Control of Quadrotor for Suspeended Load

A Project Report Submitted

in Partial Fulfillment of the Requirements

for the Course

AE640

2017-18 [Semester II]

by

ABHINAV TRIPATHI 150023

ANKISH BANSAL 17204004

to

DR MANGAL KOTHARI

DEPARTMENT OF AEROSPACE ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY,KANPUR

April 2018

Abstract

In the recent years, UAVs and micro UAVs have been employed for various automation tasks like surveillance, video monitoring and load transport.

The past few years have seen a mounting interest in research on drone delivery services, not only for military purposes but also in the distribution of consumer goods by the e-commerce companies. Sway reduction is an important aspect of any modern delivery system. This project discusses in detail the methodology of time delayed position feedback control for reducing the oscillations in the payload, which is suspended from a quadcopter.

Acknowledgements

We would like to thank our project supervisor Dr. M. Kothari for giving us a great opportunity to work in the field of sway reduction for UAV slung loads under his mentorship. We are grateful to Dr S. R. Sahoo for helping us with controller analysis. We are thankful to Souradip De, Mahathi, Mohit Sahai and Kuldeep Dhiman for all the discussions and support in this project.

Contents

List of Figures

Chapter 1

Introduction

This project addresses the problem of using unmanned aerial vehicles for transportation of suspended loads. The main contribution of this project is development of a novel control strategy that can be easily implemented on autonomous quadrotors in order to suppress the vibrations of the slung load while carrying it to its desired location.

1.1 Motivation

The effect of time delay controllers on the stability of oscillatory systems is of great importance. Studies show that the introduction of time delay controller can have stabilizing or destabilizing impact on a oscillations depending upon the gain and delay values chosen.
In [5], the authors show that large pendulations (caused by the wave-induced motions of a ship) of cargo being hoisted by a ship-mounted crane can be damped out using a control law of delayed position feedback. A planar pendulum with a rigid massless cable and massive point load is used to model the system.
The idea is extended to a spherical pendulum in [1]. We will extend the idea of time delayed position feedback to a payload hanging from a quadcopter instead of a ship-mounted crane.

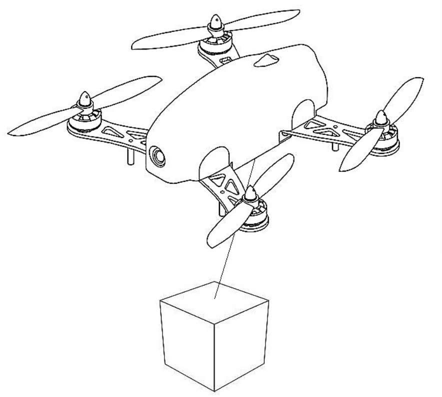

Figure 1.1: Schematic diagram of a typical delivery drone

1.2 Organization of the Report

In the next chapter, we will analyze the controller which is used for controlling the vibrations in our system. In chapter 3, we discuss position and attitude tracking of a quadrotor. In chapter 4, we discuss the control of quadcopter with slung load. Simulations for the same are shown in the last chapter.

Chapter 2

Time-Delayed Position Feedback Controller

In this chapter, we will show that a delayed position feedback for oscillating systems can produce stabilizing or destabilizing effects based on the values of the parameters (gain and delay) chosen.

We will derive our results for stabilizing the oscillations in a simple pendulum and later on show that the mathematics behind a 3-dimensional pendulum can be broken down into equations which closely resemble the ones for simple pendulum.

2.1 Sway Reduction in Simple Pendulum

For the understanding of the readers, we explain the working of time delayed position feedback for a simple pendulum. A detailed analysis of the same for simple and 3-dimensional pendulum can be found in [5] and [1].

Consider a pendulum P hanging from a cable of length l from a movable support Q as shown in figure 2.1. Let the position vector of the pendulum be $\mathbf{r_P}$, and that of the point of suspension be $\mathbf{r_Q}$, then we can write

$$\mathbf{r_P} = \mathbf{r_Q} + l(\sin\theta\,\mathbf{i} - \cos\theta\,\mathbf{k}) \tag{2.1}$$

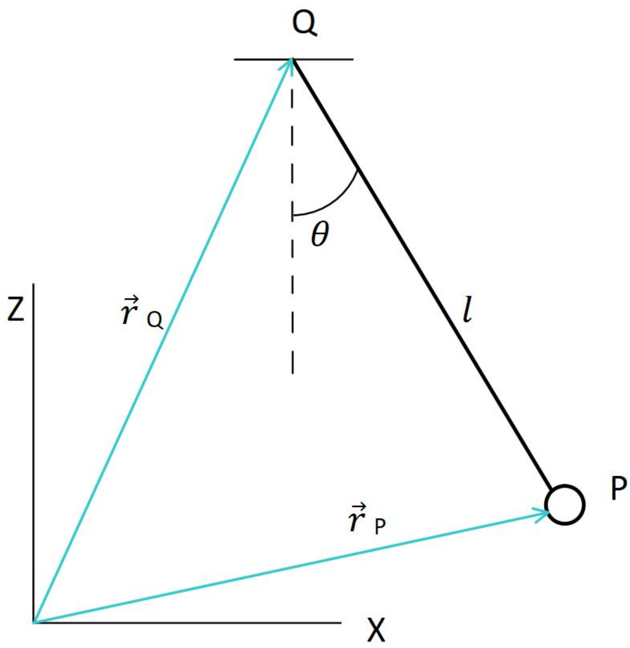

Figure 2.1: A Simple Pendulum

Take $\mathbf{r_Q} = x_Q\mathbf{i} + z_Q\mathbf{k}$, the above equation can be rewritten as

$$\mathbf{r_P} = (x_Q + l\sin\theta)\,\mathbf{i} + (z_Q - l\cos\theta)\,\mathbf{k} \tag{2.2}$$

Define $L = T - V$ where L denotes the Lagrangian, T stands for total kinetic energy and V stands for total potential energy. For our case, $T = \frac{1}{2}m|\dot{\mathbf{r_P}}|^2$ and $V = mg\,(z_Q - l\cos\theta)$.

We use Euler-Lagrange Equation,

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\dot{\theta}}\right) = \frac{\partial L}{\partial\theta} \tag{2.3}$$

It can be easily shown that on substituting the value of L in the above equation, we get

$$\ddot{x_Q}\cos\theta + \ddot{z_Q}\sin\theta + l\ddot{\theta} + g\sin\theta = 0 \tag{2.4}$$

For simplification, we shall assume that the acceleration of the point of suspension Q in vertical direction is zero. This reduces our equation to

$$\ddot{x_Q}\cos\theta + l\ddot{\theta} + g\sin\theta = 0 \tag{2.5}$$

Note that θ can be assumed to be small enough to make small angle approximations. Also, x_Q is the X coordinate of the position of point of suspension of the pendulum. Moving this point introduces pseudo force on the pendulum in Q frame. This is the only control input which available at hand for control of oscillations of P. Take

$$x_Q(t) = x_o(t) + kl\sin\theta\,(t-\tau) \tag{2.6}$$

Here k is the gain, l is the length of the pendulum and τ is known as delay. $x_o(t)$ gives us the variation in position of Q when the feedback is zero. In simplest terms, equation 2.6 tells us to vary the current position of Q taking into account the angular displacement of the pendulum τ time ago. This is the most basic form of time delayed position feedback.

From equations 2.5 and 2.6, we get

$$\ddot{\theta}(t) + \frac{g}{l}\theta(t) + k\ddot{\theta}(t-\tau) + \frac{\ddot{x_o}(t)}{l} = 0 \tag{2.7}$$

Such equations which have delays and derivatives of the same term are popularly know as *neutral delay differential equations*. Since the direct analysis of such equations is difficult, we will use other methods to analyze the variation of $\theta(t)$.

We can find out the transfer function of $\Theta(s)$ vs $X_o(s)$ by taking Laplace Transform of equation 2.6:

$$\frac{\Theta(s)}{X_o(s)} = \frac{-\frac{1}{l}\,s^2}{s^2\,(1+k\exp(-\tau s)) + \frac{g}{l}} \tag{2.8}$$

2.1.1 Stability Analysis

In this section, we try to estimate the values of gain k and time delay τ for which the system will remain stable. The characteristic equation for the system is

$$E(s) = s^2\,(1+k\exp(-\tau s)) + \frac{g}{l} \tag{2.9}$$

In [1], it was shown that the amount of damping (the darker the region, the more the damping) depends on the values of gain and delay as shown in figure 2.2. The rest of the region is unstable.

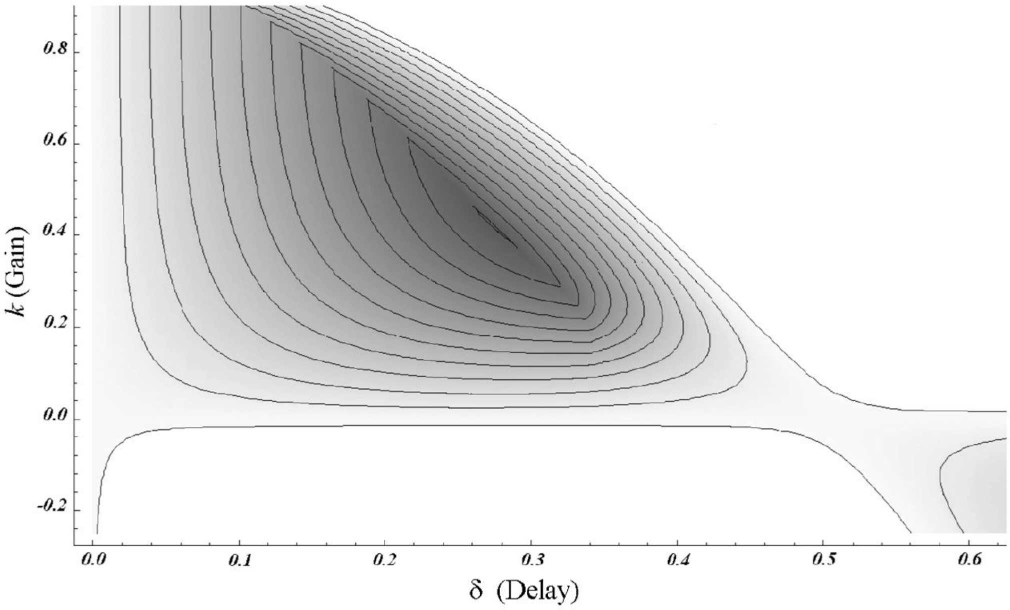

Figure 2.2: Damped Oscillations for Darker Regions (Ref: [1])

2.1.2 Simulations

In this section, simulations have been carried out for

$$\text{Mass of Pendulum} = m = 200 \text{ g}$$
$$\text{Length of Pendulum} = l = 0.5 \text{ m}$$
$$\text{Trajectory without position feedback controller } x_o(t) = 4t - 8 \text{ m}, \; t \geq 2 \text{ s}$$
$$\text{Trajectory with position feedback controller } x_Q(t) = x_o(t) + kl\sin\theta\,(t-\tau) \text{ m}$$
$$\text{Natural Swing Time Period} = T_o = 2\pi\sqrt{\frac{l}{g}} = 1.4185 \text{ s}, \; g = 9.81 \text{ m/s}^2$$

Let us define δ as normalized delay, where

$$\delta = \frac{\tau}{T_o} \tag{2.10}$$

1. Gain $(k) = 0.5$ and Delay $(\delta) = 0.001$

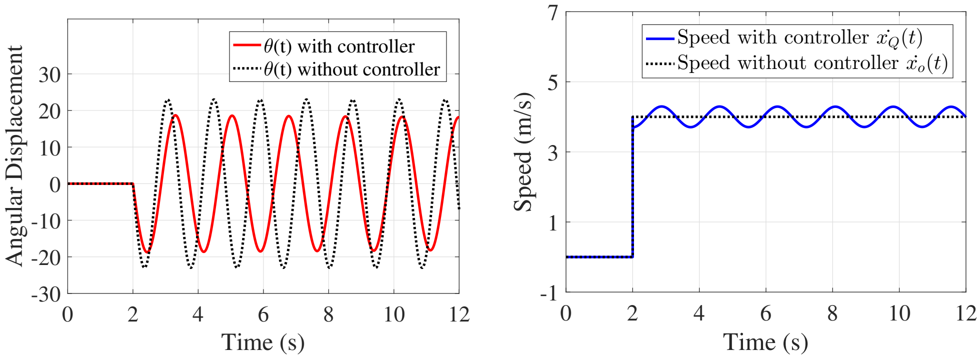

Figure 2.3: Angle and Speed for Gain $(k) = 0.5$ and Delay $(\delta) = 0.001$

2. Gain $(k) = 0.5$ and Delay $(\delta) = 0.1$

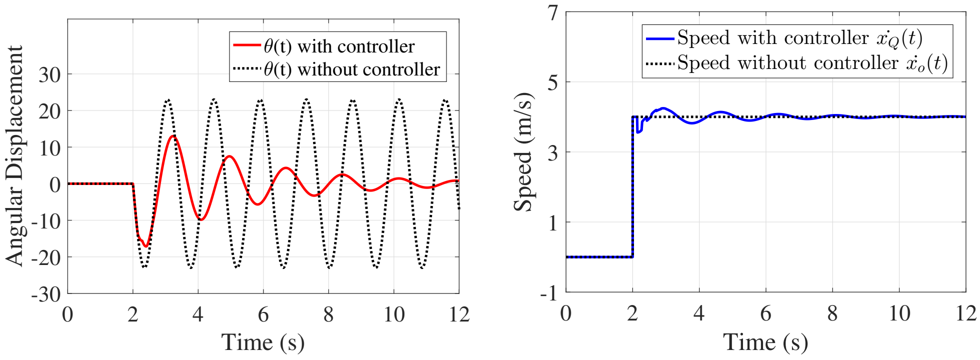

Figure 2.4: Angle and Speed for Gain $(k) = 0.5$ and Delay $(\delta) = 0.1$

3. Gain $(k) = 0.5$ and Delay $(\delta) = 0.2$

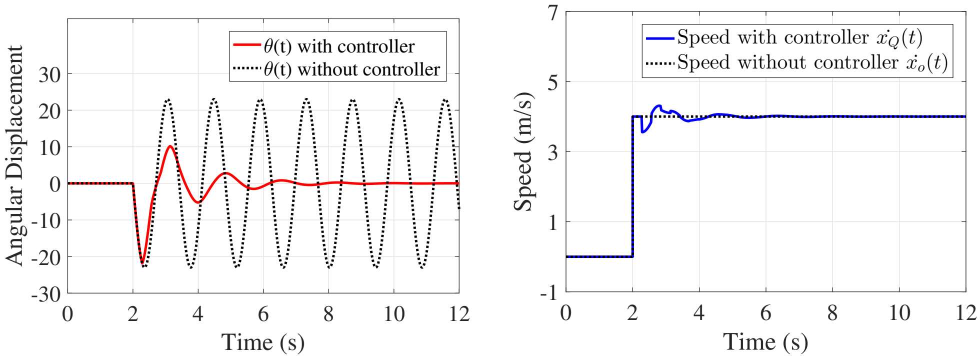

Figure 2.5: Angle and Speed for Gain $(k) = 0.5$ and Delay $(\delta) = 0.2$

4. Gain $(k) = 0.5$ and Delay $(\delta) = 0.3$

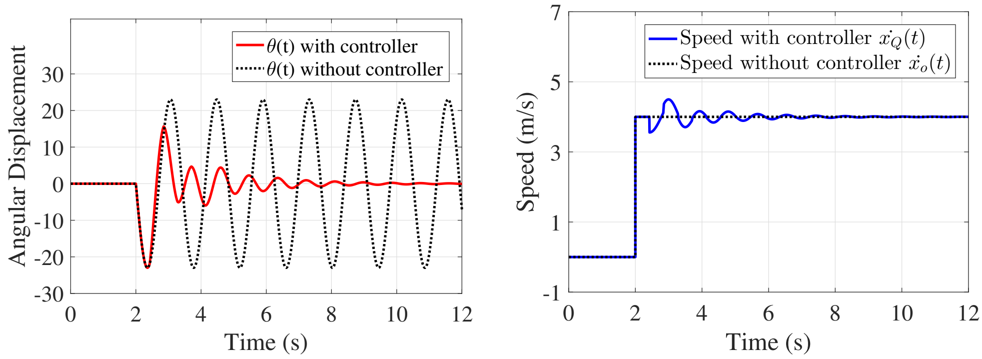

Figure 2.6: Angle and Speed for Gain $(k) = 0.5$ and Delay $(\delta) = 0.3$

5. Gain $(k) = 0.5$ and Delay $(\delta) = 0.4$

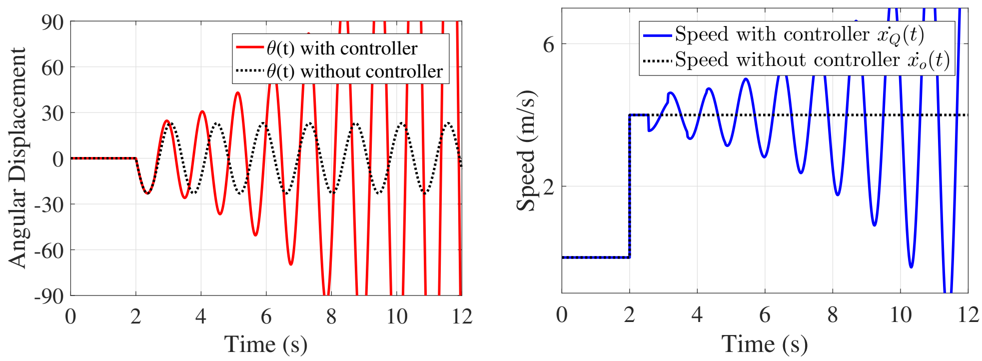

Figure 2.7: Angle and Speed for Gain $(k) = 0.5$ and Delay $(\delta) = 0.4$

2.2 Sway Reduction for a 3-dimensional Pendulum

In this section, we will show that the equations of a 3-dimensional pendulum can be broken down into equations similar to a simple pendulum as in equation 2.7.

Consider a pendulum P hanging from a movable support Q as shown in figure 2.8. In

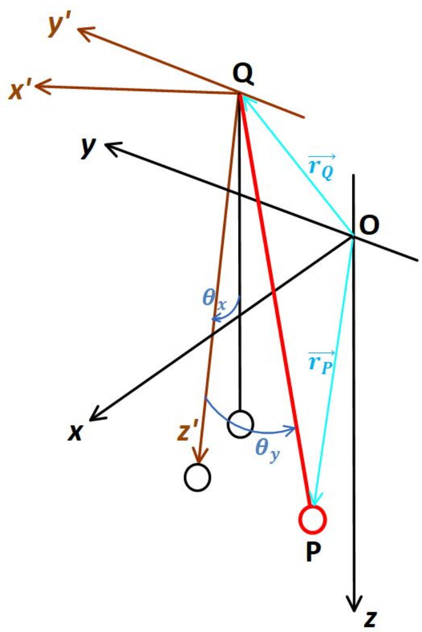

Figure 2.8: Schematic for a 3-D Pendulum

this case, the position of the pendulum is given by

$$\begin{aligned}\mathbf{r_P} &= [x_Q(t) + \sin(\theta_x(t))\cos(\theta_y(t))L]\mathbf{i} + [y_Q(t) - \sin(\theta_y(t))L]\mathbf{j} \\ &\quad + [z_Q(t) + \cos(\theta_x(t))\cos(\theta_y(t))L]\mathbf{k} \qquad (2.11)\end{aligned}$$

It can be shown using Euler-Lagrange Method that if $\dot{z_Q}(t) = 0$, the equations of motion for the pendulum are

$$\ddot{\theta_x}(t)\,\cos(\theta_y(t)) - 2\,\sin(\theta_y(t))\dot{\theta_x}(t)\dot{\theta_y}(t) + \frac{g}{L}\sin(\theta_x(t)) + \frac{1}{L}\cos(\theta_x(t))\ddot{x_Q}(t) = 0 \quad (2.12)$$

$$\ddot{\theta_y}(t) + \sin(\theta_y(t))\cos(\theta_y(t))\dot{\theta_x}^2(t) + \frac{g}{L}\sin(\theta_y(t))\cos(\theta_x(t)) -$$
$$\frac{1}{L}\sin(\theta_x(t))\sin(\theta_y(t))\ddot{x_Q}(t) - \frac{1}{L}\cos(\theta_y(t))\ddot{y_Q} = 0 \quad (2.13)$$

On making small angle approximations, these above equations can be reduced to

$$\ddot{\theta_x}(t) + \frac{g}{L}(\theta_x(t)) + \frac{1}{L}\ddot{x_Q}(t) = 0 \quad (2.14)$$
$$\ddot{\theta_y}(t) + \frac{g}{L}(\theta_y(t)) - \frac{1}{L}\ddot{y_Q} = 0 \quad (2.15)$$

These equations are very similar to those which we get for a simple pendulum.

We define our control equations for time delayed position feedback as

$$x_Q(t) = x_o(t) + k_x L\sin(\theta_x\,(t-\tau_x))\cos(\theta_y\,(t-\tau_x)) \quad (2.16)$$
$$y_Q(t) = y_o(t) - k_y L\sin(\theta_y(t-\tau_y)) \quad (2.17)$$

On substituting these values in equations of system dynamics, we get

$$\ddot{\theta_x}(t) + \frac{g}{L}(\theta_x(t)) + \frac{1}{L}\ddot{x_o}(t) + k_x\ddot{\theta_x}(t-\tau_x) = 0 \quad (2.18)$$
$$\ddot{\theta_y}(t) + \frac{g}{L}(\theta_y(t)) + \frac{1}{L}\ddot{y_o}(t) + k_y\ddot{\theta_y}(t-\tau_y) = 0 \quad (2.19)$$

These equations are same as those for simple pendulum and hence, same analysis will remain valid for this case as well.

Chapter 3

Position and Attitude Control for a Quadcopter

In this chapter, we introduce quadcopter dynamics and control to follow a trajectory.

3.1 Quadcopter Dynamics

3.1.1 Reference Frame

We introduce inertial and body frame in which our quadcopter dynamics will relate. The inertial frame is defined by ground, where gravity pointing in the negative z direction shown in left side of fig 3.1. The body frame is defined by the orientation of the quadcopter, whose rotor axis pointing in the positive z direction and arms pointing in the x and y directions shown in right side of the fig 3.1.

3.1.2 Vehicle Kinematics

We define the position and velocity of quadcopter in the inertial frame as $\mathbf{X} = (x, y, z)^T$ and $\dot{\mathbf{X}} = (\dot{x}, \dot{y}, \dot{z})^T$ respectively. Similarly, we define the angular displacement and angular velocity in the body frame as $\mathbf{\Theta} = (\phi, \theta, \psi)^T$ and $\dot{\mathbf{\Theta}} = (\dot{\phi}, \dot{\theta}, \dot{\psi})^T$ respectively, where (ϕ, θ, ψ) represent roll, pitch and yaw angle respectively. Note that, angular velocities $(\dot{\phi}, \dot{\theta}, \dot{\psi})$ are the scalar quantities, and we need angular velocity in vector form. For this, we will use following

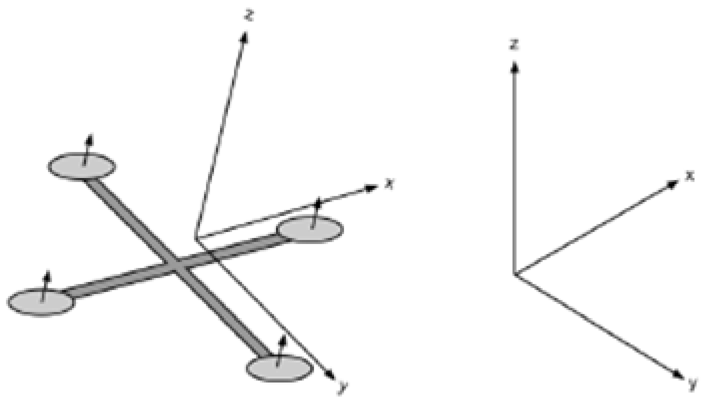

Figure 3.1: Quadcopter Inertial and Body Frame

relation:

$$\mathbf{\Omega} = \begin{bmatrix} 1 & 0 & -\sin(\theta) \\ 0 & \cos(\phi) & \cos(\theta)*\sin(\phi) \\ 0 & -\sin(\phi) & \cos(\theta)*\cos(\phi) \end{bmatrix} \dot{\mathbf{\Theta}} \tag{3.1}$$

where $\mathbf{\Omega}$ is angular velocity vector in the body frame.

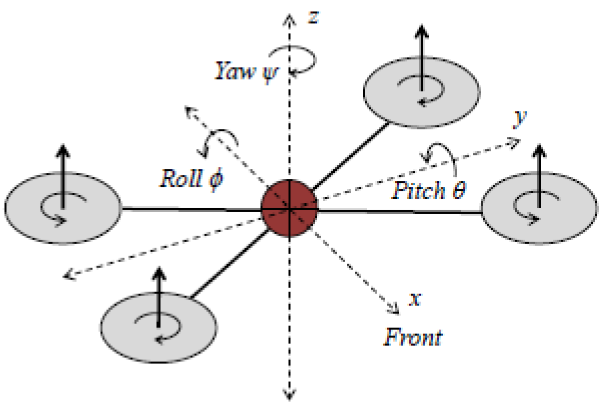

Figure 3.2: Quadcopter Rotor's directions in body frame (Ref: [2])

We are using ZYZ Euler Angle conventions, to relate the body and inertial frame. We defined the **R** matrix to define a conversion from body frame to inertial frame. Using this **R** matrix, we get any vector **A** in inertial frame by directly multiplying the vector in body frame with **R** matrix as $\mathbf{R} \times \mathbf{A}$. The **R** matrix defined as

$$\mathbf{R} = \begin{bmatrix} C_\phi C_\psi - C_\theta S_\phi S_\psi & -C_\psi S_\phi - C_\theta C_\phi S_\psi & S_\theta S_\psi \\ -C_\phi S_\psi + C_\theta C_\psi S_\phi & -S_\psi S_\phi - C_\theta C_\phi C_\psi & -S_\theta C_\psi \\ S_\psi S_\theta & C_\phi S_\theta & C_\theta \end{bmatrix} \dot{\mathbf{\Theta}} \tag{3.2}$$

where C_θ represent $\cos(\theta)$ and S_θ represent $\sin(\theta)$

The angular velocity of the rotor, denoted by ω, creates force f in the direction of the rotor axis. The general relation between force and angular velocity for each rotor is

$$f = k * \omega^2 \tag{3.3}$$

where k is lift constant, depend on the blade configuration. As all the motor configuration is exactly same, so summing over all the rotor's force, we obtain thrust T in the direction of the z direction in body frame, shown in fig 3.3.

$$T_B = k * \sum_{i=1}^{4} \omega_i^2 = k * [0 \quad 0 \quad \sum \omega_i^2]^T \tag{3.4}$$

.

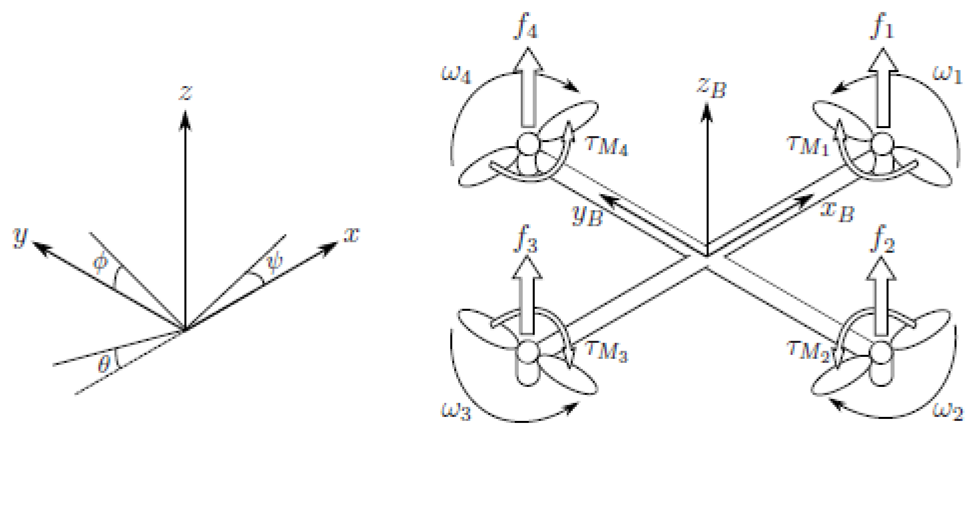

Figure 3.3: Direction of Quadcopter Thrust and Moments (Ref: [3])

Here f represents thrust force and τ_M represents moment.

We also used simplified friction model, where force is proportional to the linear velocity in each direction. There are four propeller in our model, each propeller contributes some torque about the z axis as shown in figure. This toque is required to keep the propeller spinning and providing thrust, it create the instantaneous angular acceleration and overcomes the frictional drag forces.

$$\tau_D = b * \omega^2 \tag{3.5}$$

where D represent the drag.

The torque for each rotor about z axis in body frame is given by

$$\tau_z = b * \omega^2 + I_M * \dot{\omega} \tag{3.6}$$

where I_M is the moment of inertia about the motor z axis, $\dot{\omega}$is the angular acceleration of the propeller, and b is our drag coefficient. Further, we used more simplified torque along the z axis i.e. $\tau_z = b * \omega^2$ because most of the time, propeller will not be accelerating.

The overall toque of the system in body frame is look like this

$$\begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} L * K * (\omega_1^2 - \omega_3^2) \\ L * K * (\omega_2^2 - \omega_4^2) \\ b * (\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{bmatrix} \tag{3.7}$$

where L is the distance from the center of the quadcopter to any of the propeller, ω_i is the angular velocity for each rotor i and K is constant. We have ignored some of the parameter that introduces highly non-linear dynamics such as blade flapping, wind velocity etc.

3.1.3 Equation of the Motion of quadcopter

In the inertial frame, the acceleration of the quadcopter is due to thrust, gravity, and linear friction. We can obtain the thrust vector in the inertial frame by using our rotation matrix R to map the thrust vector from the body frame to the inertial frame. Thus, the linear motion can be summarized as

$$m * \ddot{\vec{X}} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + R * \vec{T_B} + F_D \tag{3.8}$$

where $\vec{X}$ is the position of the quadcopter, g is the acceleration due to gravity, F_D is the drag force, and $\vec{T_B}$ is the thrust vector in the body frame.

Now we derive the rotational equation of quadcopter, we use Euler's equation from the rigid body dynamics. We are defining our rotational dynamics equations in the body frame, to deal with load, suspended from the center of the quadcopter. The Euler's equation is written as

$$I * \vec{\omega} + \vec{\omega} \times (I\vec{\omega}) = \vec{\tau} \tag{3.9}$$

where ω is the angular velocity vector, I is the inertia matrix and $\tau 0$ is a vector of the external torques. From this relation we got

$$\dot{\mathbf{\Omega}} = \begin{bmatrix} \dot{\omega_x} \\ \dot{\omega_y} \\ \dot{\omega_z} \end{bmatrix} = I^{-1} * (\tau - \omega \times (I * \omega)) \tag{3.10}$$

where I is an inertia matrix which is free from complex nonlinear behaviour of the system, defined as

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{3.11}$$

On simplifying the above relation for angular acceleration, we get

$$\begin{bmatrix} \dot{\omega_x} \\ \dot{\omega_y} \\ \dot{\omega_z} \end{bmatrix} = \begin{bmatrix} \tau_\phi * I_{xx}^{-1} \\ \tau_\theta * I_{yy}^{-1} \\ \tau_\psi * I_{zz}^{-1} \end{bmatrix} \tag{3.12}$$

This relation will form the basis of the control system update laws. However, in order to apply them, the orientation of the quadcopter must be known.

3.2 Quadcopter Control

Designing a good controller is most important to render a robust and accurate trajectory in the presence of simple disturbances and model uncertainties. In this section, we discuss our controller strategy. We use two different controllers in our project. First one is a PID controller for the quadcopter control and second one, is a delayed position feedback controller, which we discussed in last chapter. In this section we focus on our PID control unit.

3.2.1 General form of PID control

The general form of the PID controller is

$$e(t) = x_d(t) - x(t) \tag{3.13}$$

$$u(t) = K_P * e(t) + K_I \int_0^t e(t)dt + K_D * \frac{de(t)}{dt} \tag{3.14}$$

where u(t) is the control input and e(t) is the difference between the desired and observed state, and K_P, K_D and K_I are the parameters for the proportional, integral and derivative elements of the PID controller.

Now we explain the behaviour of the proportional, derivative and integral control in system.

- The proportional control adds springy nature in system. With high value of the proportional control, system will have overshooting behaviour.
- The derivative control is very sensitive to noise. Its response is underdamped in nature, which is desired in most cases
- The integral control deals with small disturbance and model uncertainty. Its reponse is overdamped in nature, thus it take long time to converge.
- With a good parameters setting in the combined form of proportional, derivative and inegral control, it will give the desired response.

Now we discuss our quadcopter control. There are six states in our system, three for position and three for the angles, but we will have only four control inputs moments and thrust.The quadcopter is a nested feedback loop system. The inner loop is for attitude stabilization, while the outer loop is for trajectory tracking. The complete quadcopter control is shown in figure 3.4.

3.2.2 Attitude Control

In attitude control, we find the error between desired orientation of the quadcopter $(\phi_d, \theta_d, \psi_d)$ that is calculated by the position controller and the current orientation of the quadcopter

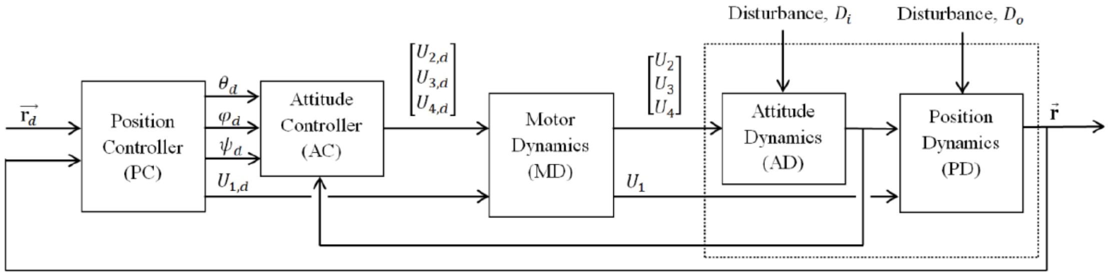

Figure 3.4: Block diagram showing entire system and controllers (Ref: [4])

(ϕ, θ, ψ) calculated by the dynamics of the model. Our PID controller give moments as the controller output, by reducing the error. The relation between the angular accelerations and thrust is defined as

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} l/2 * (1 & 1 & -1 & -1) \\ h/2 * (1 & -1 & -1 & 1) \\ a * (-1 & -1 & 1 & 1) \end{bmatrix} * \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix}$$

Now we will use the dymanics of the system to calculate the next state of the quadcopter with this control information.

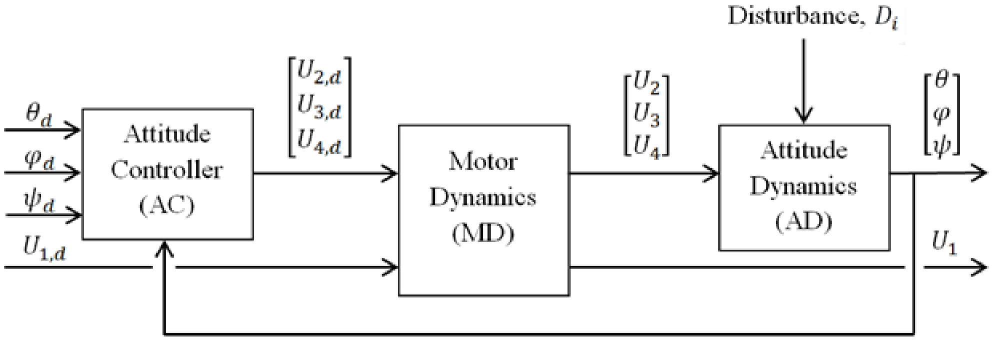

Figure 3.5: Attitude Control of quadcopter (Ref: [4])

3.2.3 Position Control

In position control, it find the error between the current state of the vehicle which is defined as $\mathbf{r}$ in block diagram fig 3.4 that is calculated by the dynamics of the vehicle and the desired trajectory $\mathbf{r_d}$. Now again we use a PID controller to supress the error, it give desired orientation of the vehicle and thrust as controlled output. This controlled information use by attitude controller to find the desired moments and this process repeat in a nested form to follow the desired trajectory.

Chapter 4

Control of Quadcopter with Load

In this chapter, we will see how to control position and attitude of a quadcopter so as to control the oscillations of suspended payload. Since the payload is attached from the quadcopter, we can take the point Q to be on quadcopter itself as shown in the figure below.

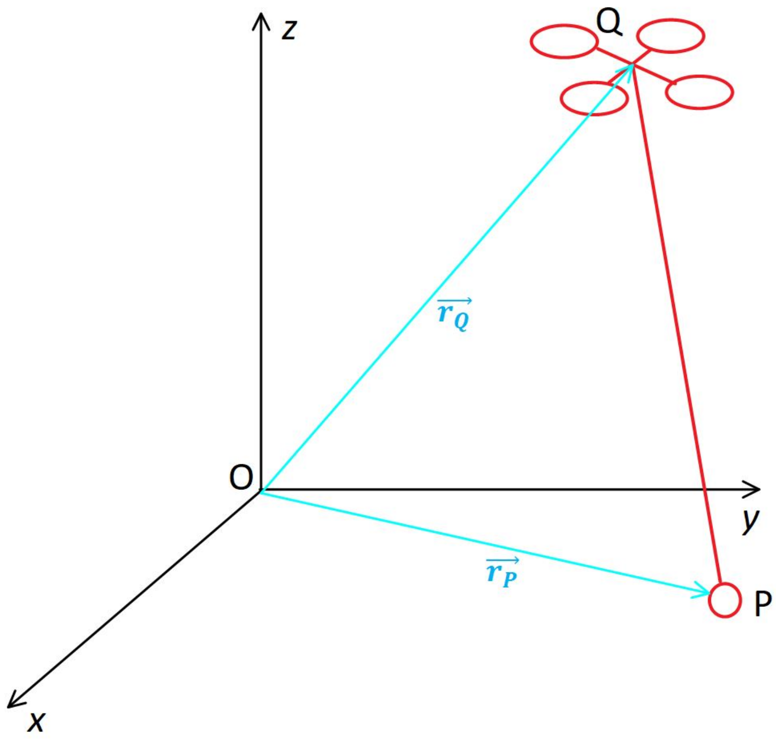

Figure 4.1: Quadcopter Suspended Load

4.1 Desired Motion of Quadrotor

The quadrotor has to be controlled such that not only the oscillations of the payload are reduced but also the quadrotor is able to carry the load to its desired location.

We have already seen that in order to stabilize a load using time delayed position feedback, the point of suspension needs to follow a trajectory given by:

$$x_{Q,des}(t) = x_{o,des}(t) + k_x L \sin(\theta_x\,(t-\tau_x))\cos(\theta_y\,(t-\tau_x)) \tag{4.1}$$

$$y_{Q,des}(t) = y_{o,des}(t) - k_y L \sin(\theta_y(t-\tau_y)) \tag{4.2}$$

These above equations define the path which we want the quadcopter to follow such that it reaches its desired location, reducing the oscillations during its flight.

4.1.1 Trajectory Tracking for Load Suspended Quadrotor

The symbols used in the equations 4.1 and 4.2 are explained below for the sake of convenience:

1. $(x_{Q,des}(t), y_{Q,des}(t))$: The desired (x, y) coordinates of the quadrotor Q which define its motion with respect to time
2. $(x_{o,des}(t), y_{o,des}(t))$: The desired motion of the quadrotor without load
3. k_x and τ_x are gain and delay parameters for oscillations in θ_x (See figure 2.8)
4. k_y and τ_y are gain and delay parameters for oscillations in θ_y (See figure 2.8)
5. L is the length of the cable from quadrotor to load. We have assumed the cable to be rigid and massless.

For tracking the trajectory given by $(x_{Q,des}(t), y_{Q,des}(t))$, we need to understand the dynamics of the system, which is discussed in the section below.

4.1.2 Coupled Dynamics

In this section, we analyze the dynamics of a quadrotor carrying a suspended load. Cooperative load transportation strategy has been designed in [6] using the method of PID control.

We will derive our results for single quad single load system in the ground frame. Suppose the total thrust force on the quadrotor-load system is $\mathbf{F}$. Let $\mathbf{g} = [0\ \ 0\ \ -9.81]^T$ be the acceleration due to gravity. If the acceleration of Q and P are $\mathbf{a_Q}$ and $\mathbf{a_P}$ respectively, we can write

$$\mathbf{F} + m_Q\mathbf{g} + m_P\mathbf{g} = m_Q\mathbf{a_Q} + m_P\mathbf{a_P} \tag{4.3}$$

$$\mathbf{F} + m_Q\mathbf{g} + m_P\mathbf{g} = m_Q\mathbf{a_Q} + m_P(\mathbf{a_Q} + \mathbf{a_{P/Q}}) \tag{4.4}$$

$$\mathbf{F} + (m_Q + m_P)\mathbf{g} = (m_Q + m_P)\mathbf{a_Q} + m_P\mathbf{a_{P/Q}} \tag{4.5}$$

This equation looks similar to the force equation for a quadcopter with mass $m_Q + m_P$, except for the additional term of $\mathbf{a_{P/Q}}$ which is the acceleration of the pendulum with respect to the quadrotor.

In [7], it has been shown that the quadrotor attitude dynamics is decoupled from the load position and attitude dynamics, while the load attitude dynamics is decoupled from the load position dynamics.

4.2 Control Strategy

The outline of our control algorithm can be summarized as follows:

1. The first step is to generate a desired trajectory for the quadrotor. This trajectory is given by $(x_{Q,des}(t), y_{Q,des}(t))$.

2. Using the desired position of the quadrotor, we can calculate the the desired euler angle configuration $(\phi_{des}, \theta_{des}, \psi_{des})$. It has been shown in [6] that the values of $(\phi_{des}, \theta_{des})$ can be calculated using the equations below.

$$T_d = (m_Q + m_P)\sqrt{(\ddot{x}_{Q,des})^2 + (\ddot{y}_{Q,des})^2 + (g - \ddot{z}_{Q,des})^2} \tag{4.6}$$

$$\phi_{des} = \sin^{-1}(u_x \sin\psi_d - u_y \cos\psi_d) \tag{4.7}$$

$$\theta_{des} = \sin^{-1}\left(\frac{u_x \cos\psi_d + u_y \sin\psi_d}{\cos\phi_{des}}\right) \tag{4.8}$$

In these set of equations, T_d represents the desired thrust force. u_x and u_y are calculated as $u_x = (m_Q + m_P)\frac{\ddot{x_{Q,des}}}{T_d}$ and $u_y = (m_Q + m_P)\frac{\ddot{y_{Q,des}}}{T_d}$. We can take $\phi_{des} = 0$.

3. To follow these desired angle configuration, we use PD control for attitude tracking. This will form the inner loop in the control block diagram.

4. In the outer loop, the position is controlled using PID control.

The details of control of quadrotor can be found in the previous chapter.

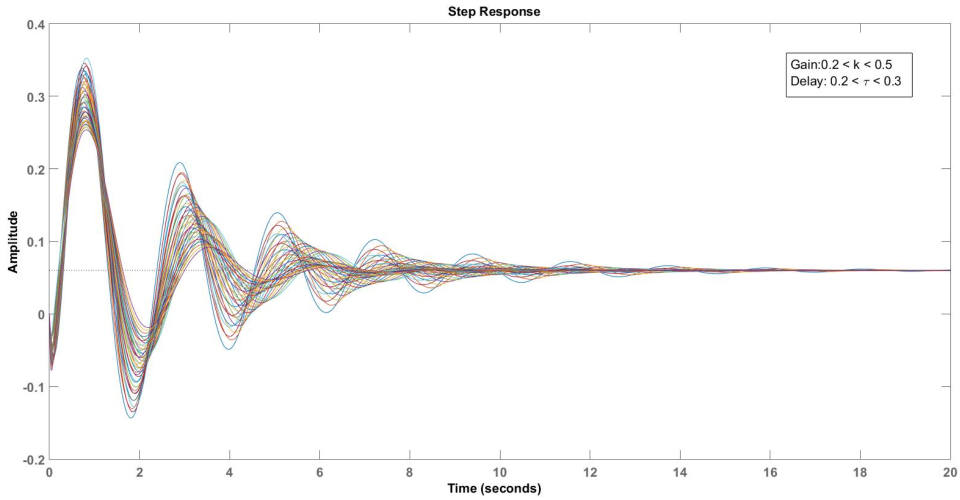

Figure 4.2: Response of the system for different k and τ using First Order Pade Approximation

Chapter 5

Simulations

In this chapter, we will simulate quadrotor carrying a suspended load. Consider a UAV of mass $M = 1$kg carrying a payload $m = 100$g attached with a cable of length $L = 0.5$m.
We simulated the system for various values of gain and delay. The best performance was achieved when the time delay was set to $\delta = 0.3$ and the gain values k_x and k_y to 1. Hence, $\tau_x = 0.3\,T_o$ and $\tau_y = 0.3\,T_o$.

Earlier we had seen in figure 2.2 that the optimum control was achieved for gain of 0.4 and delay of $0.3\,T_o$. However, through simulations, we achieve a different result when the load is attached to a quadrotor. The optimum control is achieved for gain of value 1. A possible reason behind this is the internal delay of the system. Whenever a control command is sent to the quadrotor, it is not executed instantaneously. It takes some time to actuate the control commands. This introduces an unintentional intrinsic delay in our system. This is caused due to the non-zero settling time of the position of the quadrotor.
Another possible explanation is that the quadrotor is never able to exactly track the desired trajectory for load control. Since PID control is used for tracking position and attitude, there is always some error is the desired and observed position of the quadrotor.

A detailed analysis of change in values of gain and delay for optimum control is beyond the scope of this project, but it is an interesting research problem which can be explored later.

5.1 Linear Path

Suppose we have to move the load from $(x = 0,\ y = 0)$ with an average speed of $\mathbf{v} = 4\mathbf{i} + 2\mathbf{j}$ m/s.

5.1.1 Trajectory

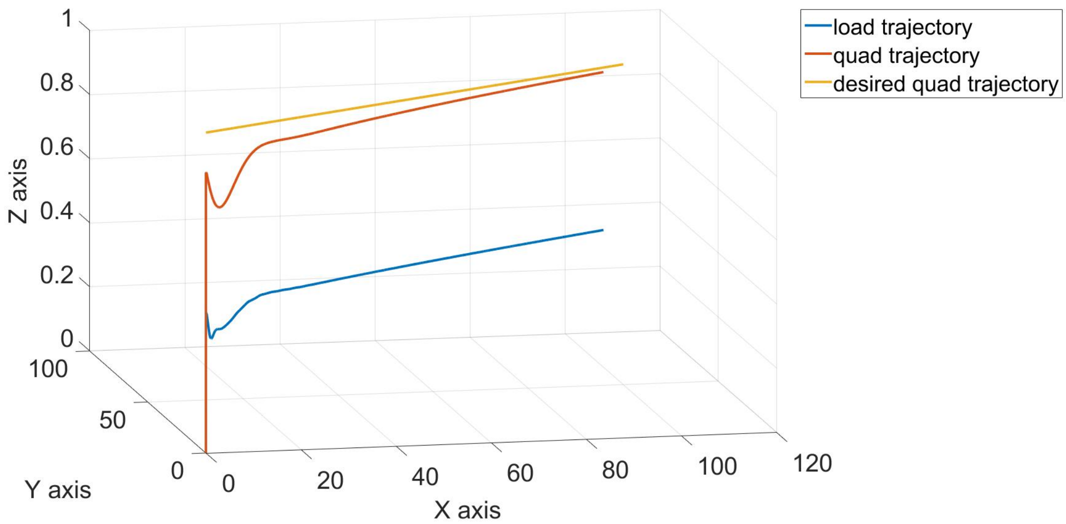

Figure 5.1: Quadcopter and Load Trajectory

5.1.2 Oscillations

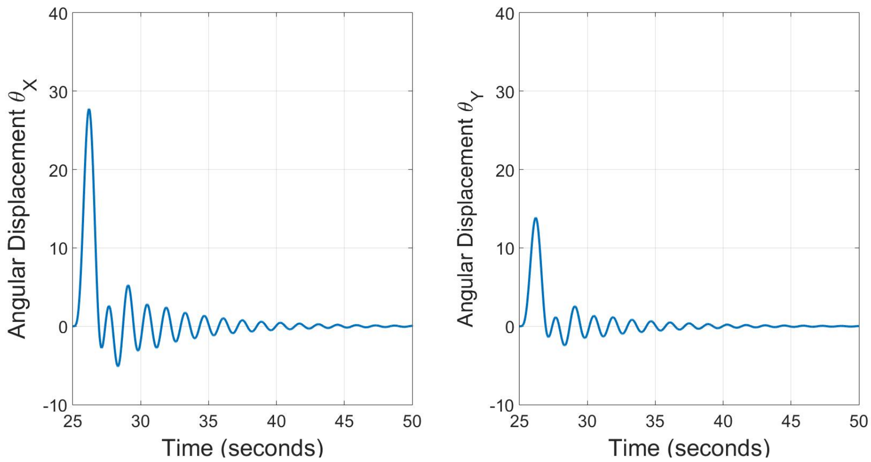

Figure 5.2: Angular Displacement of the Load

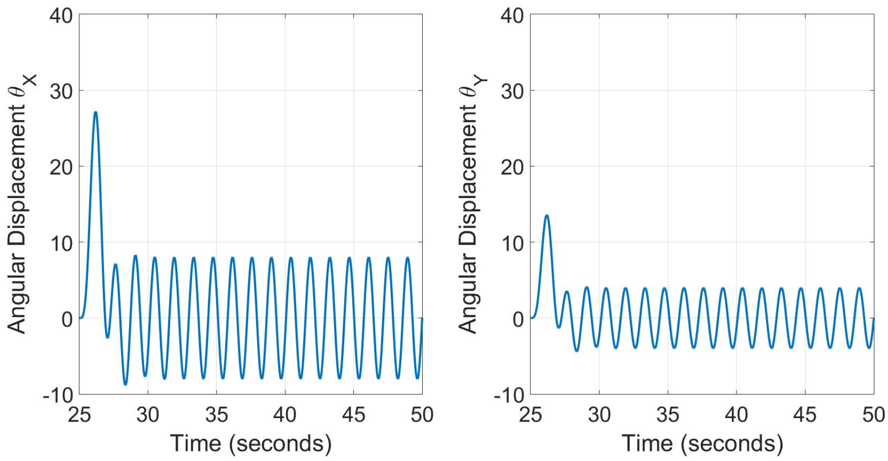

Figure 5.3: Angular Displacement of the Load without Delayed Feedback

5.1.3 Quadcopter Position and Attitude

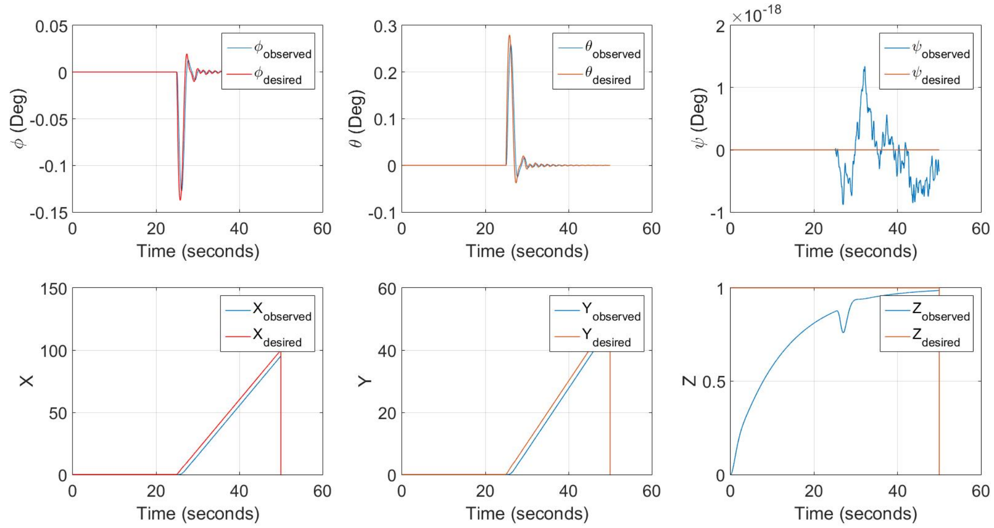

Figure 5.4: Quadcopter Position and Attitude

5.2 Circular Trajectory

In this section, we will show simulation results for the quadcopter following a circular trajectory. The speed of the quadrotor is 2.5m/s and it is tracking a circle of radius 1 with centre $(-1, -1)$.

5.2.1 Trajectory

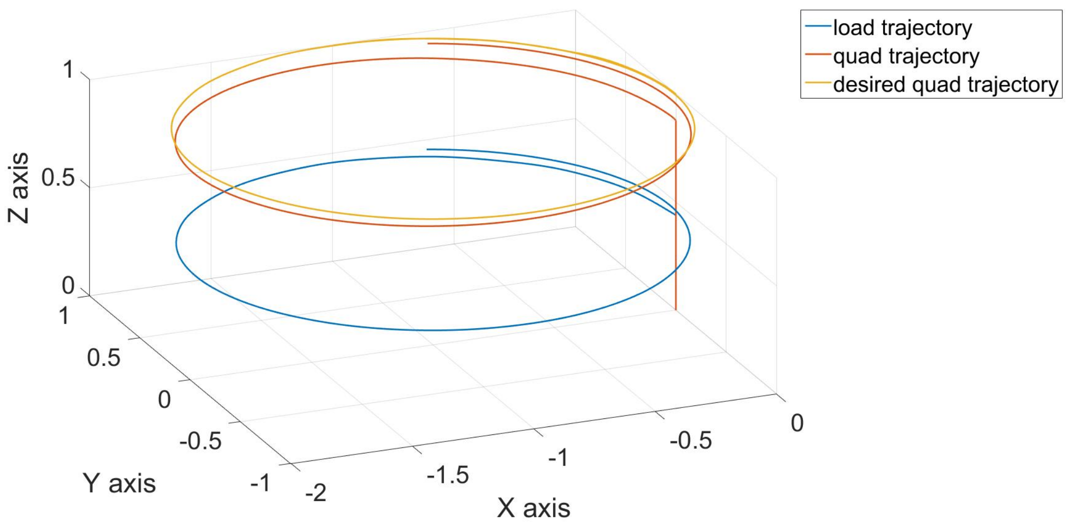

Figure 5.5: Quadcopter and Load Trajectory

5.2.2 Oscillations

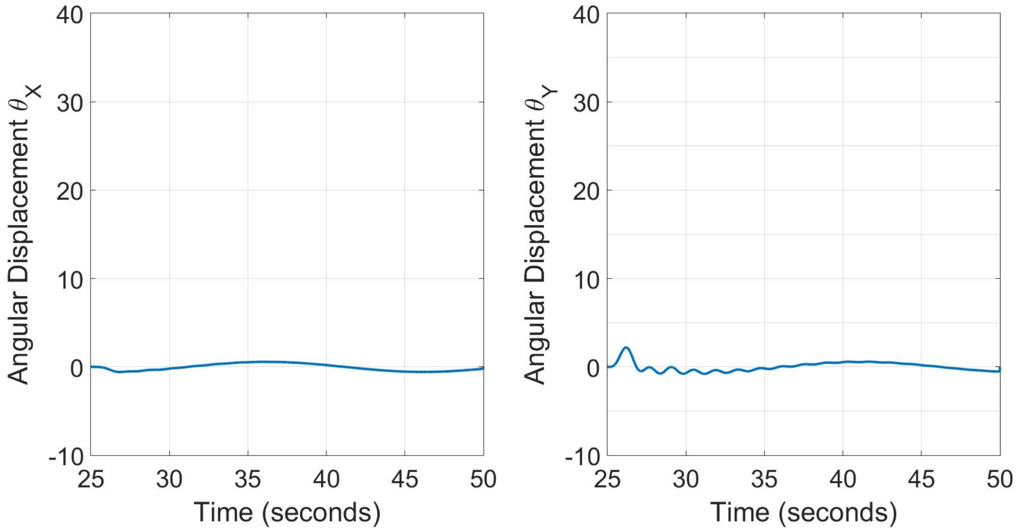

Figure 5.6: Angular Displacement of Load

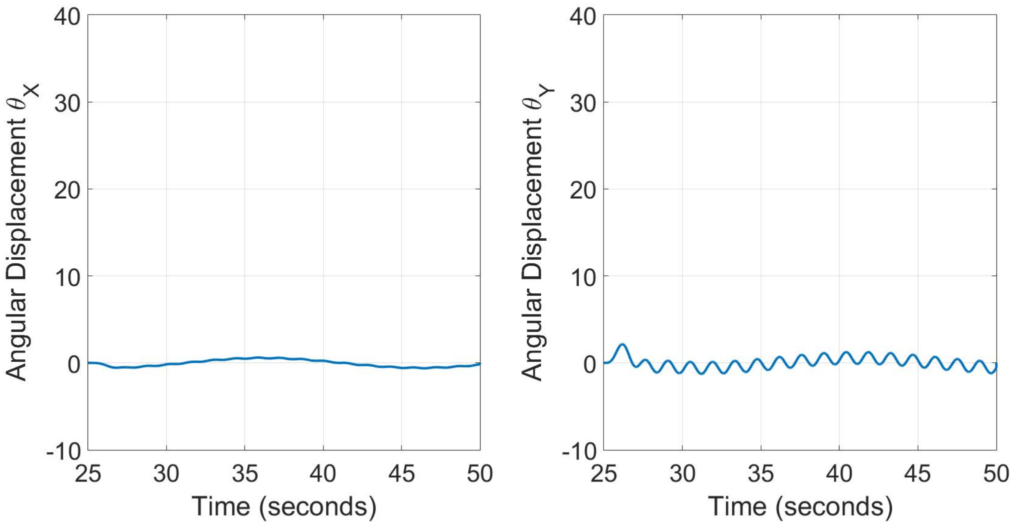

Figure 5.7: Angular Displacement of Load without Delayed Feedback

5.2.3 Quadrotor Position and Attitude

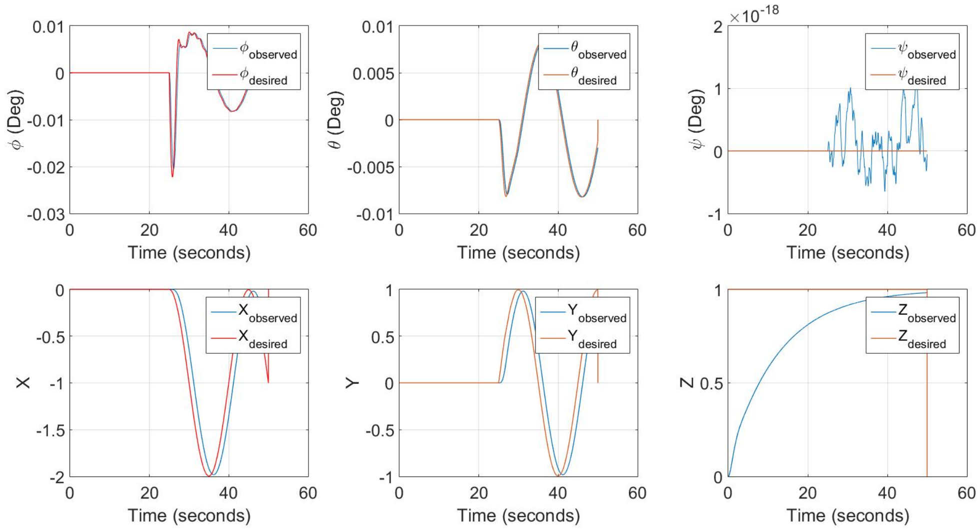

Figure 5.8: Quadcopter Position and Attitude

5.3 Performance Under Jerky Reference Path

In this section, we will show simulation results for the quadcopter following a trajectory with jerks. The speed of the quadrotor in X direction is 1 m/s and occasional jerks are given in Y direction with a speed of 50 m/s. We see that even with velocity of 50 m/s at jerks, the oscillations of load are significantly below 20°.

5.3.1 Trajectory

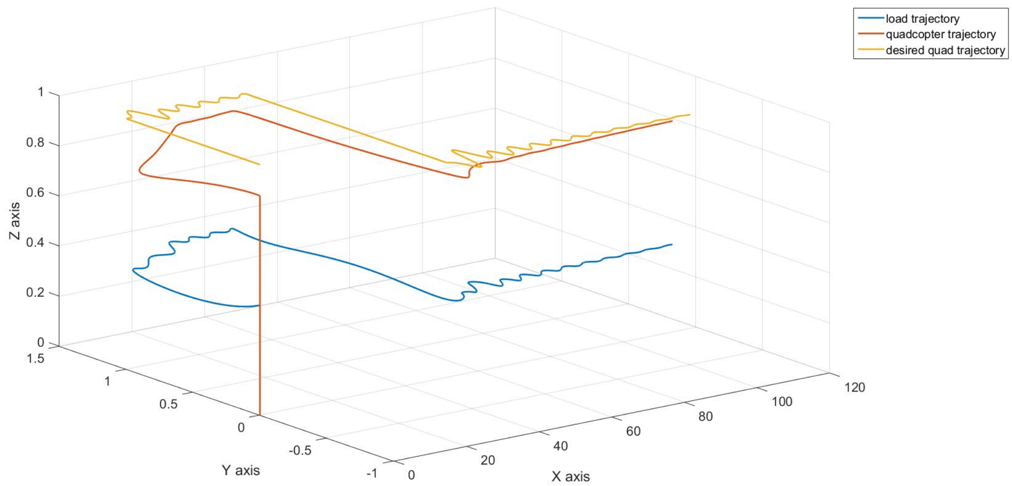

Figure 5.9: Quadcopter and Load Trajectory

5.3.2 Oscillations

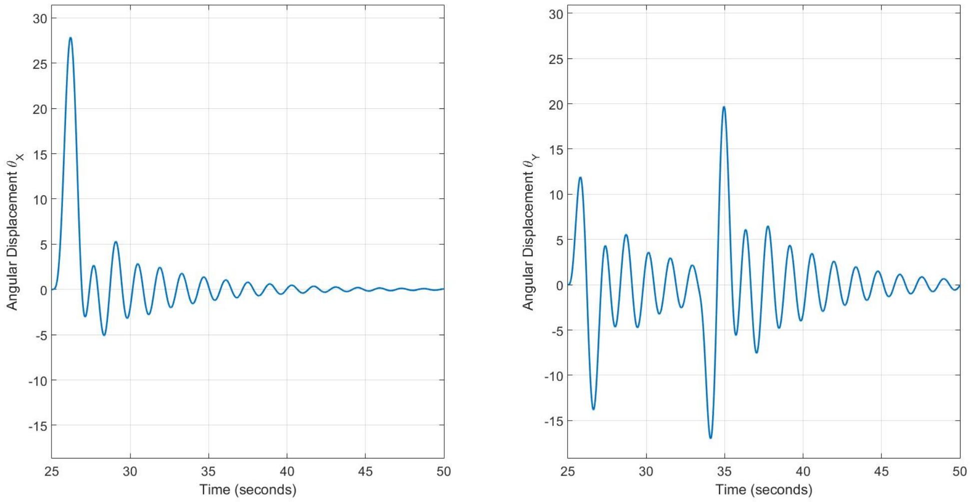

Figure 5.10: Angular Displacement of Load

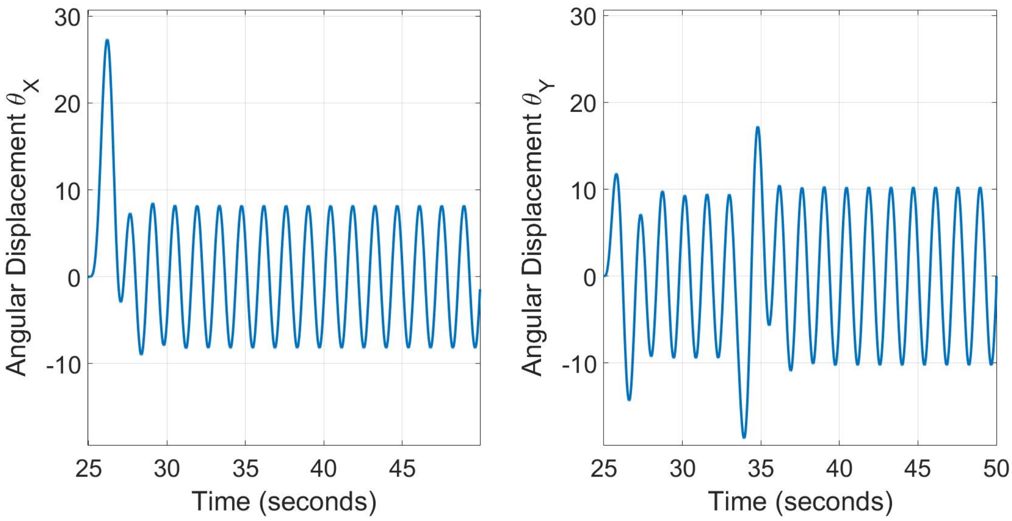

Figure 5.11: Angular Displacement of Load without Delay Controller

5.3.3 Quadrotor Position and Attitude

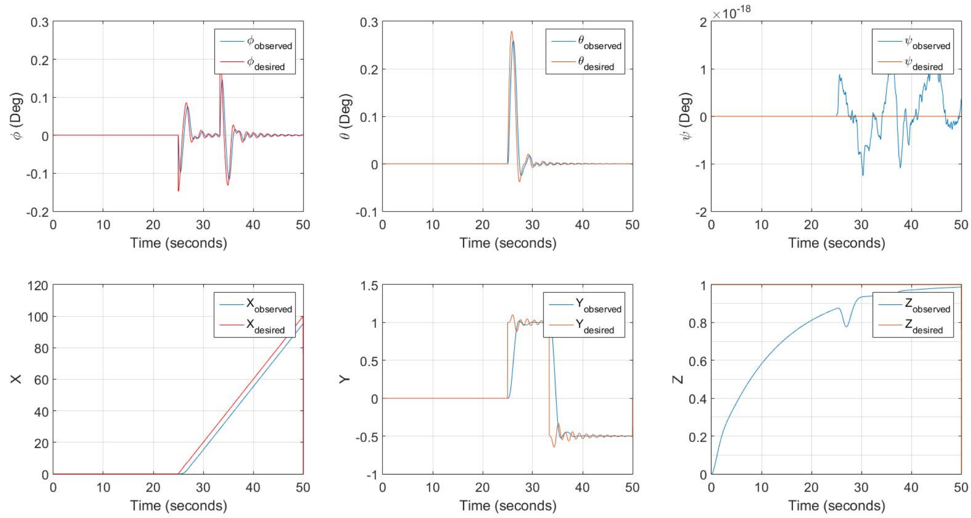

Figure 5.12: Quadcopter Position and Attitude

5.4 Conclusion

In this project, a feedback control scheme is designed in order to introduce damping in undamped oscillations of slung loads. The performance of the controller is evaluated by simulating the quad-load system in MATLAB. We find that the time delayed position feedback can damp out the vibrations quickly and does not affect the final position which the quadrotor has to reach.

5.5 Future Work

We shall analyze change in values of gain and delay for optimum control in case of quadrotor suspended load as the values are different from those seen for a spherical pendulum.
We can hypothetically state that the reason behind this is non-zero settling time for position and attitude of quadrotor and its inability to track the given path with zero error.

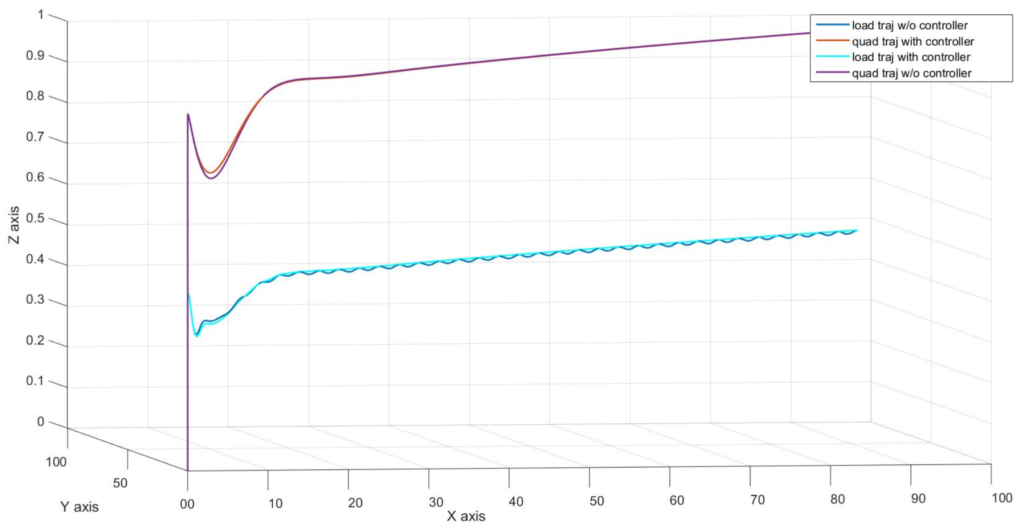

Figure 5.13: Trajectories with and without controller for Linear Path

References

[1] Z. Masoud, A. Nayfeh, and D. Mook, "Cargo pendulation reduction of ship-mounted cranes," *Nonlinear Dynamics*, vol. 35, no. 3, pp. 299–311, 2004.

[2] N. M. Zimmerman, "Flight control and hardware design of multi-rotor systems," 2016.

[3] T. Luukkonen, "Modelling and control of quadcopter," *Independent research project in applied mathematics, Espoo*, vol. 22, 2011.

[4] M. Mardan, "Attitude and position control of quadrotors: design, implementation and experimental evaluation," 2016.

[5] R. Henry, Z. Masoud, A. Nayfeh, and D. Mook, "Cargo pendulation reduction on ship-mounted cranes via boom-luff angle actuation," *Journal of Vibration and Control*, vol. 7, no. 8, pp. 1253–1264, 2001.

[6] K. K. Dhiman, A. Abhishek, and M. Kothari, "Cooperative load control and transportation," in *2018 AIAA Information Systems-AIAA Infotech@ Aerospace*, p. 0895, 2018.

[7] K. Sreenath, T. Lee, and V. Kumar, "Geometric control and differential flatness of a quadrotor uav with a cable-suspended load," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, pp. 2269–2274, IEEE, 2013.