Bibliography
Report

Robust Control of Quadrotors

Submitted in partial fulfillment of
the requirements for the award of the degree of

Master of Engineering
in
Automatic Control, Robotics & Applied Informatics

Submitted by
Subodh Mishra

Under the guidance of
Prof. Abdelhamid Chriette

Department of Automatic Control and Robotics
École Centrale de Nantes
Nantes, France – 44300

Third Semester-2016-2017

Department of Automatic Control and Robotics

École Centrale de Nantes

Certificate

This is to certify that this is a bonafide record of the bibliographic survey done by the Mr. Subodh Mishra, during the third semester, in partial fulfilment of the requirements of the degree of Master of Engineering in Automatic Control, Robotics and Applied Informatics.

Prof. Abdelhamid Chriette
(Guide)

Date:

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.

"In order to fly, all one must do is simply miss the ground."
-Douglas Adams

Contents

Chapter 1

Introduction

1.1 Preliminaries

Quadrotors are interesting platform for Aerial Robotics research. Owing to the underactuation, scarcity of reliable power source, unavailability of lighter electronics, aerial robotics has always been a challenging research area. The burgeoning demand of aerial robots in military, farming, mining, firefigting, remote sensing and meteorological obervation, etc. has provided great impetus to research and developement in this field [1].

1.1.1 Unmanned Aerial Vehicles

An Unmanned Aerial Vehicle (UAV) is an aircraft piloted by remote control or onboard computers. Also known as Unmanned Aircraft System (UAS), it is an aircraft without a human pilot aboard. The earliest recorded use of UAVs was for warfighting in 1849, when the Austrians attacked Venice with unmanned balloons loaded with explosives. During the World War 1, the US engineers developed an experimental unmanned aerial torpedo, a forerunner of present day cruise missiles. Called the Kettering Bug, this UAV was capable of striking ground targets up to 121 Kilometers. Reginald Denny, a British Royal Flying Corps Aviator pursued his interest in radio control model aircraft in the 1930s and found the Radioplane Company and manufactured nearly fifteen thousand drones for the army during World War II. The Reaper of the US Armed forces has also proved to be an effective UAV to fight terrorism without causing loss of life of the security forces. While it is generally deemed that UAVs have only military applications, the civilian UAV industry is also booming. With applications in agriculture, photography, construction, film production and rescue operations, the civilian UAV industry is expected to grow leaps and bounds. In the present era, the num-

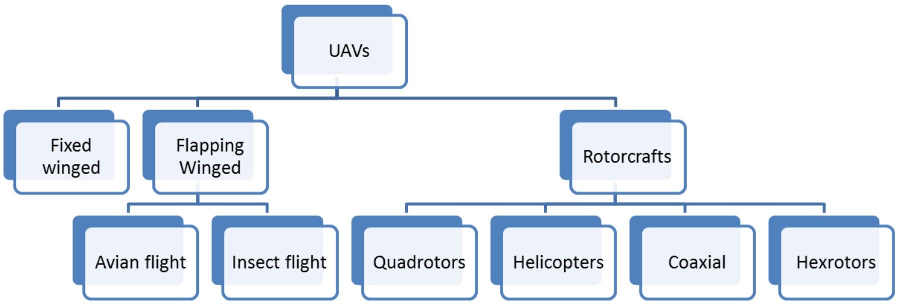

Figure 1.1: Different Types of UAVs

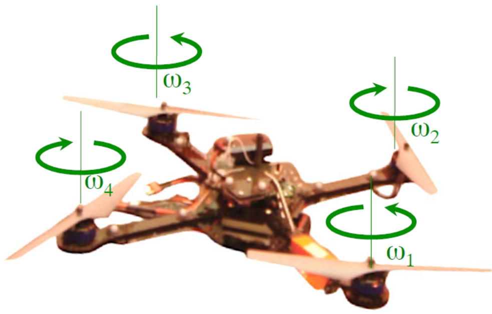

Figure 1.2: Quadrotor UAV

ber of UAVs has increased manifold and it is predicted that the UAV industry will soon be a $ 25 B industry by 2020. The different types of UAVs are listed in the Figure 1.1.

1.1.2 Quadrotors

A Quadrotor is an UAV with four rotors. Varying the speeds of the rotors helps to control the position and the orientation of the robot. As shown in Figure 1.2, the adjascent rotors have opposite sense of rotation. This is done to balance the total angular momentum of the craft, otherwise the UAV will start rotating about itself. The Quadrotor has 6 degrees of freedom but only four actuators (Rotors). Hence, Quadrotors are underactuated. The Rotors produce thrust, torque and drag force and the control input to the system is the angular velocity of the motors. A low level controller stabilizes the rotational speed of each blade. The Quadrotor can perform Vertical Take Off and

Landing (VTOL), hover and make slow precise movements. The four rotors provide a higher payload capacity. Quadrotors are relativley simpler because they do not have complicated swashplates and linkages. They are easy to assemble and can be fabricated using remote controlled toy components.[2] is a good first guide on learning about quadrotors. The coursera course on Aerial Robotics [4] is also very helpful for the uninitiated learner.

1.1.3 Components of Autonomous Flight

A good reference for studying about components of autonomous flight is [3]. Summarizing, in order to enable autonomous flight, there are a few components/modules that we need in any UAV. They are listed as below:

- **State Estimation:** Estimate the position and velocity (including rotation and anugular velocity of the robot).

- **Control:** Command motors and produce desired actions in order to navigate to the desired state.

- **Mapping:** The vehicle must have some basic capability to map its environment. If it does not know what the surrounding environment looks like, then it's incapable of reasoning about this environment and planning safe trajectories in this environment.

- **Planning:** Finally, the vehicle must be able to compute a trajectory, given a set of obstacles and a destination.

1.1.4 State Estimation

The goal of state estimation is to obtain relaible estimates of the position and velocity (linear and angular) using which control algorithms can be implemented. Described below are few techinques which are generally used for state estimation [5].

- **Motion Capture System:** The use of such systems have been discussed in many papers, one among them is [6]. Motion capture cameras allow the robot to measure its postion. The flying robot has reflective markers mounted on it and the cameras can estimate the position of each reflective marker with frequecies as high as 100-200 Hz. The measurements are very precise and accuracies are well below 1 mm. But these systems are usable only in laboratory settings.

- **Global Positioning System (GPS):** The GPS coordinates can be used outdoors but generally in urban environments, near tall buildings and indoors, GPS is not available. Some UAVs are very small and maneuverable, they can find themselves in settings where it is difficult to communicate directly with the GPS satellites. But there are technologies which also allow GPS denied navigation [7].

- **Simultaneous Localization and Mapping (SLAM):** It is difficult to find GPS indoors and near cluttered urban environments. In such settings it is advisable to use Color+Depth cameras or LASER Range finders. Using SLAM these sensors infer information about the environment and also help the robot to localize itself. The SLAM algorithm requires inputs from IMU (Inertial Measurement Unit) and LASER Range Finder/Color+Depth to map and localize at the same time. An inexpensive solution could be to use a single camera with aprilTags. Even in the absense of GPS, SLAM helps the robot to localize and estimate its state variables. In [8] a SLAM strategy for an Autonomous Quadrotor is explained.

It must be kept in mind that adding more sensors requires use of bigger motors and propellors which in turn require more powerful and heavier batteries. The solution is always a compromise when the choice of sensors is considered.

1.1.5 Applications

Quadrotors are being increasingly used in Precision Farming, Construction, Archeology, Photography, Robotic First Responders during Emergency. In precision farming, robots are being used to patrol orchads to do a visual survey in infrared spectrum to assess the quality of yield. In construction industry, quadrotors are being used for inspection and the check the progress of the work. In archaelogical sites, these robots are used to inspect the stability of old building that were built thousands of years ago. Quadrotors have the potential to be used as first respondes in case of a calamity or emergency. They can go to places where recue workers cannot go easily and provide information to the authorities to plan rescue missions. Quadrotors are also being used along side manipulators to perform cooperative mobile manipulation tasks. Engineers at the MIT have developed UAV technology that helps people navigate the MIT campus. The Quadrotor acts as a personal tour guide in difficult and disorientating environments. This Quadrotor has the capability to interface and interact with people. Another emerging area of application is the use of a swarm of aerial robots to carry out a prescribed

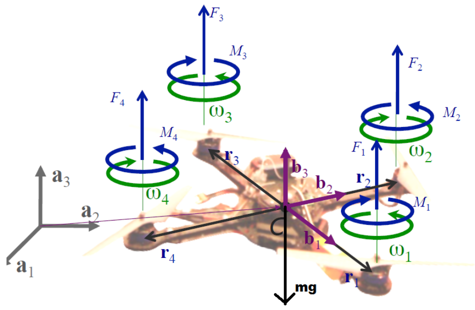

Figure 1.3: Free Body Diagram of a Quadrotor UAV

mission and to respond as a group to high level management command. A robot swarm can collectively manipulate a large load, just as small ants engage in cooperative prey retrieval carrying large, awkwardly shaped morsels of food back to their nests. It must be kept in mind that in the heart of all these applications there is a robust controller which is fault tolerant and helps the robot to carry out the desired action. Hence, robust contoller design for these flying machines is of paramount importance.

1.2 System Description

1.2.1 Basic Mechanics

In this section we explore the basic mechanics of the quadrotor. The model has been derived from [2]. The Figure 1.3 shows the free body diagram of a quadrotor. The forces acting on the system are the thrusts F_i from each of the rotor and the force of gravity -$mg\mathbf{a}_3$. The moments acting on the system are the moments due to each of the thrust and the drag moment M_i which is generated due to the propellor rotation.

If the thrust is plotted against the motor RPM, it is found that the relationship is approximately quadratic. Every time a rotor spins, there is also a drag that the rotor has to overcome and that drag moment is also quadratic. So, in hover conditions, each rotor has to support one fourth the total weight.

As can be seen in Figure 1.4, one can determine the operating motor speed (ω_0) that will be required to produce one-fourth the weight. The operating speed produces a drag moment and every rotor has to over come the drag moment. The motor provides the torque to overcome this drag. The motor torque characteristic is a property of the motor and it determines whether a

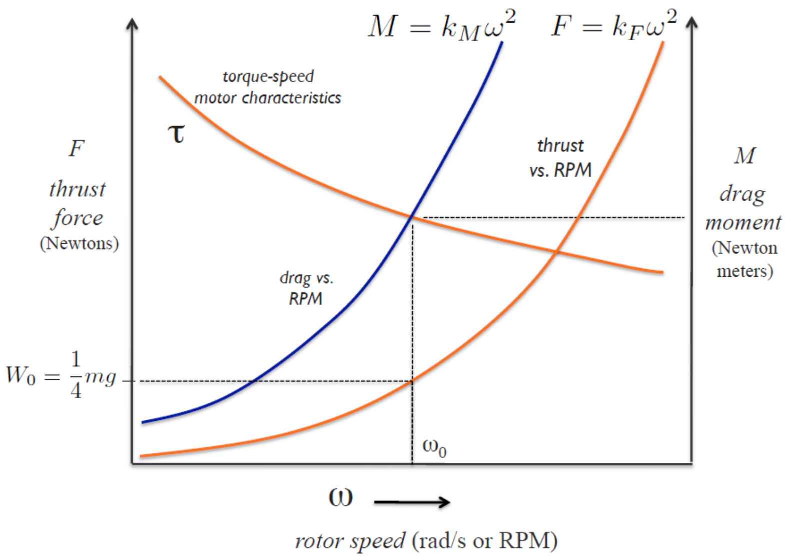

Figure 1.4: Thrust F_i, Drag Moment M_i and Motor Torque Speed Chacteristics τ vs ω

particular motor can be used or not. A motor which can compensate for the drag torque at the operating speed can be used.

- Operating motor speeds:

$$k_F\omega_0^2 = \frac{mg}{4} \tag{1.1}$$

- Motor Torques and Drag Torque (They have same magnitude but opposite signs):

$$M_i = \tau_i = k_M\omega_i^2 \tag{1.2}$$

- Thrust

$$F_i = k_F\omega_i^2 \tag{1.3}$$

- Resultant Force:

$$F = F_1 + F_2 + F_3 + F_4 - mg\mathbf{a}_3 \tag{1.4}$$

- Resultant Moment:

$$M = r_1 \times F_1 + r_2 \times F_2 + r_3 \times F_3 + r_4 \times F_4 + M_1 + M_2 + M_3 + M_4 \tag{1.5}$$

In Equilibrium, the resulant force and torque are zero. If it is non zero then the robot accelerates. A combination of motor thrusts and the weight determines the net linear and angular acceleration of the robot.

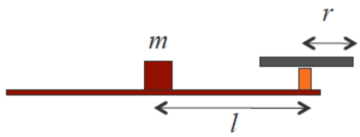

Figure 1.5: Dimensions of any arm

1.2.2 Design Considerations

In practice, the rotor thrust is limited by the peak motor torque [4]. In general the peak angular speed is known, hence the peak motor torque is also known. This data can be used to estimate the maximum climb rate of the quadrotor and the maximum weight of the payload the machine can carry without changing the motors. Changing the payload without changing the motor will change the thrust to weight ratio. A higher thrust to weight ratio is always desirable.
As a consequence of different thrust to weight ratio, the power consumption is also different. This gives us an idea about how to choose batteries.
Batteries contribute about 33% of the total mass and the motors + propellors contribute about 25 % of the total mass. Adding sensors like laser scanners and cameras increases the total mass also.

1.2.3 Agility with Scaling

We explore the effects of sizing the platform. Increasing the siwe also increases the the system weight and affects the thrust to weight ratio.

- Mass, inertia: $m \sim l^3$, $I \sim l^5$
- Thrust: $F \sim \pi r^2 \times (\omega r)^2 \Rightarrow F \sim l^2 v^2 \Rightarrow a = \frac{v^2}{l}$
- Moment: $M \sim Fl \Rightarrow M \sim l^3 v^2 \Rightarrow \alpha = \frac{v^2}{l^2}$

The relations above show that smaller quadrotors have higher maximum linear and angular accelerations. Hence, smaller aerial robots are more agile.

Chapter 2

Quadrotor Dynamics

In this chapter, the dynamics of a quadrotor using the Newton-Euler formalism is introduced. The motivation is derived from the work of Mellinger [9].

- Newton Euler Equation

$$\begin{bmatrix} F \\ \tau \end{bmatrix} = \begin{bmatrix} m\mathbf{1}_3 & 0_3 \\ 0_3 & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \alpha \end{bmatrix} + \begin{bmatrix} 0 \\ \omega \times \mathbf{I}_3 \omega \end{bmatrix} \tag{2.1}$$

In the Equation 2.1 the symbols have the following meaning, F is the net force acting on the quadrotor, $\mathbf{1}_3$ is a 3×3 indentity matrix, $\mathbf{a}$ is the linear acceleration of the centre of mass, ω is the angular velocity of the robot, m is the mass, $\mathbf{I}_3$ is the moment of inertia, $\mathbf{v}$ is the linear velocity, 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 cordinated from body frame to the world frame we use the ZXY Euler angles. It is give in the Equation below:

$$^{W}R_B = \begin{bmatrix} c\psi c\theta - s\phi s\psi s\theta & -c\phi s\psi & c\psi s\theta + c\theta s\phi s\psi \\ c\theta s\psi + s\phi c\psi s\theta & c\phi c\psi & s\psi s\theta - c\theta s\phi c\psi \\ -c\phi s\theta & s\phi & c\phi c\theta \end{bmatrix} \tag{2.2}$$

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 2.1 and 2.2.

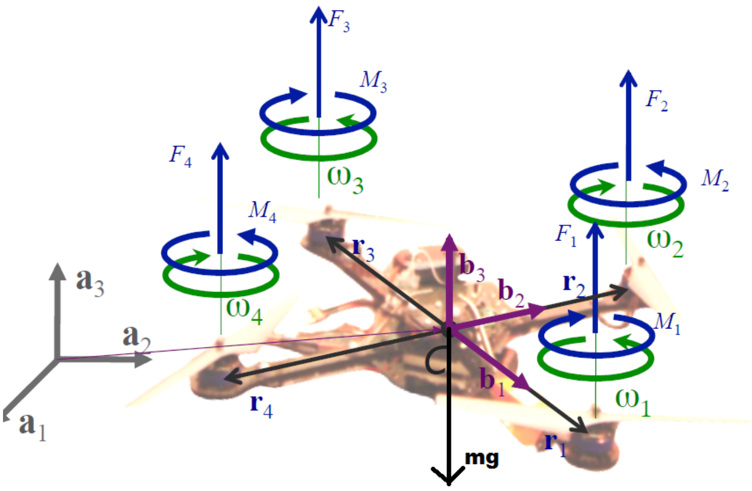

Figure 2.1: Free Body Diagram of the Quadrotor

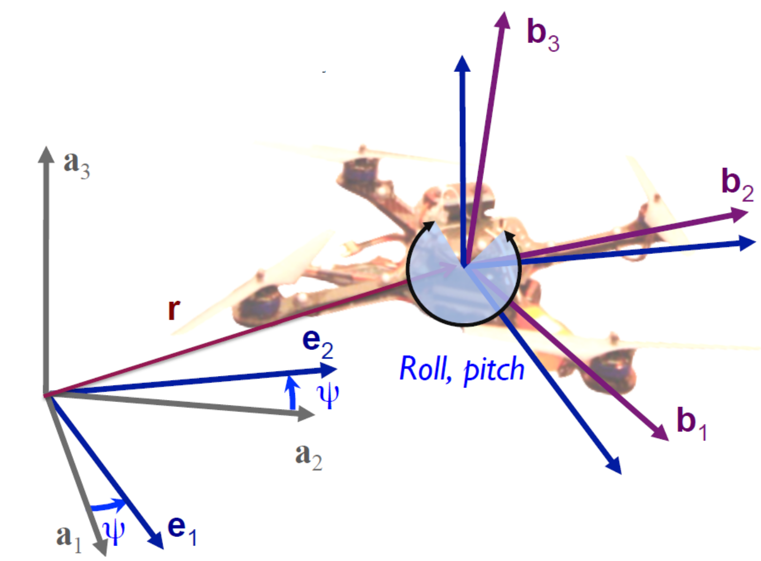

Figure 2.2: Different Frames

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

$$m\ddot{r} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} +^W R_B \begin{bmatrix} 0 \\ 0 \\ F_1 + F_2 + F_3 + F_4 \end{bmatrix} \tag{2.3}$$

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

$$\mathbf{I} \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \tau_{bx} \\ \tau_{by} \\ \tau_{bz} \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \mathbf{I} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.4}$$

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 2.1 that the rotation about b_1 is possible due 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 2.4 can written as Equation 2.5.

$$\mathbf{I} \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} L(F_2 - F_4) \\ L(F_3 - F_1) \\ M_2 + M_4 - M_1 - M_3 \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \mathbf{I} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.5}$$

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 2.6.

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} c\theta & 0 & -c\phi s\theta \\ 0 & 1 & s\phi \\ s\theta & 0 & c\phi c\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \rightarrow \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} c\theta & 0 & s\theta \\ s\theta t\phi & 1 & -c\theta t\phi \\ -\frac{s\theta}{c\phi} & 0 & \frac{c\theta}{c\phi} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.6}$$

It should be noted that in this model, the aerodynamic effects have been neglected for the sake of simplification.
The dynamic model is now establised in Equations 2.3 and 2.5. 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 2.3 and 2.5 as:

$$m\ddot{r} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} +^W R_B \begin{bmatrix} 0 \\ 0 \\ u_1 \end{bmatrix} \tag{2.7}$$

$$\mathbf{I}\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = u_2 - \begin{bmatrix}p\\ q\\ r\end{bmatrix} \times \mathbf{I}\begin{bmatrix}p\\ q\\ r\end{bmatrix} \tag{2.8}$$

Where, $u_1 = F_1 + F_2 + F_3 + F_4$ and $u_2 = \begin{bmatrix} L(F_2 - F_4) \\ L(F_3 - F_1) \\ M_2 + M_4 - M_1 - M_3 \end{bmatrix}$

Chapter 3

Control of Quadrotors

The doctoral work by Samir Bouabdallah [10] is one of best treatise on the various control strategies used for controlling quadrotors. In it, the controllers like the PID, LQ, Lyapunov, Backstepping and Integral Backstepping have been discussed lucidly with proper examples and simulations. The design process of a quadrotor is also discussed in [10]. Another master thesis work [11] is a very simple and readable reference on control of quadrotors. In this thesis, the author has presented MATLAB and SIMULINK based implementation of various control technics. This thesis has been used to simulate various control laws presented in this report. While most works have used Euler angles for modelling and control, the work [12] presents the use of quaternions. Quaternions do not suffer from gimbal locks/singularities. This work impresses upon the point that pitch and roll stabilization is much more important than yaw stabilization. Another interesting work on non linear control of quadrotors is [13]. This work is very detailed in the sense that it takes into account the dynamic models of rotors, gears and motors.

In this bibliographic work, a few control techniques have been studied and implemented in MATLAB/Simulink. The advantages and disadvantages have been discussed. The work covers very general linear and non-linear control technics. In Linear Control, the ubiquitous PD (Proportional Derivative) Control has been studied and in Non-Linear Control, technics like First Order Sliding Mode Control and the Back Stepping Control Law has been studied and implemented. It can be noted in general that the attitude and the position control can be treated as separate modules and various permutations and combinations of independent control laws can be applied to position control and attitude control. For example, the position control can be done in PD control while attitude stabilization can be done using the Lyapunov theory and vice versa.

Unless otherwise mentioned, the generalised control block diagram for

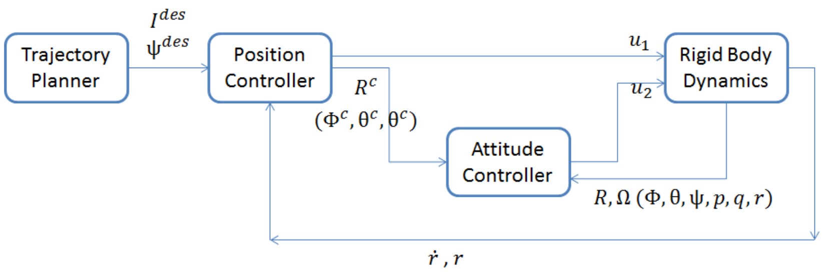

Figure 3.1: Control Block Diagram

control of a quadrotor is shown in Figure3.1. One can see in the figure that the inner loop is an attitude control loop while the outer loop is the position control loop. It is logical to infer that the dynamics of the inner loop must be faster than the dynamics of the outer loop. In hover configurations the dynamics of attitude do not matter much in general, but in cases where the robot has to make maneuvers, it is imperative to have a faster attitude control loop.

Another important aspect that is not discussed in detail is the trajectory generation process [9]. It is very important to generate a smooth trajectory between two points via several waypoints. Since the quadrotor is a dynamic entity with mass and moment of inertia, not using waypoints and higher order polynomial trajectories may result in requirement of discontinuities in the velocities and infinite accelerations which the actuators cannot deliver at any cost. Depending on what we want to minimize, we can have different kinds of trajectories, for example, minimum distance trajectory, minimum velocity trajectory (minimum distance trajectory is mathelatically the same as minimum velocity trajectory), minimum acceleration trajectory, minimum jerk trajectory or minimum snap trajectory. The idea is to use Calculus of Variations and the Euler Lagrange Equations to find a smooth trajectory with the provided boundary conditions. The idea is to design a spline for a n^{th} order system. In case of quadrotors, the relative degree of the position variables is 2 with respect to the input u_1 but it is 4 with respect to the input u_2. So, it is logical to deduce a Spline of 4th order, this is called the Minimum Snap Trajectory.

We shall not discuss more about the problem of trajectory generation because an helical trajectory is used in this case and an helical trajectory is infinitely differentiable. so, the desired acceleration ($\ddot{x}$), jerk($\dddot{x}$) and snap($\ddddot{x}$) (and other higher derivatives like pop and crackle) are all finite and bounded.

The model used in this section 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 practice, one has to take into account various actuator constraints. In the simulations presented here, the thrust input cannot be more than twice the weight of the matrix, similarly a suitable threshold is also put in the torque input. These thresholds have been put to make the control laws as practical as possible. In some cases very high inputs are generated but the actuators cannot comply with it.

In the following sections, Proportional Derivative Control, Sliding Mode Control and Backstepping Control are discussed and the results are presented. The effect of wind disturbance on the systems is presented and the performance of these control laws are discussed. The work with disturbances is motivated from [14].

The Figure shows the Trajectory being tracked in 3D and with other perspectives. This particular screens shot is taken when the Sliding Mode Control was used with the presense of wind.

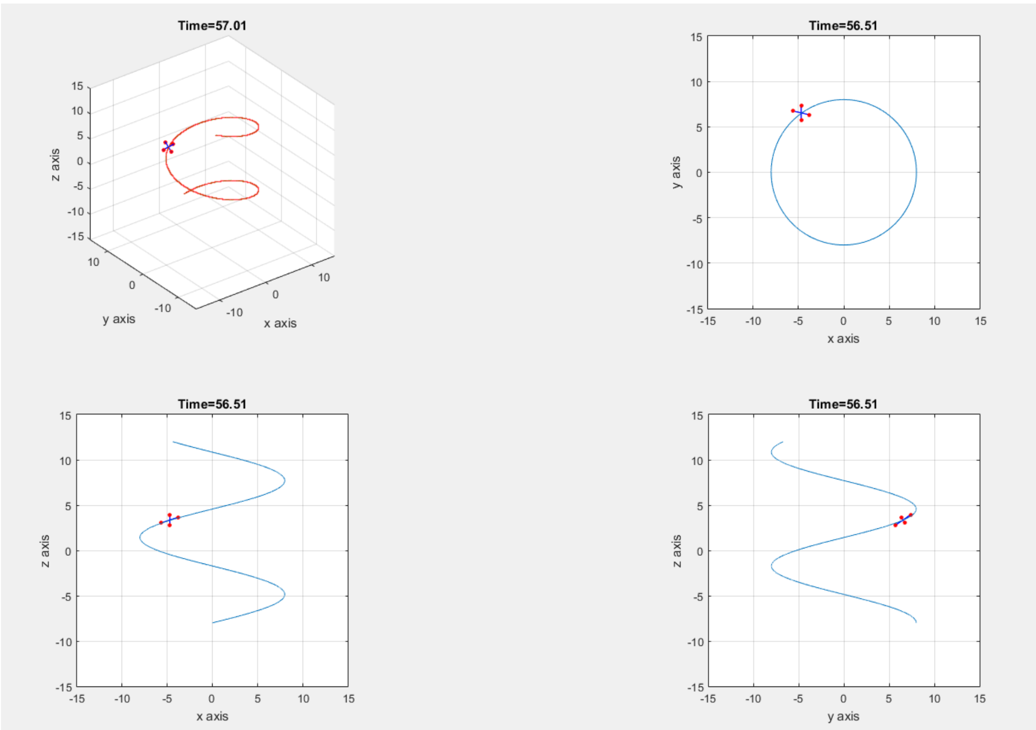

Figure 3.2: The Helical Trajectory

3.1 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.

3.1.1 Mathetmatics of PD Control of Quadrotors

The dynamic equations of the Quadrotor are given in Equations 3.1 and 3.2. The Equation 3.1 concerns the dynamics of linear equation and the Equation 3.2 concerns the dynamics of angular motion.

$$m\ddot{r} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} +^{W} R_B \begin{bmatrix} 0 \\ 0 \\ u_1 \end{bmatrix} \tag{3.1}$$

$$\mathbf{I} \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = u_2 - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \mathbf{I} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{3.2}$$

where,

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} c\theta & 0 & -c\phi s\theta \\ 0 & 1 & s\phi \\ s\theta & 0 & c\phi c\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \rightarrow \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} c\theta & 0 & s\theta \\ s\theta t\phi & 1 & -c\theta t\phi \\ -\frac{s\theta}{c\phi} & 0 & \frac{c\theta}{c\phi} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{3.3}$$

$[p, q, r]$: 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) and

$${}^{W}R_B = \begin{bmatrix} c\psi c\theta - s\phi s\psi s\theta & -c\phi s\psi & c\psi s\theta + c\theta s\phi s\psi \\ c\theta s\psi + s\phi c\psi s\theta & c\phi c\psi & s\psi s\theta - c\theta s\phi c\psi \\ -c\phi s\theta & s\phi & c\phi c\theta \end{bmatrix}$$

The desired trajectory is $\mathbf{r}_T = \begin{bmatrix} x_{des} \\ y_{des} \\ z_{des} \\ \psi_{des} \end{bmatrix}$

let us define:

$$
\begin{aligned}
e_p &= \mathbf{r}_T - \mathbf{r} \\
e_v &= \mathbf{r}_T - \mathbf{r}
\end{aligned}
$$

and we want

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

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: $u_1 \approx mg$, $\theta \approx 0$, $\phi \approx 0$, $\psi \approx \psi_0$, $u_2 \approx 0$, $u_2 \approx 0$, $p \approx 0$, $q \approx 0$ and $r \approx 0$.

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

$$
\begin{aligned}
\phi_c &= \frac{1}{g}(\ddot{r}_{1,c}sin(\psi_{des}) - \ddot{r}_{2,c}cos(\psi_{des})) \\
\theta_c &= \frac{1}{g}(\ddot{r}_{1,c}cos(\psi_{des}) + \ddot{r}_{2,c}sin(\psi_{des})) \\
\psi_c &= \psi_{des} \\
\ddot{r}_{3,c} &= \ddot{r}_{3,des} + k_{d,3}(\dot{r}_{3,des} - \dot{r}_3) + k_{p,3}(r_{3,des} - r_3)
\end{aligned}
\tag{3.4}
$$

The control laws can be written as:

$$u_1 = m(g + \ddot{r}_{3,c}) \tag{3.5}$$

$$u_2 = I \begin{bmatrix} k_{p,\phi}(\phi_c - \phi) + k_{d,\phi}(p_c - p) \\ k_{p,\theta}(\theta_c - \theta) + k_{d,\theta}(q_c - q) \\ k_{p,\psi}(\psi_c - \psi) + k_{d,\theta}(q_c - q) \end{bmatrix} \tag{3.6}$$

Using Equation 3.3 and 3.4, one can get $[p_c, q_c, r_c]^T$. Also note that $[r_1, r_2, r_3]^T = [x, y, z]^T$

3.1.2 Results

The objective of the control is to track an Helical Trajectory. The Figure 3.3 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 $V_w = 5\hat{\imath} + 5\hat{\jmath} + 5\hat{k}$ $m.s^{-1}$.

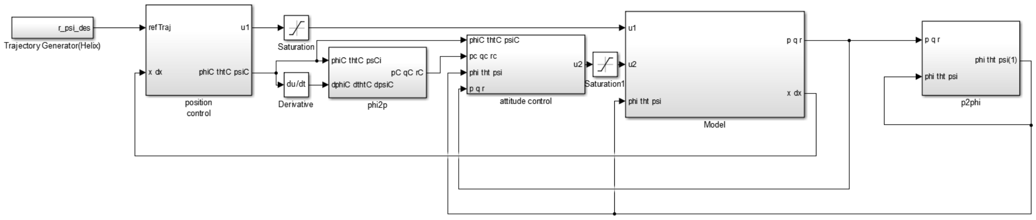

Figure 3.3: Global view of the Simulink model of the system with PD Control

Results sans Disturbances

Without any disturbance, the tracking by PD control is excellent and the tracking error is neglibible. The Figures 3.4, 3.5, 3.6 show the positions, orientations, control inputs and the tracjectory error in the absense of disturbances.

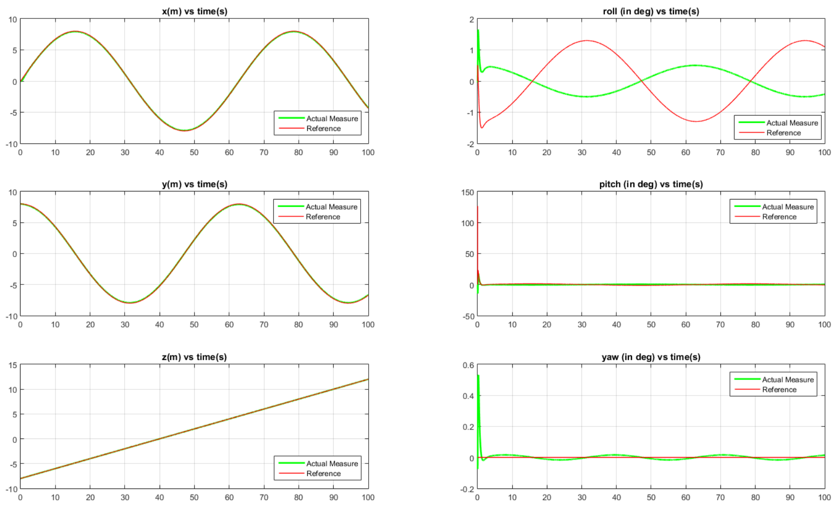

Figure 3.4: Position and Orientation vs Time

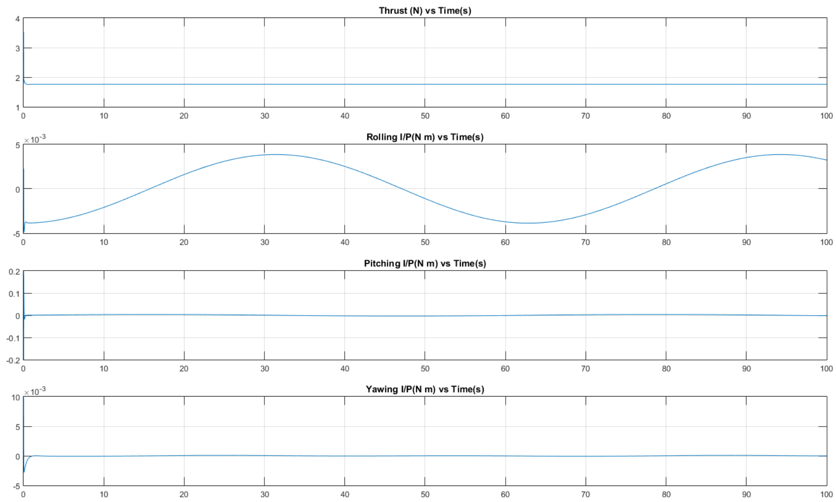

Figure 3.5: Thrust, Rolling, Pitching and Yawing Inputs vs Time

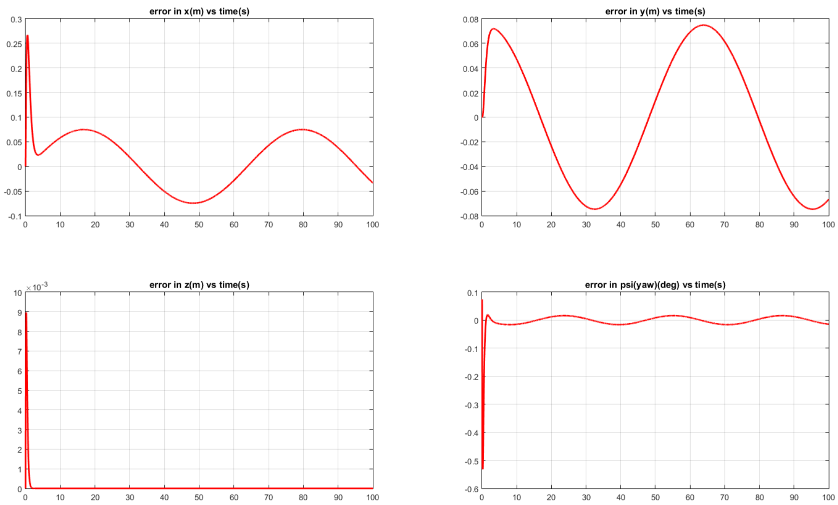

Figure 3.6: Errors in x,y,z and yaw

Results with Disturbances

The disturbance is wind of velocity $V_w = 5\hat{\imath} + 5\hat{\jmath} + 5\hat{k}$ at is applied as a step input at time $t = 25$ s. The Figures 3.7, 3.8, 3.9 show the positions, orientations, control inputs and the tracjectory error in the presence of disturbances. It can be concluded from the plots 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 ∞.

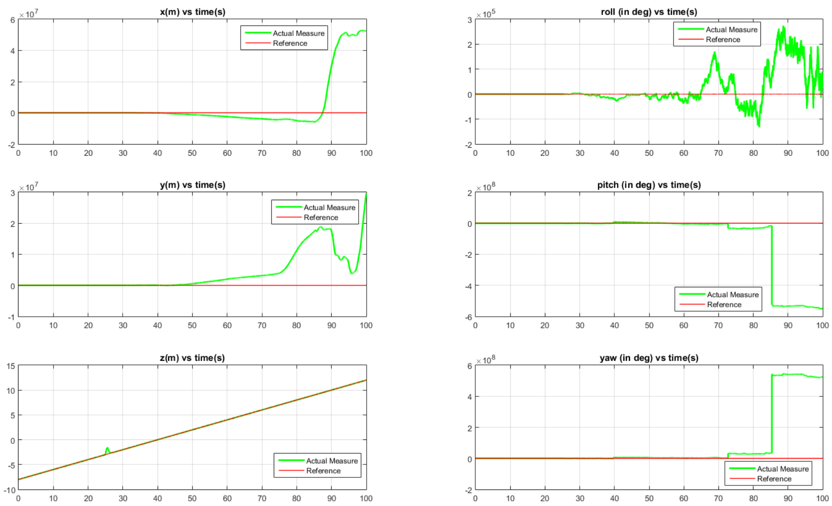

Figure 3.7: Position and Orientation vs Time [With Disturbance at 25s]

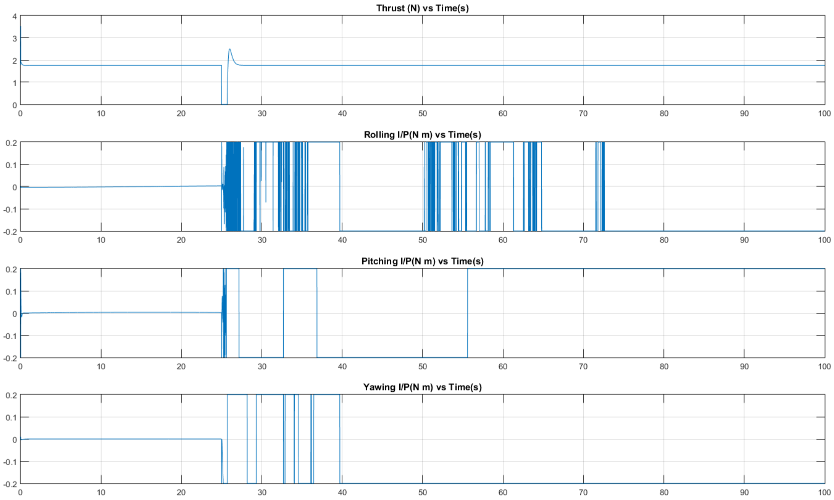

Figure 3.8: Thrust, Rolling, Pitching and Yawing Inputs vs Time [With Disturbance at 25s]

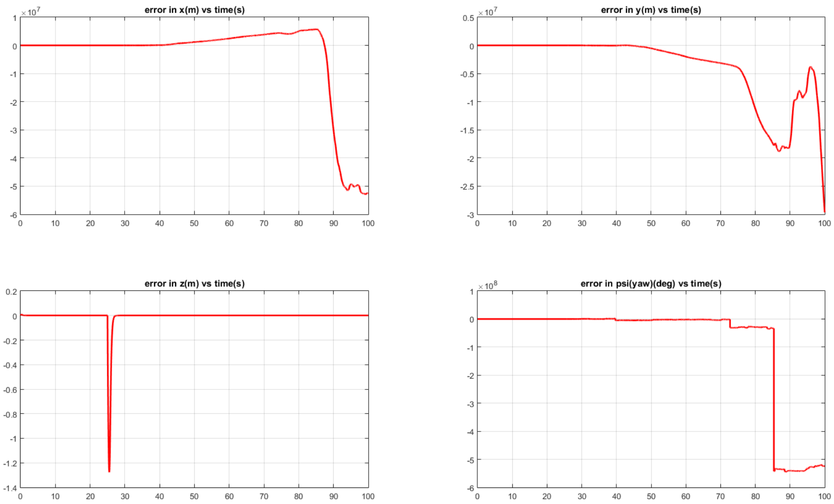

Figure 3.9: Errors in x,y,z and yaw [With Disturbance at 25s]

3.2 Sliding Mode Control

The Sliding Mode Control is a Non Linear Robust Control Technic which can be used to nullify the effect of disturbances and uncertainities. Sliding Mode Control is a type of Variable Structure Control(VSC). A high speed switching control is used to minimise the tracking error. The control used is discountinous. The control can be divided into two parts, one which does the input/output linearization and the other part helps to minimise the tracking error. The first order sliding mode control is used here. To make things simpler, the sliding mode control is just applied in the attitude control part and for the position control part the Proportional Derivative Control is used. The altitude control is fairly simple and there is no point in using complicated control laws for controlling it, hence the PD control is used for that part.

3.2.1 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 3.7:

$$\mathbf{I}\ddot{\omega} = u_2 - \dot{\omega} \times \mathbf{I}\dot{\omega} \Rightarrow \ddot{\omega} = -\mathbf{I}^{-1}\dot{\omega} \times \mathbf{I}\dot{\omega} + \mathbf{I}^{-1}u_2 \tag{3.7}$$

We have:

$$\omega = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, \omega_c = \begin{bmatrix} \phi_c \\ \theta_c \\ \psi_c \end{bmatrix}$$

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

From the Equation 3.7, it can be seen that 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,

$$\begin{aligned} \dot{s} &= \ddot{e} + \lambda \dot{e} \\ \dot{s} &= \ddot{\omega} - \ddot{\omega}_c + \lambda(\dot{\omega} - \dot{\omega}_c) \\ \dot{s} &= -\mathbf{I}^{-1}\dot{\omega} \times \mathbf{I}\dot{\omega} + \mathbf{I}^{-1}u_2 - \ddot{\omega}_c + \lambda(\dot{\omega} - \dot{\omega}_c) \\ \dot{s} &= -\mathbf{I}^{-1}\dot{\omega} \times \mathbf{I}\dot{\omega} - \ddot{\omega}_c + \lambda(\dot{\omega} - \dot{\omega}_c) + \mathbf{I}^{-1}u_2 \\ \dot{s} &= \alpha_{SM} + \beta_{SM}u_2 \end{aligned} \tag{3.8}$$

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 has to 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:

$$u_2 = \mathbf{I}(-(-\mathbf{I}^{-1} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \times \mathbf{I} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} - \begin{bmatrix} \ddot{\phi}_c \\ \ddot{\theta}_c \\ \ddot{\psi}_c \end{bmatrix} + \lambda(\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} - \begin{bmatrix} \dot{\phi}_c \\ \dot{\theta}_c \\ \dot{\psi}_c \end{bmatrix}) - K \begin{bmatrix} sign(s_1) \\ sign(s_2) \\ sign(s_3) \end{bmatrix}) \tag{3.9}$$

Where,

$$\lambda = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix}$$

$$K = \begin{bmatrix} K_1 & 0 & 0 \\ 0 & K_2 & 0 \\ 0 & 0 & K_3 \end{bmatrix}$$

3.2.2 Results

The objective of the control is to track an Helical Trajectory. The Figure 3.10 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 $V_w = 5\hat{\imath} + 5\hat{\jmath} + 5\hat{k}$ $m.s^{-1}$.

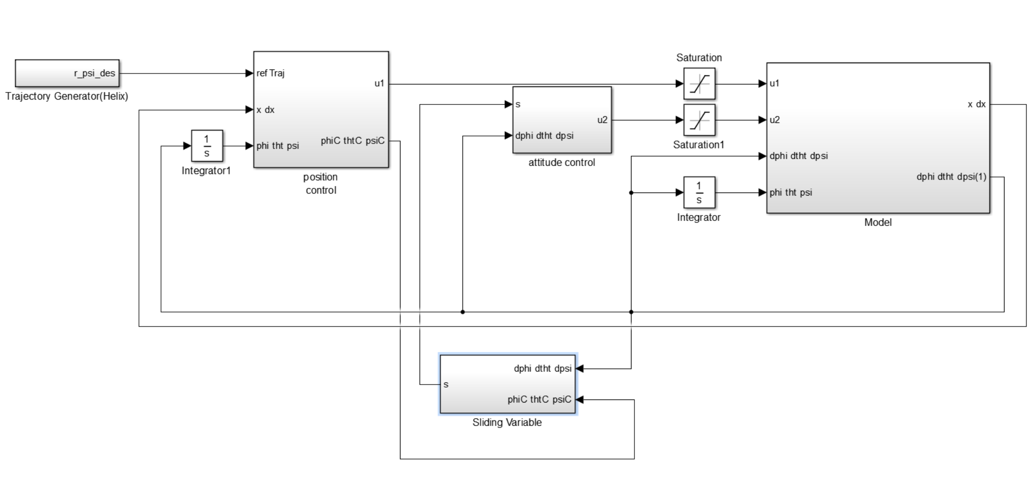

Figure 3.10: Global view of the Simulink model of the system with SMC Control

Results sans Disturbances

Without any disturbance, the tracking by SMC control is better than PD control. The magnitudes of errors is lesser than the case of PD control. But the control is discontinous and switches very frequently. The Figures 3.11, 3.12, 3.13 show the positions, orientations, control inputs and the tracjectory error in the absense of disturbances.

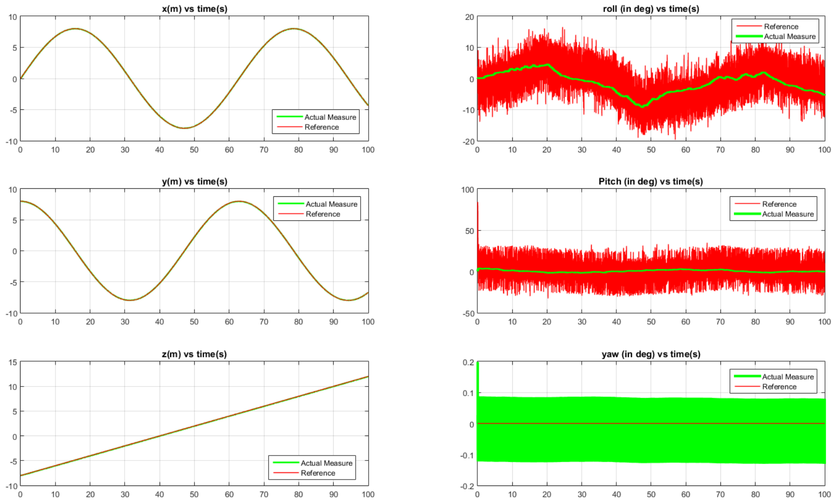

Figure 3.11: Position and Orientation vs Time

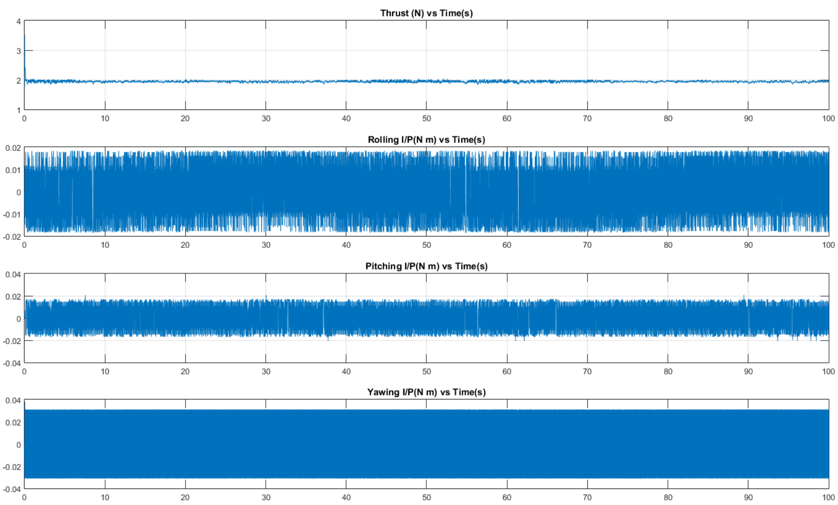

Figure 3.12: Thrust, Rolling, Pitching and Yawing Inputs vs Time

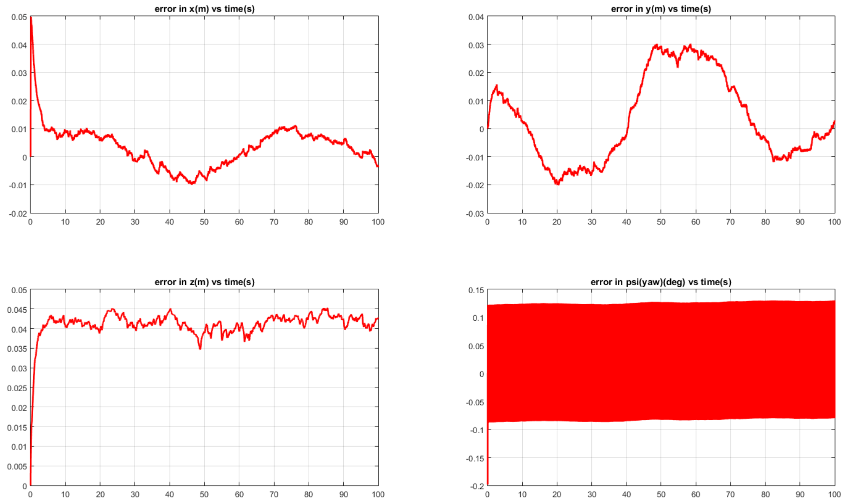

Figure 3.13: Errors in x,y,z and yaw

Results with Disturbances

The disturbance is wind of velocity $V_w = 5\hat{\imath} + 5\hat{\jmath} + 5\hat{k}$ at is applied as a step input at time $t = 25$ s. The Figures 3.14, 3.15, 3.16 show the positions, orientations, control inputs and the tracjectory 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.

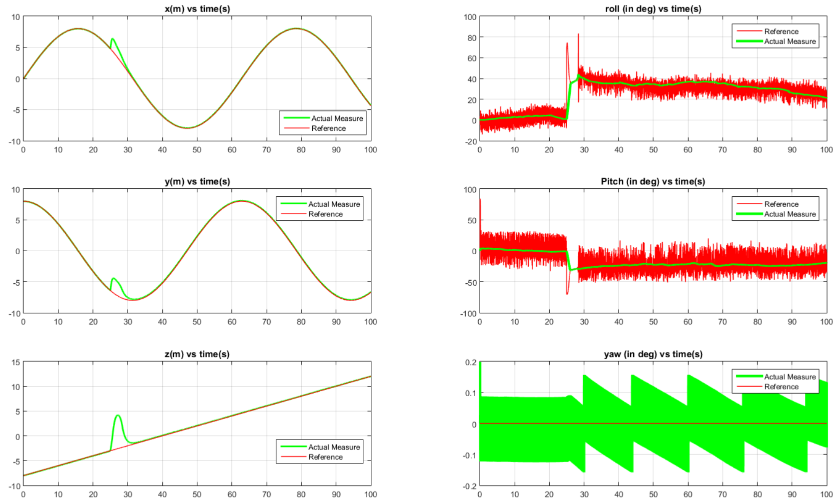

Figure 3.14: Position and Orientation vs Time [With Disturbance at 25s]

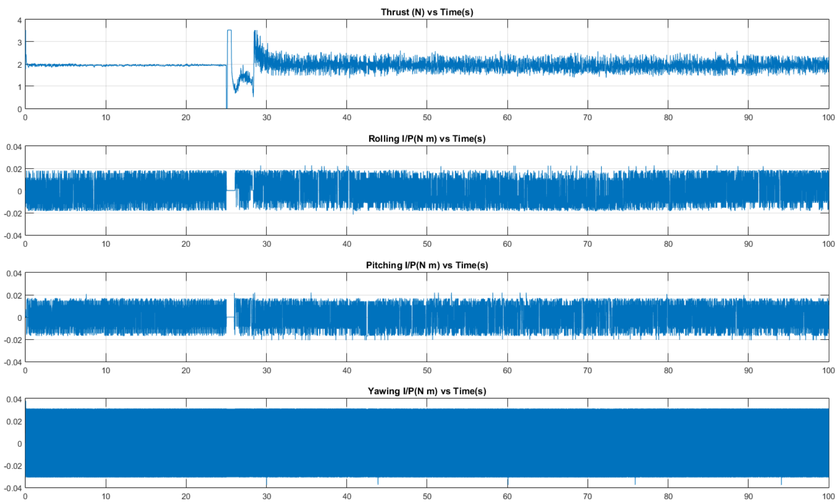

Figure 3.15: Thrust, Rolling, Pitching and Yawing Inputs vs Time [With Disturbance at 25s]

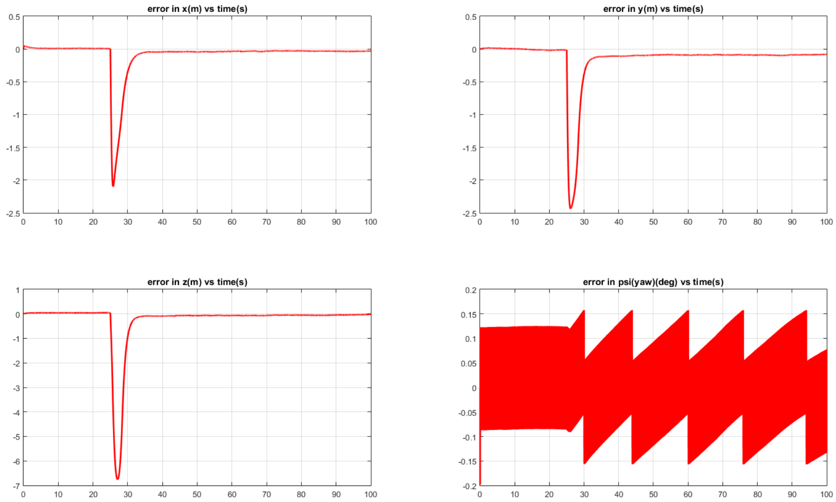

Figure 3.16: Errors in x,y,z and yaw [With Disturbance at 25s]

3.3 Back Stepping Control

Backstepping controller is used to control the 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.

3.3.1 Mathematics of Backstepping Control

The dynamic model presented in Equations 3.1 and 3.2 are re-written for the sake of convenience. The state vairables are renamed as follows: $x_1 = \phi$, $x_2 = \dot{x}_1$, $x_3 = \theta$, $x_4 = \dot{x}_2$, $x_5 = \psi$, $x_6 = \dot{x}_5$, $x_7 = z$, $x_8 = \dot{z}$, $x_9 = x$, $x_{10} = \dot{x}_9$, $x_{11} = y$ 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: $a_1 = \frac{I_{yy}-I_{xx}}{I_{xx}}$, $a_2 = \frac{I_{zz}-I_{xx}}{I_{yy}}$, $a_3 = \frac{I_{xx}-I_{yy}}{I_{zz}}$, $b_1 = \frac{1}{I_{xx}}$, $b_2 = \frac{1}{I_{yy}}$ and $b_3 = \frac{1}{I_{zz}}$.

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

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \\ \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{bmatrix} = \begin{bmatrix} x_2 \\ a_1 x_4 x_6 + b_1 u_{21} \\ x_4 \\ a_2 x_2 x_6 + b_2 u_{22} \\ x_6 \\ a_3 x_2 x_4 + b_3 u_{23} \\ x_8 \\ -g + b_4(\cos x_1 \cos x_3) u_1 \\ x_{10} \\ b_4(\cos x_5 \sin x_3 + \sin x_5 \cos x_3 \sin x_1) u_1 \\ x_{12} \\ b_4(\sin x_5 \sin x_3 - \cos x_5 \cos x_3 \sin x_1) u_1 \end{bmatrix} \tag{3.10}$$

The model developed in Equation 3.10 is used to implement the backstepping control.

Roll Controller

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

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ a_1 x_4 x_6 + b_1 u_{21} \end{bmatrix} \tag{3.11}$$

The roll angle equation is strictly in feedback form, that is to say 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:

$$V_1 = \frac{1}{2} z_1^2 \tag{3.12}$$

Here z_1 is the error between the desited and the actual roll angle defined as:

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

The time derivative of the function defined in Equation 3.12 is;

$$\dot{V}_1 = z_1 \dot{z}_1 = z_1(\dot{x}_{1c} - \dot{x}_1) = z_1(\dot{x}_{1c} - x_2) \tag{3.13}$$

According to Krasovaskii-LaSalle principle, the system is guatanted 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 an bound on $\dot{V}_1$ as given in Equation

$$\dot{V}_1 = z_1(\dot{x}_{1c} - x_2) \leq -c_1 z_1^2 \tag{3.14}$$

Here c_1 is a positive constant. To satisfy inequality 3.14 the virtual control input is chosen to be:

$$(x_2)_{desired} = \dot{x}_{1c} + c_1 z_1 \tag{3.15}$$

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

$$z_2 = x_2 - \dot{x}_{1c} - c_1 z_1 \tag{3.16}$$

Rewriting Equation 3.13

$$\begin{aligned} \dot{V}_1 &= z_1 \dot{z}_1 = z_1(\dot{x}_{1d} - x_2) = z_1(\dot{x}_{1d} - (z_2 + \dot{x}_{1d} + c_1 z_1)) \\ \Rightarrow \dot{V}_1 &= -z_1 z_2 - c_1 z_1^2 \end{aligned} \tag{3.17}$$

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.

$$\begin{aligned} V_2 &= V_1 + \frac{1}{2} z_2^2 \\ \Rightarrow \dot{V}_2 &= \dot{V}_1 + z_2 \dot{z}_2 \\ \Rightarrow \dot{V}_2 &= -z_1 z_2 - c_1 z_1^2 + z_2(\dot{x}_2 - \ddot{x}_{1c} - c_1 \dot{z}_1) \end{aligned} \tag{3.18}$$

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

$$-z_1 z_2 - c_1 z_1^2 + z_2(a_1 x_4 x_6 + b_1 u_{21} - \ddot{x}_{1d} - c_1 \dot{z}_1) \leq -c_1 z_1^2 - c_2 z_2^2 \tag{3.19}$$

Using the equality case of Equation 3.19, we get:

$$u_{21} = \frac{1}{b_1}(\ddot{x}_{1d} + c_1 \dot{z}_1 - a_1 x_4 x_6 + z_1 - c_2 z_2) \tag{3.20}$$

Pitch Controller

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

$$\begin{bmatrix} \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} x_4 \\ a_2 x_2 x_6 + b_2 u_{22} \end{bmatrix} \tag{3.21}$$

The pitch angle equation is strictly in feedback form, that is to say 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:

$$V_3 = \frac{1}{2} z_3^2 \tag{3.22}$$

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

$$z_3 = x_{3c} - x_3$$

The time derivative of the function defined in Equation 3.22 is;

$$\dot{V}_3 = z_3 \dot{z}_3 = z_3(\dot{x}_{3c} - \dot{x}_3) = z_3(\dot{x}_{3c} - x_4) \tag{3.23}$$

According to Krasovaskii-LaSalle principle, the system is guatanted 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 an bound on $\dot{V}_3$ as given in Equation

$$\dot{V}_3 = z_3(\dot{x}_{3c} - x_4) \leq -c_3 z_3^2 \tag{3.24}$$

Here c_3 is a positive constant. To satisfy inequality 3.24 the virtual control input is chosen to be:

$$(x_4)_{desired} = \dot{x}_{3c} + c_3 z_3 \tag{3.25}$$

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

$$z_4 = x_4 - \dot{x}_{3c} - c_3 z_3 \tag{3.26}$$

Rewriting Equation 3.23

$$\begin{aligned} &\dot{V}_3 = z_3 \dot{z}_3 = z_3(\dot{x}_{3d} - x_3) = z_3(\dot{x}_{3d} - (z_4 + \dot{x}_{3d} + c_3 z_3)) \\ \Rightarrow &\dot{V}_3 = -z_3 z_4 - c_3 z_3^2 \end{aligned} \tag{3.27}$$

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.

$$\begin{aligned} &V_4 = V_3 + \frac{1}{2} z_4^2 \\ \Rightarrow &\dot{V}_4 = \dot{V}_3 + z_4 \dot{z}_4 \\ \Rightarrow &\dot{V}_4 = -z_3 z_4 - c_3 z_3^2 + z_4(\dot{x}_4 - \ddot{x}_{3c} - c_3 \dot{z}_3) \end{aligned} \tag{3.28}$$

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

$$-z_3 z_4 - c_3 z_3^2 + z_4(a_2 x_2 x_6 + b_2 u_{22} - \ddot{x}_{3d} - c_3 \dot{z}_3) \leq -c_3 z_3^2 - c_4 z_4^2 \tag{3.29}$$

Using the equality case of Equation 3.49, we get:

$$u_{22} = \frac{1}{b_2}(\ddot{x}_{3d} + c_3 \dot{z}_3 - a_2 x_2 x_6 + z_3 - c_4 z_4) \tag{3.30}$$

Yaw Controller

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

$$\begin{bmatrix} \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} x_6 \\ a_3 x_2 x_4 + b_3 u_{23} \end{bmatrix} \tag{3.31}$$

The yaw angle equation is strictly in feedback form, that is to say 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:

$$V_5 = \frac{1}{2} z_5^2 \tag{3.32}$$

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

$$z_5 = x_{5c} - x_5$$

The time derivative of the function defined in Equation 3.32 is;

$$\dot{V}_5 = z_5 \dot{z}_5 = z_5(\dot{x}_{5c} - \dot{x}_5) = z_5(\dot{x}_{5c} - x_6) \tag{3.33}$$

According to Krasovaskii-LaSalle principle, the system is guatanted 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 an bound on $\dot{V}_5$ as given in Equation

$$\dot{V}_5 = z_5(\dot{x}_{5c} - x_6) \leq -c_5 z_5^2 \tag{3.34}$$

Here c_5 is a positive constant. To satisfy inequality 3.34 the virtual control input is chosen to be:

$$(x_6)_{desired} = \dot{x}_{5c} + c_5 z_5 \tag{3.35}$$

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

$$z_6 = x_6 - \dot{x}_{5c} - c_5 z_5 \tag{3.36}$$

Rewriting Equation 3.33

$$\begin{aligned} \dot{V}_5 &= z_5 \dot{z}_5 = z_5(\dot{x}_{5d} - x_5) = z_5(\dot{x}_{5d} - (z_6 + \dot{x}_{5d} + c_5 z_5)) \\ \Rightarrow \dot{V}_5 &= -z_5 z_6 - c_5 z_5^2 \end{aligned} \tag{3.37}$$

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.

$$\begin{aligned} V_6 &= V_5 + \frac{1}{2} z_6^2 \\ \Rightarrow \dot{V}_6 &= \dot{V}_5 + z_6 \dot{z}_6 \\ \Rightarrow \dot{V}_6 &= -z_5 z_6 - c_5 z_5^2 + z_6(\dot{x}_6 - \ddot{x}_{5c} - c_5 \dot{z}_5) \end{aligned} \tag{3.38}$$

Choosing a positive definite bounding function and substituing the model ($\dot{x}_6$) leads to the following:

$$-z_5 z_6 - c_5 z_5^2 + z_6(a_3 x_2 x_4 + b_3 u_{23} - \ddot{x}_{5d} - c_5 \dot{z}_5) \leq -c_5 z_5^2 - c_6 z_6^2 \tag{3.39}$$

Using the equality case of Equation 3.39, we get:

$$u_{23} = \frac{1}{b_3}(\ddot{x}_{5d} + c_5 \dot{z}_5 - a_3 x_2 x_4 + z_5 - c_6 z_6) \tag{3.40}$$

Altitude Controller

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

$$\begin{bmatrix} \dot{x}_7 \\ \dot{x}_8 \end{bmatrix} = \begin{bmatrix} x_8 \\ -g + b_4(\cos x_1 \cos x_3) u_1 \end{bmatrix} \tag{3.41}$$

The altitude equation is strictly in feedback form, that is to say 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:

$$V_7 = \frac{1}{2} z_7^2 \tag{3.42}$$

Here z_7 is the error between the desired and the actual roll angle defined as:

$$z_7 = x_{7c} - x_7$$

The time derivative of the function defined in Equation 3.22 is;

$$\dot{V}_7 = z_7 \dot{z}_7 = z_7(\dot{x}_{7c} - \dot{x}_7) = z_7(\dot{x}_{7c} - x_8) \tag{3.43}$$

According to Krasovaskii-LaSalle principle, the system is guatanted 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 an bound on $\dot{V}_7$ as given in Equation

$$\dot{V}_7 = z_7(\dot{x}_{7c} - x_8) \leq -c_7 z_7^2 \tag{3.44}$$

Here c_7 is a positive constant. To satisfy inequality 3.24 the virtual control input is chosen to be:

$$(x_8)_{desired} = \dot{x}_{7c} + c_7 z_7 \tag{3.45}$$

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

$$z_8 = x_8 - \dot{x}_{7c} - c_7 z_7 \tag{3.46}$$

Rewriting Equation 3.43

$$\begin{aligned} \dot{V}_7 &= z_7 \dot{z}_7 = z_7(\dot{x}_{7d} - x_7) = z_7(\dot{x}_{7d} - (z_8 + \dot{x}_{7d} + c_7 z_7)) \\ \Rightarrow \dot{V}_7 &= -z_7 z_8 - c_7 z_7^2 \end{aligned} \tag{3.47}$$

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.

$$\begin{aligned} V_8 &= V_7 + \frac{1}{2} z_8^2 \\ \Rightarrow \dot{V}_8 &= \dot{V}_7 + z_8 \dot{z}_8 \\ \Rightarrow \dot{V}_4 &= -z_3 z_4 - c_7 z_7^2 + z_8(\dot{x}_8 - \ddot{x}_{7c} - c_7 \dot{z}_8) \end{aligned} \tag{3.48}$$

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

$$-z_7 z_8 - c_7 z_7^2 + z_8(-g + b_4(\cos x_1 \cos x_3) u_1 - \ddot{x}_{7d} - c_7 \dot{z}_7) \leq -c_7 z_7^2 - c_8 z_8^2 \tag{3.49}$$

Using the equality case of Equation 3.49, we get:

$$u_{23} = \frac{1}{b_4 \cos x_1 \cos x_3}(\ddot{x}_{7d} + c_7 \dot{z}_7 - c_8 z_8 + g + z_7) \tag{3.50}$$

3.3.2 Results

The objective of the control is to track an Helical Trajectory. The Figure 3.17 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 $V_w = 5\hat{\imath} + 5\hat{\jmath} + 5\hat{k}$ $m.s^{-1}$.

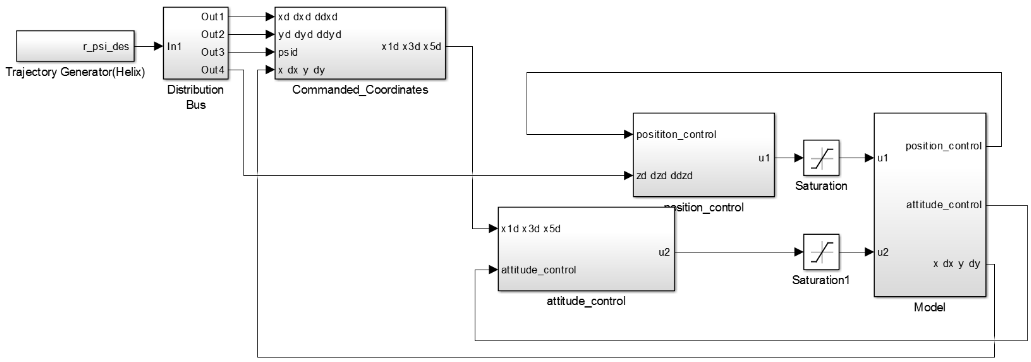

Figure 3.17: Global view of the Simulink model of the system with BSC Control

Results sans Disturbance

Without any disturbance, the tracking by Backstepping control is better than both Proportional Derivative and Sliding Mode Control. The magnitudes of errors is the least. The control is smooth and continous. The Figures 3.21, 3.22, 3.23 show the positions, orientations, control inputs and the tracjectory error in the absense of disturbances.

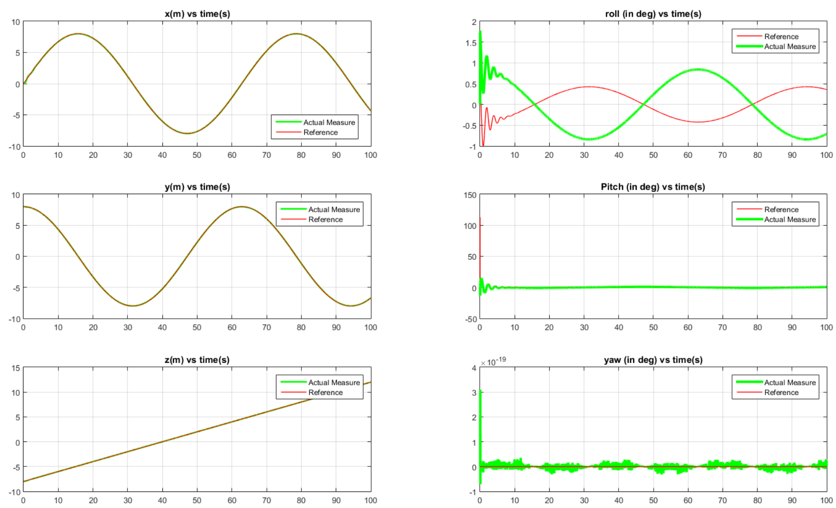

Figure 3.18: Position and Orientation vs Time

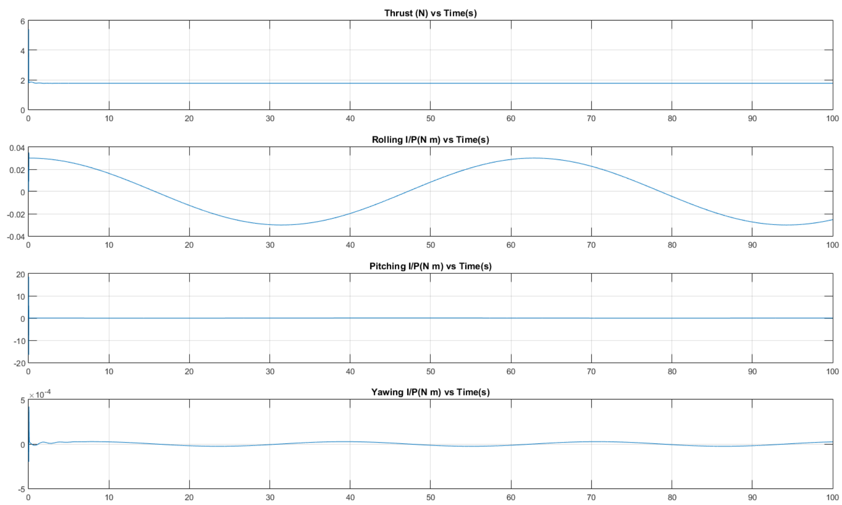

Figure 3.19: Thrust, Rolling, Pitching and Yawing Inputs vs Time

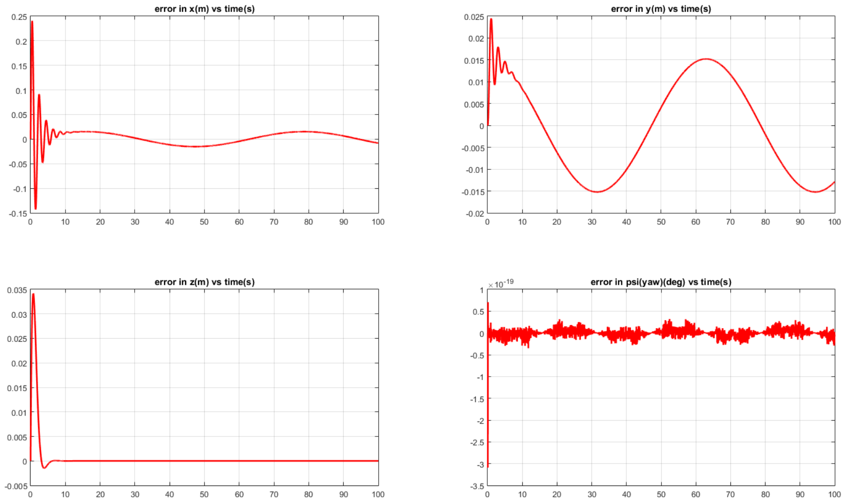

Figure 3.20: Errors in x,y,z and yaw

Results with Disturbance

The disturbance is wind of velocity $V_w = 5\hat{\imath} + 5\hat{\jmath} + 5\hat{k}$ at is applied as a step input at time $t = 25$ s. The Figures 3.21, 3.22, 3.23 show the positions, orientations, control inputs and the tracjectory 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.

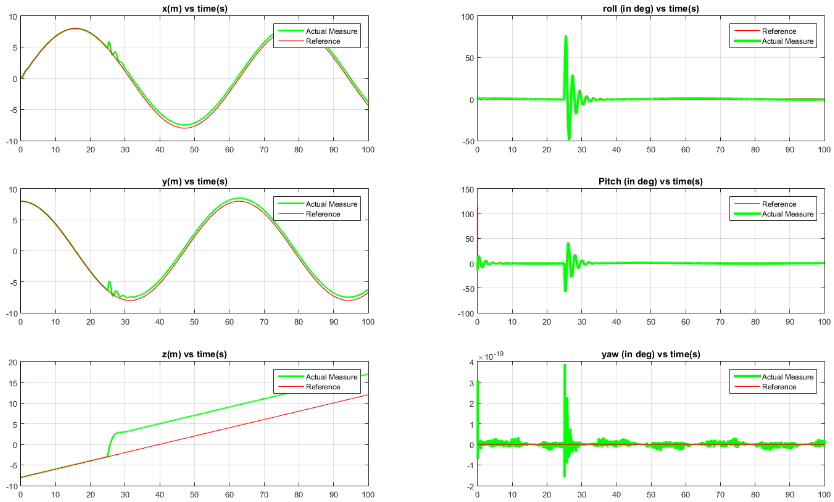

Figure 3.21: Position and Orientation vs Time [With Disturbance at 25s]

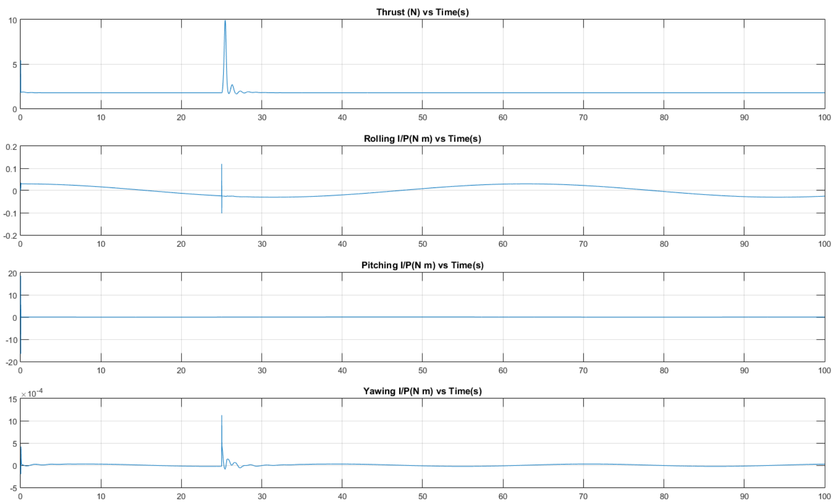

Figure 3.22: Thrust, Rolling, Pitching and Yawing Inputs vs Time [With Disturbance at 25s]

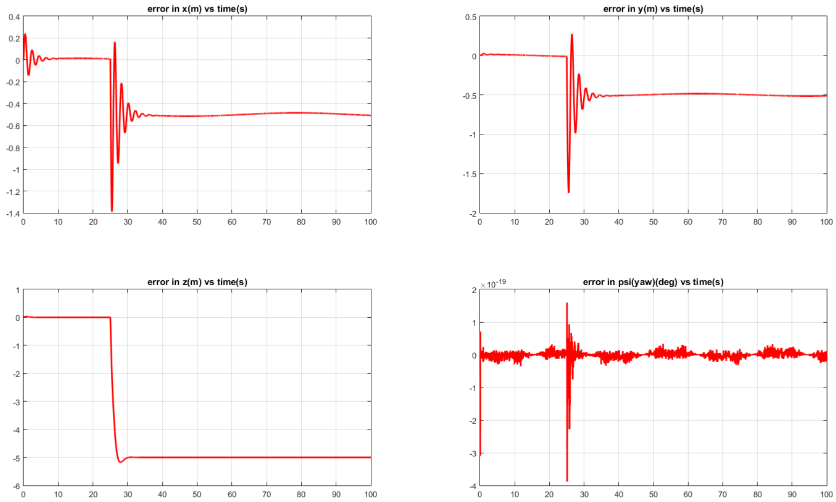

Figure 3.23: Errors in x,y,z and yaw [With Disturbance at 25s]

Chapter 4

Conclusion

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

4.1 Summary of Proportional Derivative Control

The advantages of Proportional Derivative (PD) Control are::

a. Easy to understand and implement.

b. Less 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.

4.2 Summary of 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.

4.3 Summary of 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 a number of 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.

[Disturbance/Control]	$max\|e_x\|(m)$	$max\|e_y\|(m)$	$max\|e_z\|(m)$	$max\|e_\psi\|$(deg)	Control
Without Disturbance/PD	0.0747	0.0747	0	0.016	Low and Smooth
Without Disturbance/SMC	0.01	0.03	0	0.123	Discontinous
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	Discontinous
With Disturbance/BSC	0.5	0.5	5	0	Low and Smooth

Table 4.1: Comparison Table

References

[1] Farid Kendoul, Survey of Advances in Guidance, Navigation, and Control of Unmanned Rotorcraft Systems. Journal of Field Robotics 29(2), 315378 (2012)

[2] Robert Mahony, Vijay Kumar and Peter Corke, Modeling, Estimation, and Control of Quadrotor. Digital Object Identifier 10.1109/MRA.2012.2206474, Date of publication: 27 August 2012

[3] H. Jin Kim and David H. Shim, A flight control system for aerial robots:algorithms and experiments. Control Engineering Practice 11 (2003) 13891400

[4] Vijay Kumar, Robotics: Aerial Robotics.

[5] Daniel Cremers and Jurgen Sturm, Autonomous Navigation for Flying Robots.

[6] Guillaume Ducard and Raffaello D'Andrea, Autonomous Quadrotor Flight Using a Vision System And Accommodating Frames Misalignment.

[7] Abraham Bachrach, Samuel Prentice, Ruijie He and Nicholas Roy, RANGE - Robust Autonomous Navigation in GPS-denied Environments. Published in the Journal of Field Robotics (JFR 2011).

[8] Pratik Agarwal and Tom Brady, SLAM Strategy for an Autonomous Quadrotor.

[9] Daniel Warren Mellinger, Trajectory Generation and Control for Quadrotors. Publicly Accessible Penn Dissertations. Paper 547.

[10] Samir Bouabdallah, Design and Control of the Quadrotors with Application to Autonomous Flying. Doctoral Thesis, EPFL

[11] Heba Talla Mohamed Nabil ElKholy, Dynamic Modelling and Control of a Quadrotor Using Linear and Non Linear Approaches.

[12] Dario Brescianini, Markus Hehn and Raffaello D'Andrea, Nonlinear Quadrocopter Attitude Control. Technical Report, Institute for Dynamic Systems and Control (IDSC), ETH Zurich.

[13] Patrick Adigbli, Nonlinear Attitude and Position Control of a Micro Quadrotor using Sliding Mode and Backstepping Techniques. 3rd US-European Competition and Workshop on Micro Air Vehicle Systems (MAV07) & European Micro Air Vehicle Conference and Flight Competition (EMAV2007), 17-21 September 2007, Toulouse, France

[14] Oualid Araar and Nabil Aouf, Quadrotor Control for Trajectory Tracking in Presence of Wind Disturbances. 2014 UKACC International Conference on Control 9th - 11th July 2014, Loughborough, U.K