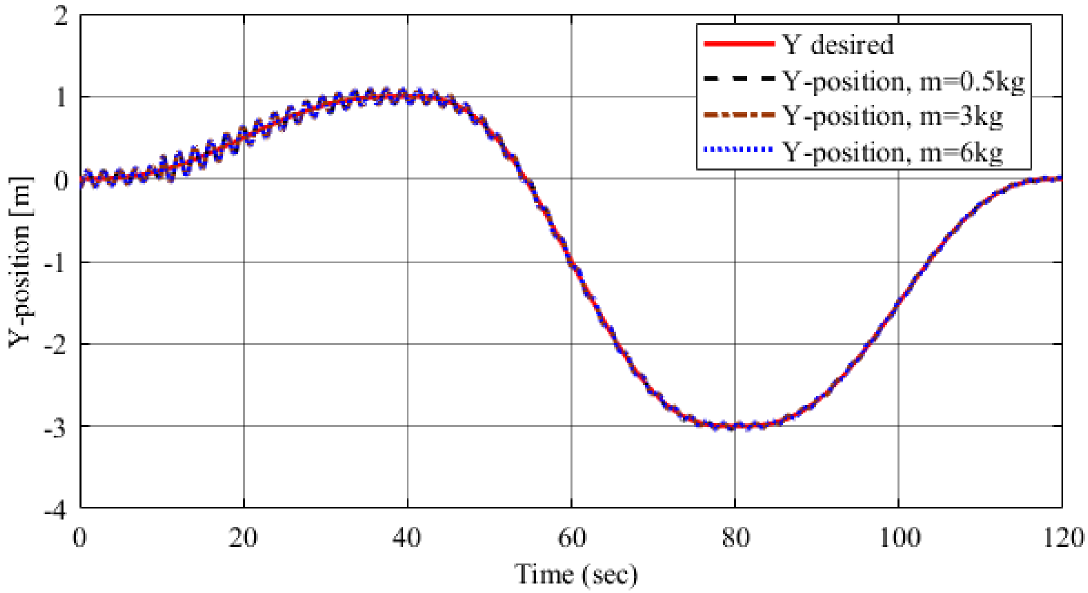


Figure 5.25 Position *Y* trajectory tracking when there is wind gust disturbance

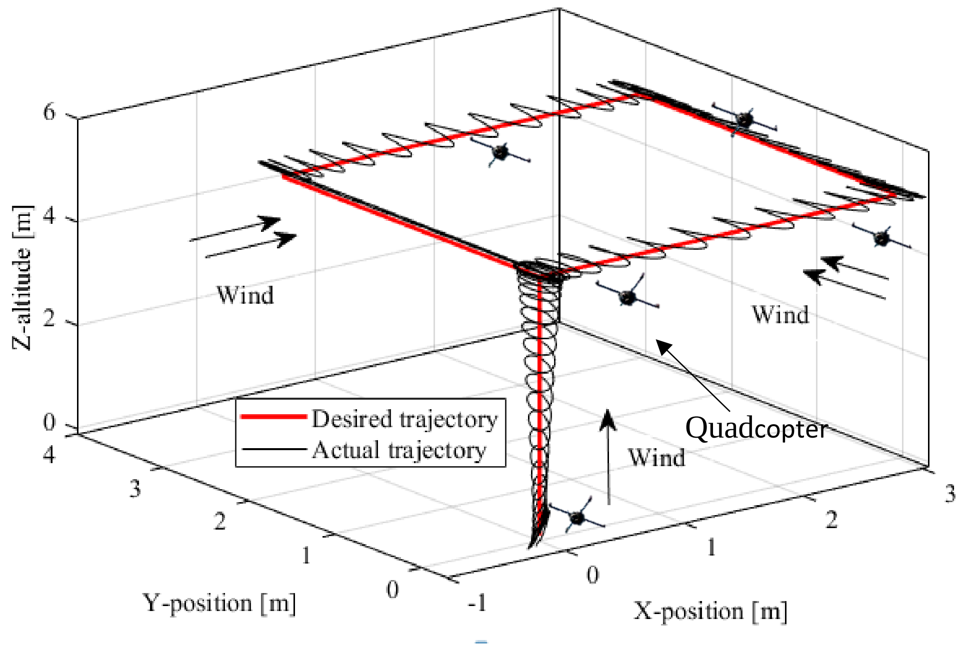


Figure 5.26 *3D* trajectory tracking when there is wind gust disturbance

As the response of the test results of altitude *Z* shown in figure 5.23, in presence of external disturbances, a little bit increasing of error is observed (about 0.4 m) at a mass of 6 kg; which is a small change as compared to the nominal one. Despite the increase of the very small error, the performance of the system is still acceptable and reasonable. The results *X* and *Y* positions are shown in Figures 5.24 and 5.25, which indicates the controller successfully compensates the effects of disturbances maintaining the stability of the system by regaining all positions to the desired trajectory values. And the *3D* tracking result in Figure 5.26 also shows as the wind external disturbance applied to the quadcopter how it regains its tracking position with small tracking error. This implies that the proposed controller is properly performed and also output results demonstrated that the proposed controllers have good pilot performance.

5.5. Sliding Mode Controller results

5.5.1. Trajectory tracking performance of SMC with parametric uncertainty

A) Altitude (*Z*) controller tracking performance: Altitude Z tracking of the quadcopter system is shown in Figure 5.27. The result shows that as there is parameter uncertainty in the system model the SMC fails. The black broken line shows the altitude of the quadcopter at the designed mass of 0.1kg. Unlike ASMC the SMC does not track the given trajectory as shown in the result by a black straight line. For the tracking mass value of 0.1kg, the tracking error is negligible.

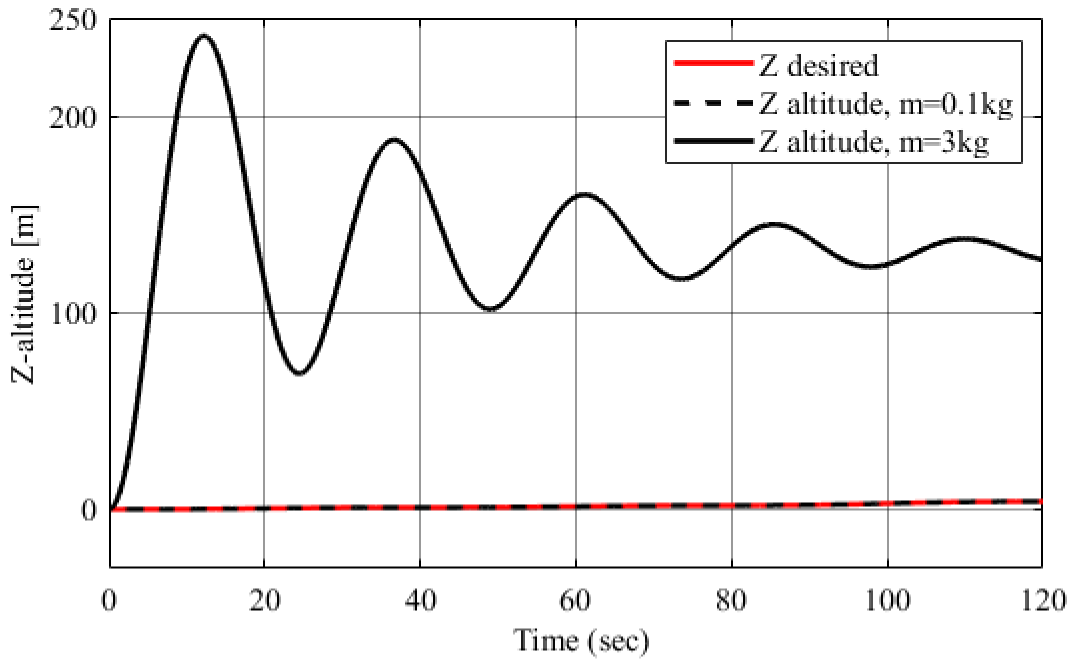


Figure 5.27 Altitude *Z* reference trajectory tracking controller performance for different values of *m* using SMC

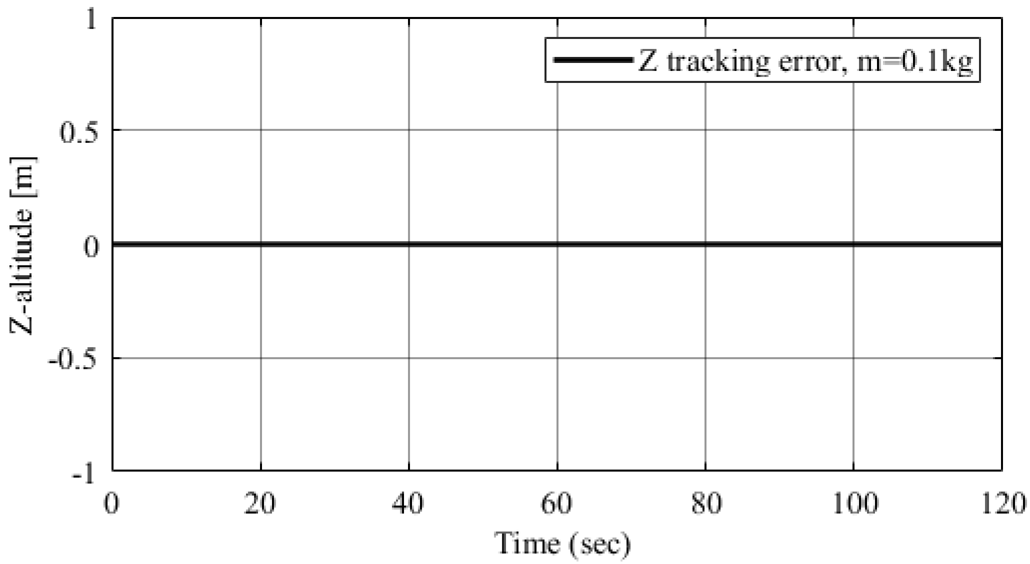


Figure 5.28 Altitude Z reference trajectory tracking error using SMC

B) Position (X) controller tracking performance: Figure 2.29 shows how SMC fails when there is parameter uncertainty in the system. When there is a variation of parameters in the system, at that value of the parameter the SMC cannot track the desired trajectory. But, for the tracking of the generated trajectory, there is a peak error of 0.15m which is shown in Figure 5.30.

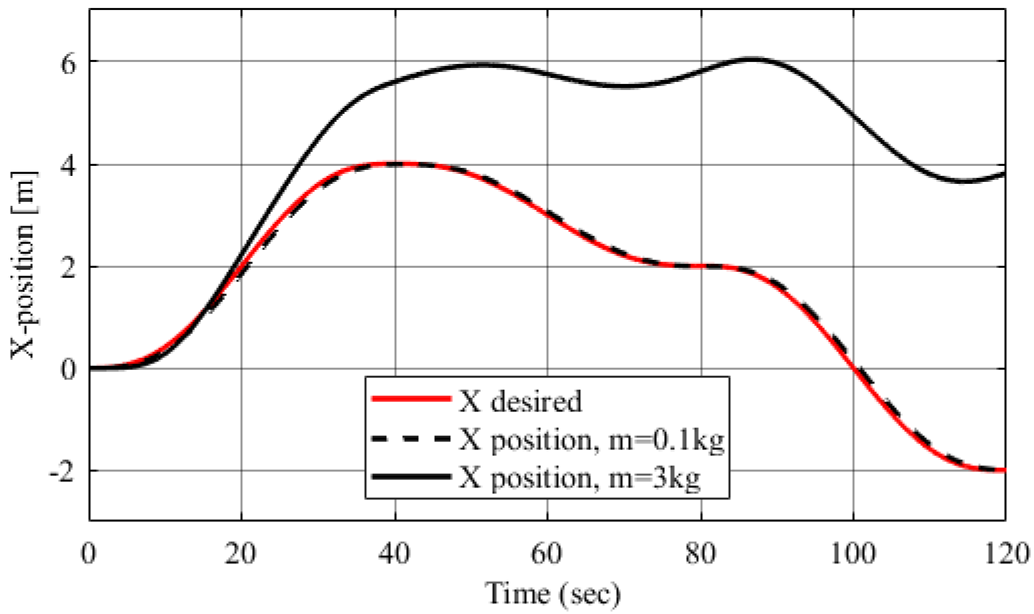


Figure 5.29 Position X reference trajectory tracking controller performance for different values of m using SMC

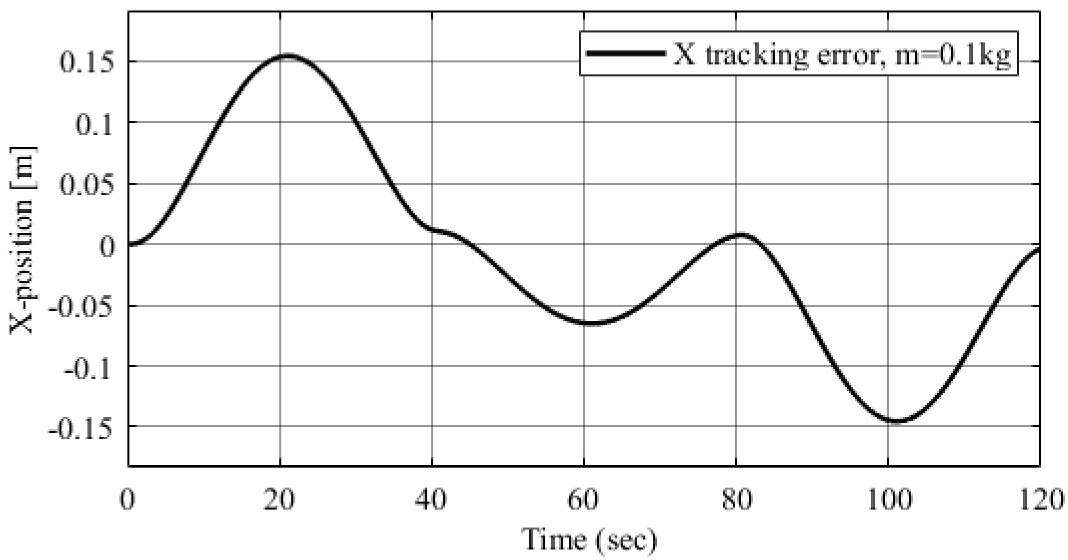


Figure 5.30 Position X reference trajectory tracking error using SMC

C) Position (*Y*) controller tracking performance: Y position tracking performance and tracking error using SMC is shown in Figure 5.31 and Figure 2.32 respectively. As the mass of the system vary from the designed value of mass that is 0.1kg, the system couldn't track the reference trajectory.

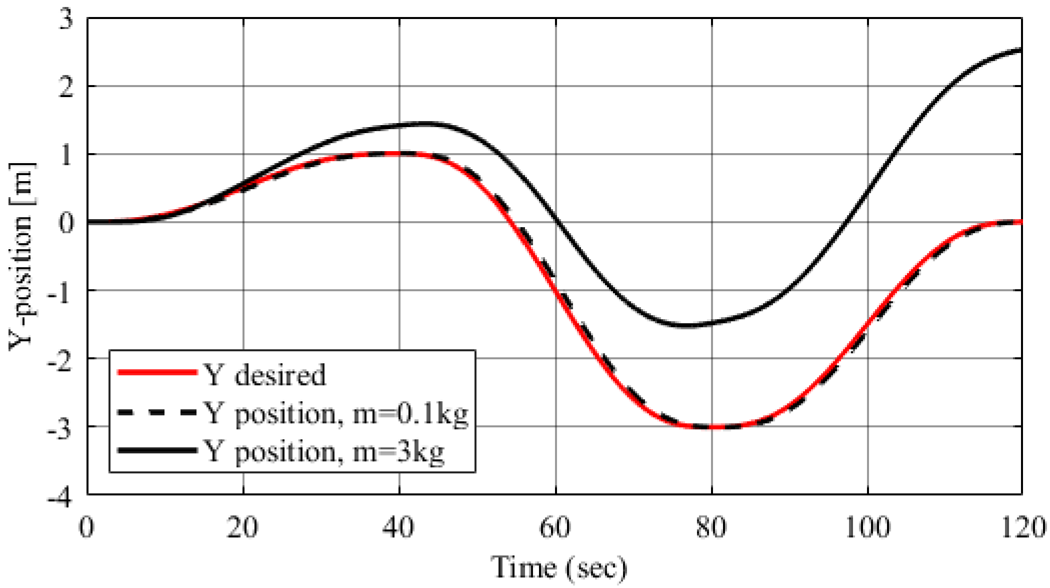


Figure 5.31 Position Y reference trajectory tracking controller performance for different values of m using SMC

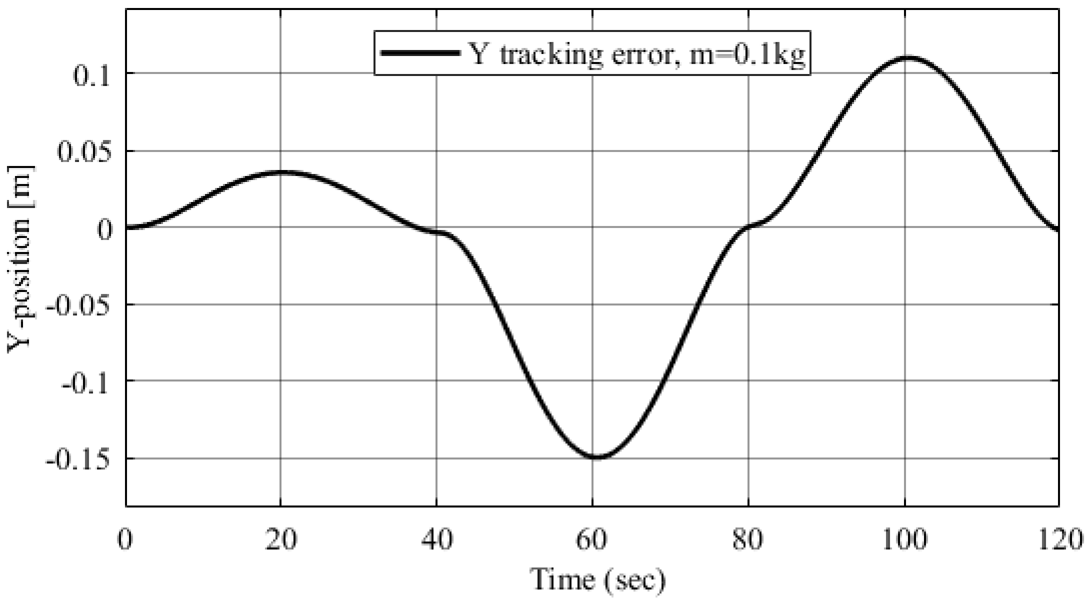


Figure 5.32 Position X reference trajectory tracking error using ASM

6. CONCLUSION AND RECOMMENDATION

6.1. Conclusion

This research work introduced the application of ASMC to quadcopter stabilization problems for minimum jerk trajectory. The nonlinear dynamic model of the quadcopter system is formulated using Euler-Newton formalization. The model contains two parts namely, translational and rotational dynamics (Euler- angle dynamics). After driving the dynamic model of the system, a minimum jerk trajectory is designed to be used as a reference trajectory and SMC & ASMC controllers are designed for the model.

To verify the performance and efficiency of the controller, a simulation is obtained using MATLAB/Simulink. Both controllers are designed for four output control variables separately and for two output variables in terms of the other two variables The controlled outputs are altitude, position, attitude, and heading.

Finally, the obtained results shows that the system tracks the desired trajectory with small tracking error. And the quadcopter back to tracking position even if there would be some parameter uncertainty (mass variation of 0.1kg-6kg) and external disturbance (wind gust disturbance) inhibiting it.

After observing the simulation results, it can be concluded that the quadcopter system is stabilized by controlling the altitudes, position, attitude, and heading of it by using ASMC.

The developed ASMC algorithm possess all the good properties of the sliding mode systems while minimizing unnecessary discontinuity of the control and thus reducing chattering.

6.2. Recommendation

For future researchers, this research work recommends the following:

- Modelling a quadcopter that tracks a reference trajectory in which only the initial points are given to it and the next waypoints are decided by the quadcopter.
- Model and control the quadcopter by considering the motor dynamics of the quadcopter.
- Hardware implementation of the modelled quadcopter.

Acknowledgements

The research for this paper was financially supported by Adama Science and Technology University under the grant number of **ASTU/SM-R/170/20**, Adama, Ethiopia.

Reference

[1] P. Pal, V. Mukherjee, H. Alemayehu, G. G. Jin, G. J. M. Feyisa, and C. i. Simulation, "Generalized adaptive backstepping sliding mode control for synchronizing chaotic systems with uncertainties and disturbances," vol. 190, pp. 793-807, 2021.

[2] G. Qi, S. Ma, X. Guo, X. Li, and J. Guo, "High-Order Differential Feedback Control for Quadrotor UAV: Theory and Experimentation," vol. 9, no. 12, p. 2001, 2020.

[3] M. K. Habib, W. G. A. Abdelaal, and M. S. Saad, "Dynamic modeling and control of a quadrotor using linear and nonlinear approaches," 2014.

[4] I. M. Khairuddin, A. P. Majeed, A. Lim, J. A. M Jizat, and A. A. Jaafar, "Modelling and PID Control of a Quadrotor Aerial Robot," in *Advanced Materials Research*, 2014, vol. 903, pp. 327-331: Trans Tech Publ.

[5] D. Giordan *et al.*, "The use of unmanned aerial vehicles (UAVs) for engineering geology applications," *Bulletin of Engineering Geology and the Environment*, vol. 79, no. 7, pp. 3437-3481, 2020/09/01 2020.

[6] P. Angelov, *Sense and avoid in UAS: research and applications*. John Wiley & Sons, 2012.

[7] H. Hou, J. Zhuang, H. Xia, G. Wang, and D. Yu, "A simple controller of minisize quad-rotor vehicle," in *2010 IEEE International Conference on Mechatronics and Automation*, 2010, pp. 1701-1706: IEEE.

[8] X. Zhang, X. Li, K. Wang, and Y. Lu, "A Survey of Modelling and Identification of Quadrotor Robot," *Abstract and Applied Analysis*, vol. 2014, p. 320526, 2014/10/20 2014.

[9] A. Basri, A. Husain, and K. A. Danapalasingam, "Backstepping Controller with Intelligent Parameters Selection for Stabilization of Quadrotor Helicopter," *Journal of Engineering Science and Technology Review*, vol. 7, pp. 66-74, 06/01 2014.

[10] M. Labbadi, M. Cherkaoui, and M. h. Guisser, "Modeling and robust integral sliding mode control for a quadrotor unmanned aerial vehicle," in *2018 6th International Renewable and Sustainable Energy Conference (IRSEC)*, 2018, pp. 1-6: IEEE.

[11] A. Noordin, M. A. M. Basri, Z. Mohamed, I. M. J. A. J. f. S. Lazim, and Engineering, "Adaptive PID controller using sliding mode control approaches for quadrotor UAV attitude and position stabilization," vol. 46, no. 2, pp. 963-981, 2021.

[12] F. Ahmad, P. Kumar, A. Bhandari, and P. P. J. M. T. P. Patil, "Simulation of the Quadcopter Dynamics with LQR based Control," vol. 24, pp. 326-332, 2020.

[13] R. Ayad, W. Nouibat, M. Zareb, Y. B. J. I. J. o. S. Sebanne, and T. o. E. E. Technology, "Full control of quadrotor aerial robot using fractional-order FOPID," vol. 43, no. 1, pp. 349-360, 2019.

[14] W. Jie, W. Xiaozhou, and W. Jinhuan, "Trajectory tracking controller design for a quadrotor aircraft based on fuzzy sliding-mode control," in *2017 36th Chinese Control Conference (CCC)*, 2017, pp. 1354-1358: IEEE.

[15] W.-T. Xue, S.-J. Tao, and X.-F. Yang, "Trajectory tracking controller design for a quadrotor aircraft based on cascade sliding mode control," in *2018 IEEE 27th International Symposium on Industrial Electronics (ISIE)*, 2018, pp. 994-999: IEEE.

[16] H. Razmi, S. J. A. S. Afshinfar, and technology, "Neural network-based adaptive sliding mode control design for position and attitude control of a quadrotor UAV," vol. 91, pp. 12-27, 2019.

[17] M. Usman, "Quadcopter Modelling and Control With MATLAB/Simulink Implementation," 2020.

[18] Ş. Tüzel, "Attitude and position control of a quadrotor using on board vision system," 2019.

[19] R. Niemiec and F. J. J. o. A. Gandhi, "Multirotor Controls, Trim, and Autonomous Flight Dynamics of Plus-and Cross-Quadcopters," vol. 54, no. 5, pp. 1910-1920, 2017.

[20] T. J. M. t. Bresciani, "Modelling, identification and control of a quadrotor helicopter," 2008.

[21] R. Beard, "Quadrotor dynamics and control rev 0.1," 2008.

[22] B. Etkin, *Dynamics of atmospheric flight*. Courier Corporation, 2012.

[23] O. QUADROTOR and M. YILDIZ, "EFFECTS OF ACTIVE LANDING GEAR ON THE ATTITUDE DYNAMICS," 2015.

[24] C. Balas, "Modelling and linear control of a quadrotor," 2007.

[25] E. Frazzoli, "Dynamic Systems and Control Lecture 6: Dynamical Systems Chapter 6," 2011.

[26] R. Shadmehr and S. Wise, "Supplementary documents for “Computational Neurobiology of Reaching and Pointing”," ed: Cambridge, MA: MIT Press, 2005.

[27] P. Freeman, *Minimum jerk trajectory planning for trajectory constrained redundant robots*. Washington University in St. Louis, 2012.

[28] J. Liu, Y. Gao, Y. Yin, J. Wang, W. Luo, and G. Sun, *Sliding mode control methodology in the applications of industrial power systems*. Springer, 2020.

[29] B. Bandyopadhyay, F. Deepak, and K.-S. Kim, *Sliding mode control using novel sliding surfaces*. Springer, 2009.

[30] J. Lee, H. Dallali, M. Jin, D. G. Caldwell, and N. G. J. A. R. Tsagarakis, "Robust and adaptive dynamic controller for fully-actuated robots in operational space under uncertainties," vol. 43, no. 4, pp. 1023-1040, 2019.

[31] V. Bobál, J. Böhm, J. Fessl, and J. Machácek, *Digital self-tuning controllers: algorithms, implementation and applications*. Springer Science & Business Media, 2006.

[32] L. Wang, C. He, P. J. I. J. o. A. Zhu, and C. Engineering, "Adaptive sliding mode control for quadrotor aerial robot with I type configuration," vol. 3, pp. 20-26, 2014.

[33] C. T. V. P. B. Morcego, "Position-Heading Quadrotor Control using LPV Techniques."

[34] H. Bouadi, S. S. Cunha, A. Drouin, and F. Mora-Camino, "Adaptive sliding mode control for quadrotor attitude stabilization and altitude tracking," in *2011 IEEE 12th International Symposium on Computational Intelligence and Informatics (CINTI)*, 2011, pp. 449-455: IEEE.

[35] F. Ahmad, P. Kumar, A. Bhandari, and P. P. Patil, "Simulation of the Quadcopter Dynamics with LQR based Control," Materials Today: Proceedings, vol. 24, pp. 326-332, 2020.

[36] S. A. Frank, "Adaptive Control," in Control Theory Tutorial(SpringerBriefs in Applied Sciences and Technology, 2018, pp. 85-89.

[37] T. Huang, D. Huang, Z. Wang, and A. Shah, "Robust Tracking Control of a Quadrotor UAV Based on Adaptive Sliding Mode Controller," Complexity, vol. 2019, pp. 1-15, 2019.

[38] E. Kuantama, I. Tarca, S. Dzitac, I. Dzitac, and R. Tarca, "Flight Stability Analysis of a Symmetrically-Structured Quadcopter Based on Thrust Data Logger Information," Symmetry, vol. 10, no. 7, 2018.

[39] H. Le Nhu Ngoc Thanh and S. K. Hong, "Quadcopter Robust Adaptive Second Order Sliding Mode Control Based on PID Sliding Surface," IEEE Access, vol. 6, pp. 66850-66860, 2018.

[40] N. Suthanthira Vanitha, L. Manivannan, T. Meenakshi, and K. Radhika, "Stability Analysis of Quadrotor using State Space Mathematical Modeling," Materials Today: Proceedings, vol. 33, pp. 4040-4043, 2020.

[41] S. Nadda and A. Swarup, "On adaptive sliding mode control for improved quadrotor tracking," Journal of Vibration and Control, vol. 24, no. 14, pp. 3219-3230, 2017.

[42] J. Xiao, "Trajectory planning of quadrotor using sliding mode control with extended state observer," Measurement and Control, vol. 53, no. 7-8, pp. 1300-1308, 2020.

[43] P. Pal, V. Mukherjee, and S. Bhakta, "Adaptive synchronisation of unified chaotic system via sliding mode control with cubic reaching law," Australian Journal of Electrical and Electronics Engineering, vol. 18, no. 2, pp. 104-117, 2021.

[44] Y. Park, W. Kim, and H. Moon, "Time-Continuous Real-Time Trajectory Generation for Safe Autonomous Flight of a Quadrotor in Unknown Environment," Applied Sciences, vol. 11, no. 7, 2021.

Reference Links

[L1] https://www.element14.com/ applications/robotics/blog/2015/03/07/building-a-quadcopter-part-3

[L2] https://www.mydronelab.com/best-pick/quadcopter-for-gopro.html

[L3] https://www.rs-online.com/designspark/tuning-a-drone-flight-control-system without- flight

[L4] https://fdocuments.in/document/video-111-vijay-kumar-prod-property-of-university- of-pennsylvania-vijay-kumar.html

Appendices

A MATLAB script for ASMC design

function [u_1,del_hat_dot]= fcn(phi,theta,ez,ez_dot,ez_int,del_hat)

kp=350; ki=400; kd=200; lamda=0.001;k=4.75;

g=9.81;

m=0.1;

s_=kp*e + ki*e_int + kd*e_dot;

del_hat_dot=lamda*s;

u_1=-m*(-g - k*s_z - del_hat)/(cos(theta)*cos(phi));

u_2=I_xx*(-k*s_phi - theta_dot*si_dot*(I_yy-I_zz) - J*omega*theta_dot- del_hat);

u_3=I_yy*(-k*s_theta - phi_dot*si_dot*(I_zz-I_xx) + J*omega*phi_dot - del_hat);

u_4=I_zz*(-k*s_theta - phi_dot*theta_dot*(I_xx -I_yy) - del_hat);

B MATLAB/SIMULINK Block Diagrams

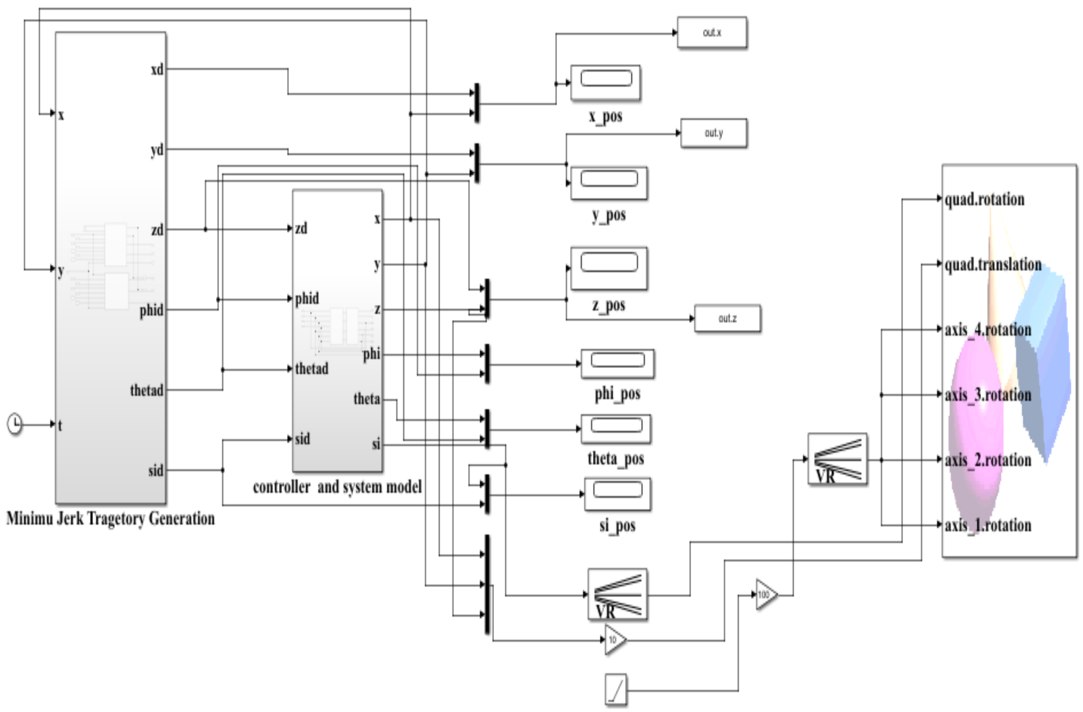


Figure B.1 Overall system

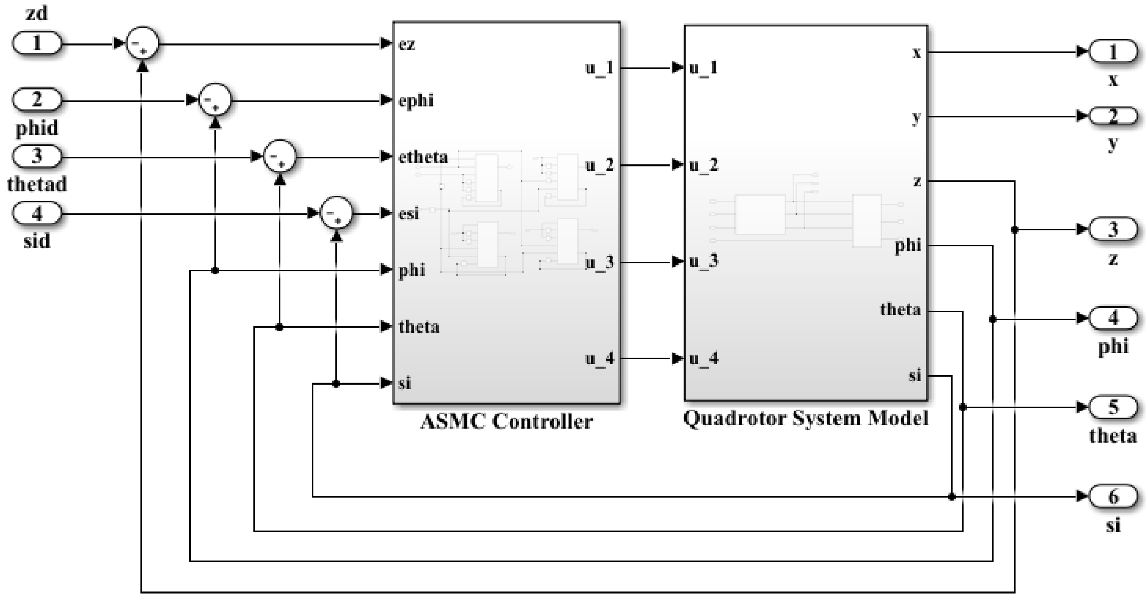


Figure B.2 ASMC and quadcopter system

C 3D Animation Using VR Sink

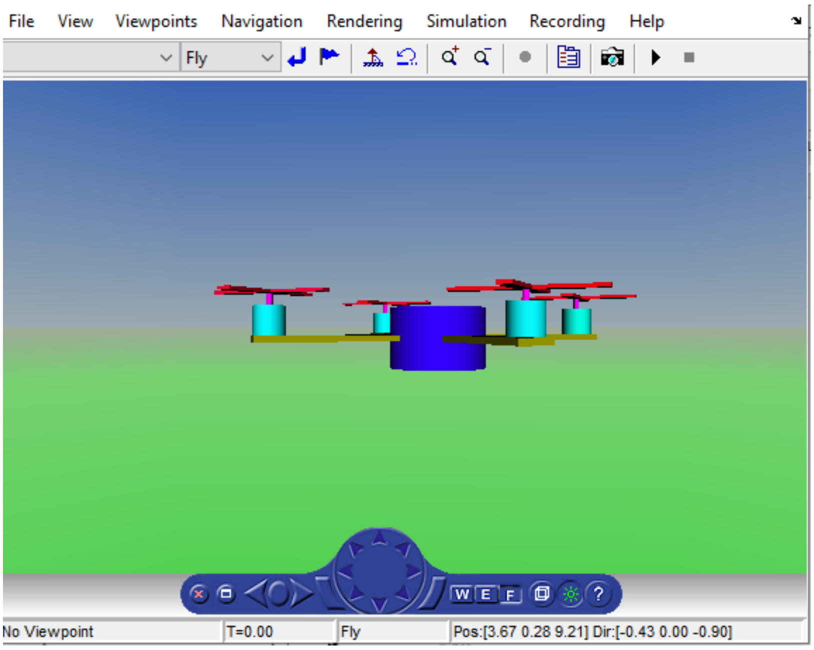


Figure B.3 Quadcopter model on VR sink