People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

University M'Hamed BOUGARA – Boumerdes

Institute of Electrical and Electronic Engineering
Department of Power and Control

Final Year Project Report Presented in Partial Fulfilment of
the Requirements for the Degree of

MASTER

In **Control**

Option: **Control**

Title:

Modelling and Control of a Quadrotor

Presented by:

- **BAITECHE Amin**
- **KENNANE Oussama**

Supervisor:

Dr. BOUSHAKI Razika

Registration Number:………/2019

Abstract

The report presents the state-of-the-art methodologies used to control Quadrotor UAVs. Prior to the discussion of the control methodologies, a detailed description of the dynamic modelling of the Quadrotor is presented. Various control strategies like the Proportional Derivative Control, the Sliding Mode Control and the Backstepping Control methods have been elucidated and implemented in MATLAB and SIMULINK. Simulations have been carried out and the results have been presented.

Acknowledgements

Before all, we would like to thank our supervisor Dr. BOUSHAKI Razika for accepting to do this project with us and for all the advice she gave us during this year. Many people have contributed and helped us in achieving this work that it would be difficult to cite them all! Special thanks to BOUGHELLABA Mouaad who helped us move forward. Finally, our thanks go to our families for their support and patience.

List of tables

Table of Contents

List of Figures

Introduction

"*A helicopter is a collection of vibrations held together by differential equations*."

John Watkinson

1. Motivations and Objectives

This work will focus on the modelling and control of a quadrotor type UAV. The reason for choosing the quadrotor is in addition to its advantages that will be addressed later, the research field is still facing some challenges in the control field because the quadrotor is a highly nonlinear, multivariable system and since it has six Degrees of Freedom (DOF) but only four actuators, it is an underactuated system [1]. Underactuated systems are those having a smaller number of control inputs compared to the system's degrees of freedom. They are very difficult to control due to the nonlinear coupling between the actuators and the degrees of freedom [2]. Although the most common flight control algorithms found in literature are linear flight controllers, these controllers can only perform when the quadrotor is flying around hover, they suffer from a huge performance degradation whenever the quadrotor leaves the nominal conditions or performs aggressive manoeuvres [3]. The contributions of this work are: deriving an accurate and detailed mathematical model of the quadrotor UAV, developing linear and nonlinear control algorithms and applying those on the derived mathematical model in computer-based simulations. The thesis will be concluded with a comparison between the developed control algorithms in terms of their dynamic performance and their ability to stabilize the system under the effect of possible disturbances.

2. State of the art

Since the advances in technologies and the ability to manufacture miniature sensors and controllers using the Micro-Electro-Mechanical Systems (MEMS) technologies, there have been a lot of advances in the UAVs area. A lot of the research conducted focused on the quadrotor due to its previously mentioned advantages of easier manufacturing, compactness and manoeuvrability among others. Some literature focused only on developing a control algorithm to be applied in a simulation environment while others developed quadrotors models to test their proposed flight algorithm on. This chapter discusses some of the most commonly used control techniques and some of the hardware platforms used in research.

2.1. Control

This project focuses on surveying existing control methods and modelling techniques with the objective of determining capabilities and effectiveness of algorithms for unmanned autonomous flight, navigation, obstacle avoidance, and performance of acrobatics. The surveyed control techniques can be fit into one of three categories: linear, nonlinear and model-free. After summarizing each controller, and its application, each control approach is categorized accordingly in the comparative table shown in Figure 1.1. The linear methods are divided into single-input single-output (SISO) methods and multi-input multi-output (MIMO) methods. Proportional-Integral-Derivative (PID) controllers fall under the SISO linear control category. MIMO linear controllers consist of linear feedback controllers, such as linear quadratic regulators (LQG) and linear quadratic Gaussian (LQG), $H_{infinity}$ controllers, and gain scheduling controllers that may utilize synthesis techniques. Nonlinear methods are divided into linearized and fully nonlinear methods. Linearized techniques start with a nonlinear model, and utilize various techniques to linearize the system dynamics, including input/output feedback linearization. Other methods can then be applied, including adaptive control, model predictive control (MPC), and nested saturation loops, backstepping control approaches utilize fully nonlinear models. Lastly, model free and learning-based methods include neural networks (NN), fuzzy logic, and human-based learning techniques. Human-based learning techniques include differential dynamic programming (DDP) and reinforcement learning.

2.1.1. Linear flight control systems

They are the most common and conventional flight control systems, typically based on PID, Linear Quadratic (LQ) or $H_{infinity}$ algorithms. It was reported that in the late 1960's, a full-scale helicopter achieved autonomous waypoint navigation using a classical linear control technique [3] .**PID and LQ** Bouabdallah et al. proposed the usage of PID and LQ control techniques to be applied on an indoor micro quadrotor, it was found out that these two types of controllers performed comparably and were able to stabilize the quadrotor's attitude around its hover position when it undergoes little disturbances [4] [5]. Li and Yuntang used the classical PID to control the position and orientation of a quadrotor and it was able to stabilize in a low speed wind environment [6] Yang et al. used a self-tuning PID controller based on adaptive pole placement to control the attitude and heading of a quadrotor. Simulation showed that the proposed controller performed well with online tuning of the parameters [7].$H_{infinity}$ Raffo et al. used an $H_{infinity}$ controller to stabilize the rotational angles together with a Model Predictive Controller (MPC) to track the desired position [8]. The effect of wind and model uncertainties was added to the simulated model and it performed robustly with a zero steady-state error. $H_{infinity}$ is a linear robust controller; robust controllers are those parametric uncertainty and unmodeled dynamics. It is reported that it is used for control of full-scaled helicopters [3] **Switched Dynamics and Gain Scheduling** To use a linear controller to control a nonlinear system like a quadrotor, the nonlinearity of the system can be modelled as a collection of simplified linear models. This is the concept of gain scheduling and it is commonly used to design flight controllers. Gillula et al. divided the state space model of a STARMAC quadrotor to a set of simple hybrid modes and this approach enabled the quadrotor to carry out aerobatic maneuvers [9] [3]Also, Ataka et al. used gain scheduling on a linearized model of the quadrotor around some equilibrium points and tested the controllability and observability of the resulting system [10]. Amoozgar et al. used a gain scheduled PID controller with the latter's parameters tuned using a fuzzy logic inference scheme to control a quadrotor. The system was tested under actuator fault conditions and compared with the performance of the conventional PID controller. The results showed a better performance for the gain scheduled PID controller [11] Sadeghzadeh et al. also use a gain scheduled PID controller applied to a quadrotor dropping a

carried payload at a designated time. The algorithm was able to stabilize the system during the dropping operation [12].

2.1.2. Non-linear flight control systems

Due to the fact that the dynamics of the quadrotor is of a nonlinear nature, developing nonlinear control algorithms to be used as flight controllers was necessary. There is a variety of nonlinear control algorithms applied to quadrotors including: feedback linearization, model predictive control, backstepping and sliding-mode. **Backstepping and Sliding-mode** Backstepping is a recursive control algorithm that can be applied to both linear and nonlinear systems [3]. In a more recent paper, Bouabdallah and Siegwart proposed the use of backstepping and sliding-mode nonlinear control methods to control the quadrotor which gave better performance in the presence of disturbances [13]. Waslander et al. proposed developing controllers that can stabilize the quadrotor in an outdoors environment, they compared the performance of an integral sliding-mode controller vs. a reinforcement learning controller. They reached a conclusion that both controllers were able to stabilize the quadrotor outdoors with an improved performance over classical control techniques [14]. Madani and Benallegue used a backstepping controller based on Lyapunov stability theory to track desired values for the quadrotor's position and orientation. They divided the quadrotor model into 3 subsystems: underactuated, fully actuated and propeller subsystems. Their proposed algorithm was able to stabilize the system under no disturbances [15]. Fang and Gao proposed merging a backstepping controller with an adaptive controller to overcome the problems of model uncertainties and external disturbances. The proposed adaptive integral backstepping algorithm was able to reduce the system's overshoot and response time and eliminate steady state error [16] .Lee et al. used a backstepping controller to control the position and attitude of a quadrotor, the proposed controller was tested in a noisy environment and gave a satisfactory performance [17]. Zhen et al. combined a backstepping controller with an adaptive algorithm to control the attitude of a quadrotor. A robust adaptive function is used to approximate the external disturbances and modeling errors of the system. Simulations showed the success of the proposed controller in overcoming disturbances and uncertainties [18]. Gonz´alez et al. proposed using a chattering free sliding mode controller to control the altitude of a quadrotor. The proposed controller performed well in both simulations and on a real system in the presence of disturbances [19]. **Feedback Linearization** Feedback linearization is a control

techniques that uses a nonlinear transformation between the system's nonlinear state variables to linear ones. Linear algorithms can be then used to stabilize the transformed linear system which will then be inversely transformed back into the original state variables. Kendoul et al. was able to control a quadrotor in several flight tests based on the concept of feedback linearization [20] [3].**Model Predictive Control (MPC)** Model predictive control relies on predicting the future states of the system and tracking the error to give an improved performance [3].Alexis et al. relied on a MPC to control the attitude of a quadrotor in the presence of atmospheric disturbances. The proposed algorithm behaved well in performing rough manoeuvres in a wind induced environment as was able to accurately track the desired attitude [21].Sadeghzadeh et al. also used a MPC applied to a quadrotor in dropping a carried payload, the MPC was able to stabilize the system with a promising performance [12].

2.1.3. Learning based flight control systems

Opposing the previous control techniques, learning based flight control systems do not need a precise and accurate dynamic model of the system to be controlled. On the other hand, several trials are carried out and flight data are used to \train" the system. There are many types of learning-based flight control systems, the most widely used are: neural networks, fuzzy logic and human-based learning. **Neural Network** Efe used a Neural Network to simplify the design of a PID controller and decrease the computational time and complexity [22] **Fuzzy Logic** The idea behind fuzzy logic is to translate the knowledge and actions of skilled human beings to a set of rules that can be used by a machine to execute a certain task usually executed by humans. So, for flight control systems, a skilled pilot is usually the one doing the training for a fuzzy logic system [3].

2.1.4. Hybrid flight control systems

In recent literature, it was found out that using only one type of flight control algorithms was not sufficient to guarantee a good performance specially when the quadrotor is not flying near its nominal condition, so researchers are now proposing using more than one type of flight control algorithms. Azzam and Wang used a PD controller for altitude and yaw rotation and a PID controller integrated with a backstepping controller for the pitch and roll control. An optimization algorithm was used instead of the pole placement technique to overcome the difficulty of pole placement in a nonlinear time variant system. The system was divided into rotational and translation subsystems where the translation subsystem stabilizes the quadrotor position in flight and generates the needed roll and pitch angles to be fed to the rotational subsystem [23].Nagaty et al. proposed the usage of a nested loop control algorithm; the outer loop consists of a PID controller responsible for the generation of the desired attitude angles that would achieve the desired position. These attitude angles are then fed to the inner loop. The inner loop stabilization controller relies on the backstepping algorithm to track the desired altitude, attitude and heading [24].

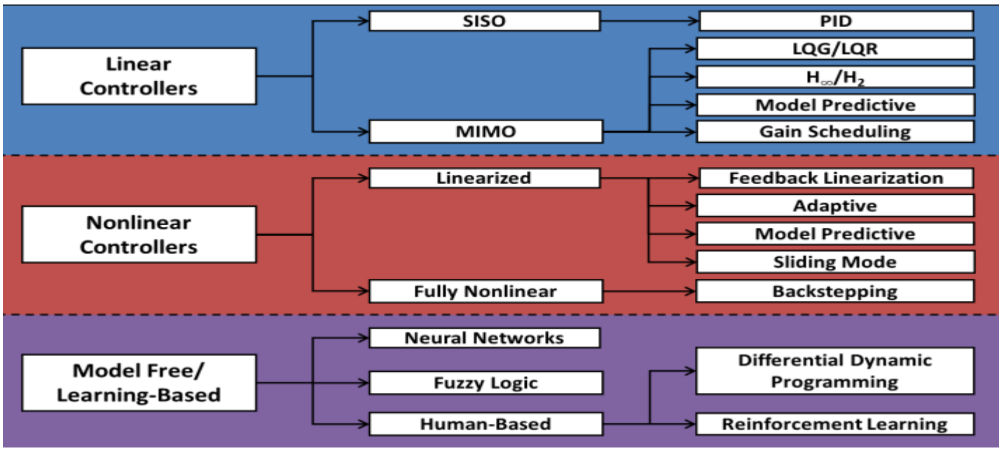


Figure 1.1 Control techniques

3. Project structure

This thesis is organized as follow, Chapter 1 presents the mathematical modelling of a quadrotor UAV based on the Newton-Euler formalism. Chapter 2,3,4 shows three developed control techniques to control and stabilize the attitude, heading altitude and position of the quadrotor in space. The controllers are verified using computer simulations and the results of these simulations are shown. Finally, a conclusion and discussion of the results acquired in the previous chapters is presented in the conclusion.

Chapter 1 Quadrotor dynamics

In this chapter, the kinematics and dynamics models of a quadrotor will be derived based on a Newton-Euler formalism with the following assumptions:

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

The model used in this work is a simple one with no Coriolis forces or aerodynamic non linearities like drag, blade flapping and ground effects. In controlled hover like conditions these nonlinearities can be assumed to be absent. The main objective of this work is to present different control technics currently used by researchers to control small UAVs, so the controller output is directly fed into the dynamic model without making any mapping in the actuator space

In this chapter, the dynamics of a quadrotor using the Newton-Euler formalism is introduced. The motivation is derived from the work of Mellinger [25].

NEWTON EULER EQUATION

$$\begin{bmatrix} F \\ \tau \end{bmatrix} = \begin{bmatrix} m1_3 & 0_3 \\ 0_3 & I_3 \end{bmatrix} \begin{bmatrix} \mathrm{a} \\ \alpha \end{bmatrix} + \begin{bmatrix} 0 \\ \omega \times I_3\omega \end{bmatrix} \tag{1.1}$$

In the Equation 1.1 the symbols have the following meaning, F is the net force acting on the quadrotor, $\mathbf{1}_3$ is a 3 × 3 identity matrix, **a** is the linear acceleration of the centre of mass, w is the angular velocity of the robot, m is the mass, $\mathbf{I}_3$ is the moment of inertia, τ is the net torque and α is the angular acceleration. Now we will write the linear and the angular equations of motion separately. For transforming coordinated from body frame to the world frame we use the ZXY Euler angles. It is given in the Equation below:

$$W_{R_B} = \begin{bmatrix} c\Psi c\Theta - \mathrm{s}\Phi\mathrm{s}\Psi s\Theta & -c\Phi\mathrm{s}\Psi & c\Psi s\Theta + \mathrm{c}\Theta\mathrm{s}\Phi\mathrm{s}\Psi \\ s\Psi c\Theta + \mathrm{s}\Phi\mathrm{c}\Psi\mathrm{s}\Theta & c\Phi\mathrm{c}\Psi & s\Psi s\Theta - \mathrm{c}\Psi c\Theta\mathrm{s}\Phi \\ -\mathrm{c}\Phi\mathrm{s}\Theta & \mathrm{s}\Phi & \mathrm{c}\Phi\mathrm{c}\Theta \end{bmatrix} \tag{1.2}$$

Here, Φ is the angle of rotation about the x axis (roll angle), θ is the angle of rotation about the y axis (pitch angle) and Ψ is the angle of rotation about the z axis. We shall write the dynamics of linear motion in the world frame $\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3$ and the dynamics of angular motion in the body frame of reference $\mathbf{b}_1\mathbf{b}_2\mathbf{b}_3$. Refer Figures 1.1 and 1.2

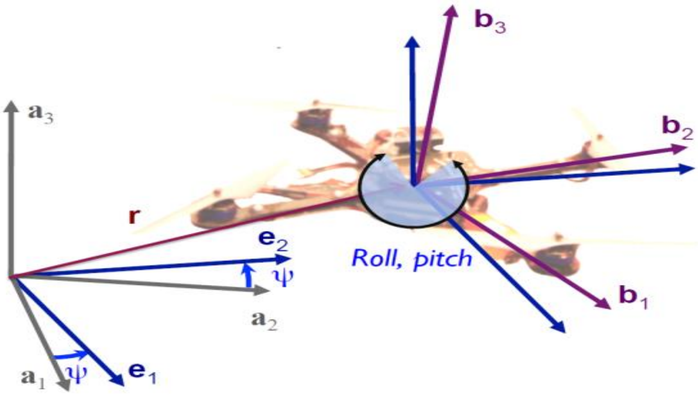


Figure 1.1 Free Body Diagram of the Quadrotor

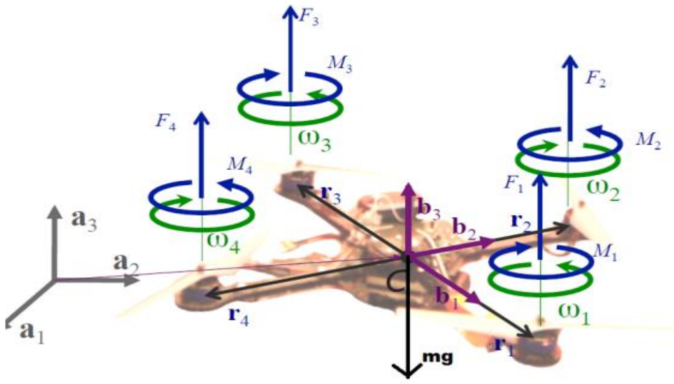


Figure 1.2 Inertial frame & Body frame

1. Linear Motion Equation in World Frame $\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3$

$$m\ddot{r} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + W_{R_B} \begin{bmatrix} 0 \\ 0 \\ F_1 + F_2 + F_3 + F_4 \end{bmatrix} \tag{1.3}$$

2. Angular Motion Equation in the Body Frame $\mathbf{b}_1\mathbf{b}_2\mathbf{b}_3$

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \tau_{bx} \\ \tau_{by} \\ \tau_{bz} \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{1.4}$$

τ_{bx} is the torque about the body-x axis b_1, τ_{by} is the torque about the body-y axis b_2 and τ_{bz} is the torque about the body-z axis b_3. It can be seen in the Figure 1.1 that the rotation about b_1 is possible due to the difference between thrusts exerted by the rotors numbered 2 and 4. Similarly, the rotation about b_2 is possible due the difference between the thrusts exerted by the rotors numbered 1 and 3 and the rotation about the axis b_3 can be brought about by changing the motor torques which changes the drag moments exerted on the quadrotor. With this information, the Equation 1.4 can be written as Equation 1.5.

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} L(F_2 - F_4) \\ L(F_3 - F_1) \\ M_2 + M_4 - M_1 - M_3 \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{1.5}$$

L is the arm length and $[p\ q\ r]^T$ is the angular velocity vector in the body frame. The rate of change of roll, pitch and yaw angles can be found from the knowledge of $[p\ q\ r]^T$ using the Equation 1.6.

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} c\Theta & 0 & -c\Phi s\Theta \\ 0 & 1 & s\Phi \\ s\Theta & 0 & c\Phi c\Theta \end{bmatrix} \begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} \rightarrow \begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} c\Theta & 0 & s\Theta \\ s\Theta t\Phi & 1 & -c\Theta t\Phi \\ -\frac{s\Theta}{c\Phi} & 0 & \frac{c\Theta}{c\Phi} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{1.6}$$

It should be noted that in this model, the aerodynamic effects have been neglected for the sake of simplification. The dynamic model is now established in Equations 1.3 and 1.5. It is known as that the control inputs are the rotor thrusts F_i and the rotor drag moments M_i and these quantities depend on the rotor speed. To adopt a modular approach in controller design, we write the Equations 1.3 and 1.5 as:

$$m\ddot{r} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + w_{R_B} \begin{bmatrix} 0 \\ 0 \\ u_1 \end{bmatrix} \tag{1.7}$$

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = u_2 - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad u_2 = \begin{bmatrix} L(F_2 - F_4) \\ L(F_3 - F_1) \\ M_2 + M_4 - M_1 - M_3 \end{bmatrix} \&\& \; u_1 = F_1 + F_2 + F_3 + F_4 \tag{1.8}$$

The formulated quadrotor model will be used in open-loop simulations to verify the mathematical model, an open loop simulation was carried out using MATLAB/Simulink. The quadrotor's parameters were taken from Bouabdallah's PhD thesis which is based on the OS4 hardware [26]. The block diagram for the simulation, the desired inputs and their corresponding unstable outputs is shown respectively in Figure 1.3, Figure 1.4 and Figure 1.5.

Table 1.1 parameters used throughout the project

m(mass)	0.18 kg
I (Inertia matrix)	[0.00025 0 2.55$e-6$ 0 0.000232 0 2.55$e-6$ 0 0.0003738]
g (gravity)	9.8 m/s^2

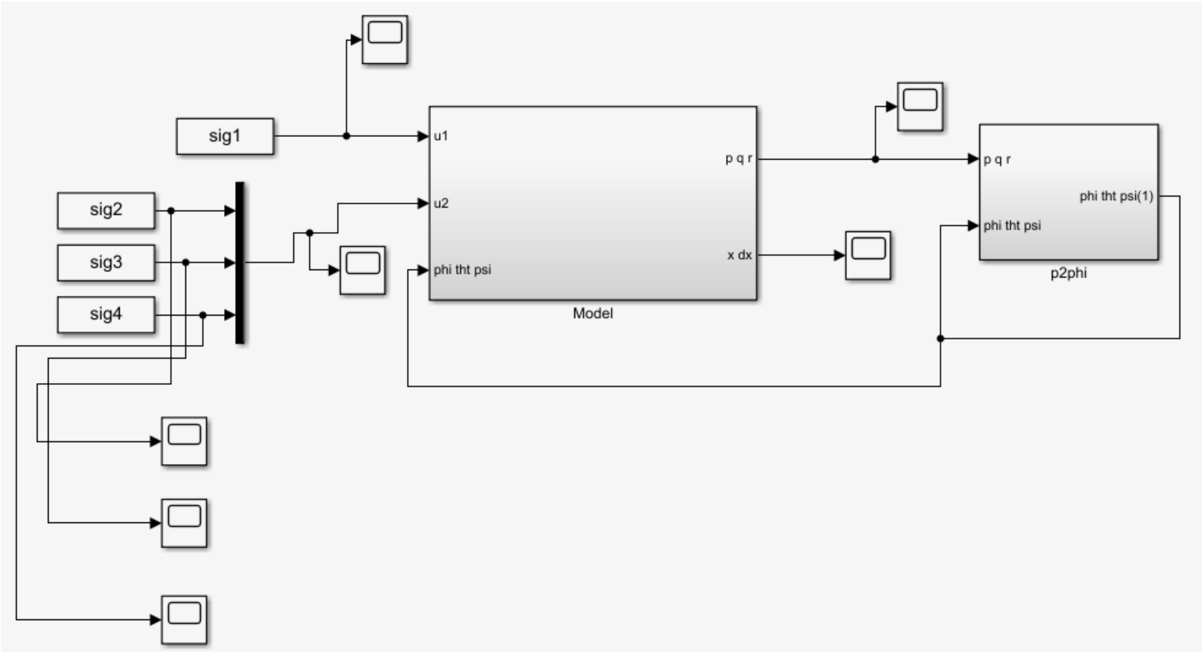


Figure 1.3 Block diagram for the open loop simulation

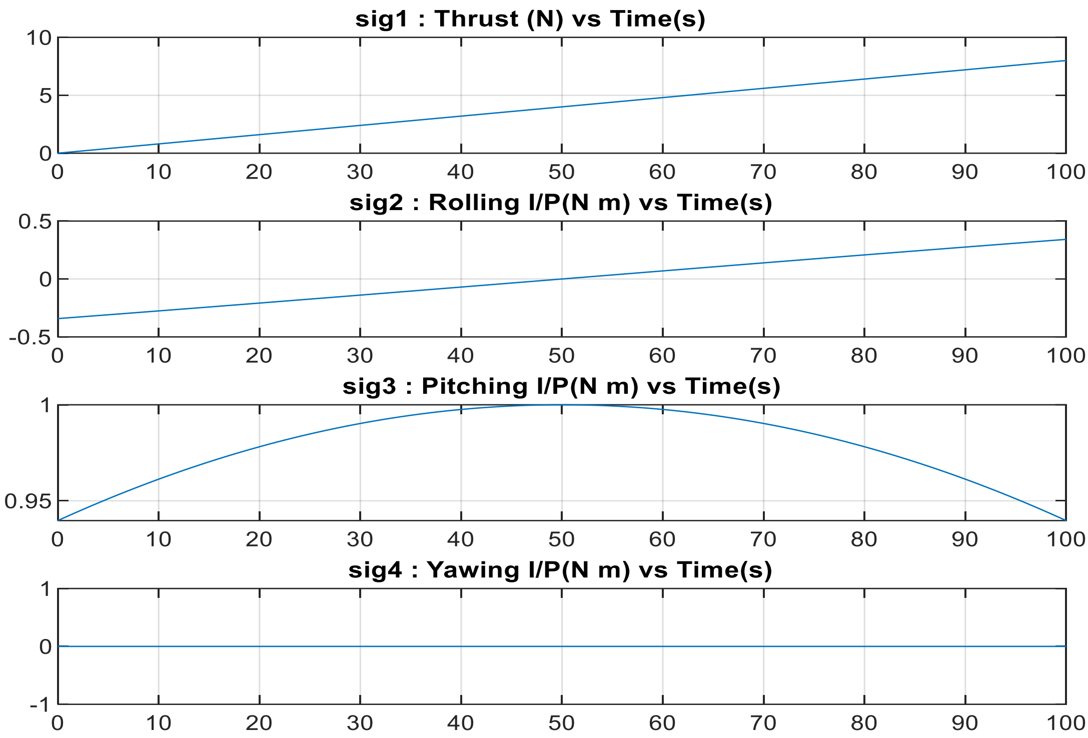


Figure 1.4 The desired trajectory for the open loop simulation

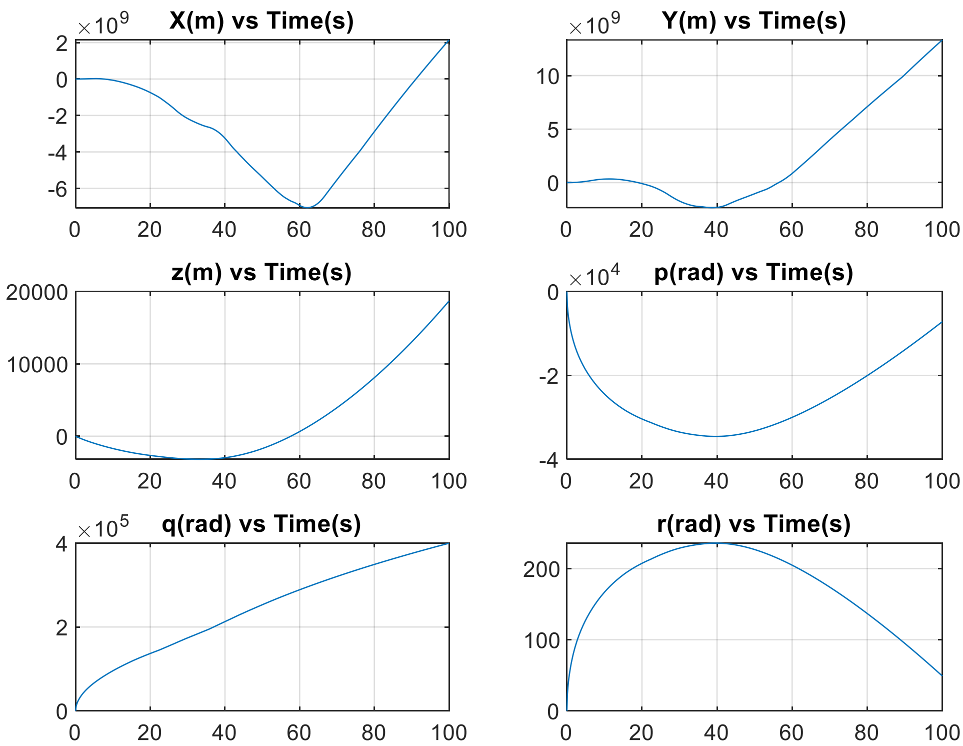


Figure 1.5 outputs of the open loop simulation.

After the derived mathematical model of the quadrotor was verified using the open loop simulation, the simulation environment will be extended to include an altitude, attitude, heading and position controllers in closed loop to stabilize and to track the desired helical trajectory.

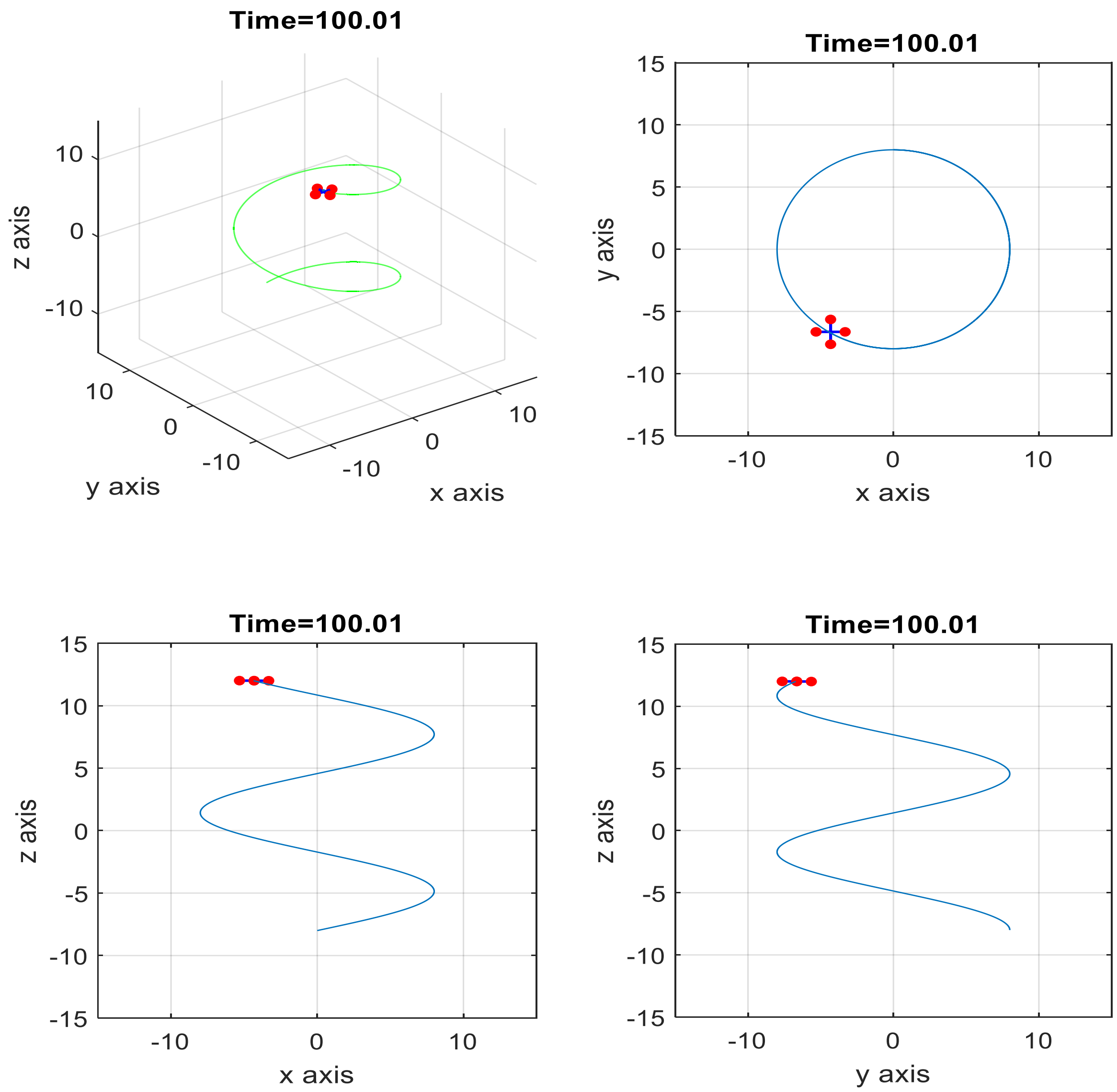


Figure 1.6 the helical Trajectory being tracked in 3D and with other perspectives

In the following chapter we're going to present one of the control techniques basically PD controller.

Chapter 2 Proportional derivative control

After the mathematical model of the quadrotor along with its open loop simulation are verified, a PID controller was developed. The PID controller generates the desired control inputs for the quadrotor. A PD control is used instead of PID for the sake of simplicity. The block diagram for a PID controller is shown in Figure 2.1 where Ki = 0 in that case.

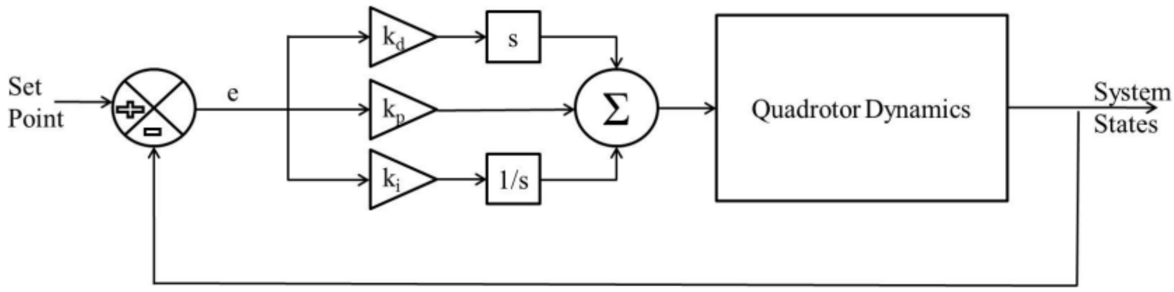


Figure 2.1 PID controller block diagram

2.1 Introduction of Proportional derivative control

The Proportional Derivate Control is one of the simplest linear control laws. It is very simple and computationally efficient. It can be easily implemented in real time systems using microcontrollers. The mathematical simplicity and ease of understanding makes it one of the most used control laws in Aerial Robotics.

2.2 Mathematics of PD Control of Quadrotors

The dynamic equations of the Quadrotor are given in Equations 2.1 and 2.2. The Equation 2.1 concerns the dynamics of linear equation and the Equation 2.2 concerns the dynamics of angular motion.

$$m\ddot{r} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + w_{R_B} \begin{bmatrix} 0 \\ 0 \\ u_1 \end{bmatrix} \tag{2.1}$$

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = u_2 - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.2}$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} c\Theta & 0 & -c\Phi s\Theta \\ 0 & 1 & s\Phi \\ s\Theta & 0 & c\Phi c\Theta \end{bmatrix} \begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} \rightarrow \begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} c\Theta & 0 & s\Theta \\ s\Theta t\Phi & 1 & -c\Theta t\Phi \\ -\frac{s\Theta}{c\Phi} & 0 & \frac{c\Theta}{c\Phi} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.3}$$

$[p; q; r]$	body angular accelerations measured by the gyroscope.
$[\Phi; \Theta; \Psi]$	Roll, Pitch and Yaw angles.
m	system mass.
I	system moment of inertia.
u_1	The thrust input.
u_2	The moment input (3 × 1 vector).

$$W_{R_B} = \begin{bmatrix} c\Psi c\Theta - s\Phi s\Psi s\Theta & -c\Phi s\Psi & c\Psi s\Theta + c\Theta s\Phi s\Psi \\ s\Psi c\Theta + s\Phi c\Psi s\Theta & c\Phi c\Psi & s\Psi s\Theta - c\Psi c\Theta s\Phi \\ -c\Phi s\Theta & s\Phi & c\Phi c\Theta \end{bmatrix}$$

The desired trajectory is $\boldsymbol{r}_T = \begin{bmatrix} \boldsymbol{x}_{des} \\ \boldsymbol{y}_{des} \\ \boldsymbol{z}_{des} \\ \Psi_{des} \end{bmatrix}$

Let us define: $e_p = \boldsymbol{r}_T - r$ and $e_v = \dot{\boldsymbol{r}}_T - \dot{\boldsymbol{r}}$

and since we've a tracking error problem we want:

$$\ddot{\boldsymbol{r}}_T - \ddot{\boldsymbol{r}}_c + k_{d,r} e_v + k_{p,r} e_p = 0$$

Here $\ddot{r}_c$: is the commanded acceleration, calculated by the controller. We design the control for hovering and linearize the dynamics at the hover configuration, where we have:

$u_1 \approx mg$ $\quad \Theta \approx 0$ $\quad \Phi \approx 0$ $\quad \Psi \approx \Psi_0$ $\quad u_2 \approx 0$ $\quad p \approx 0$ $\quad q \approx 0$ $\quad r \approx 0$

Using these approximations and all the previous data, one can deduce the following equations:

$$\Phi_c = \frac{1}{g}\left(\ddot{r}_{1,c} sin(\Psi_{des}) - \ddot{r}_{2,c} cos(\Psi_{des})\right)$$

$$\Theta_c = \frac{1}{g}\left(\ddot{r}_{1,c} cos(\Psi_{des}) + \ddot{r}_{2,c} sin(\Psi_{des})\right) \tag{2.4}$$

$$\Psi_c = \Psi_{des}$$

$$\ddot{r}_{3,c} = \ddot{r}_{3,des} + k_{d,3}(\dot{r}_{3,des} - \dot{r}_3) + k_{p,3}(r_{3,des} - r_3)$$

The control laws can be written as:

$$u_1 = m\,(g + \ddot{r}_{3,c}) \tag{2.5}$$

$$u_2 = I \begin{bmatrix} k_{p,\varphi}(\varphi_c - \varphi) + k_{d,\varphi}(p_c - p) \\ k_{p,\theta}(\theta_c - \theta) + k_{d,\theta}(q_c - q) \\ k_{p,\psi}(\psi_c - \psi) + k_{d,\theta}(r_c - r) \end{bmatrix} \tag{2.6}$$

Using Equation 1.3 and 1.4, one can get $[p_c, q_c, r_c]^T$. Also note that $[r_1 \quad r_2 \quad r_3]^T = [x \quad y \quad z]^T$

2.3 Results

The objective of the control is to track a Helical Trajectory. The Figure 2.2 shows the SIMULINK model of the Quadrotor with the position and attitude control blocks in closed loop. The system has been tested with and without disturbances. The disturbance is wind with velocity vector $V_w = 5i + 5j + 5k\ m.s^{-1}$

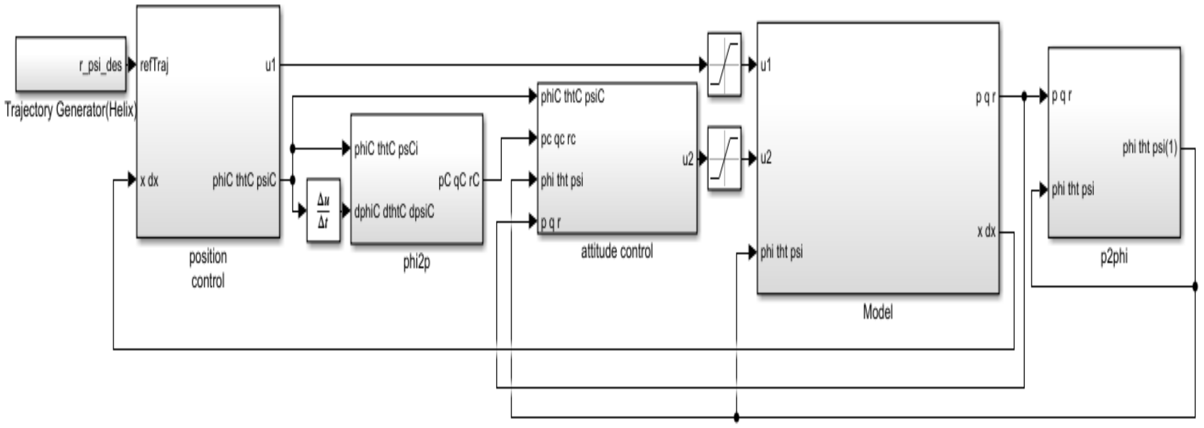


Figure 2.2 Global view of the Simulink model of the system with PD Control

Table 2.1 Tuning parameters for PD

With respect to	Kp	Kd
X	4	4
Y	4	4
Z	25	10
Phi	625	50
Theta	625	50
Psi	625	50

2.3.1 Results without disturbance

Without any disturbance, the tracking by PD control is excellent and the tracking error is negligible. The Figures 2.3, 2.4, 2.5 show the position and orientation, control inputs and the trajectory error in the absence of disturbances.

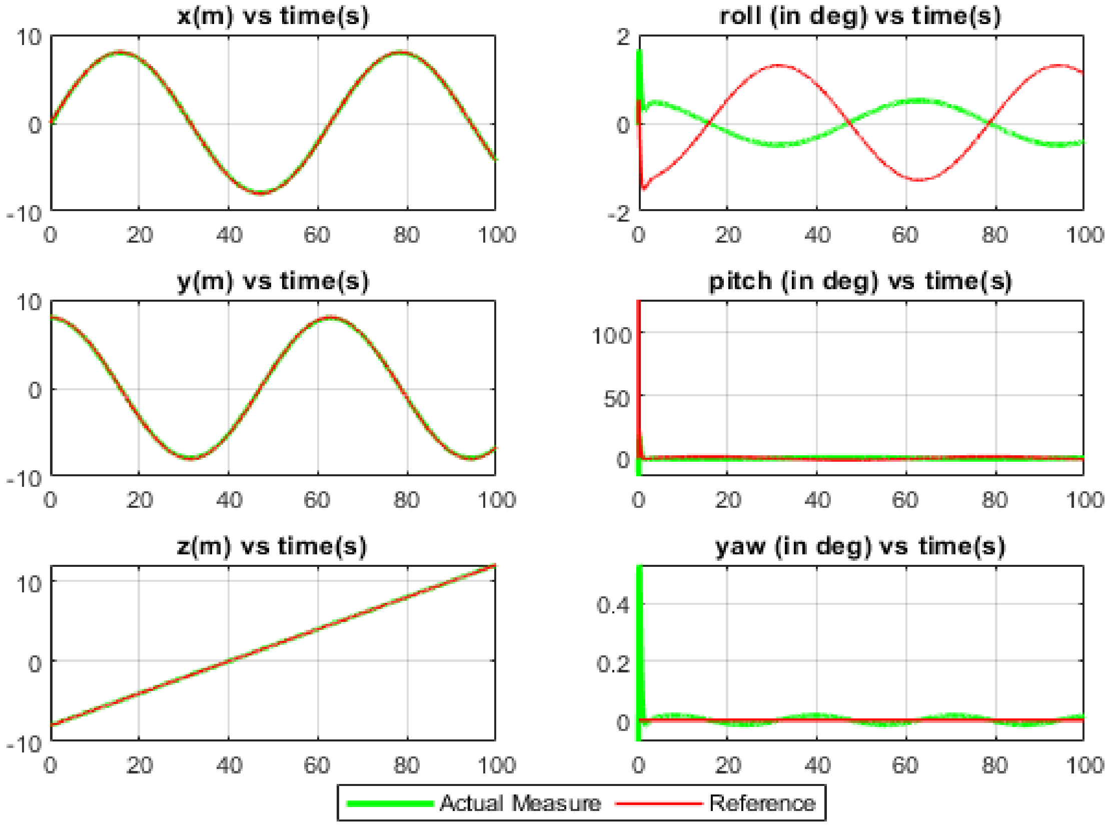


Figure 2.3 Position and Orientation vs Time

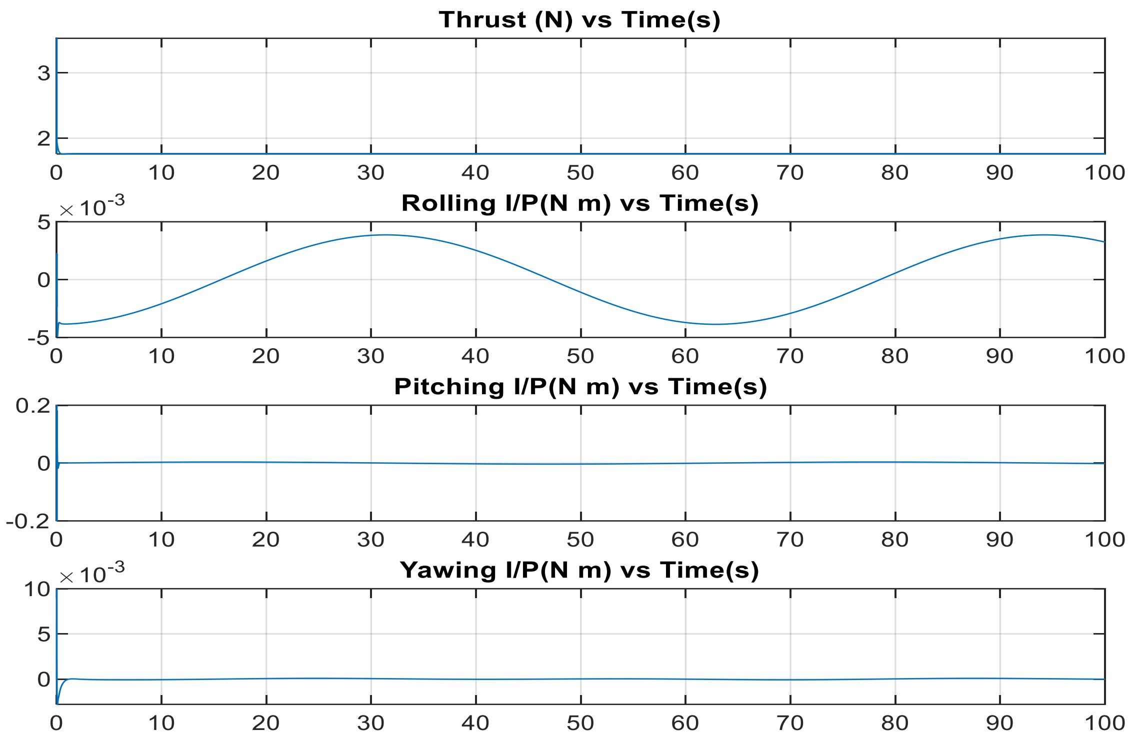


Figure 2.4 Thrust, Rolling, Pitching and Yawing Inputs vs Time

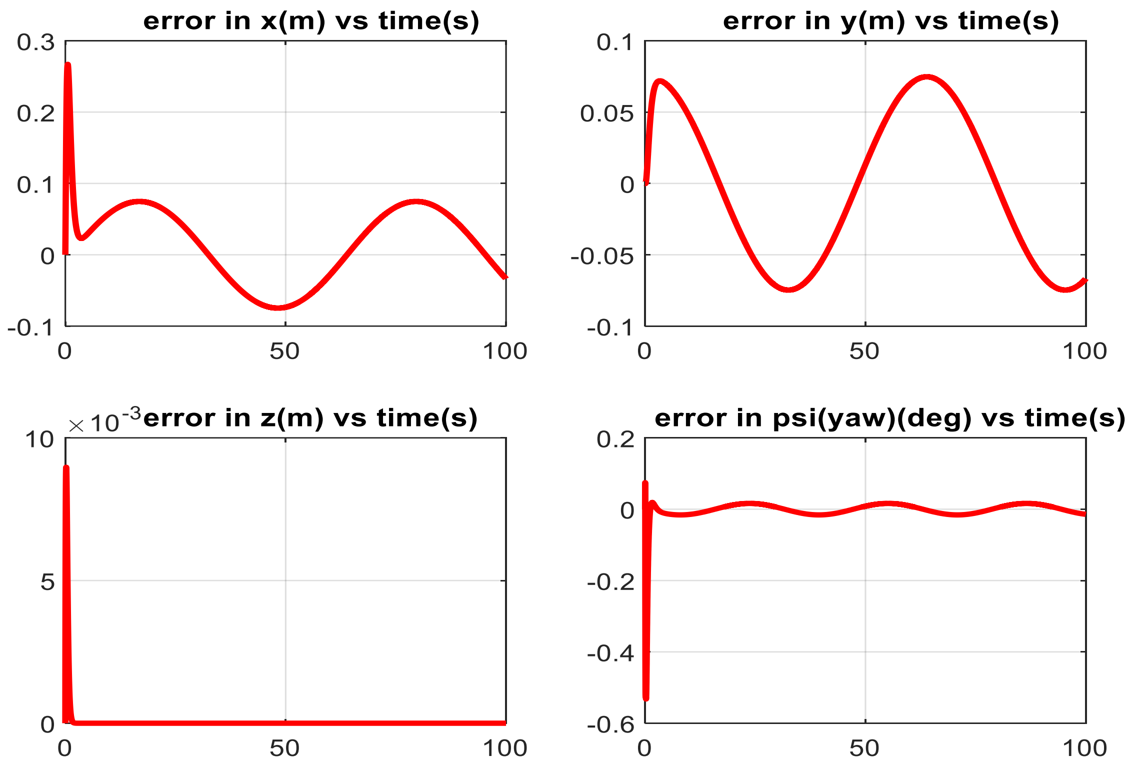


Figure 2.5 Errors in x,y,z and yaw

From figures 2.3, 2.4,2.5 the results were satisfying “small error”

2.3.2 Results with Disturbances

The disturbance is wind of velocity $\boldsymbol{V_w} = 5i + 5j + 5k$ $m.s^{-1}$ is applied as a step input at time t = 25 s. The Figures 2.6, 2.7, 2.8 show the position and orientation, control inputs and the trajectory error in the presence of disturbances.

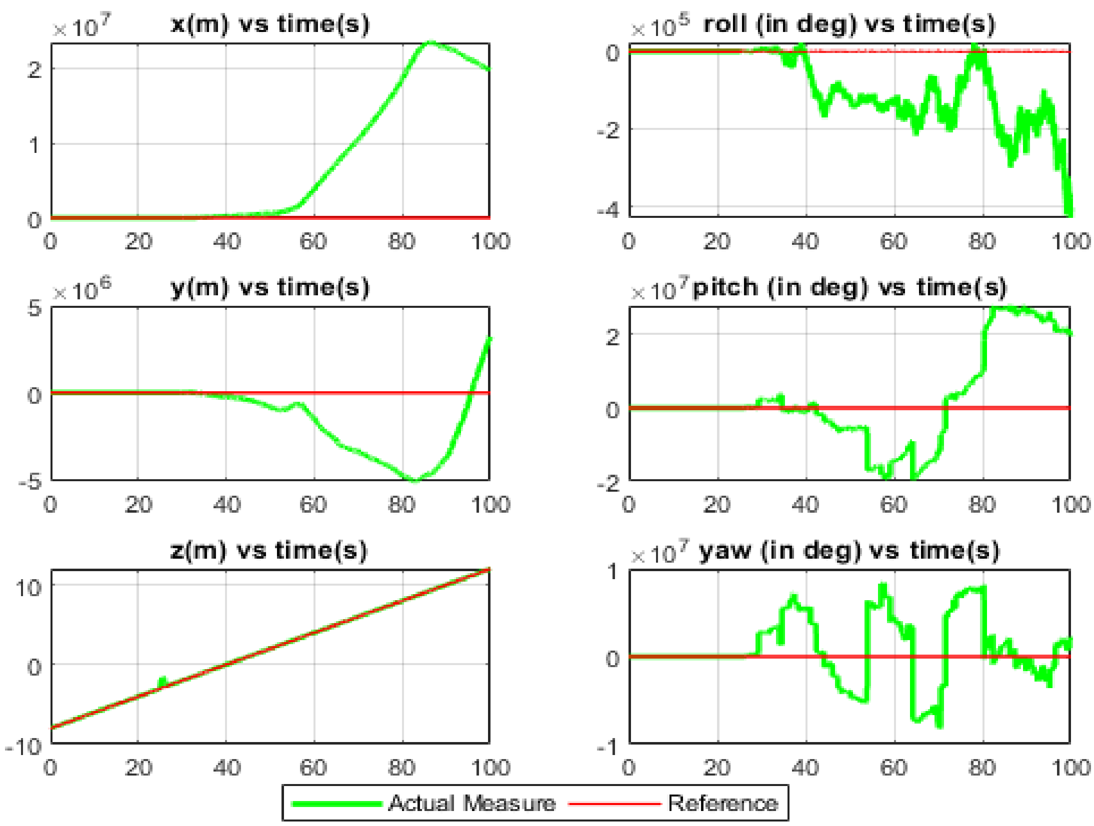


Figure 2.6 Position and Orientation vs Time [With Disturbance at 25s]

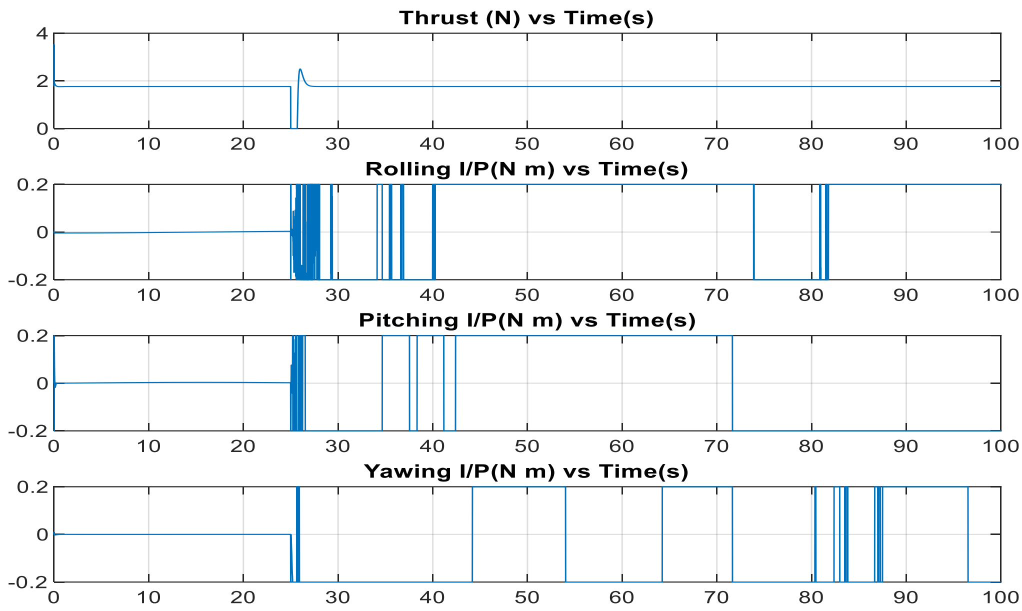


Figure 2.7 Thrust, Rolling, Pitching and Yawing Inputs vs Time [With Disturbance at 25s]

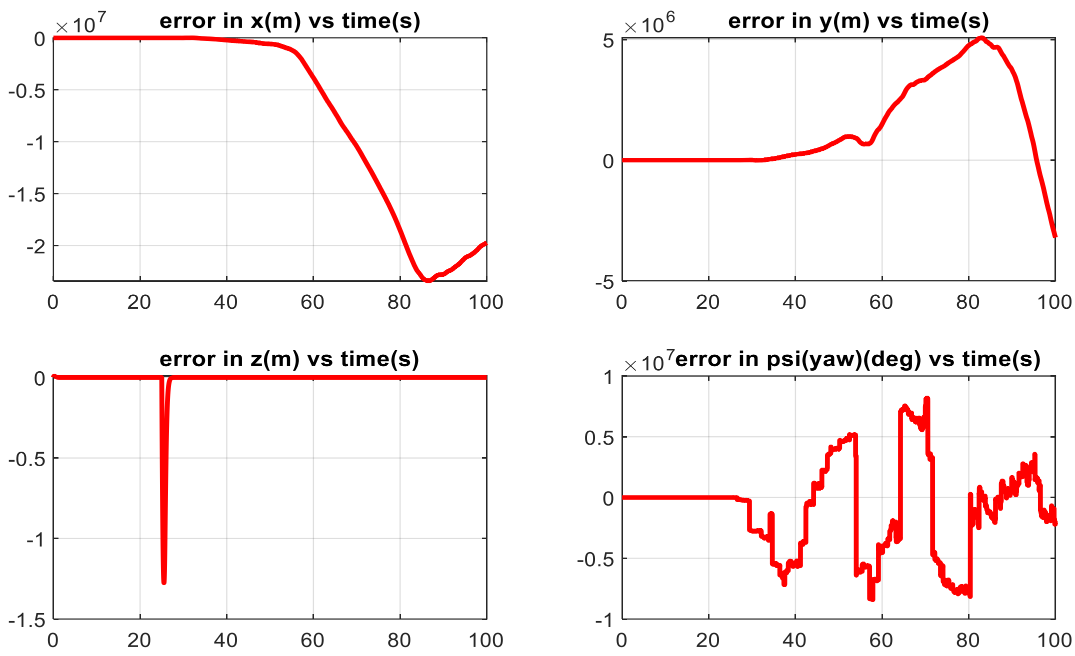


Figure 2.8 Errors in x,y,z and yaw [With Disturbance at 25s]

. It can be concluded from the plots 2.6, 2.7, 2.8 that the control is not robust enough to tolerate the effect of winds. The system becomes completely unstable with the addition of wind and all the coordinates except z diverge to ∞.

Chapter 3 Sliding Mode Control

Since the quadrotor system is a nonlinear type system, we proposed using a nonlinear Sliding Mode Controller (SMC) to control the states of the quadrotor.

3.1 Introduction to SMC

An SMC is a type of Variable Structure Control (VSC). It uses a high speed switching control law to force the state trajectories to follow a specified, user defined surface in the states space and to maintain the state trajectories on this surface [28]. The control law for an SMC consists of two parts; a corrective control part and an equivalent control part. The corrective control function is to compensate any variations of the state trajectories from the sliding surface in order to reach it. The equivalent control on the other hand, makes sure the time derivative of the surface is maintained to zero, so that the state trajectories would stay on the sliding surface.

3.2 Mathematics of Sliding Mode Control

We consider near hover configurations, so we can have the following approximation: $p \approx \dot{\Phi}$, $q \approx \dot{\Theta}$ and $r \approx \dot{\Psi}$. The dynamics of angular motion is given in the following Equation:

$$\mathbf{I}\ddot{\omega} = u_2 - \dot{\omega} \times \mathbf{I}\dot{\omega} \Rightarrow \ddot{\omega} = -\boldsymbol{I}^{-1}\dot{\omega} \times \mathbf{I}\dot{\omega} + \boldsymbol{I}^{-1}u_2 \tag{3.1}$$

We've

$$W = \begin{bmatrix} \Phi \\ \Theta \\ \Psi \end{bmatrix} \quad \text{and} \quad w_c = \begin{bmatrix} \Phi c \\ \Theta c \\ \Psi c \end{bmatrix}$$

The error is defined as $e(t)$ as $e(t) = \omega - w_c$

From the equation 3.1, the relative degree of the system is 2, so the First Order Sliding Mode Control variable can be defined as: $s = \dot{e} + \lambda e$

Differentiating once,

$$
\begin{aligned}
\dot{s} &= \ddot{e} + \lambda\dot{e} \\
\dot{s} &= \ddot{\omega} - \ddot{\omega}_c + \lambda(\dot{\omega} - \dot{\omega}_c) \\
\dot{s} &= -\mathbf{I}^{-1}\dot{\omega} \times \mathbf{I}\dot{\omega} + \mathbf{I}^{-1}u_2 - \ddot{\omega}\, c + \lambda(\dot{\omega} - \dot{\omega}_c) \\
\dot{s} &= -\mathbf{I}^{-1}\dot{\omega} \times \mathbf{I}\dot{\omega} - \ddot{\omega}_c + \lambda(\dot{\omega} - \dot{\omega}_c) + \mathbf{I}^{-1}u_2 \\
\dot{s} &= \alpha_{SM} + \beta_{SM}u_2
\end{aligned}
\tag{3.2}
$$

Where $\alpha_{SM} = -\mathbf{I}^{-1}\dot{\omega} \times \mathbf{I}\dot{\omega} - \ddot{\omega}c + \lambda(\dot{\omega} - \dot{\omega}_c)$ and $\beta_{SM} = \mathbf{I}^{-1}$. For nominal control, the control law will be chosen as: $u_2 = \beta_{SM}^{-1}(-\alpha_{SM} + v)$

For sliding mode, v must be chosen as: $v = -K * sign(s)$

Here, K is the sliding mode gain matrix. Using the above equations, the control input for controlling the altitude is: $u_2 = \beta_{SM}^{-1}(-\alpha_{SM} - K * sign(s))$

For the given problem at hand:s_1

$$
u_2 = -\mathbf{I}(-\mathbf{I}^{-1}\begin{bmatrix}\dot{\Phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} \times \mathbf{I}\begin{bmatrix}\dot{\Phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} - \begin{bmatrix}\ddot{\Phi}c\\ \ddot{\theta}c\\ \ddot{\psi}c\end{bmatrix} + \lambda\,(\begin{bmatrix}\dot{\Phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} - \begin{bmatrix}\dot{\Phi}c\\ \dot{\theta}c\\ \dot{\psi}c\end{bmatrix}) - k\begin{bmatrix}\text{sign}(s_1)\\ \text{sign}(s_2)\\ \text{sign}(s_3)\end{bmatrix})
\tag{3.3}
$$

Where $\lambda = \begin{bmatrix}\lambda_1 & 0 & 0\\ 0 & \lambda_2 & 0\\ 0 & 0 & \lambda_3\end{bmatrix}$ and $k = \begin{bmatrix}k_1 & 0 & 0\\ 0 & k_2 & 0\\ 0 & 0 & k_3\end{bmatrix}$

Table 3.1 Tuning parameters for SMC control

With respect to	kp	Kd
X	20	30
Y	10	10
z	4	4

$$k = \begin{bmatrix} 55 & 0 & 0 \\ 0 & 55 & 0 \\ 0 & 0 & 55 \end{bmatrix} \qquad \lambda = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 100 \end{bmatrix}$$

3.3 Results

The objective of the control is to track a Helical Trajectory. The Figure shows the SIMULINK model of the Quadrotor with the position and attitude control blocks in loop. The system has been tested with and without disturbances. The disturbance is wind with velocity Vector: V_w=5ˆi+5ˆj+5ˆk$m.s^{-1}$

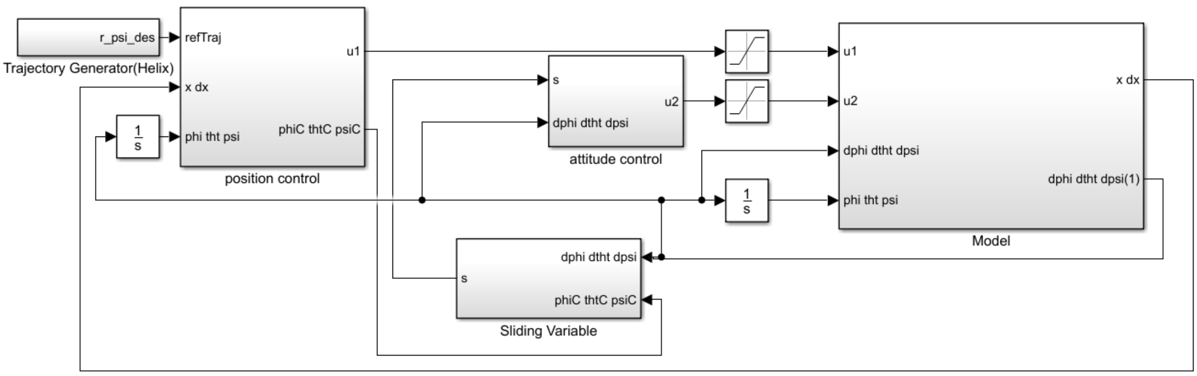
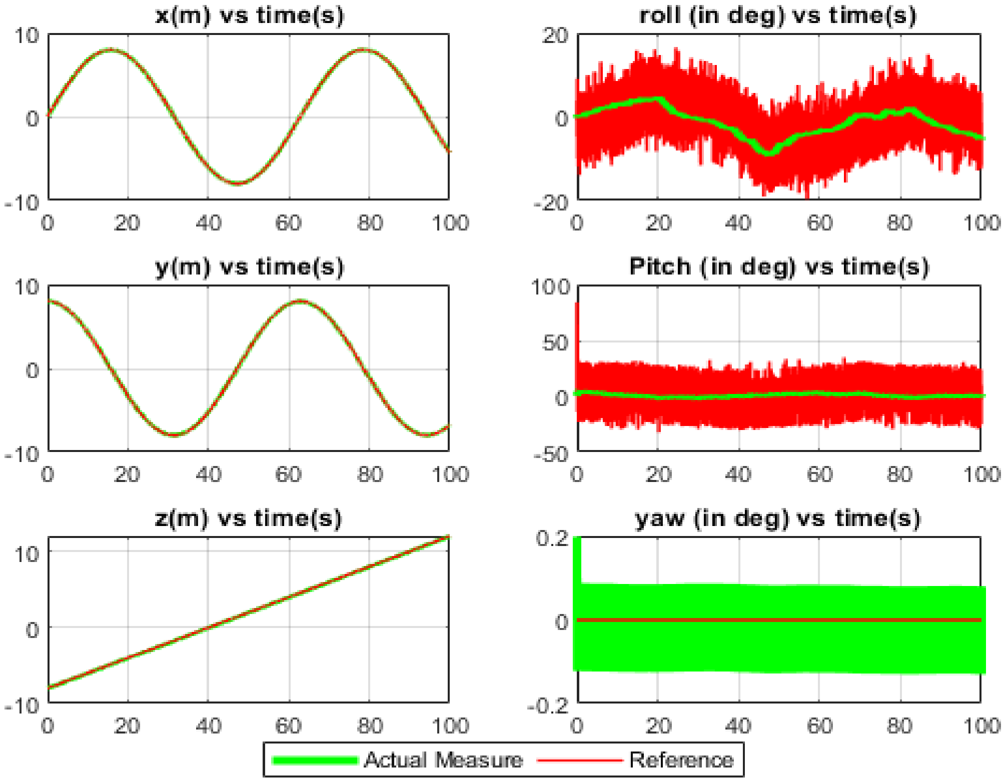


Figure 3.1 Global view of the Simulink model of the system with SMC control

3.3.1 Results without disturbances

Without any disturbance, the tracking by SMC control is better than PD control. The magnitudes of errors are less than the case of PD control. But the control is discontinuous and switches very frequently. The Figures 3.2, 3.3, 3.4 show the position and orientation, control inputs and the trajectory error in the absence of disturbances



.

Figure 3.2 Position and Orientation vs Time

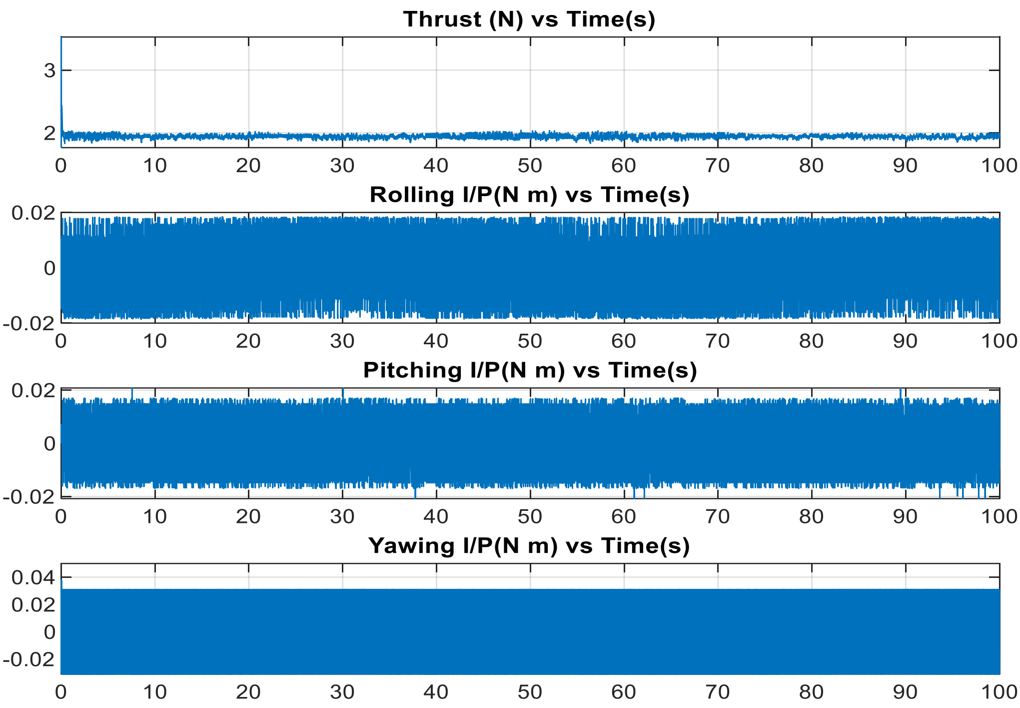


Figure 3.3 Thrust, Rolling, Pitching and Yawing Inputs vs Time

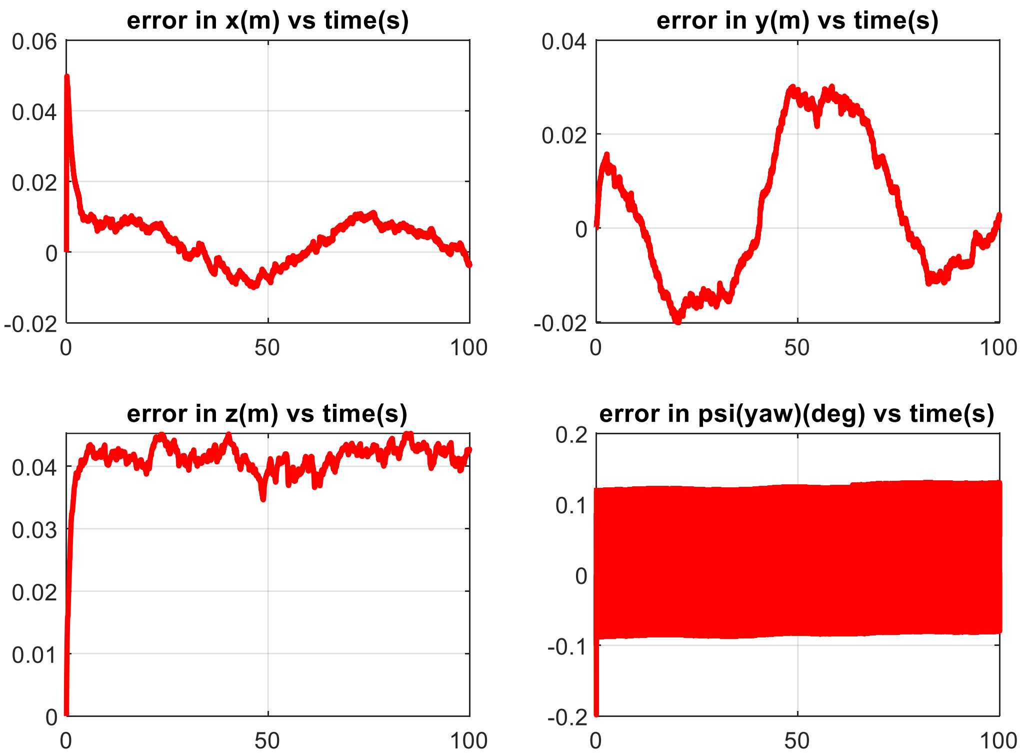


Figure 3.4 Errors in x,y,z and yaw

Without any disturbance, the tracking by SMC control is better than PD control. The magnitudes of errors are less than the case of PD control. But the control is discontinuous and switches very frequently.

3.3.2 Results with disturbances

The disturbance is wind of velocity $V_w = 5\hat{i} + 5\hat{j} + 5\hat{k}$ is applied as a step input at time t = 25 s. The Figures 3.5, 3.6, 3.7 show the position and orientation, control inputs and the trajectory error in the presence of disturbances

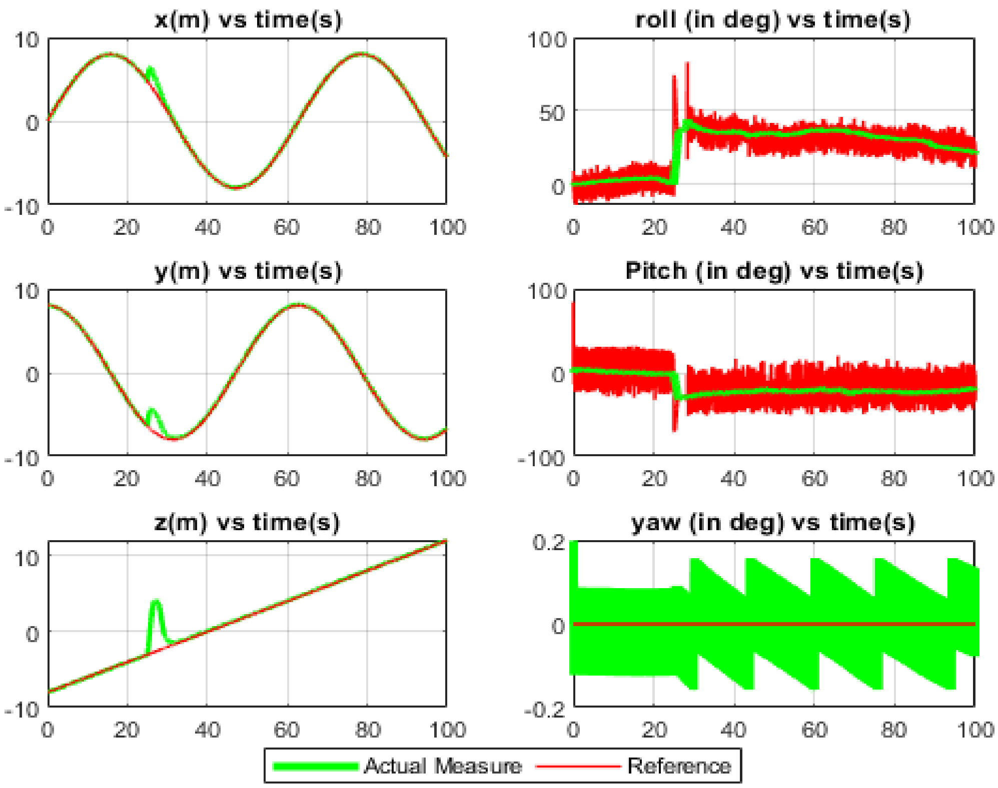


Figure 3.5 Position and Orientation vs Time [With Disturbance at 25s]

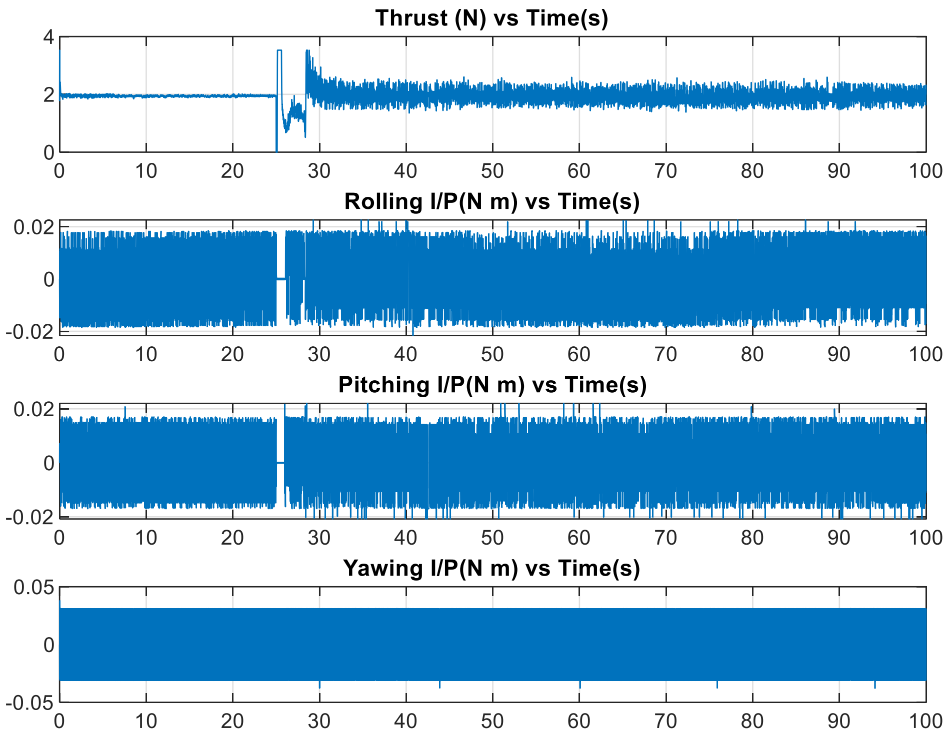


Figure 3.6 Thrust, Rolling, Pitching and Yawing Inputs vs Time [With Disturbance at 25s]

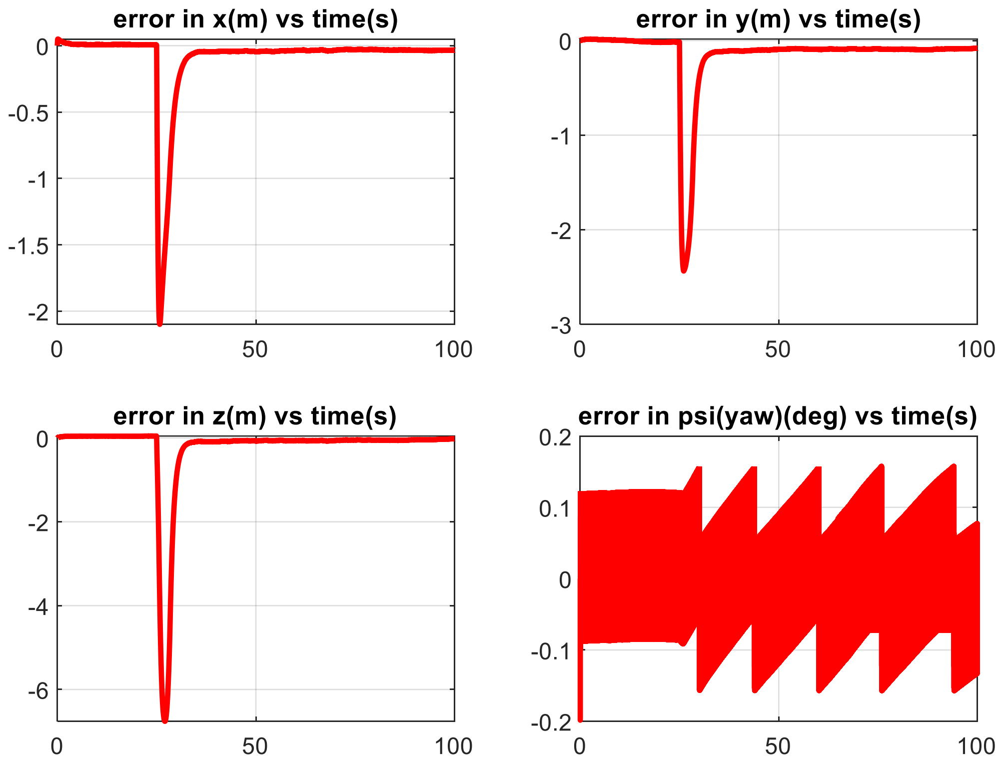


Figure 3.7 Errors in x,y,z and yaw [With Disturbance at 25s]

The Figures 3.6, 3.7 , 3.8 show the position and orientation, control inputs and the trajectory error in the presence of disturbances. It can be concluded from the plots that the control is robust enough to tolerate the effect of winds. The system becomes unstable with the addition of wind but gains control in approximately 5 seconds and all the coordinates converge to the desired values.

Chapter 4 Back Stepping Control

Backstepping controller is used to control the attitude, heading and the altitude of the quadrotor. This is a recursive control algorithm that works by designing intermediate control laws for some of the state variables.

4.1 Introduction to Backstepping

Backstepping is a recursive control algorithm that works by designing intermediate control laws for some of the state variables. These state variables are called "virtual controls" for the system [29]. Unlike other control algorithms that tend to linearize nonlinear systems such as the feedback linearization algorithm, backstepping does not work to cancel the nonlinearities in the system. This leads to more flexible designs since some of the nonlinear terms can contribute to the stability of the system. An example of such terms that add to the stability of the system are state variables taking the form of negative terms with odd powers (e.g. $-x_3$), they provide damping for large values of x [29] [30].

4.2 Mathematics of Backstepping Control

The dynamic model presented in Equations 2.1 and 2.2 are re-written for the sake of convenience. The state variables are renamed as follows:

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

The control inputs are:

1. Thrust=u_1
2. Rolling Input=u_{21}
   Pitching Input=u_{22}
   Yawing Input=u_{23}

The following parameters are used:

$$a_1 = \frac{I_{yy}-I_{zz}}{I_{xx}} \quad a_2 = \frac{I_{zz}-I_{xx}}{I_{yy}} \quad a_3 = \frac{I_{xx}-I_{yy}}{I_{zz}} \quad b_1 = \frac{1}{I_{xx}} \quad b_2 = \frac{1}{I_{yy}} \quad b_3 = \frac{1}{I_{zz}}$$

Using these state variables and the parameters, the dynamic model can be written as:

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \\ \dot{x_5} \\ \dot{x_6} \\ \dot{x_7} \\ \dot{x_8} \\ \dot{x_9} \\ \dot{x_{10}} \\ \dot{x_{11}} \\ \dot{x_{12}} \end{bmatrix} = \begin{bmatrix} x_2 \\ a_1 x_4 x_6 + b_1 u_{21} \\ x_4 \\ a_2 x_2 x_6 + b_2 u_{22} \\ x_6 \\ a_3 x_2 x_4 + b_3 u_{23} \\ x_8 \\ -g + b_4(\cos x_1 \cos x_3) u_1 \\ x_{10} \\ b_4(\cos x_5 \sin x_3 + \sin x_5 \cos x_3 \sin x_1) u_1 \\ x_{12} \\ b_4(\sin x_5 \sin x_3 - \cos x_5 \cos x_3 \sin x_1) u_1 \end{bmatrix} \tag{4.1}$$

The model developed in Equation 4.1 is used to implement the backstepping control.

Table 4.1 tuning parameter for backstepping control

With respect to	kp	kd
x	10	1
y	10	1

Table 4.2 list of constants that is used in the following equations

C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
100	100	100	100	100	100	1	1

4.2.1 Roll Controller

The states x_1 and x_2 are the roll and its rate of change. Considering the first two state variables only:

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} x_2 \\ a_1 x_4 x_6 + b_1 u_{21} \end{bmatrix} \tag{4.2}$$

The roll angle equation is strictly in feedback form, the second equation is influenced by an input. This makes the choice of the Lyapunov function easier. A simple positive definite Lyapunov Function is picked:

$$v_1 = \frac{1}{2} z_1^2 \tag{4.3}$$

Here $\boldsymbol{z_1}$ is the error between the desired and the actual roll angle defined as:

$$\boldsymbol{z_1 = x_{1c} - x_1}$$

The time derivative of the function defined in Equation 4.3 is;

$$\dot{v}_1 = z_1 \dot{z}_1 = z_1(\dot{x_{1c}} - \dot{x}_1) = z_1(\dot{x_{1c}} - x_2) \tag{4.4}$$

According to Krasovaskii-LaSalle principle, the system is guaranteed to be stable if the time derivative of the positive definite Lyapunov function is negative semidefinite. A positive definite bounding function is picked which is a bound on $\dot{v}_1$ as given in Equation

$$\dot{v}_1 = z_1 \dot{z}_1 = z_1(\dot{x_{1c}} - x_2) \le -c_1 z_1^2 \tag{4.5}$$

Here c_1 is a positive constant. To satisfy inequality 4.5 the virtual control input is chosen to- be:

$$(x_2)_{desired} = \dot{x}_{1c} + c_1 z_1 \quad (4.6)$$

A new error variable z_2 is defined which is the deviation of the state x_2 from its desired value.

$$z_2 = x_2 - \dot{x}_{1c} - c_1 z_1 \quad (4.7)$$

Rewriting equation 6.4 ➔

$$\dot{V}_1 = z_1 \dot{z}_1 = z_1(\dot{x}_{1d} - x_2)$$

$$= z_1 (\dot{x}_{1d} - (z_2 + \dot{x}_{1d} + c_1 z_1)) \quad (4.8)$$

$$\dot{V}_1 = -z_1 z_2 - c_1 z_1{}^2$$

The next step is to augment the first Lyapunov function V_1 with a quadratic term in the second variable z_2 to get a positive definite V_2.

$$V_2 = V_1 + \frac{1}{2} z_2{}^2$$

$$\dot{V}_2 = \dot{V}1 + z_2 \dot{z}_2 \quad (4.9)$$

$$V_2 = -z_1 z_2 - c_1 z_1{}^2 + z_2 (\dot{x}_2 - \ddot{x}_{1c} - c_1 \dot{z}_1)$$

Choosing a positive definite bounding function and substituting the model ($\dot{x}_2$) leads to the following:

$$-z_1 z_2 - c_1 z_1^2 + z_2(a_1 x_4 x_6 + b_1 u_{21} - \ddot{x}_{1d} - c_1 \ddot{z} \leq -c_1 z_1^2 - c_2 z_1^2 \quad (4.10)$$

Using the equality case of Equation (4.10), we get:

$$u_{21} = \frac{1}{b1} (\ddot{x}_{1d} + c_1 \dot{z}_1 - a_1 x_4 x_6 + z_1 - c_2 z_2) \quad (4.11)$$

4.2.2 Pitch controller

The states x_3 and x_4 are the pitch and its rate of change. Considering the third and the fourth state variables only:

$$\begin{bmatrix} \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} x_4 \\ a_2 x_2 x_6 + b_2 u_{22} \end{bmatrix} \quad (4.12)$$

The pitch angle equation is strictly in feedback form, only the second equation is influenced by an input. This makes the choice of the Lyapunov function easier. A simple positive definite Lyapunov Function is picked:

$$V_3 = \frac{1}{2} z_3^2 \quad (4.13)$$

Here z_3 is the error between the desired and the actual roll angle defined as:

$z_3 = x_{3c} - x_3$

The time derivative of the function defined in Equation 6.13 is;

$$\dot{V}_3 = z_3\dot{z}_3 = z_3(\dot{x}3c - \dot{x}3) = z_3(\dot{x}_{3c} - x_4) \quad (4.14)$$

According to Krasovaskii-LaSalle principle, the system is guaranteed to be stable if the time derivative of the positive definite Lyapunov function is negative semidefinite. A positive definite bounding function is picked which is bound on $\dot{\boldsymbol{V}}_3$ as given in Equation

$$\dot{V}_3 = z_3(\dot{x}_{3c} - x_4) \leq -c_3 z_3^2 \quad (4.15)$$

Here c_3 is a positive constant. To satisfy inequality 4.15 the virtual control input is chosen to be:

$$(x_4)\, desired = \dot{x}_{3c} + c_3 z_3 \quad (4.16)$$

A new error variable z_4 is defined which is the deviation of the state x_4 from its desired value.

$$z_4 = x_4 - \dot{x}_{3c} - c_3 z_3 \quad (4.17)$$

Rewriting Equation 6.14

$$\dot{V}_3 = z_3\dot{z}_3 = z_3(\dot{x}_{3d} - x_3)$$

$$= z_3(\dot{x}_{3d} - (z_4 + \dot{x}_{3d} + c_3 z_3)) \quad (4.18)$$

$$\dot{V}_3 = -z_3 z_4 - c_3 z_3^2$$

The next step is to augment the first Lyapunov function $V_3$with a quadratic term in the second variable z_4 to get a positive definite V_4.

$$V_4 = V_3 + \frac{1}{2} z_4^2$$

$$\dot{V}_4 = \dot{V}_3 + z_4\dot{z}_4 \quad (4.19)$$

$$\dot{V}_4 = -z_3 z_4 - c_3 z_3^2 + z_4(\dot{x}_4 - \ddot{x}_{3c} - c_3\dot{z}_3)$$

Choosing a positive definite bounding function and substituting the model ($\dot{x}_4$) leads to the following:

$$-z_3 z_4 - c_3 z_3^2 + z_4(a_2 x_2 x_6 + b_2 u_{22} - \ddot{x}3d - c_3\dot{z}_3) \leq -c_3 z_3^2 - c_4 z_4^2 \quad (4.20)$$

Using the equality case of Equation 4.20, we get:

$$u_{22} = \frac{1}{b2}(\ddot{x}_{3d} + c_3\dot{z}_3 - a_2 x_2 x_6 + z_3 - c_4 z_4) \quad (4.21)$$

4.2.3 Yaw controller

The states x_5 and x_6 are the yaw and its rate of change. Considering the fifth and the sixth state variables only:

$$\begin{bmatrix} \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} x_6 \\ a_3 x_2 x_4 + b_3 u_{23} \end{bmatrix} \quad (4.22)$$

The yaw angle equation is strictly in feedback form, that only the second equation is influenced by an input. This makes the choice of the Lyapunov function easier. A simple positive definite Lyapunov Function is picked:

$$V_5 = \frac{1}{2} z_5^2 \quad (4.23)$$

Here z_5 is the error between the desired and the actual roll angle defined as:

$z_5 = x_{5c} - x_5$

The time derivative of the function defined in Equation 4.23 is;

$$\dot{V}_5 = z_5\dot{z}_5 = z5(\dot{x}_{5c} - \dot{x}_5) = z_5(\dot{x}_{5c} - x_6) \quad (4.24)$$

According to Krasovaskii-LaSalle principle, the system is guaranteed to be stable if the time derivative of the positive definite Lyapunov function is negative semidefinite. A positive definite bounding function is picked which is a bound on $\dot{V}_5$ as given in Equation

$$\dot{V}_5 = z_5(\dot{x}_{5c} - x_6) \leq -c_5 z_5^2 \quad (4.25)$$

Here $c5$ is a positive constant. To satisfy inequality 4.25 the virtual control input is chosen to be:

$$(x_6)\ desired = \dot{x}_{5c} + c_5 z_5 \quad (4.26)$$

A new error variable z_6 is defined which is the deviation of the state x_6 from its desired value.

$$z_6 = x_6 - \dot{x}_{5c} - c_5 z_5 \quad (4.27)$$

Rewriting Equation 6.24

$$\begin{aligned} \dot{V}_5 &= z_5\dot{z}_5 = z_5(\dot{x}_{5d} - x_5) \\ &= z_5(\dot{x}_{5d} - (z_6 + \dot{x}_{5d} + c_5 z_5)) \quad (4.28) \\ \dot{V}_5 &= -z_5 z_6 - c_5 z_5^2 \end{aligned}$$

The next step is to augment the first Lyapunov function $V_5$with a quadratic term in the second variable z_6 to get a positive definite V_6.

$$\begin{aligned} V_6 &= V_5 + \frac{1}{2} z_6^2 \\ \dot{V}_6 &= \dot{V}_5 + z_6\dot{z}_6 \quad (4.29) \\ V_6 &= -z_5 z_6 - c_5 z_5^2 + z_6(\dot{x}_6 - \ddot{x}_{5c} - c_5\dot{z}_5) \end{aligned}$$

Choosing a positive definite bounding function and substituting the model ($\dot{x}_4$) leads to the following:

$$-z_5 z_6 - c_5 z_5^2 + z_6(a_3 x_2 x_4 + b_3 u_{23} - \ddot{x}_{5d} - c_5\dot{z}_5) \leq -c_5 z_5^2 - c_6 z_6^2 \quad (4.30)$$

Using the equality case of Equation 4.30, we get:

$$u_{23} = \frac{1}{b3}\,(\ddot{x}_{5d} + c_5\dot{z}_5 - a_3x_2x_4 + z_5 - c_6z_6) \tag{4.31}$$

4.2.4 Altitude controller

The states x_7 and x_8 are the altitude and its rate of change. Considering the seventh and the eighth state variables only:

$$\begin{bmatrix}\dot{x}_7\\ \dot{x}_8\end{bmatrix} = \begin{bmatrix} x_8 \\ -g + b_4(\cos x_1 + \cos x_3)\, u_1\end{bmatrix} \tag{4.32}$$

The altitude equation is strictly in feedback form, that only the second equation is influenced by an input. This makes the choice of the Lyapunov function easier. A simple positive definite Lyapunov Function is picked:

$$V_7 = \frac{1}{2}\, z_7^2 \tag{4.33}$$

Here Z_7 is the error between the desired and the actual altitude angle defined as:

$z_7 = x_{7c} - x_7$

The time derivative of the function defined in Equation 4.33 is;

$$\dot{V}_7 = z_7\dot{z}_7 = z_7(\dot{x}_{7c} - \dot{x}_7) = z_7(\dot{x}_{7c} - x8) \tag{4.34}$$

According to Krasovaskii-LaSalle principle, the system is guaranteed to be stable if the time derivative of the positive definite Lyapunov function is negative semidefinite. A positive definite bounding function is picked which is a bound on $\dot{V}_7$ as given in Equation

$$\dot{V}_7 = z_7(\dot{x}_{7c} - x_8) \le -c_7z_7^2 \tag{4.35}$$

Here c_7 is a positive constant. To satisfy inequality 4.35 the virtual control input is chosen to be:

$$(x_8)\ desired = \dot{x}_{7c} + c_7z_7 \tag{4.36}$$

A new error variable z_8 is defined which is the deviation of the state x_8 from its desired value.

$$z_8 = x_8 - \dot{x}_{7c} - c_7 z_7 \tag{4.37}$$

Rewriting Equation 4.34

$$\begin{aligned}
\dot{V}_7 &= z_7\dot{z}_7 = z_7(\dot{x}_{7d} - x_7) \\
&= z_7(\dot{x}_{7d} - (z_8 + \dot{x}_{7d} + c_7 z_7)) \\
\dot{V}_7 &= -z_7 z_8 - c_7 {z_7}^2
\end{aligned} \tag{4.38}$$

The next step is to augment the first Lyapunov function V_7 with a quadratic term in the second variable z_8 to get a positive definite V_8.

$$\begin{aligned}
V_8 &= V_7 + \frac{1}{2}\, {z_8}^2 \\
\dot{V}_8 &= \dot{V}_7 + z_8\dot{z}_8 \\
V_8 &= -z_7 z_8 - c_7 {z_7}^2 + z_8(\dot{x}_8 - \ddot{x}_{7c} - c_7\dot{z}_7)
\end{aligned} \tag{4.39}$$

Choosing a positive definite bounding function and substituting the model ($\dot{x}_8$) leads to the following:

$$-z_7 z_8 - c_7 z_7^2 + z_8(-g + b_4(\cos x_1 + \cos x_3)\, u_1 - \ddot{x}_{7d} - c_7\dot{z}_7) \le -c_7 z_7^2 - c_8 z_8^2 \tag{4.40}$$

Using the equality case of Equation 4.40, we get:

$$u_{23} = \frac{1}{b_4 \cos x_1 \cos x_3} (\ddot{x}_{7d} + c_7\dot{z}_7 - c_8 z_8 + g + z_7) \tag{4.41}$$

4.3 Results

The objective of the control is to track a Helical Trajectory. The Figure 4.1 shows the SIMULINK model of the Quadrotor with the position and attitude control blocks in loop. The system has been tested with and without disturbances. The disturbance is wind with velocity vector $V_w = 5i + 5j + 5 km.s^{-1}$.

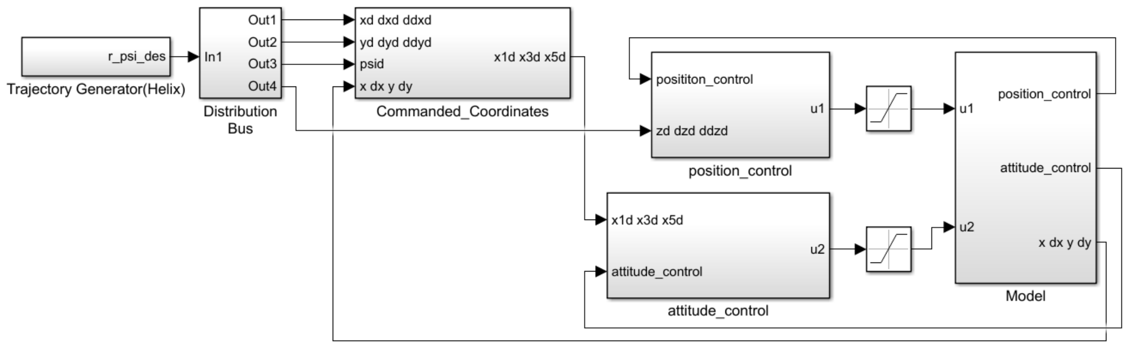


Figure 4.1 Global view of the Simulink model of the system with BSC control

4.3.1 Results without disturbance

Without any disturbance, the tracking by Backstepping control is better than both Proportional Derivative and Sliding Mode Control. The magnitudes of errors are the least. The control is smooth and continuous. The Figures 4.2, 4.3, 4.5 show the positions, orientations, control inputs and the trajectory error in the absence of disturbances.

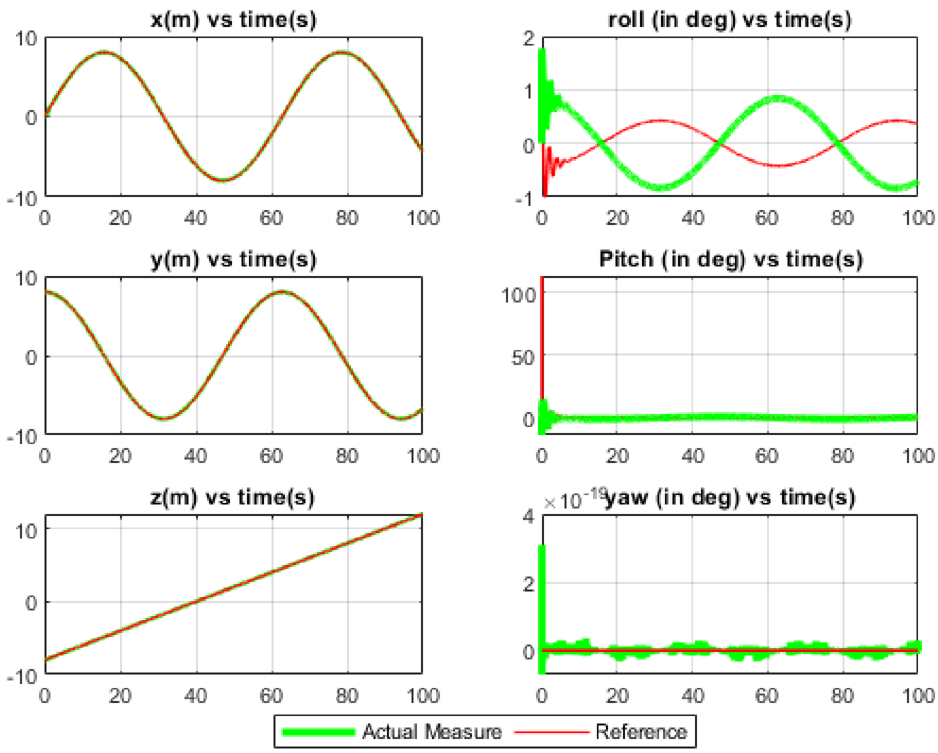


Figure 4.2 Position and Orientation vs Time

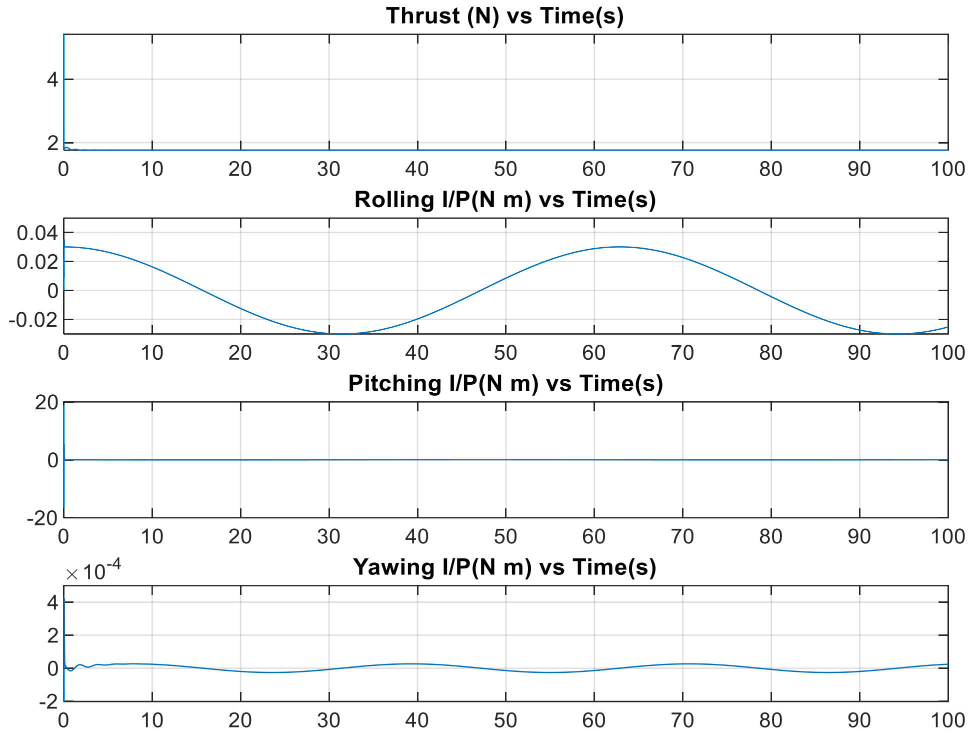


Figure 4.3 Thrust, Rolling, Pitching and Yawing Inputs vs Time

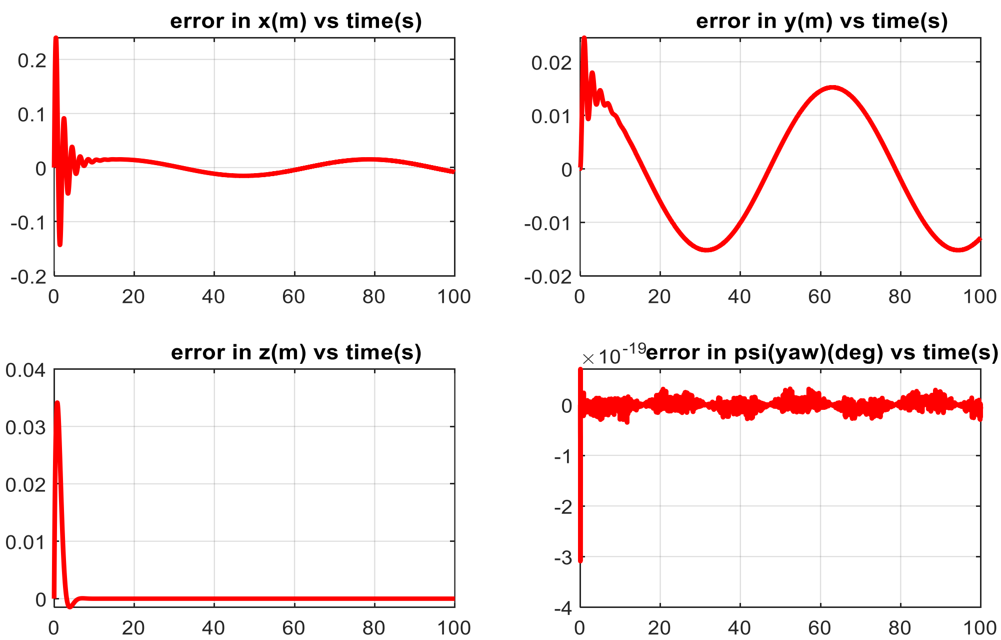


Figure 4.4 Errors in x,y,z and yaw

Without any disturbance, the tracking by Backstepping Control is better than both Proportional Derivative and Sliding Mode Control. The magnitudes of errors are the least. The control is smooth and continuous.

4.3.2 Results with disturbance

The disturbance is wind of velocity $V_w = 5i + 5j + 5k$ at is applied as a step input at time $t = 25$ s. The Figures 4.5, 4.6, 4.7 show the positions, orientations, control inputs and the trajectory error in the presence of disturbances.

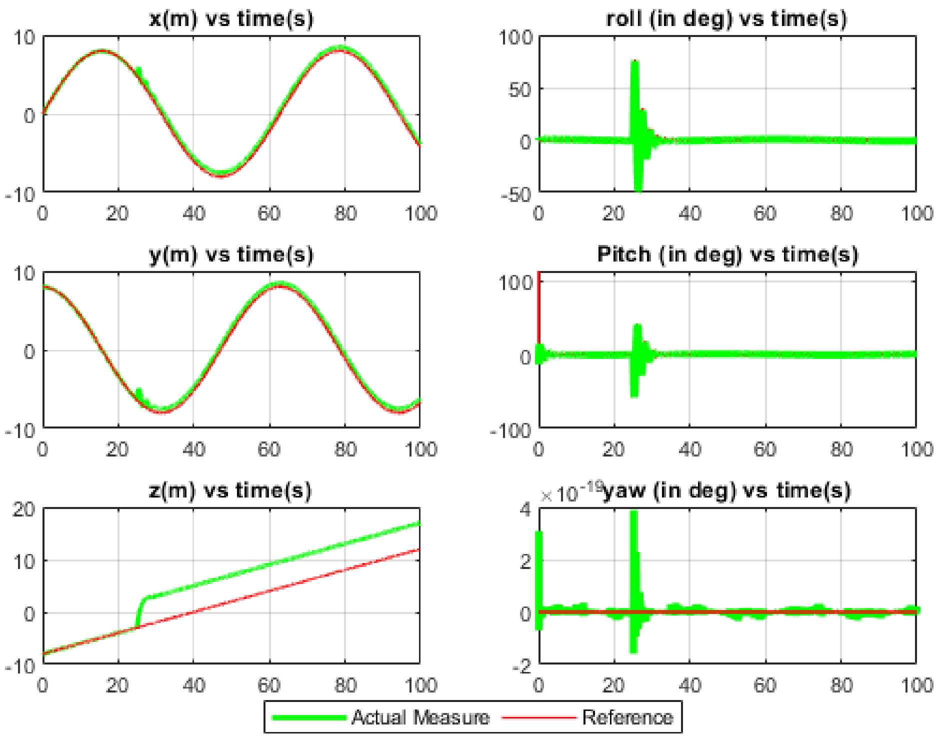


Figure 4.5 Position and Orientation vs Time [With Disturbance at 25s]

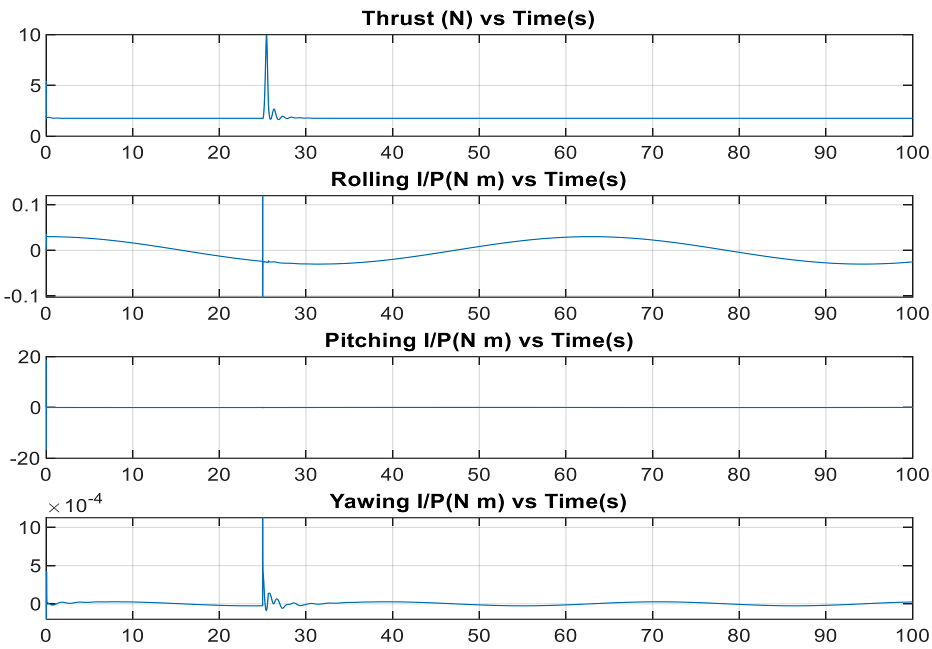


Figure 4.6 Thrust, Rolling, Pitching and Yawing Inputs vs Time [With disturbance at 25 s]

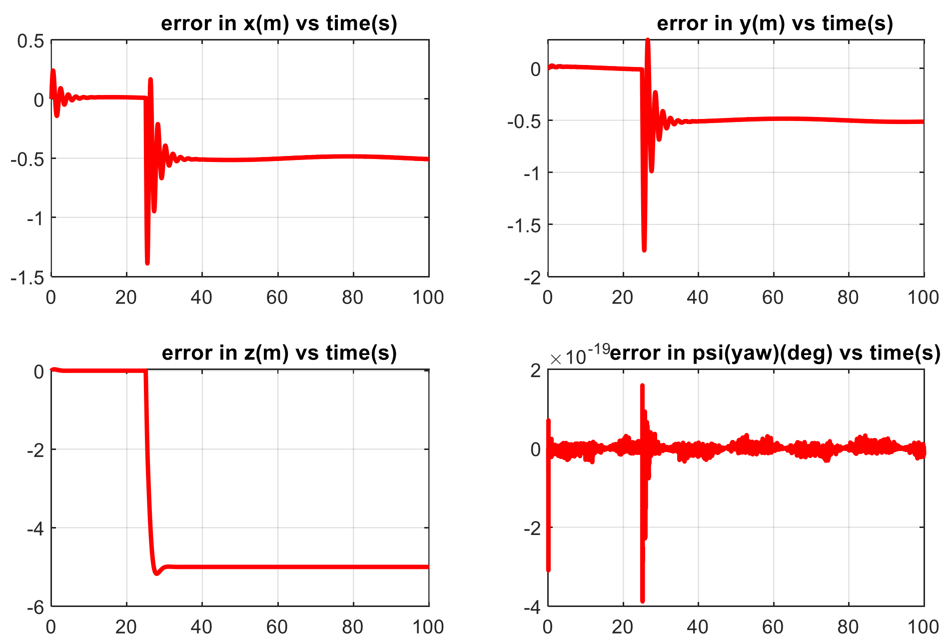


Figure 4.7 Errors in x,y,z and yaw [With Disturbance at 25s]

The Figures 4.5, 4.6, 4.7 show the positions, orientations, control inputs and the trajectory error in the presence of disturbances. It can be concluded from the plots that the control does not perform as well as SMC, there is significant steady state error and the system does not converge to the desired trajectory but has steady state errors. The system does not become unstable with but does not converge to the desired values either

In this chapter, the summary of all the control methods is presented. The advantages and the disadvantages of the various techniques is explained.

4.4 Discussions

1. Proportional Derivative Control

The advantages of Proportional Derivative (PD) Control are:

a. Easy to understand and implement.
b. fewer tuning parameters.
c. tuning process is not complicated

The disadvantages of PD Control are:

a. The derivative term creates problem in real systems. It amplifies the noise.
b. The control is not robust to disturbances and parameter variations.

2. Sliding Mode Control

The advantages of Sliding Mode Control (SMC) are:

a. Easy to understand and implement.
b. Robust to parameter variations and disturbances.

The disadvantages of SMC are:

a. The control law is discontinuous, and this may adversely affect the actuators.
b. The gains are very high, and it can cause actuator saturation.

3. Backstepping Control

The advantages of Backstepping Control (BSC) are:

a. Ensures Lyapunov Stability.
b. It does not involve cancelling of system non-linearities by feedback linearization, hence it is not system dependent.

The disadvantages of BSC are:

a. The theory is mathematically exacting.
b. There are several gains to tune.
c. Although it does not become unstable with introduction of disturbances, there is considerable finite steady state error. The solution can be to use integral backstepping.

Table 4.3 comparison table

[Disturbance/Control]	***max*** $\|e_x\|$**(*m*)**	***max*** $\|e_y\|$**(*m*)**	***max*** $\|e_z\|$**(*m*)**	***max*** $\|e_\psi\|$**(*m*)**	**Control**
Without Disturbance/PD	0.0747	0.0747	0	0.016	Low and Smooth
Without Disturbance/SMC	0.01	0.03	0	0.123	Discontinuous
Without Disturbance/BSC	*0.0151*	*0.0151*	0	*0*	Low and Smooth
With Disturbance/PD	∞	∞	0	∞	Saturated
With Disturbance/SMC	0.03	0.08	0.06	0.15	Discontinuous
With Disturbance/BSC	0.5	0.5	5	0	Low and Smooth

General Conclusion

The goal of this work was to derive a mathematical model for the quadrotor. Three control techniques were then developed; a linear (proportional-Derivative **PD**), a nonlinear (Sliding Mode and Backstepping) controllers. A complete simulation is then implemented on MATLAB/Simulink successfully relying on the derived mathematical model of the quadrotor. The simulation environment is used to evaluate the mentioned controllers and compare their dynamic performances. The tuning of the parameters of the three used controllers is done using the genetic algorithm GA. The three controllers performed comparably in near hovering operation of the quadrotor in the range of $0\sim20°$ of attitude and heading and gave better performance outside the linear hovering region when the system is perturbed.

Future work

one valuable addition would be the robustification of the developed control techniques against wind as this is a common problem with quadrotors control and our simulation results showed a huge degradation of the performance of the controllers when the system is exposed to wind. Moreover, in our work it is assumed that all the model parameters are known accurately without any uncertainties, which is not the case in reality, thus, developing adaptive control algorithms to count for the system uncertainties would enhance the performance of the quadrotor when operating in a real environment. Adding an integral action to the developed Backstepping and sliding mode (SMC) controllers will lead to the formulation of an adaptive control algorithm robust to system uncertainties and to eliminate the shattering effect for the SMC.

Bibliography

[1] J. Z. H. X. G. W. a. D. Y. Hongning Hou, "A simple controller of minisize quad-rotor vehicle," in *2010 IEEE International Conference on Mechatronics and Automation*, Xi'an, China, 4-7 Aug. 2010.

[2] M.-S. K. a. S. P. Jinhyun Kim, "Accurate modeling and robust hovering control for a quadrotor VTOL aircraft," in *Journal of Intelligent and Robotic Systems, 57(1-4):9{26, 2010 . ISSN 0921-0296. doi: 10.1007/s10846-009-9369-z. URL http://dx.doi.org/10.1007/s10846-009-9369-z*.

[3] K. Farid, "Survey of advances in guidance, navigation, and control of un-manned rotorcraft systems," Journal of Field Robotics, 29(2):315{378, 2012.

[4] P. E. I. Pounds, "Design, construction and control of a large quadrotor micro air vehicle," Phd. thesis, Australian National University, 2007.

[5] A. N. a. R. S. S. Bouabdallah, "PID vs LQ control techniques applied to an indoor micro quadrotor," In Intelligent Robots and Systems, 2004 . (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on, volume 3, pages 2451{2456 vol.3, 2004. doi: 10.1109/IROS.2004.1389776.

[6] J. L. a. Y. Li, "Dynamic analysis and PID control for a quadrotor," In Mechatronics and Automation (ICMA), 2011 International Conference on, pages 573{578, 2011. doi: 10.1109/ICMA.2011.5985724.

[7] Z. C. Q. L. a. Y. W. Jinpeng Yang, “Self-tuning pid control design for quadrotor uav based on adaptive pole placement control,” In chinese Automation Congress (CAC), 2013, pages 233{237. IEEE, 2013.

[8] M. G. O. a. F. R. R. Guilherme V Raffo, “An integral predictive/nonlinear h control structure for a quadrotor helicopter,” Automatica, 46(1):29{39, 2010.

[9] G. M. H. H. H. M. P. V. a. C. J. T. Jeremy H Gillula, “Applications of hybrid reachability analysis to robotic aerial vehicles,” The International Journal of Robotics Research, 30(3):335{354, 2011.

[10] H. T. R. I. M. A. H. P. A. C. a. Y. Y. A. Ataka, “Controllability and observability analysis of the gain scheduling based linearization for uav quadrotor,” In Robotics, Biomimetics, and Intelligent Computational Systems (ROBIONETICS), 2013 IEEE International Conference on, pages 212{218, Nov 2013. doi: 10.1109/ROBIONETICS.2013.6743606.

[11] A. C. a. Y. Z. Mohammad Hadi Amoozgar, “Fault tolerant fuzzy gain scheduled pid for a quadrotor helicopter testbed in the presence of actuator faults,” In IFAC Conference on Advances in PID Control, Brescia, Italy (March 2012), 2012.

[12] M. A. a. Y. M. Z. Iman Sadeghzadeh, “payload drop application of unmanned quadrotor helicopter using gain-scheduled pid and model predictive control techniques,” In Intelligent Robotics and Applications,pages 386{395. Springer, 2012..

[13] S. B. a. R. Siegwart, “Backstepping and sliding-mode techniques applied to an indoor micro quadrotor,” In Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on, pages 2247-2252, 2005. doi: 10.1109/ROBOT.2005.1570447.

[14] G. H. J. S. J. a. C. T. S.L. Waslander, “Multi-agent quadrotor testbed control design: integral sliding mode vs. reinforcement learning,” In Intelligent Robots and Systems,

2005. (IROS 2005). 2005 IEEE/RSJ International Conference on, pages 3712{3717, 2005. doi: 10.1109/IROS.2005.1545025.

[15] T. M. a. A. Benallegue, "Backstepping control for a quadrotor helicopter," In Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on,pages 3255{3260, 2006. doi: 10.1109/IROS.2006.282433.

[16] Z. F. a. W. Gao, "Adaptive backstepping control of an indoor micro quadrotor," Research Journal of Applied Sciences, 4, 2012.

[17] S. K. T. R. a. H. J. K. Hyeonbeom Lee, " Backstepping control on se (3) of a micro quadrotor for stable trajectory tracking," In Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on, pages 4522-4527. IEEE, 2013.

[18] X. Q. a. H. D. Hongtao Zhen, "An adaptive block backstepping controller for attitude stabilization of a quadrotor helicopter," WSEAS Transactions on Systems & Control, 8(2), 2013.

[19] S. S. a. R. L. Iv´an Gonz´alez, "Chattering-free sliding mode altitude control for a quad-rotor aircraft: Real-time application," Journal of Intelligent & Robotic Systems, 73(1-4):137{155, 2014.

[20] Z. Y. a. K. N. Farid Kendoul, "Guidance and nonlinear control system for autonomous flight of minirotorcraft unmanned aerial vehicles," Journal of Field Robotics, 27(3):311-334, 2010.

[21] G. N. a. A. T. Kostas Alexis, "Switching model predictive attitude control for a quadrotor helicopter subject to atmospheric disturbances," Control Engineering Practice, 19(10):1195-1207, 2011.

[22] M. Efe, "Neural network assisted computationally simple pid control of a quadrotor uav," Industrial Informatics, IEEE Transactions on, 7(2):354-361 2011. ISSN 1551-3203. doi: 10.1109/TII.2011.2123906.

[23] A. A. a. X. Wang, “Quad rotor arial robot dynamic modeling and configuration stabilization,” In Informatics in Control, Automation and Robotics (CAR), 2010 2nd International Asia Conference on, volume 1, pages 438-444,2010. doi: 10.1109/CAR.2010.5456804.

[24] S. S. C. T. M. S. a. H. L. Amr Nagaty, “control and navigation framework for quadrotor helicopters,” Journal of Intelligent and Robotic Systems, 70(1-4):1{12, 2013. ISSN 0921-0296. doi: 10.1007/s10846-012-9789-z. URL http://dx.doi.org/10.1007/s10846-012-9789-z.

[25] D. W. Mellinger, “Trajectory Generation and Control for Quadrotors,” Publicly Accessible Penn Dissertations. Paper 547.

[26] Samir BOUABDALLAH, “design and control of quadrotors with application to autonomous flying,” in *THÈSE NO 3727 (2007),à LA FACULTÉ DES SCIENCES ET TECHNIQUES DE L'INGÉNIEUR*.

[27] O. A. a. N. Aouf, “Quadrotor Control for Trajectory Tracking in Presence of Wind Disturbances,” in *2014 UKACC International Conference on Control 9th - 11th July 2014, Loughborough, U.K*.

[28] J. L. a. X. Wang, “Advanced Sliding Mode Control for MechanicalSystems: Design, Analysis and MATLAB Simulation,” in *Springer, 2012*.

[29] P. V. K. a. I. K. Miroslav Krstic, “Nonlinear and adaptive control design,” in *John Wiley & Sons, Inc., 1995*.

[30] H. J. Marquez, “Nonlinear control systems: analysis and design,” in *John Wiley,2003*.