Impact of Magnetic Fields due to High Tension Transmission Lines on Multirotor Vehicles

A Graduate Project Report submitted to Manipal Academy of Higher Education in partial fulfilment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY
In
Electronics and Communication Engineering

Submitted by

Utkarsh Tripathi
Reg. No: 150907396

Under the guidance of

Simit Pradhan
Technical Expert,
Corporate Technology,
Siemens Technology and Services Private Ltd.,
Bangalore

&

A. Gopalakrishna Pai
Assistant Professor,
Department of Electronics & Communication Engineering,
Manipal Institute of Technology,
Manipal

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
MANIPAL INSTITUTE OF TECHNOLOGY
(A Constituent Institution of Manipal Academy of Higher Education)
MANIPAL – 576104, KARNATAKA, INDIA
MAY 2019

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

MANIPAL INSTITUTE OF TECHNOLOGY

(A Constituent Institution of Manipal Academy of Higher Education)

MANIPAL – 576 104 (KARNATAKA), INDIA

Manipal
6th May 2019

CERTIFICATE

This is to certify that the project titled **Impact of Magnetic Fields due to High Tension Transmission Lines on Multirotor Vehicles** is a record of the bonafide work done by **Utkarsh Tripathi** (*Reg. No. 150907396*) submitted in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology (BTech) in **ELECTRONICS AND COMMUNICATION ENGINEERING** of Manipal Institute of Technology, Manipal, Karnataka, (A Constituent Institution of Manipal Academy of Higher Education), during the academic year 2018 - 2019.

A. Gopalakrishna Pai
Assistant Professor,
Dept. of Electronics &
Communication Engineering,
Manipal Institute of
Technology, Manipal

Simit Pradhan
Technical Expert,
Corporate Technology,
Siemens Technology and
Services Private Ltd.,
Bangalore

Prof. Dr. M. Sathish Kumar
Head of Department,
Dept. of Electronics &
Communication Engineering,
Manipal Institute of
Technology, Manipal

Department of
Electronics & Communication Engg.
M.I.T., MANIPAL - 576 104

SIEMENS

Corporate Research and Technologies India

Bangalore
30.04.2019

CERTIFICATE

This is to certify that the project entitled **Impact of Magnetic Fields due to High Tension Transmission Lines on Multirotor Vehicles** was carried out by **Utkarsh Tripathi** (*Reg. No. 150907396*) at **Siemens Technology & Services Pvt. Ltd.** under my guidance during **January 2019** to **May 2019.**

Simit Pradhan
Technical Expert,
Corporate Technology,
Siemens Technology & Services Pvt. Ltd.,
Bangalore

Siemens Technology and Services Pvt. Ltd.
Corporate Research & Technologies India
105. Prestige Alecto, Keonics Electronic City,
Hosur Road, Bangalore - 560 100.

Siemens Technology and Services Pvt. Ltd.
Management: Klaus Trescher
Corporate Research and Technologies India Management: Ravi Madipadaga

No. 105, Prestige Alecto,
Keonics Electronics City
Hosur Road
Bangalore 560100
Karnataka, India

Tel .: +91 80 33135618
Fax: +91 80 33134503
Web: www.siemens.co.in/sts
E-mail:
girish.suryanarayana@siemens.com

Registered Office: 130, Pandurang Budhkar Marg, Worli, Mumbai 400 018. Telephone +91 22 39677000. Fax +91 22 39677075.
Corporate Identity Number: U99999MH1986PLC093854. Former name: Siemens Information Systems Ltd.
Other Offices: Bangalore, Chennai, Gurgaon, Kolkata, Noida, Pune.

ACKNOWLEDGMENTS

I wish to express my gratitude towards the people who helped me and contributed to this project. Thank you, Simit Pradhan, for his valuable technical support on this project. Also, thanks to A. Gopalakrishna Pai for his support and mentorship during my internship in Siemens. Also, thanks to Prof. Dr. M Sathish Kumar for his guidance and support to both the students and the teachers of the department.

I would also like to thank the Chief Librarian, for letting me use the library resources through proxy when I was living out of station.

Thanks to my parents for the care and guidance they have given me throughout my life.

ABSTRACT

The project is based on developing a stable navigation and control system for a multirotor based drone. The deliverable of the project is a drone system that is stable in scenarios with high magnetic field noise on the drone's magnetic sensor system. This system would enable operation of drones in many useful scenarios. One specific scenario is to inspect transmission lines using drones with the help of visual as well as actual measurements of the current in transmission lines using flux sensors. Therefore, the objective of the work is to simulate the effects of magnetic field with noise from transmission lines, and propose a robust system that can help to mitigate the effect of noise.

The projects starts with a simulation of the drone in a 3D environment. This is followed demonstration of the capabilities of simulation. Then, a noise source is introduced and its effect on the sensor systems are studied. Further, the drone's dynamics and control system are simulated on Simulink. Finally, a mitigation algorithm is put in place to help to drone remove the noise from raw sensor data.

The setup uses a variety of open source tools and framework. These tools are Robot Operation System (ROS), Gazebo, mavros, DroneCode, ArduPilot and PX4 flight stack. These tools are integrated in the Ubuntu flavor of GNU Linux on an Intel machine with amd64 architecture. The dynamic simulation was done on a Microsoft Windows machine on Simulink. Simulink provides multiple toolboxes where the programmer use pre-made blocks. In this project, the aerospace toolbox, control systems toolbox, and digital signal processing toolbox were used.

LIST OF FIGURES

Contents

CHAPTER 1
INTRODUCTION

1.1 Introduction

Drones are ubiquitous in today’s age, serving multiple purposes from aerial photography to industrial automation and inspection. One of the use cases of drones may be inspection and current sensing in high tension transmission lines. Specifically, a drone may be used obtain current measurements from the high-tension lines via a flux sensor. However, such a scenario insinuates a peculiar problem. Large, periodic voltage variations in the transmission line produce varying magnetic field which may affect the drone’s sensor, especially, the magnetometer. This can lead to erroneous state estimation which may drive the drone’s system to instability. This project aims to simulate such a scenario, observe its effect of the sensor readings, and propose solutions to mitigate the problem.

This chapter introduces the reader to the framework used by most popular drones today. It gives some insights about the drone sensor system. Finally, an introduction to the problem is given.

1.2 Introduction to the area of work

Drones are ubiquitous in today’s age, serving multiple purposes from aerial photography to industrial automation and inspection. One of the use cases of drones may be inspection and current sensing in high tension transmission lines. Specifically, a drone may be used obtain current measurements from the high-tension lines via a flux sensor. However, such a scenario insinuates a peculiar problem. Large, periodic voltage variations in the transmission line produce varying magnetic field which may affect the drone’s sensor, especially, the magnetometer. This can lead to erroneous state estimation which may drive the drone’s system to instability. This project aims to simulate such a scenario, observe its effect of the sensor readings, and propose solutions to mitigate the problem.

The dynamics of a multirotor drone is a complicated one. A quadcopter drone is a 6 DOF vehicle, however, has only four actuators. Thus, a quadcopter is an underactuated system. That is, it uses only 4 actuators to maneuver in the 6 degrees of freedom, namely, the translational X, Y and Z axes and the roll, pitch, and yaw Euler angles. Adding to the complicated dynamics, an aerial drone is also an unstable system, closely representing the dynamics of an inverted pendulum, which is non-linear in nature.

Thus, a quadcopter needs to employ efficient control loops to stabilize the system. These control loops are generally robots feedback loops relying heavily on IMU sensors. In a perfect world, only a gyroscope is enough to stabilize the drone. A gyroscope outputs the measured angular velocities about all the three axes. One can numerically integrate the angular velocities to get angular positions, namely, yaw, pitch, and roll. These integrated values can be used as feedback to stabilize the drone. A numerical integration is given by the equation:

$$\Theta^{(\mathrm{n+1})} = \Theta^{(n)} + \dot{\Theta}\mathrm{T_s}$$

Where, Θ represents the vector of Euler angles, $\dot{\Theta}$ represents the derivative of Euler angles output by the gyroscope, Ts is the sampling time period and the superscript *(n)* represents the *n*th time instant.

In the real world, however, integration of any signal poses the problem of error build-up. Tiny fluctuations from the sensor readings can add up to become larger error. In the case of gyroscope, it is known as gyroscopic drift. This phenomenon can slowly have an offset on the measured state of the vehicle, and the vehicle can become difficult to control. This phenomenon is illustrated in Figure 1, where a comparison is drawn between the actual state and estimated state of some angle theta from a gyroscope.

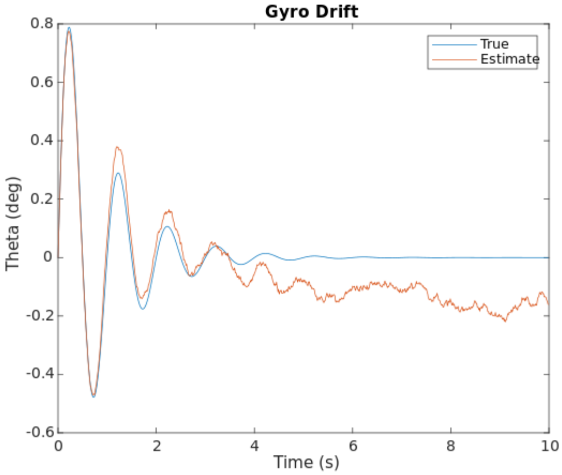


Figure 1: Gyroscopic bias drift.

To prevent gyroscopic drift, often extra sensors is employed. Most often, an accelerometer is used for state estimation. An accelerometer is a 3 DOF device capable of measuring accelerations in the body frame. This includes acceleration exhibited by the gravitational forces. Being a 3 DOF device, an accelerometer, in principle, measure gravitational acceleration in the three axes of the body frame. If the accelerometer does not align perfectly with the gravity vector, the gravitational vector is simply projected onto the 3 unit-vectors forming the basis in the accelerometer body frame. This is illustrated in Figure 2.

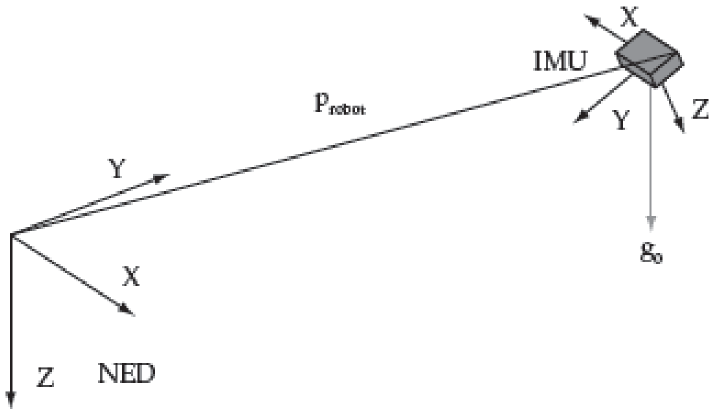


Figure 2: Gravity vector in the NED frame is projected onto the accelerometer body frame.

From the measurements in the three axes, it is possible to deduce the roll and pitch angles by using the following algorithm.

```
rotX = atan2(accelY, accelZ) * 180/pi;

rotY = atan2(-accelX, sqrt(accelY*accelY + accelZ*accelZ)) * 180/pi;
```

Where rotX is the angle in degrees of the rotation about X axis (roll), and rotY is the angle in degrees of the rotation about Y axis (pitch), accelX, accelY, accelZ are measurements from the accelerometer in the respective axes, and atan2 is the arctangent function which includes the special case where both numerator and denominator being zero.

A plot of atan2 function is illustrated in Figure 3.

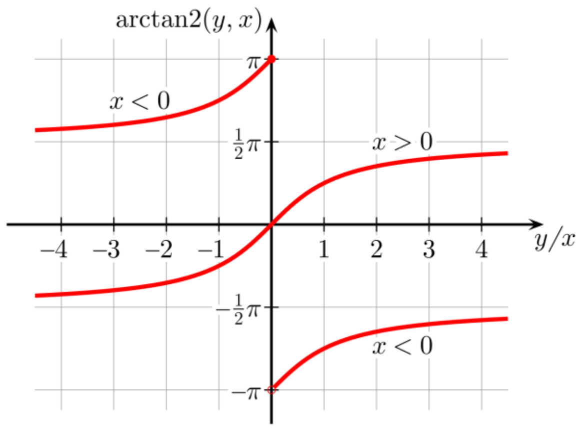


Figure 3: A plot of atan2(y,x) function

Thus, it is possible to deduce roll and pitch angle by just using an accelerometer. However, it in not possible to calculate yaw angle from an accelerometer in all the cases. In a steady state, a quadcopter has both roll angle and pitch angle equal to zero. In such a case, for all yaw angles, the accelerometer readings do not change. To estimate the yaw, along with roll and pitch, it is quite a common practice to employ both accelerometer and gyroscope in conjunction to each other. When used in conjunction with each other, usually a complementary filter is employed to mix the roll and pitch outputs of the sensors. A complementary filter is just a weighted sum of the respective readings from the output vectors of the sensor such that the mapping of sensor input, and state estimation is preserved. For yaw estimation, only gyroscope readings are used post numerical integration. Thus, it is still possible for yaw drift to occur.

To prevent the problem of yaw drift, a third sensor, a magnetometer is used. A magnetometer can measure the vector of the earth magnetic field in the body frame of the sensor. It is also a 3 DOF device, but it only measures yaw angle. Unlike gyroscope, there is no numerical integration required to get the state estimate, thus, it poses no problem of drift.

These three sensors: accelerometer, gyroscope and magnetometer, form the basic sensor system in most drones. Of course, there are highly advanced autonomous drones which employ even more sensors such as GPS, Optical Flow Sensors, Radars, Lidars, etc. However, for this project, just an analysis of a drone with these three sensors is enough.

The problem comes when the drone comes in proximity of high-tension electricity cables. A drone may be used to obtain current measurements from the high tension lines via a flux sensor. However, such a scenario insinuates a peculiar problem. Large, periodic voltage variations in the transmission line produces varying magnetic field which may affect the drone’s sensor, especially, the magnetometer. This can lead to erroneous state estimation which may drive the drone’s system to instability. A detailed study of such a disturbance is given in Chapter 2. In Chapter 3, such effects are observed on a popular drone platform known as PX4. The platform is simulated on Gazebo and the effects are noted.

In Chapter 4, a drone system is modeled in Simulink to more closely observe the effects and to facilitate the development of the mitigation algorithms. Finally in Chapter 5, the mitigation algorithms are tested in Simulink and observations are drawn.

1.3 Motivation for the project

Generally, linemen have to go to proximity of transmission lines in order to make observations, which apart from being a life risk is also a logistically and economically taxing. The project aims to use drones to perform checks on the transmission lines, which can serve the following purpose:

1. Dramatically improve safety of linemen
2. Speedup the process
3. Automate logistical requirements
4. Reduce cost associated with such an operation

Figure 4 shows an image of a lineman using a helicopter specifically design to hover near a high power line.

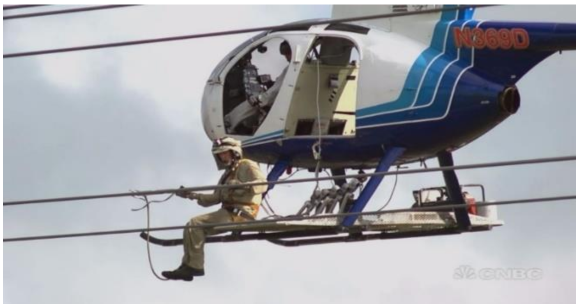


Figure 4: Lineman performing maintenance task on a helicopter. (Courtesy: CNBC)

As evident from Figure 4, it is a life risking task. Therefore, use of drones may dramatically improve the safety of linemen most situations where the task is limited to observation and visual inspections.

1.4 Objective of the work

The following points briefly summarise the objective of the project.

1. Set up a simulation of a drone operating in a 3D world
2. Create a scenario with high tension transmission lines producing magnetic field
3. Observe the effects of magnetic interference due to the transmission lines on the drone and its sensors
4. Propose an algorithm to mitigate the effects of magnetic interference on the drone using concepts from signal processing and control theory.

5. Implement the mitigation algorithm to the simulated scenario and observe the effects. Analyse the stability of the drone after implementing mitigation algorithm

1.5 Target Specification

The deliverable of the project is a plug & play system block that readily plugs in from of the sensor block. The function of the block should be as follows.

1. Reject any disturbances caused due to magnetic fields induced by high voltage transmission lines.

2. If the disturbances cannot be rejected, then manipulate the state estimation algorithm to correctly estimate state either though other sensors or some other methodology

1.5.1 Importance of end-result

The proposed mitigation algorithm block should enable operations of drones near high tension transmission lines. This will open the door to a whole new application for drones. Drones may then be used for analysis and inspection tasks of high-tension transmission lines without the risk of damage and crashing of the drones. This will also increase the safety of linemen who will not have to risk their lives performing inspection of high-tension transmission lines manually.

1.6 Project Work schedule

January 2019	Work with the IT team to convey software requirements and setup the simulation software.
February 2019	Perform literature survey on magnetic fields due to high tension wires. Modify the simulation to include magnetic fields due to high tension transmission lines.
March 2019	Design a mathematical model of the drone system in Simulink and perform simulation to validate it. Get step-response data and tune the system parameters
April 2019	Implement noise source and the mitigation algorithms. Perform validation tests. Draw comparison
May 2019	Documentation and final conclusion

1.6 Organization of Project Report

The report is divided into 5 chapters. The first chapter gives the reader an overview of the drone system. It also introduces the reader to the area of the problem. It provides insights on the current scenario of drone use in the industry as well as the practices done by linemen. The second chapter is a review of the background theory, and literature review. The magnetic field experienced by the drone are derived from the first principles. A study has been done on the shape of the conductor. Also, the drone dynamics have been discussed. The results from chapter 2 are then used in Chapter 3, where the methodology of the study is discussed. Simulink was chosen as the platform of choice for modelling. The dynamic equations were modelled, and the system was simulated using a joystick controller. Subsequently, the step responses of various components of the system were plotted. The sensors were also modelled. The magnetometer was modelled by system identification based on frequency response studies of existing sensors. Gyroscope and accelerometer models were taken from the Aerospace toolbox. The controller block is also discussed in this chapter. Mitigation algorithms are proposed. These algorithms were tested.

The step responses and study of the noise model is done in Chapter 4. Firstly, the dynamic stability of the drone is observed through various step responses. Then, multiple conditions, through which the noise source affects the drone sensor system are created. The effects of these simulations are observed, and the sensors response are plotted. A qualitative study has been done to propose a mitigation algorithm. Finally, a few mitigation algorithm blocks are proposed. These blocks too are tested, and the results are plotted. A brief discussion of the real-life implications of such blocks is also presented,

In the final chapter, the project concludes with a recap of the study done so far. A brief section discusses the methodology used and the challenges faced. Future scope of the work is presented following a conclusion section which contains final remark about the project work.

CHAPTER 2
BACKGROUND THEORY

2.1 Introduction

This chapter will discuss the background concepts for the proposed simulation framework as well as methodology to induce noise simulated from high tension transmission lines to the magnetic sensor. Furthermore, it will go into the details of the publisher subscriber system used by both ROS and PX4 and how do they differ with each other.

2.2 An Introduction to the Project

Simulation of magnetic fields due to a current carrying conductor has been a classical problem since a long time. In the case of this problem however, the shape of the current carrying wire has a distinct shape. Since the current carrying conductor is suspended from two points, it naturally hangs in the middle due to weight of the wire itself. The exact shape can be approximated using the following formula.

$$y = a\cosh\left(\frac{x}{a}\right) = \frac{a\left(e^{\frac{x}{a}} + e^{-\frac{x}{a}}\right)}{2}$$

Where *a* is a constant. The plot of the y vs x is given by Figure 5:

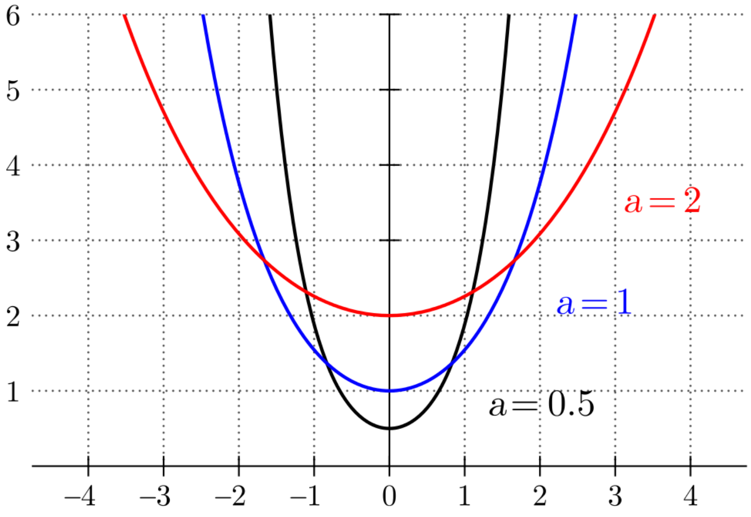


Figure 5: Mathematical plot of a catenary shape

Indeed this shape can be found in nature quite frequently as shown in Figure 6.

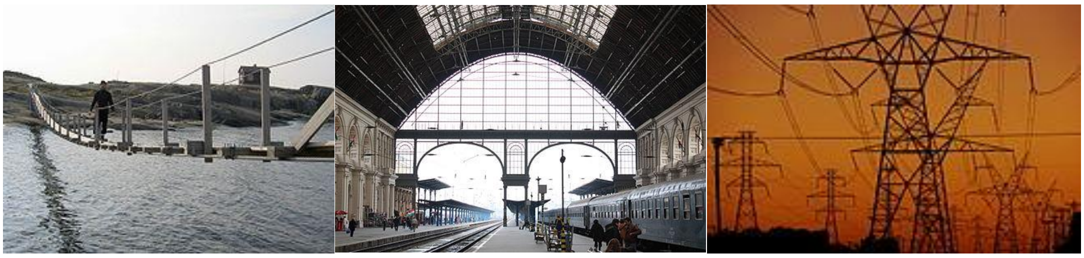


Figure 6: Examples of the shape of catenary are readily available in nature. (Courtesy: Wikipedia)

2.3 Literature Review

The magnetic fields due to a catenary shaped conductor is studied intensively by Lipscombe et al. Their study shows that a current-carrying wire hanging between two suspension points in a transverse magnetic field adopts a shape intermediate between a circle and a hyperbolic cosine. This magneto-catenary can be analytically calculated as a novel extension of the standard hanging chain problem in an intermediate mechanics course.

$$\pm X = \frac{1}{1-\beta}\left[\sqrt{\frac{2Y - Y^2 + \beta Y^2}{1+\beta}} - \frac{2\beta}{\sqrt{\beta^2 - 1}}\ln\left(\sqrt{1 + \frac{(\beta-1)Y}{2}} + \sqrt{\frac{(\beta-1)Y}{2}}\right)\right]$$

However, it was decided that for the sake of simplicity and generality, a simpler formula would be used that defines the magnetic field about a point near an infinitely long current carrying conductor.

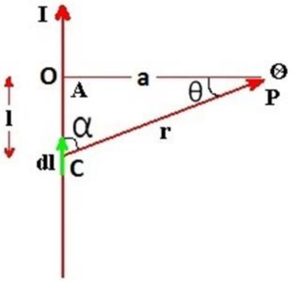


Figure 7: Infinite current carrying conductor exhibiting magnetic field at point P

From Figure 7, one can deduce that magnetic field due to a current carrying conductor of some length, where the ends of the conductors are making the angles α_1 and α_2 is given by,

$$B = \frac{\mu I}{4\pi R}(sin\alpha_2 + sin\alpha_1)$$

However, for our case, we assume that the power line is infinite. This assumption is valid if the dimensions of the drones are very small compared to the length of the wire. Thus our magnetic field equation becomes.

$$B = \frac{\mu I}{2\pi R}$$

We know that I is a sinusoidal function of time, with a frequency of 50 Hertz. Thus the magnetic field too, is a sinusoidal function of time.

$$B = \frac{\mu I_0 sin(2\pi 50t)}{2\pi R}$$

Where I_o is the amplitude of current.

$$B = A sin(2\pi 50t)$$

Where *A* is the constant given by

$$B = \frac{\mu I_0}{2\pi R}$$

2.3.1 General Analysis

Since the current is varying, B is actually a function of I. Current is varying as a sine wave with a fixed frequency of 50Hz. The amplitude of the current is assumed to be constant if the position of the aircraft is not moved and there is no variation in the current. This B becomes a sinusoidal wave with constant amplitude and a frequency of 50 Hz. This is the noise that is assumed to be induced on the magnetic field. Thus,

$$B = A\sin(100\pi t)$$

Where A is a constant.

Therefore, the plot of the magnetic noise induced at the sensor is given by Figure 8.

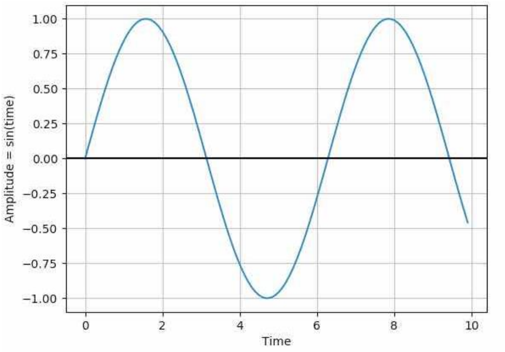


Figure 8: Noise disturbance

2.3.2 General Note

It can be observed that an oversimplification has been done to model the noise source. This is done due to the following reasons:

i. Simulation with real equations of magnetic field due to catenary was both difficult and taxing on the machine resources.

ii. The primary reason of oversimplification is that the aim of the project was not to model real world noise source, rather, to develop a general and robust algorithm capable of rejecting noise is any and all environment. Such an algorithm can be validated after development in real world scenarios. As for the simulation now, this was not needed.

2.4 Dynamic Model

The dynamics block is the most important block of the simulation. It models the quadcopter dynamics as a set of mathematical equations. Two frames of reference are introduced:
1. Space Frame
2. Body Frame
Thus, the orientation and position of the body are written respectively, in body frame and space frame:

$$Orientation, \quad \eta = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}$$

$$Position, \quad \xi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Where η represents the Euler angles in the body frame and ξ represents the Cartesian coordinates of the vehicle in the space frame.

The equations of motion are derived from Newton-Euler formalism.

$$mI_{3\times3}\dot{v} + \omega \times mv = \Sigma F$$

$$J\dot{\omega} + \omega \times J\omega = \Sigma M$$

Where m is the mass of the multirotor

I is a 3x3 Identity matrix.

v is the velocity of the aircraft in the Cartesian co-ordinate system

ω is the vector of angular velocities about X, Y and Z axes respectively

F is the force on the body

J is a 3x3 matrix given by

$$J = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}$$

Ixx, Iyy, and Izz are moment of inertias in the X, Y and Z axes respectively.
And M is the angular moment on the body.
From [1],[2], we can obtain,

In the body frame,

$$\ddot{\phi} = \frac{\left(\dot{\theta}\dot{\psi}\left(J_{yy} - J_{zz}\right) - k_{ax}\dot{\phi}^2 + u_2\right)}{J_{xx}}$$

$$\ddot{\theta} = \frac{\left(\dot{\phi}\dot{\psi}\left(J_{zz} - J_{xx}\right) - k_{ay}\dot{\theta}^2 + u_3\right)}{J_{yy}}$$

$$\ddot{\psi} = \frac{\left(\dot{\phi}\dot{\theta}\left(J_{xx} - J_{yy}\right) - k_{az}\dot{\psi}^2 + u_4\right)}{J_{zz}}$$

Where k_{ax}, k_{ay}, *and* k_{az}, are the rotational coefficient of drag in the X, Y and Z axes respectively, $u_{2,}$ u_3 and u_4 are roll input, pitch input and yaw input respectively after being mapped through equation described through A.

And in the space frame,

$$\ddot{x} = \frac{-k_x\dot{x}}{m} + \frac{u_1\left(\cos\phi\cos\psi\sin\theta + \sin\phi\sin\psi\right)}{m}$$

$$\ddot{y} = \frac{-k_y\dot{y}}{m} + \frac{u_1\left(\cos\phi\sin\theta\sin\psi + \sin\phi\cos\psi\right)}{m}$$

$$\ddot{z} = \frac{-k_z\dot{z}}{m} + \frac{1}{m}\left(\cos\phi\cos\theta\right)u_1 - g$$

Where k_x, k_y, k_z represent translation coefficient of drag in the X, Y, and Z axes respectively,
m is the mass of the aircraft and
u_1 represents the throttle input after being remapped by A

These six equations are used for the mathematical model of the quadcopter dynamics

2.5 Conclusion

Magnetic noise of a fixed frequency was used to model the noise source. This was done after oversimplification of the equations of magnetic field due to a catenary shaped conductor. In the next chapter, the tools used will be discussed and how the mathematical noise model is injected into the simulation stack.

CHAPTER 3
METHODOLOGY

3.1 Introduction

The methodology of the project was restricted to software testing. The effect of noise on the drone's system were observed on the drone using a popular drone platform known as PX4. This platform was then simulated in a feature rich physics engine-based simulator known as Gazebo. The features of PX4 were extended using ROS to facilitate the noise modelling.

More information about ROS, Gazebo, and PX4 can be found in [3][4][5]. Figure 9 presents the logos of the respective softwares.

Figure 9: Logos of ROS, Gazebo and PX4 respectively

In this chapter, the detailed discussion of the methodology will be discussed. In Section 3.2, the effect of magnetic interference due to high tension transmission lines on the sensor and state estimation system of the drone is observed. In the following sections, a model of the drone system is proposed, and individual components are studied. Further, we look at some assumptions made while modelling the quadcopter system and, in the noise, modelling due to high tension transmission lines. Finally, we draw some conclusions based on the studies done so far.

3.2 Methodology

The methodology of the project was roughly divided into two parts. The first part was to observe the effects of the noise source, the second part was to implement the model in Simulink and create a disturbance source.

The project is carried out in the following parts:

- Set up a simulation of a drone operating in a 3D world
    1. The simulation is to be set up using ROS and Gazebo
    2. The dynamics and 3D environment will be sourced from existing software

    Simulate magnetic fields due to high tension transmission line:
    1. Perform literature survey to determine the best method to simulate such fields
    2. Modify the existing simulated scenario to include magnetic interference

- Make observations on the drone sensors and create the model on Simulink
    1. Make observations and comparisons between different scenarios
    2. Perform literature survey to model the dynamics of the aircraft
    3. Model the dynamics and implement a noise source.

  Implement the proposed solution on the simulated drone

    1. Draw comparisons between the modified and unmodified setups
    2. Conclude the study with a final mitigation algorithm

In the next subsections, we discuss these parts in detail.

3.2.1 Simulation Setup

To observe the effects of the noise in Gazebo, a noise source was setup as explained in Section 2.3.1. Multiple open source tools and software were used in the simulation setup. This can be explained using the block diagram as shown below. The details of each components will be discussed in the next section.

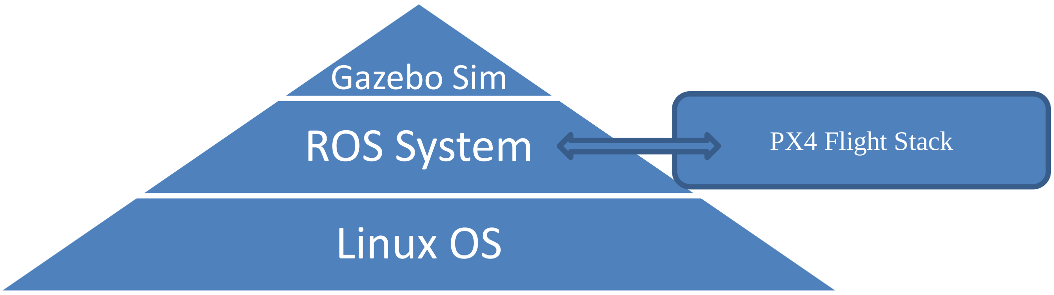


Figure 10: Simulation setup layers

The whole Linux OS is run on an Intel machine with amd64 architecture. On top of linux, a version of ROS runs with includes capabilities like RTPS based inter node communication and logging and debugging features. This ROS layer communicates with the PX4 flight stack firmware which is built for a Linux machine. Finally, the 3D world is visualized and simulated on Gazebo with features a rich physics engine.

Different variations of the publisher subscriber node system will also be discussed.

3.2.2 Tools Used

The following tools were used:

i. **Ubuntu Linux:** Ubuntu is a complete Linux operating system, freely available with both community and professional support.

ii. **ROS:** ROS (Robot Operating System) is a framework for robotics which provide libraries and tools to help software developers create robot applications. It supports hardware abstraction, device drivers, libraries, publisher/subscriber messaging, package management, etc.

The following diagram gives overview of ROS architecture:

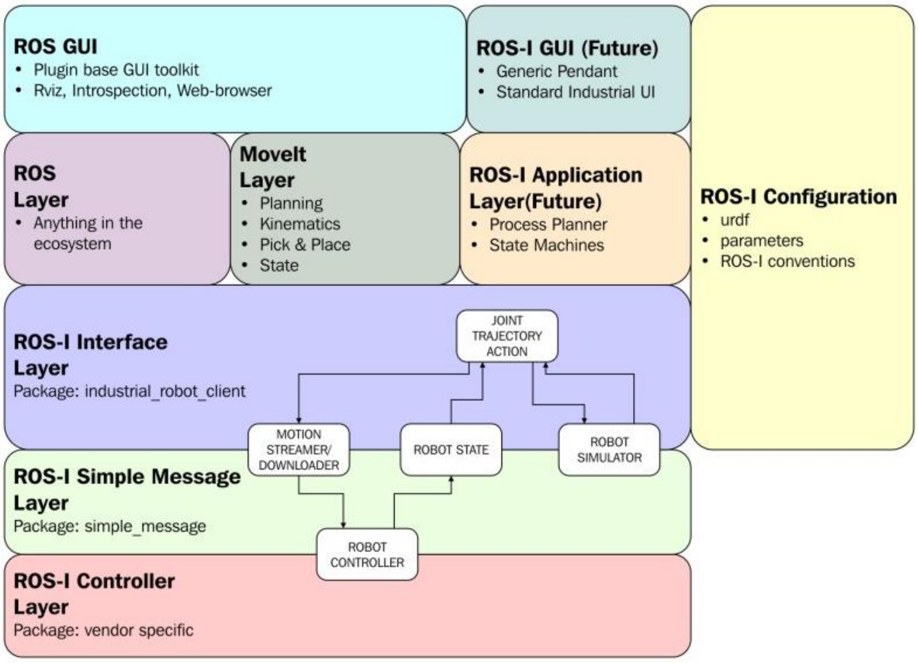


Figure 11: ROS Block Diagram (Courtesy: ros.org)

iii. **Gazebo Sim:** Gazebo is a 3D robotics simulator. It is open source. Gazebo uses ODE as its primary physics engine, OpenGL, and support code for sensor simulation and actuator control.

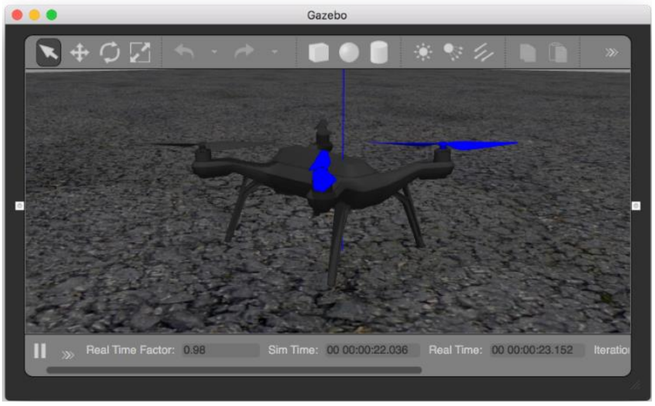


Figure 12: Gazebo Simulation running PX4 Flight Stack

iv. PX4 Flight Stack: PX4 is an open source flight control software for drones and other unmanned vehicles. PX4 provides a toolset for drone developers, and engineers to create technologies tailored for drone applications. The PX4 flight stacks block diagram is given in figure 13.

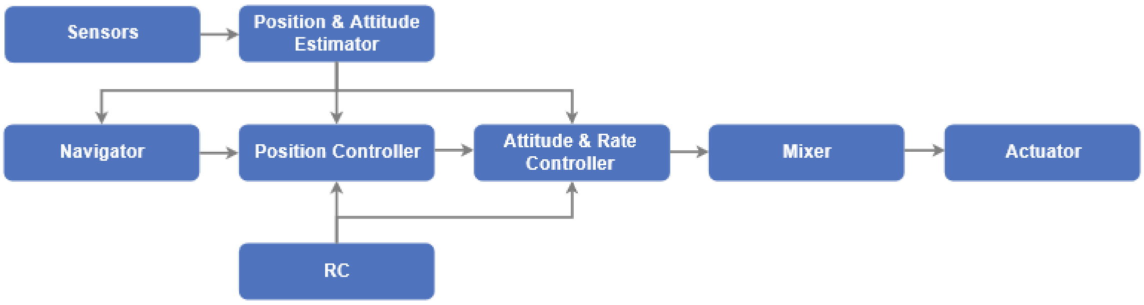


Figure 13: PX4 flight stack. (Courtesy: PX4)

The simulation can be shown in Figure 14. It shows a UAV based on PX4 flight stack simulated using Gazebo

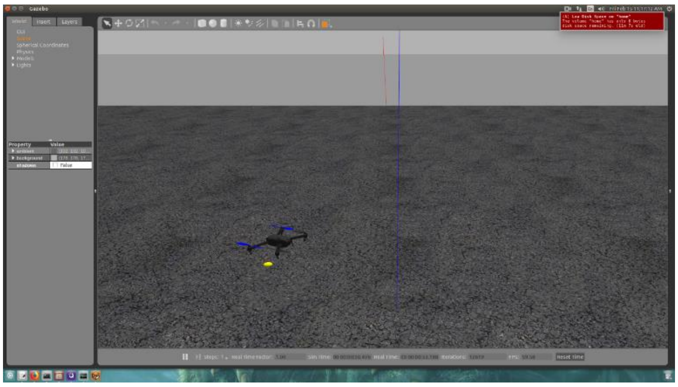


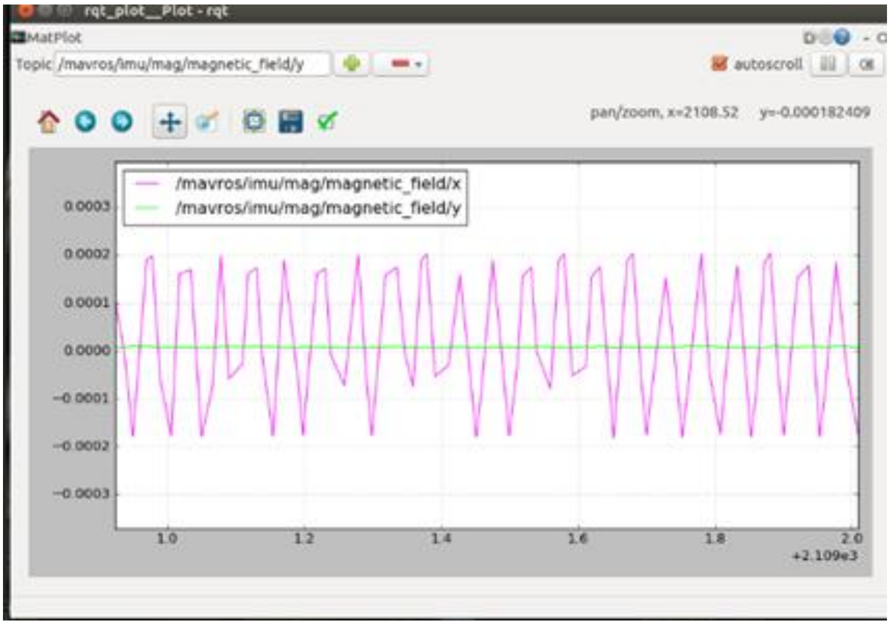
Figure 14: Gazebo Sim running PX4 Flight Stack Simulation

The PX4 flight stack uses the uORB publisher subscriber system, while the ROS uses Real Time Publisher Subscriber system (RTPS). This protocol is an industry standard and a default in all version of ROS. It enables inter node, inter thread communication with high speed. The middleware between PX4 and ROS converts uORB messages from PX4 and RTPS messages from ROS system. Therefore, the interface between PX4 and ROS can be demonstrated in the following diagram.

3.2.3 Noise observed in the Gazebo Simulation

Figure 15 shows the simulated interference from high voltage source on the magnetometer. The graph represents an approximate sine wave noise with the frequency of 50

Hz. At this point, the simulated drone declined to take off due to high variations in the sensor readings.


Topic /mavros/imu/mag/magnetic_field/y

autoscroll

pan/zoom, x=2108.52 y=-0.000182409

Figure 15: Noise observed in the Gazebo Simulation

3.2.3 Simulink Setup: Drone Dynamics

The drone system was setup as a classical control system. Concepts from modern control theory were also used, for example, state space modelling. The model is depicted in Figure 16. The best approach to discuss the following block diagram is to discuss each block from left to right.

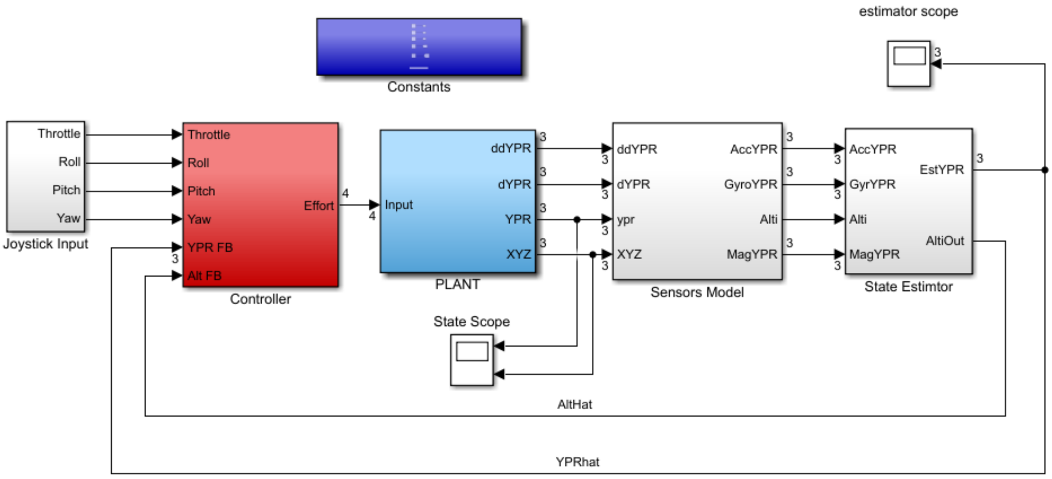


Figure 16: Drone System Block Diagram

Joystick Input Block: To control the simulated system, step inputs can be used. To make the system even more interactive, a joystick controller input were used. This was done to simply the process of testing. The joystick input block simply consists of the 'Joystick Input Block' from the Simulink Aerospace Toolbox. The outputs of the Joystick Input Block are remapped to suit our system. One can, thus connect a joystick to the simulation and do the testing. A typical commercially available joystick is shown in Figure 17.

Figure 17: The joystick used for simulation has two analog sticks to control the aircraft

The two analog inputs are primarily used to control the drone. This is a standard control schemes from multi-rotor vehicles. The mappings is shown in Figure 18.

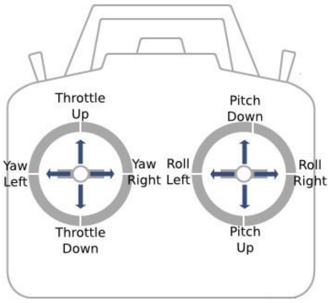


Figure 18: Control scheme for a quadcopter

Controller Block: The controller block contains PID controller for controlling the aircraft. The quadrotor dynamics are inherently an unstable system. Analogy can be drawn from the classical control problem of controlling an inverted pendulum, the quadcopter closely resembles an inverted problem. Moreover, a quadcopter is a 6DOF vehicle, with only four actuators. This makes the quadcopter an unstable, under actuated system. This makes it necessary to introduce some control schemes that will stabilize the drone. The PX4, like many other drones, use the PID control system. PID stands for proportional, integral, derivative control. As the name implies, the error signal is passed into a gain block, a gain block followed by integral block and a gain block followed by a derivative block. These three components of the signal are then added to form the *effort* signal. Thus the PID controller can be described by the mathematical equation given below.

$$effort = K_p e + K_i \int e\,dt + K_d \dot{e}$$

Where effort is the output of the PID controller, Kp, Ki, and Kd are the gains of the Proportional, Integral and Derivative components respectively. The dot notation represents derivative with respect to time. The e represents the error signal given by

$$e = setpoint - feedback$$

Where setpoint is the desired state and the feedback represents the current state from the sensors.

The integration component of the PID controller tends to build up, and thus dramatically increase the *effort*. This case is generally observed when the effort signal is not connected to the next block. To prevent this, a technique called Integral Anti-windup is used. It basically acts as a saturation block in front of the integral block, which prevents it from becoming too large. The PID block was utilized from the Control system Toolbox of Simulink. The PID gains were tuned manually by visual inspection of the step responses and incrementally varying the gains until a decent step response is obtained. The controller block is expanded in Figure 20.

Plant Block: This is the most important block in the system. It contains the dynamic model of the system along with other sub-blocks. The dynamics block is expanded as shown in figure 19.

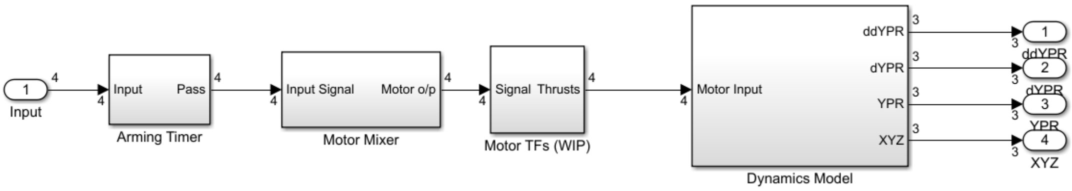


Figure 19: Plant Block

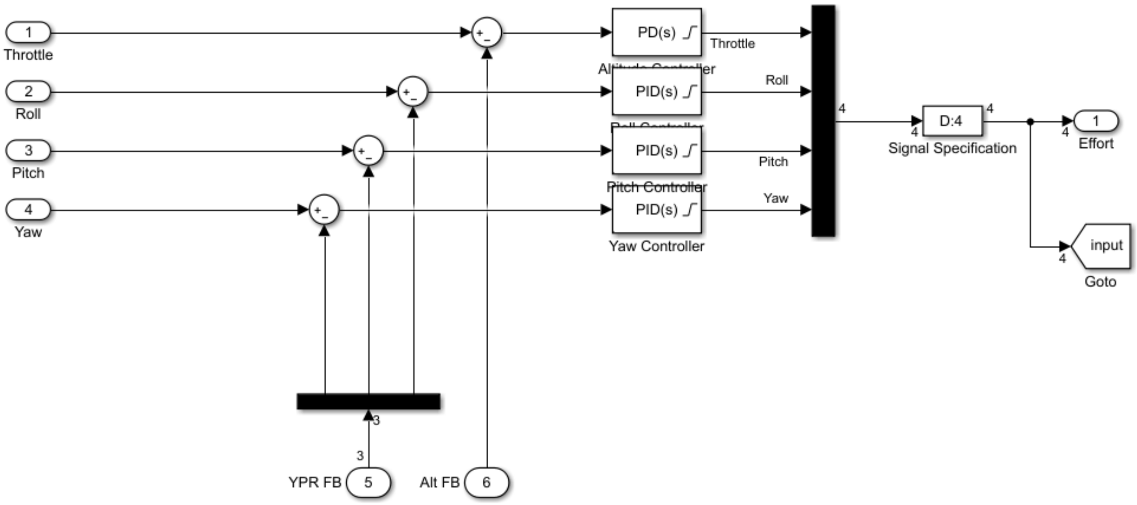


Figure 20: Controller Block. Roll, pitch and yaw controllers use a PID controller while the throttle controller use a PD controller. Feedback comes from the State Estimator block.

The arming timer block is simply a timer that stops any input from passing through for the first 5 seconds. This was done to ensure that the dynamics and the sensors are initialized properly before operation. This is similar to the arming mechanism used by PX4 and other popular drone platforms. The second block is the motor mixed block. This is an important block which maps the inputs to the necessary actuators. The motor mixer block makes sure that the inputs like roll, pitch and yaw are mapped to correct actuators. For example, when the user gives roll input, the actuators on one side receive higher input and the actuators on the other side receive lower input, effectively causing a misbalance in actuator forces, resulting in roll maneuver. Essentially, this block is a matrix multiplication with the input vector. The matrix is in a way a transform from the input space to the actuator space. The motor mixer block does the following mathematical operation.

$$out = Ai$$

Where *i* is the vector of all inputs and A is given by

$$A = \frac{1}{4}\begin{bmatrix} 1 & c^{-1} & -c^{-1} & d^{-1} \\ 1 & -c^{-1} & -c^{-1} & -d^{-1} \\ 1 & c^{-1} & c^{-1} & d^{-1} \\ 1 & -c^{-1} & c^{-1} & -d^{-1} \end{bmatrix}$$

Here, c is the drag constant on roll and pitch (due to symmetry) and d is the drag constant due to yaw maneuver. The values of c and d were taken from [3]

Dynamics Block: The dynamics block uses equations from Section 2.4 to model the dynamics of a quadrotor. The Dynamics block is shown in figure 21.

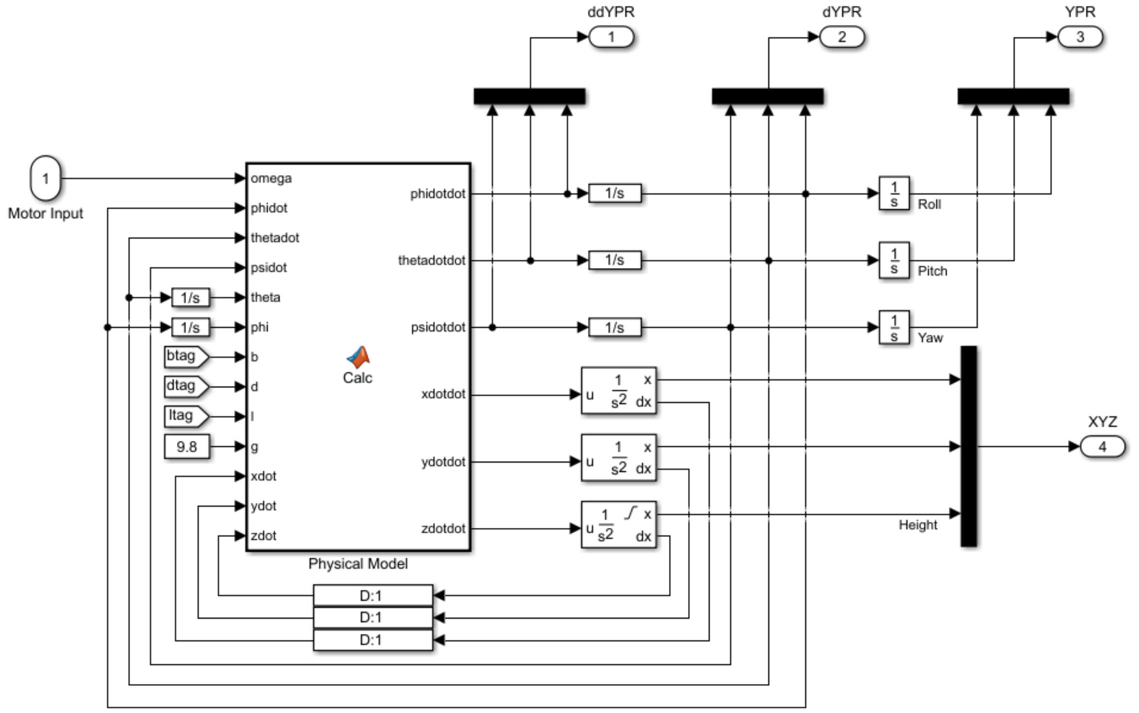


Figure 21: The Dynamics Block

The dynamics block uses the equations of motion described earlier to calculate the state of the vehicle in MATLAB's continuous time step. The equations themselves output the angular accelerations of the body frame and the accelerations of quadcopter in the space frame. The Euler angular accelerations are integrated to obtain angular velocities and are fed back to the physical model. Then, these Euler angular velocities are further integrated to get Euler angles.

Similarly, the Cartesian space frame accelerations are passed through a second order integrator to obtain Cartesian coordinates of the vehicle. The derivatives of the integrator are also fed back to the dynamics block.

The algorithm used for the dynamics block can be seen in the appendix section.

Sensor Block: Four sensors were modeled in the sensor block. The block diagram expansion is given in figure 22.

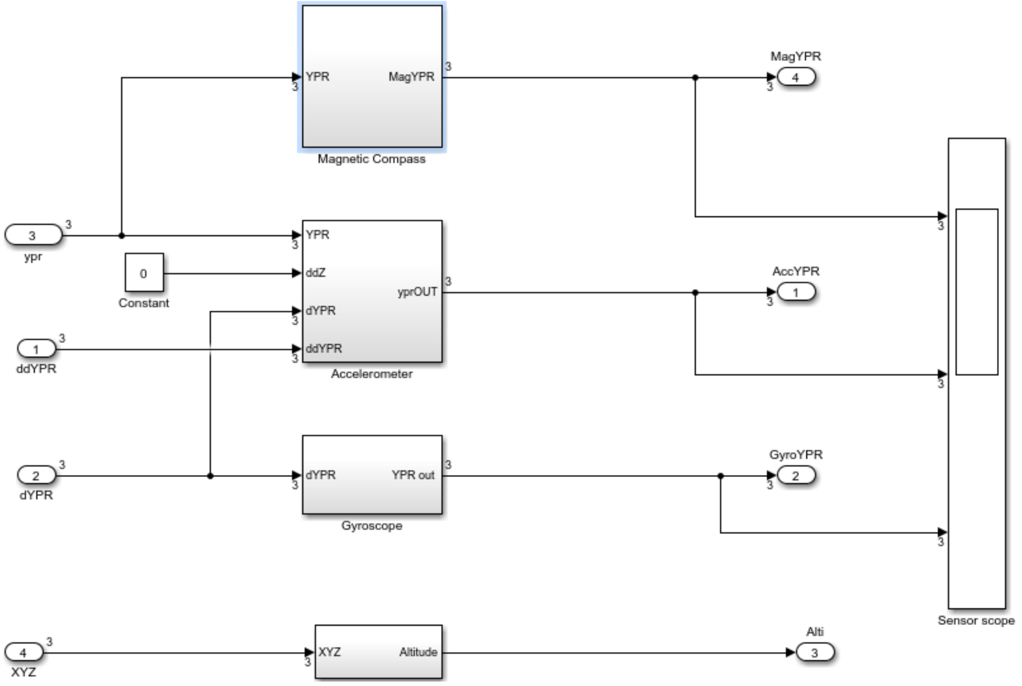


Figure 22: Sensor block diagram. The block consists of a magnetic compass, gyroscope, accelerometer and an altimeter

The accelerometer sensor and the gyroscope blocks were taken from the Simulink Aerospace toolbox library. The magnetic compass was modelled using [5]

The magnetometer has a delayed operation. That is to say, the response to change in magnetic field is not immediate. This kind of behavior can be captures using a transfer function. However, available literature on the magnetometer model was limited. In a few papers, frequency responses of magnetometers were analyzed. Thus system modelling for the magnetometer was based on approximate pole placement according to those paper. For our magnetometer, [5]'s analysis of the frequency response was used. Thus, the magnetometer block was modelled and the block diagram expansion is illustrated in Figure 23.

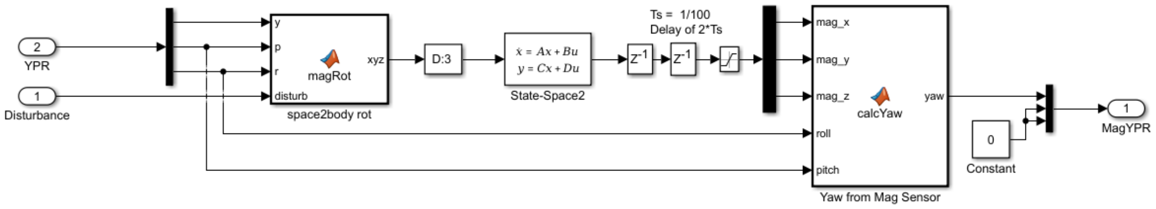


Figure 23: The magnetometer block

The transfer function has a pole at s=10. This implies that there is a slope of -20 dB per decade. This was well captured in the transfer function as shown in the Bode plot in figure 24.

For reference, the frequency response from analytical data are shown in figure 25.

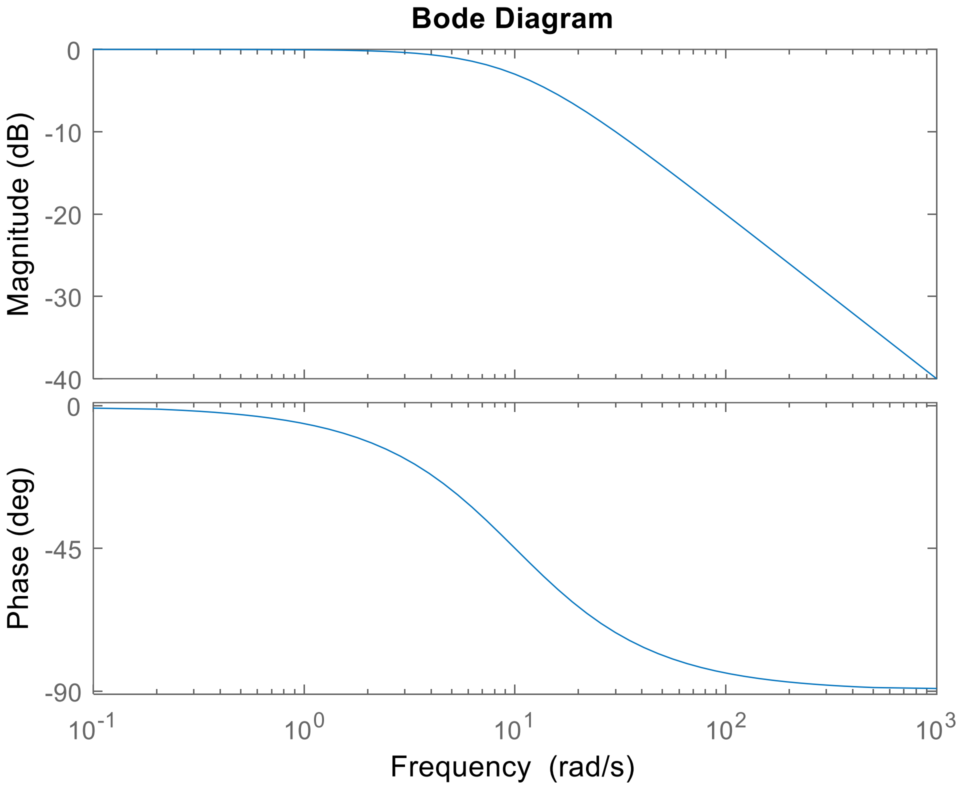


Figure 24: Magnetometer sensor bode plot

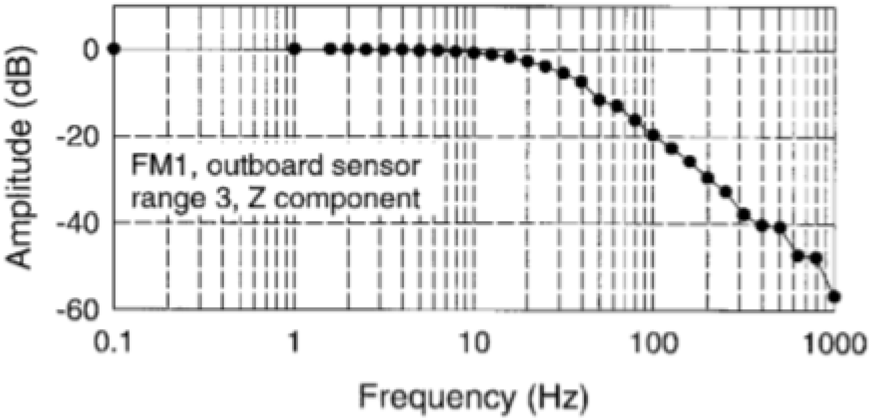


Figure 25: Experimental frequency response. Courtesy: [5]

State Estimator block: The state estimator block is just a complementary filter which takes inputs from all the sensors and outputs a weighted sum of all inputs. The weights can be adjusted based on the reliability of the sensor.

A complementary filter is just a weighted sum of the sensor outputs. The mathematical equation is,

$$\hat{x} = (1 - c)y_1 + cy_2 \;\; ; \;\; c \in [0,1]$$

This filter can help reduce noise and in used in some of the non-autonomous drone platforms like BetaFlight, LibrePilot, CC3D etc. [7][8][9]

3.2.4 Simulink Setup: Disturbance Modelling in the Magnetometer

The disturbance was modelled as shown in figure 26

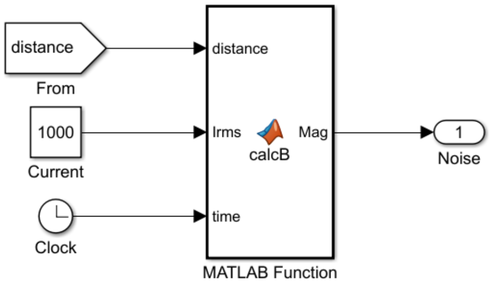


Figure 26: The noise source model

Where the outport connects to the magnetometer disturbance port.

Amplitude of the disturbance sine wave is a function of distance to the disturbance source. Again, the disturbance source is modelled as an infinite current carrying conductor with sinusoidally varying current. The expression of amplitude is given by.

$$B = \frac{\mu I_0}{2\pi R}$$

Assuming a 11KV line on a 1000KVA single phase distribution system, then, for $\mu = \mu_0$, the current flowing through the line is 90.9A. Then the induced magnetic field at 1m distance is 9E-6 Tesla. This is equal to 0.18 Gauss. If the drone is 0.5 meters away from the wire, the magnetic field increases to 0.36 Gauss. In contrast, the earth's magnetic field ranges from 0.3 Gauss to 0.6 Gauss. This quantity is comparable to the amplitude of disturbance. A graph of distance versus magnetic field, with 1000KVA line and 10MVA line carrying 10 KV is given is given in figure 27.

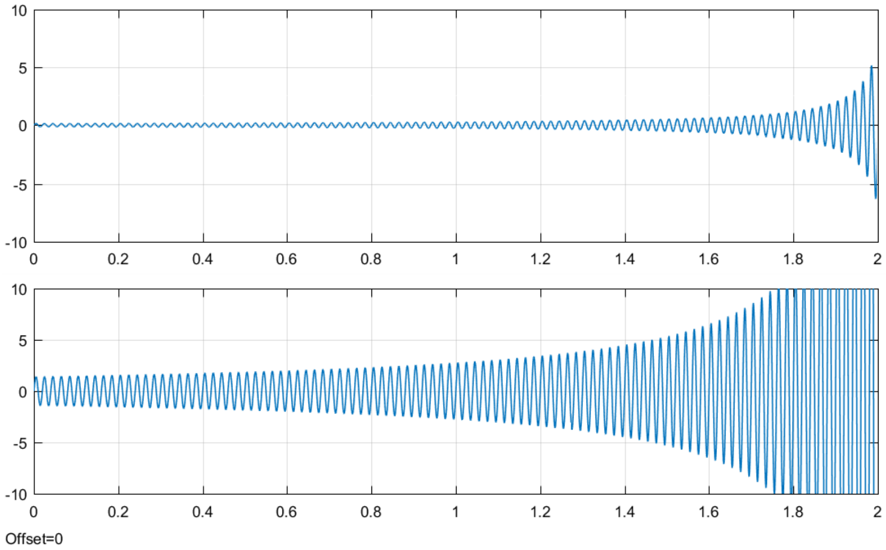


Figure 27: Disturbance visualization between 1000KVA line and 10MVA line.

The yaw state estimation obtained from the magnetometer sensor measurement also varies similar to figure 28.

3.3 The Mitigation Algorithm

3.3.1 Purpose of the mitigation algorithm

The purpose of the mitigation algorithm is to either reject the disturbance in the Yaw sensor output. However, if the disturbances are too much, that is, the amplitude of the sinusoidal noise has saturated the yaw sensor output, then the task of the mitigation algorithm is to stop the magnetic sensor and push other sensor outputs to the state estimator block, for instance, the gyroscope output. The reason a magnetometer might saturate is due to mechanical constraint or technical specification. The methods used by the mitigation algorithm is given in the following sub-sections. The block diagram of the mitigation algorithm is given in Figure 29.

3.3.2 Block Diagram of the mitigation algorithm

The mitigation algorithm itself is divided into two subsystem. The first block calculates the energy of the input signals for a 100 data points at its time. The energy function itself is just the sum of squares of all the samples. This energy functions can help determine if the magnetometer block is saturated or not. The plot of energy function as the disturbance increases is given in Figure 29.

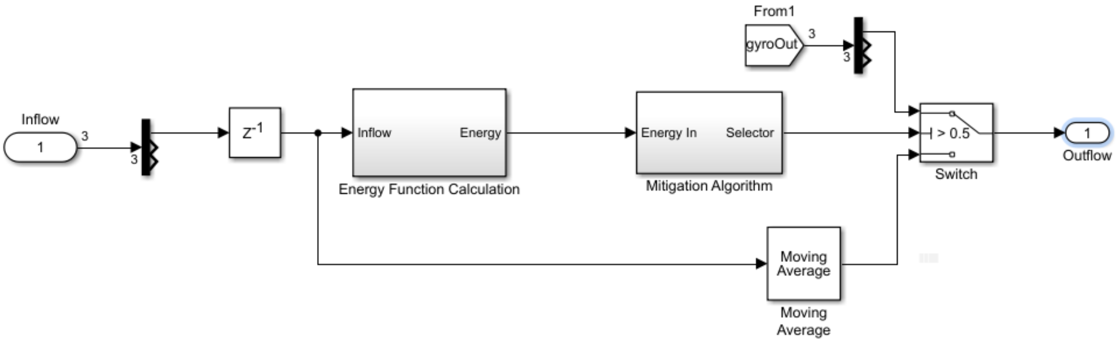


Figure 28: Mitigation Algorithm Block Diagram

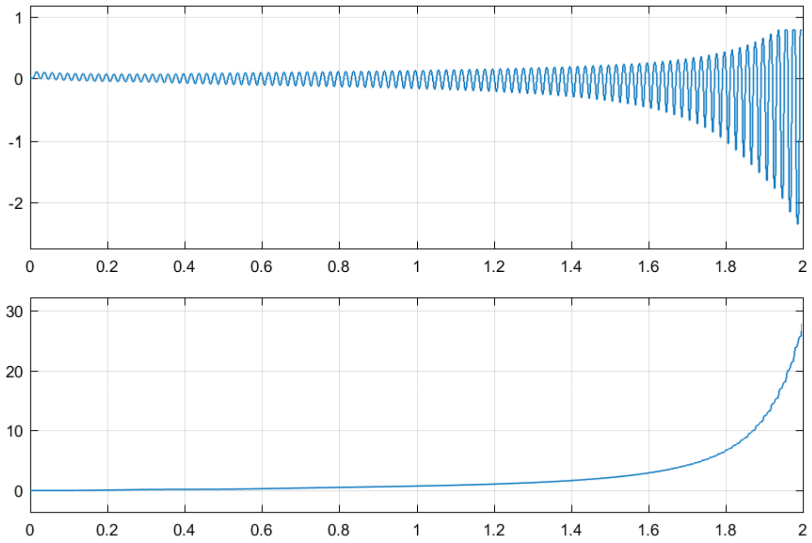


Figure 29: Plot of yaw when as disturbance increases due to reduction in distance. The second plot shows the corresponding energy function

From here on, the working of the mitigation algorithm can be explained in two cases. From figure 29, it was found that an appropriate value to consider the sensor has saturated would be around 10. Then the procedure of the mitigation algorithm can be explained by the following paragraphs.

If energy is less than 10: The sensor has not saturated. This implies that the additive noise can be removed. Since the noise is just a single frequency, the use of an FIR or IIR filter would be unnecessary. The idea here is to subtract the periodic variation of the noise from itself. If we using a sliding window averaging filter, we can easily average out the noise component of the signal. Since the noise has zero DC component, (noise is a sigle frequency), basically we eliminate all of the noise. Thus, the moving average block output is passed to the state estimator.

If the energy is more than 10: this implies that the sensor might be saturated. This means that the sensor output will represent a square wave oscillating at then minimum and maximum saturation points of the sensor. In this case, the moving average filter will just give a DC output. The actual yaw has been drowned by the noise. In this case, it would be impossible to recover the signal by using any filter at all. Thus, the pass-through will be from a gyroscope. It will reduce the stability of the system, but still the aircraft will still be airborne to perform escape maneuvers.

3.3.3 Unexpected Finding

The sampling frequency has also to be taken into account in the mitigation algorithm. There is an interesting result that come up as part of the analysis. This is because the sine wave with a frequency of 50 Hz maybe at the borderline of Nyquist criterion. This means, that the sensor output will not have a decent reconstruction of the sine wave signal if the sampling frequency of the magnetometer of about 100 Hz. Below 100 Hz, the noise will be aliased. Above 100 Hz, the mitigation algorithm will work better as we go higher in frequency.

It is possible to make the control loop run at a frequency of more than 100 Hz. However, the bottleneck might be the sensor itself. For example, if the sensor is hardwired to work at 100 Hz, then the noise will be sampled at evenly at every period. That means, the sampling might happen when the sine is zero, but also when sine is at its peak. If it's the latter case, then, the noise will appear as a DC offset on the actual signal. There is no way for the mitigation algorithm to differentiate between the two components of the signal in such a case. Hence, it would be best to not trust the magnetometer in such a case.

Since, this is a hardware limitation, the mitigation algorithm still has to make sure the system is stable, even when in proximity of a noise environment. Here, unfortunately, the user has to interfere. The user has to be careful to switch on mitigation algorithm when operating in a noisy environment. Then magnetometer reading will not pass through the sensor.

This can be done if the certain feature is mapped to a button on the user's remote controller.

CHAPTER 4
RESULT ANALYSIS

4.1 Introduction

This chapter discusses the validations for the model. Validation includes the stability analysis of the drone model, analysis of the noise model, as well as, the mitigation algorithm. In the next sections, various step responses will be plotted. The noise model will be analyzed. Finally, the performance of the mitigation algorithm will be tested.

4.2 Result analysis

In the following two sub-section, a detailed analysis of the results obtained so far has been done. In Section 4.2.1, general results are plotted. In section 4.2.2, brief discussion about the plots obtained is done.

4.2.1 Graphical / tabular form

In this section, the plots are observed and commented on. These plots give us insights about the actual performance of the mathematical model of the drone, the disturbance characteristics, and the performance of the mitigation algorithm. Firstly, we discuss the characteristics of the mathematical model and the dynamic equations. Further, we will see the sensor response. We will then take a look at the disturbance model and how disturbance is observed if the sampling frequency of the sensor is varied. Finally, we take a look at the mitigation algorithm, and the performance of the quadrotor after mitigation algorithm has been put to place.

Brushless motors have a delay in operation. Usually, brushless dc motors take up to 1 to 2 seconds to reach the settling point. Thus the brushless motor was modelled as shown in Figure 30.

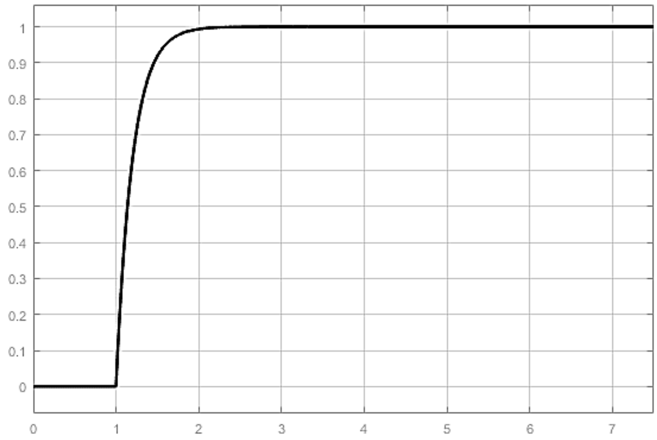


Figure 30: Step Response of the Motor. The scale is the angular velocity of the motor linearly mapped by the mapping [0, ωmax] → [0, 1]

This response is approximated from the experiments done by [8]. As we can see in Figure 31, the settling time for a brushless DC motor is about 0.5 seconds to 1 second.

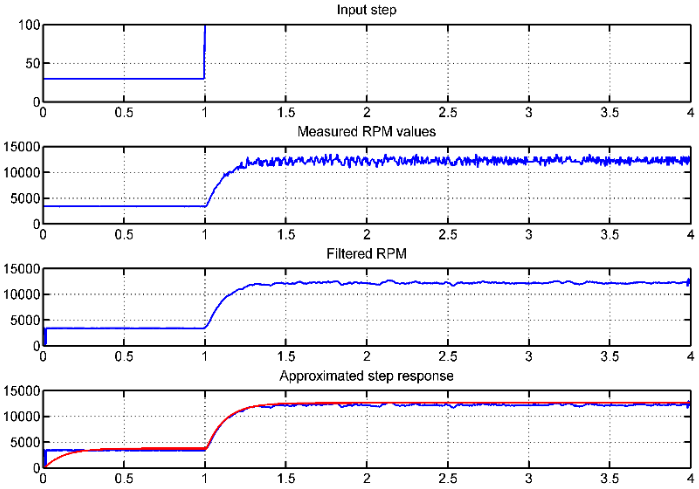


Figure 31: Experiment to obtain the step response of the brushless DC motor.

Now, we take a look at the dynamic model of the quadcopter. In Figure 32 and Figure 33, the model has first been given an altitude set point. It takes some time for the quadcopter to settle at that time considering the motors saturate at a maximum symbolic angular velocity. In Figure 32, after the set point has been reached, a Roll step input is given. Similarly, in Figure 33, a yaw step input is given post the settling of the altitude. The step responses show a minor overshoot followed by quick settling. This shows that the PID controllers are working properly. Thus the system is stable. A point to be noted is the quadcopter model is equivalent in roll and pitch axes. Therefore, only roll step response is plotted, since, pitch step response is identical to roll step response.

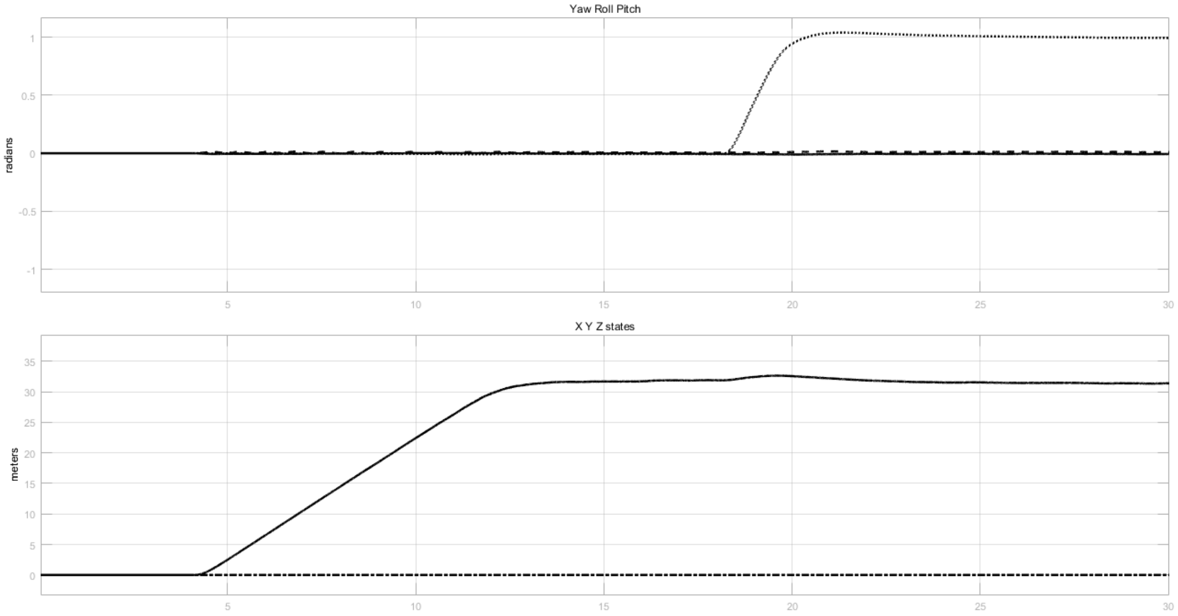


Figure 32: Altitude Step Response followed by Roll Step Response

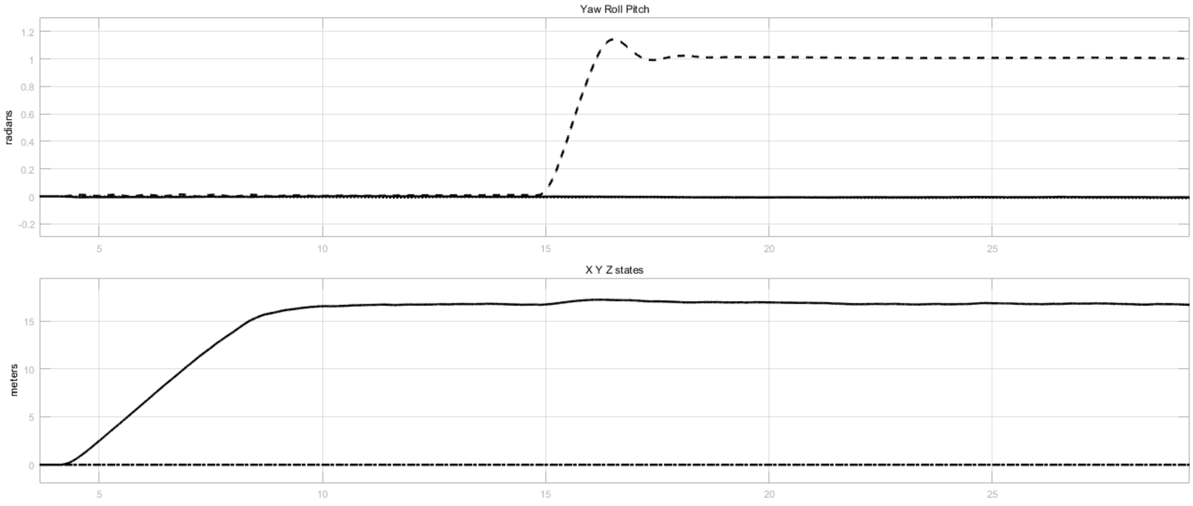


Figure 33: Altitude Step Response followed by yaw Step Response

In the idle state, sensors still produce some output. This is due to the introduced white Gaussian noise, numerical inaccuracies as well as quantization noise being built up due to integrators. Figure 34 shows the plot of the yaw sensor when the system is idle. One must note the scale of the Y axis. This plot shows that, while there is noise, it is minimal and does not affect much the estimated state of the graph.

In figure 35, the disturbance model is showcased. This graph represents the drone moving closer to the transmission line as time passes. Specifically, the noise received by the sensor is plotted in the first subplot, as the drone moves from 2 meters away from the source to 0.1 meters away from the source. The current in the conductor was assumed to be 100 A. With that current, it can be said that the disturbance magnitude surpasses the typical earth's magnetic field magnitude at around 1 m away from the noise source. For reference, the earth's magnetic field varies from 0.3 Gauss to 0.6 Gauss depending on the geographical location.

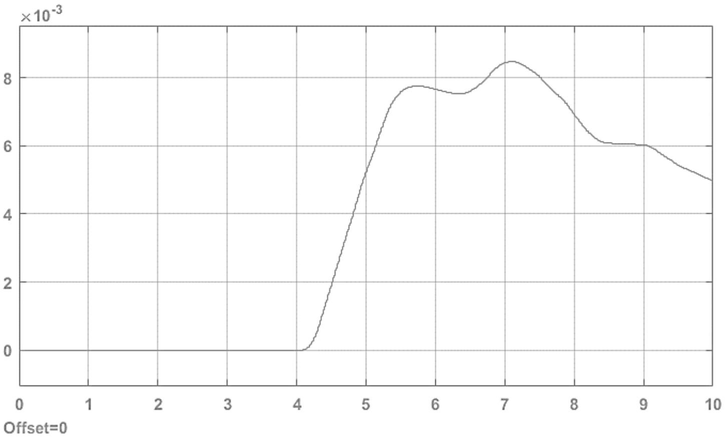


Figure 34: Yaw response when Idle. Notice the x axis scale. The minute error is probably due to numerical inaccuracy, or quantization noise

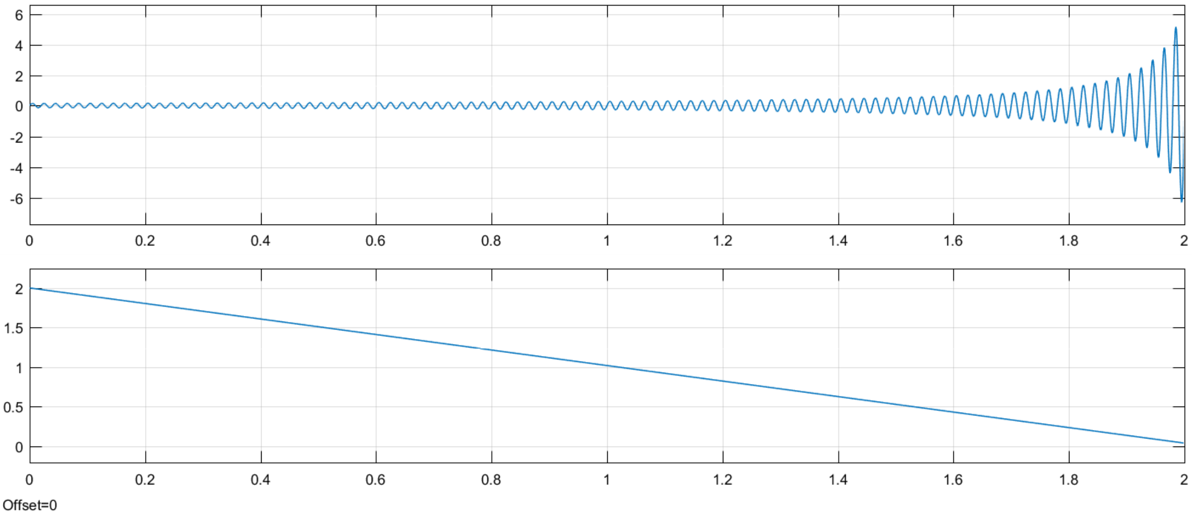


Figure 35: Noise vs Distance. The distance is decreased from 2 meters to 0.05 meters towards the noise source. The current through the wire is 100 A

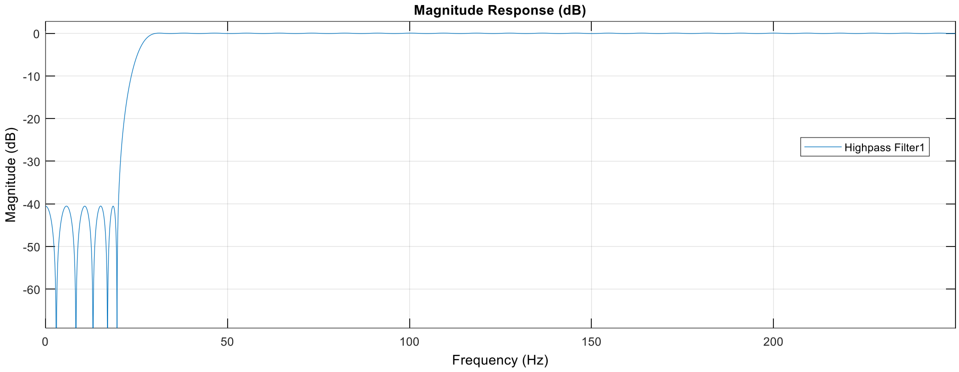


Figure 36: A high pass filter was added before the energy calculation block to get rid of the DC component of the signal. The DC components may be the current yaw position.

In figure 36, the noise can be visualized as a function of distance to the noise source.

The disturbance in yaw can be observed in figures 37, figure 38 and figure 39.

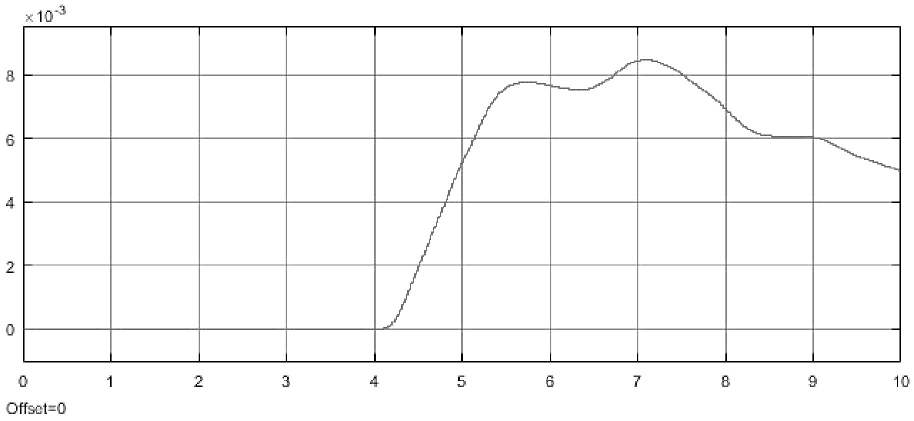


Figure 37: Reference Yaw trajectory of 10 seconds

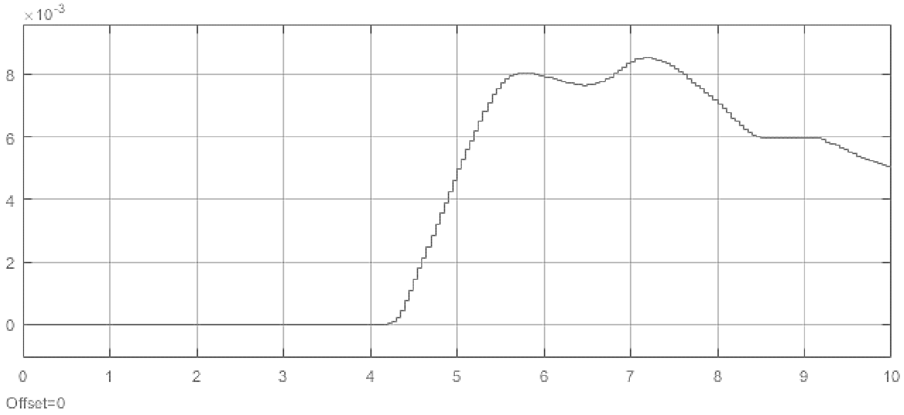


Figure 38: Yaw trajectory with 50 Hz noise. Note that the noise has been aliased out.

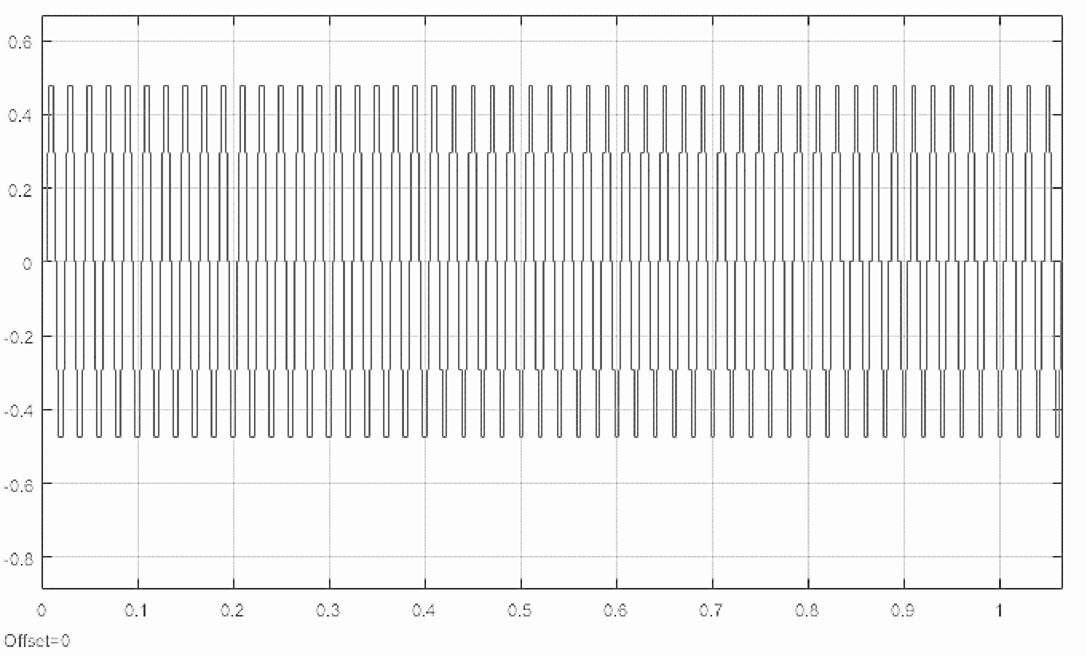


Figure 39: Magnetometer working at 500 Hz with the 50 Hz sinusoidal noise. Noise greatly overpowers the yaw state.

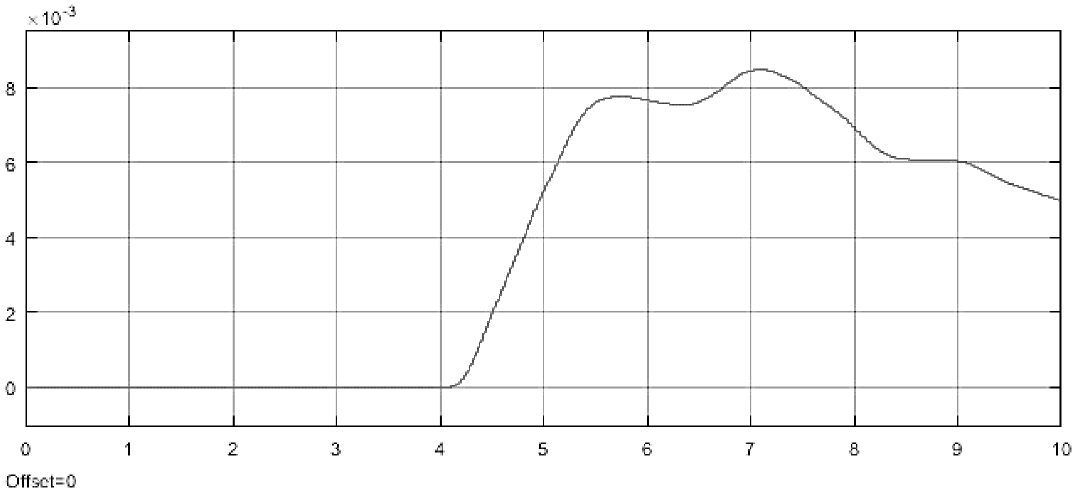


Figure 40: Yaw estimated state after passing through the mitigation algorithm. Since the sampling frequency of the sensor was 500 Hz, the moving average filter was used with size 500 Hz / 50 Hz = 10. This completely eliminates the noise signal.

4.3 Significance of the result obtained

Figures 37 through 40 demonstrate the capability of the mitigation algorithm in a noisy condition. This phenomenon can be further verified by the presence of a notch in the frequency response of a 10 taps moving average filter for a sampling frequency of 500 Hz. This bode plot is shown in figure 41. The frequency axis is in rad/s. For 50 Hz, the formula is given as

$$\frac{2\pi 50}{500} = \quad 0.6283$$

Hence the presence of the notch at 50 Hz.

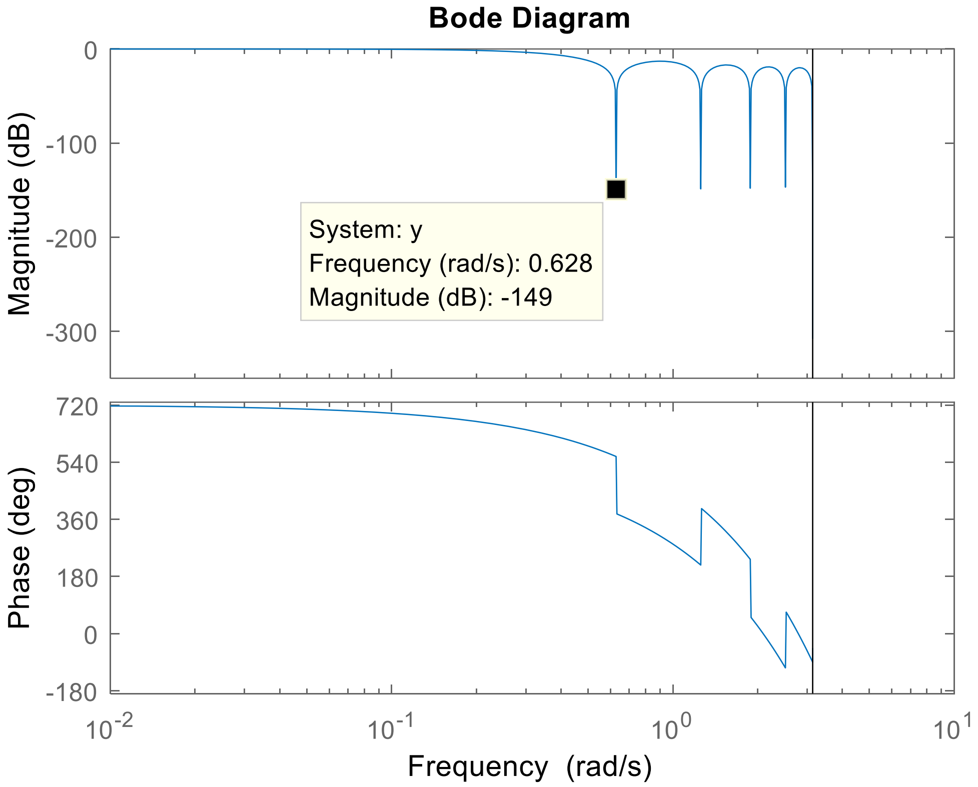


Figure 41: Bode plot of the moving avegage filter showing a notch at the frequency of 0.628 rad/s. This correspondes to 50 Hz at a 500 Hz sampling rate.

4.4 Deviation from expected result

The sampling frequency has also to be taken into account in the mitigation algorithm. There is an interesting result that come up as part of the analysis. This is because the sine wave with a frequency of 50 Hz maybe at the borderline of Nyquist criterion. This means, that the sensor output will not have a decent reconstruction of the sine wave signal if the sampling frequency of the magnetometer of about 100 Hz. Below 100 Hz, the noise will be aliased. Above 100 Hz, the mitigation algorithm will work better as we go higher in frequency.

This phenomenon is explained in detail in section 3.3.3.

4.5 Conclusions

In this chapter, first we validated the dynamics of the quadcopter’s mathematical model. We gave various step inputs to the dynamics model and observed their outputs.

The step responses and study of the noise model was done. The dynamic stability of the drone is observed through various step responses. Then, multiple conditions, through which the noise source affects the drone sensor system are created. The effects of these simulations are observed, and the sensors response are plotted.

A qualitative study has been done on the mitigation algorithm. Finally, the performance of the mitigation algorithm is done. The detailed analysis of the result obtained is also done by plotting the bode plot of the mitigation algorithm and observing the notch on the magnitude at the desired point.

Finally, a small note is given for the sensors with a lower sampling frequency than the Nyquist criteria for the sinusoidal noise.

CHAPTER 5
CONCLUSION AND FUTURE SCOPE OF WORK

5.1 Summary

In this chapter, a brief summary of the work done so far will be discussed.

5.1.1 Problem statement

One of the use cases of drones may be inspection and current sensing in high tension transmission lines. Specifically, a drone may be used obtain current measurements from the high-tension lines via a flux sensor. However, such a scenario insinuates a peculiar problem. Large, periodic voltage variations in the transmission line produce varying magnetic field which may affect the drone's sensor, especially, the magnetometer. This can lead to erroneous state estimation which may drive the drone's system to instability. This project aims to simulate such a scenario, observe its effect of the sensor readings, and propose solutions to mitigate the problem.

5.1.2 Work methodology

The methodology of the project was roughly divided into two parts. The first part was to observe the effects of the noise source, the second part was to implement the model in Simulink and create a disturbance source.

The first part of the project was to set up a simulation of a drone operating in a 3D world. In this part, the simulation was set up using ROS and Gazebo. The dynamics and 3D environment were sourced from existing popular drone platform known as PX4. Simulation of magnetic fields due to high tension transmission line were performed using ROS and Gazebo. The noise model used was derived from the first principles. Further the simulation was modified to include magnetic interference due to the noise source.

The next part of the project was to make a replica of the drone model on Simulink. Here, approximate and reduced order models of the drone dynamics, the motor transfer functions, the sensor systems were mathematically modelled. Toolboxes that Simulink provides were also exploited to model PID controllers, gyroscope and the accelerometer. Step responses, and other plots were plotted to study the behavior of the drone in cartesian space. Magnetometer's system was obtained through system identification through frequency response [5]. A noise model injected the sine wave noise to the magnetometer block. The effects were observed for different parameters. Finally, the potential mitigation algorithm and mitigation block was proposed and compared with bare-bones drone's system.

5.2 Conclusions

The project was done in two parts, the first part was to set up a simulation 3D scenario of the PX4 platform using Gazebo and ROS. In the 3D Simulation scenario, noise sources were introduced as described earlier in this report. The projects starts with a simulation of the drone in a 3D environment. This is followed my demonstration of the capabilities of simulation. Then, a noise source is introduced and its effect on the sensor systems are studied.

Magnetic noise of a fixed frequency was used to model the noise source. This was done after oversimplification of the equations of magnetic field due to a catenary shaped conductor. In the next chapter, the tools used were discussed and how the mathematical noise model is injected into the simulation stack. The same was simulated on the dynamics model on Simulink.

In Simulink, the drone's control system was modelled so as to replicate the models used by commercially available drone platforms like PX4, Ardupilot etc. This includes dynamic modelling of the equations of motions, sensor modelling and state estimation block modelling. The disturbance block was also modelled.

Finally, a mitigation algorithm was implemented. The block was designed so that it would fit between the state estimation block and the sensor block. The mitigation algorithm block was designed to work with sensors with any given sampling frequency. If the sampling frequency of the sensor is 100 Hz or less, then the mitigation algorithm cannot work and has to pass through other sensor readings to the state estimation block. In this case, an input from the user is required. If the sampling frequency of the sensor is more than 100 Hz, at about 200 Hz or more, then the mitigation algorithm can work properly by designing a moving average filter of length Fs/50, to get rid of the 50 Hz sinusoidal noise.

The mitigation algorithm was tested in Chapter 4 and an analysis was done on the frequency response of the filter. Finally, a note was made for sensors with lesser sampling frequency. Finally, the project concludes with a summary of the work done so far, including the methodology and qualitative assessment. Some insight for the furture scope of the research is also given in the next section.

5.3 Future scope of work.

The future of the projects opens a lot of windows for further research and development. First and foremost, the algorithm still needs to be improved. Research can be done for sensors with lesser sampling frequency. Further extensive testing needs to be done in an accurate simulation setup to validate both the algorithm and the simulation of the noise source. Thus, the simulation setup can be further improved to include other phenomenon like noise from multiple current carrying conductors etc.

Finally, a real world testing of the hardware should be performed. Such a test will assess the mitigation algorithm in a real world scenario and will completely validate the mitigation algorithm, following which, the drone can be deployed for operation near high tension magnetic noise sources as transmission lines.

REFERENCES

[1] Artale, V. ,Milazzo,C.L.R. , and Ricciardello,A. (2013). “Mathematical modelling of hexacopter.” 392 Applied Mathematical Sciences,8

[2] Balas, C. (2007). “Modelling and linear control of quadrotor.” MSc Thesis, Cranfield University

[3] Robot Operating System Documentation, *https://www.ros.org/*

[4] Gazebo Simulator Documentation, *https://gazebosim.org/*

[5] PX4 Documentation, *https://px4.io/*

[6] Balogh, A., M. W. Dunlop, S. W. H. Cowley, D. J. Southwood, J. G. Thomlinson, K. H. Glassmeier, G. Musmann et al. "The Cluster magnetic field investigation." Space Science Reviews 79, no. 1-2 (1997): 65-91.

[7] Lipscombe, Trevor C., and Carl E. Mungan. "The magnetocatenary." European Journal of Physics 33.6 (2012): 1667.

[8] Website: Squadrone (Swarm Operating Quadcopter) *http://squadrone.hagenberg.servus.atl*

ANNEXURES

Dynamics Calculation

```
function [phidotdot,thetadotdot,psidotdot,xdotdot,ydotdot, zdotdot] =
Calc(omega, phidot, thetadot, psidot, theta, phi, b, d, l, g, xdot, ydot,
zdot)
    % Rotational coeffecient of drag about respective axes
    kax = 1;
    kay = 1;
    kaz = 1;

    % Translational coefficient of drag in respective axes
    kx = 20;
    ky = 20;
    kz = 20;


    % Other coefficients
    m = 1;
    Ixx = 1/6;
    Iyy = (sqrt(2)*l)^3/12;
    Izz = (sqrt(2)*l)^3/6;


    % Calculation of states
    phidotdot = (kax/Ixx) * ((thetadot*psidot*(Iyy - Izz)) -kax*phidot +
(b*l/sqrt(2))*(omega(1)^2 - omega(2)^2 - omega(3)^2 + omega(4)^2));

    thetadotdot = (kay/Iyy) * ((phidot*psidot*(Izz- Ixx)) -kay*thetadot +
(b*l/sqrt(2))*(-omega(1)^2 - omega(2)^2 + omega(3)^2 + omega(4)^2));

    psidotdot = (kaz/Izz) * ((thetadot*phidot*(Ixx- Iyy)) -kaz*psidot +
d*(omega(1)^2 - omega(2)^2 + omega(3)^2 - omega(4)^2));

    xdotdot = -kx*xdot/m;
    ydotdot = -ky*ydot/m;
    zdotdot = (1/m) * cos(theta)*cos(phi) * b*20 * (omega(1)^2 +
omega(2)^2 + omega(3)^2 + omega(4)^2) - kz*zdot/m - g;
```

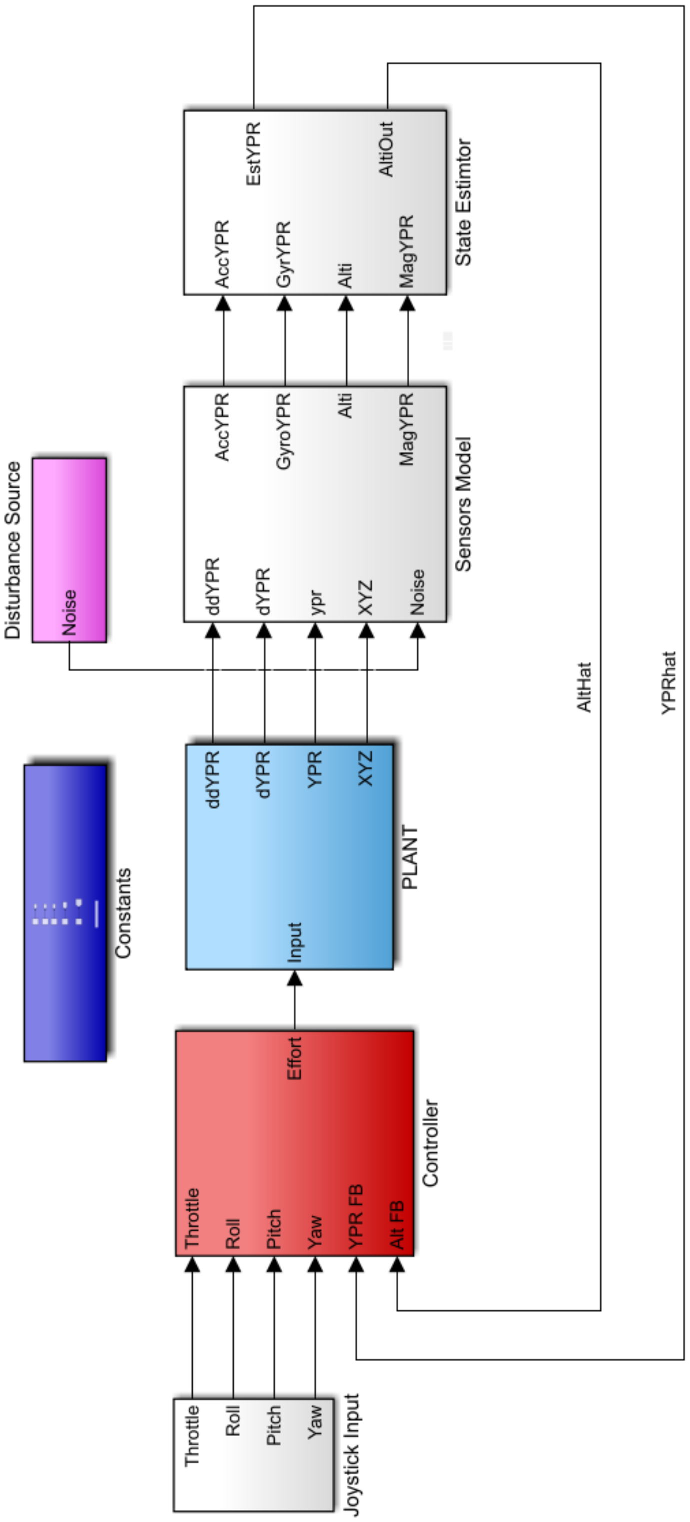
Magnetometer

```
% space to body transformation matrix

function xyz = magRot(y,p,r, disturb)
  rot1 = [1 0 0 0; 0 cos(r) -sin(r) 0; 0 sin(r) cos(r) 0; 0 0 0 1];
  rot2 = [cos(p) 0 sin(p) 0; 0 1 0 0; -sin(p) 0 cos(p) 0; 0 0 0 1];
  rot3 = [cos(y) -sin(y) 0 0; sin(y) cos(y) 0 0; 0 0 1 0; 0 0 0 1];
  rotsb = rot1*(rot2*rot3);
  temp = rotsb * ([0.6; 0; 0; 0] + disturb);
  xyz = temp(1:3);

% Calculation of yaw from magnetic sensors

function yaw = calcYaw(mag_x, mag_y, mag_z, roll, pitch)
x = mag_x / sqrt(mag_x^2 + mag_y^2);
y = mag_y / sqrt(mag_x^2 + mag_y^2);
z = mag_z / sqrt(mag_x^2 + mag_y^2);
yaw = atan2(y/(cos(roll)),x/cos(pitch));
```

Mitigation Algorithm

```
% calculate energy from 100 taps

function energy = energy(taps)
    energy = sum(taps.^2);


%The algorithm itself

function outflow = mitAlgo(Fs, energy, miti)
outflow = 0;
% If the sample time is 1/200
% Turn off magnetometer
if miti==1;
    if Fs <= 200
        outflow = 0;

    % If energy is less than threshold,
    % use filter.
    elseif energy < 5
        outflow = 1;

    else
        outflow = 0;
    end
end
if miti==0
    outflow = 1;
end
```

Block Diagram of the System



PROJECT DETAILS

Student Details			
Student Name	**Utkarsh Tripathi**		
Register Number	150907396	Section / Roll No	57
Email Address	tri.utkarsh7596@gmail.com	Phone No (M)	7607368279
Project Details			
Project Title	**Impact of Magnetic Fields due to High Tension Transmission Lines on Multirotor Vehicles**		
Project Duration	4 Months	Date of reporting	7th January 2019
Organization Details			
Organization Name	**Siemens Technology & Services Pvt. Ltd.**		
Full postal address with pin code	84, Hosur Road, Electronic City Phase 1, Bangalore 560100		
Website address	www.siemens.com		
Supervisor Details			
Supervisor Name	**Simit Pradhan**		
Designation	Technical Expert		
Full contact address with pin code	84, Hosur Road, Electronic City Phase 1, Bangalore 560100		
Email address	simit.pradhan@siemens.com	Phone No (M)	+91 96634716171
Internal Guide Details			
Faculty Name	**A Gopalakrishna Pai**		
Full contact address with pin code	Department of Electronics & Communication Engineering, Manipal Institute of Technology, Manipal 576104 Karnataka, India		
Email address	gopalkrishna.pai@manipal.edu		