People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

University M'Hamed BOUGARA – Boumerdès

Institute of Electrical and Electronic Engineering

Department of Electronics

Project Report Presented in Partial Fulfilment of

the Requirements of the Degree of

'MASTER'

In Control Engineering

Title:

Identification & Implementation of 3DOF Small Scale Helicopter Control system design

Presented By:

- **BENAZIZA Youcef**
- **MAOUCHE Taha Moncef**
- **OULDACHE Wassim Abdessamed**

Supervisor:

Dr. OUADI Abderahmane

ABSTRACT

Considerable attention has been attracted to the analysis and control of helicopters due to their potential military and civil applications as well as scientific significance. In order to build a testbed for implementing newly developed linear and nonlinear control techniques, a device of 3-DOF helicopter is developed in our laboratory (Signaux et systèmes) similar to that produced by Quanser Consulting Inc. for laboratory use. Such a platform emulates the longitudinal motion of actual helicopters and presents significant similarities, in terms of dynamics and underactuation properties, with six-DOF multicopters; hence, it constitutes a prime experimental testbed and is widely used for the design of nonlinear feedback controllers.

The 3-DOF laboratory helicopter consists of a base that carries a long arm capable of rotating about the elevation axis. One end of the arm is attached to a counterweight, while two bldc motors with propellers are installed at the other end to create forces that drive the propellers. Two motors' axes are parallel, and the thrust vector is normal to the frame. Three encoders are connected to the helicopter in order to measure the elevation, pitch, and travel angles of the body, and two voltage controlling electronic speed controllers are used to realize the control action in the system. The front-and back-motor voltages are the control input of the system. Since there are more degrees of freedom than actuators, this is an under-actuated system.

The 3DOF helicopter is modeled using Newton-Euler, and three equation of motions were considered. Different experimental setups were developed in our laboratory for identifying the unknown motions equations parameters. The first experiment established the bldc motor-propeller thrust force-voltage relation-ship parameters. Three other experiments are based on time-motion recorded measurement for each axis, while the others axis were fixed. Using these measurement data, the requested model parameters were generated using optimization technique with MATLAB MINUNC function.

A linearized 3DOF helicopter MIMO state space model, obtained from nonlinear model, around an equilibrium point is used to design an LQR and LQG (Linear Quadratic Guassian) controllers which are simulated using SIMULINK. A 3D image (3DOF helicopter) with the LQR controller is implemented using LABVIEW, allowing real-time simulation of a 3D view of travel, yaw and elevation variations.

The designed LQR controller algorithm is successfully implemented using LABVIEW to control the 3DOF platform in real-time in order to assess the controller performances and compare it with the simulated ones. The NI PCI6221 daq board is used to generate the control voltages of the front and back motors. While the three axis positions are sensed using the 1000 p/r encoders. All variable parameters, including the 3 axis helicopter variable and generated output are data-logged using developed recording functions within the developed software for complete analysis and performance evaluation

Keyword: 3DOF helicopter, Model identication, MIMO LQR design, Kalman filter, LQR real-time implementation.

ACKNOWLEDGEMENT

First of all, we would like to start this letter by expressing our deepest respect and gratitude to our dear mentor and supervisor Dr. OUADI, who took the place of the 4th man in this project, and who gave us the opportunity to work on a project that offered valuable lessons and challenges in this field of study.

For students who lack of practical experience, we had the chance to be surrounded by wise and patient teachers, Dr. MAACHE and Dr. TOUZOUT who dedicated some of their valuable time to help and suggest ideas to achieve the goals set in the beginning of the year and even to surpass them.

Working on a project that resumed all the hardwork of the past five years was not easy, luckily, we had our dear friends: Taib Amine, Benbouali Amine, Chiker Nesrine and Chouider Naziha who contributed to this project by showing their support, offering help on the technicalities of the project and for providing the necessary educational environment.

DEDICATION

Dedication In this special day, I want to dedicate my work to my family members. My father, my mother, my three brothers Mohamed, Abdelkrim Yassin and my sister, my uncles and all my cousins. Through all these years you were by my side showing your love and support toward me so in this blessed day I want to say you are my gift from god and my source of strength thank you for being in my life. I want also to thank the people whom I met during my study journey, my friends: Ouldache Wassim, Maouche Taha Moncef, Chiker Nesrine, Chouider Nezha, Taib Amine, Nouri Mohamed, Ayad Abdou, Douadi Walid, Aziz makesse, Khetab Amine, Anis Mahiou, Delmi Racim, Zitouni Farid. Thank you for you being always with me.

Youcef,

DEDICATION

Growing up, I always have been close to my family; close to my father who taught me the most in this life, to my mother who sacrificed a lot for me and my siblings and to my sisters who never hesitated to show me love. Today, this family is the reason behind my success, because yes, I consider myself successful for being rewarded for all the hardwork I put in these twenty-two years of existence.

It is in the nature of a human to make mistakes in order to learn, and I feel extremely lucky and grateful for all the tests that Allah has given me, the health issues that hit me in the last couple of months and the delicate situation that I have been through thirteen months ago.

Finally, I want to thank the few friends that I am blessed to have: Abdessamed, Adlane, Amine, Anes, Ines, Meriem, Raouf, Samy, Wassim and Youcef. These people showed me the right way in many occasions, brought up the best of me and insured my weall-being.

Taha Moncef,

DEDICATION

I dedicate this humble work to the people I cherish the most, a special feeling of gratitude to my loving parents. To my mother whose unconditional love and kindness lit and paved the paths for me. I should also not forget my father's wisdom and support whose words of encouragement and push for tenacity ring in my ears. I will always appreciate all what you have done for me. My brother Karim and my sisters Nour and Nawal have never left my side and are very special. Lucky to say I have siblings like yourselves. I dedicate this dissertation as well to my friends who have supported me throughout the process. This work couldn't have been accomplished without the help, guidance, and commitment of my teachers. Thank you for being there throughout the entire master's years.

Wassim Abdessamed,

CONTENT

List of Figures

List of Tables

ABREVIATION

3DOF: Three degrees of freedom.

LQR: Linear Quadratic Regulator.

LQI: Linear Quadratic Integrator.

LQG: Linear Quadratic Gaussian.

ODE45: Order Differential Equation using Runge-Kutta (4,5).

DAQ: Data Acquisition board.

Fminunc: Unconstrained Function minimum optimization.

Fmincon: Constrained Function minimum optimization.

FBD: Free Body Diagram.

RPM: Rounds Per Minute (unit of measurement for rotational speed).

MIMO: Multi input Multi output.

SISO: Single input Single output.

PID: Proportional Intgrator Derivative.

PWM: Pulse Width Modulation.

MPC: Model Predictive Control.

MSE: Mean Square Error

General Introduction

Helicopters and other aerial vehicles have been of significant research interest over the past few decades due to their multi-disciplinary applications. Their commercial demand has expanded globally because of their flexible nature to reach remote destinations [1].

Throughout history, humans have created numerous mechanisms, prototypes, and models in various fields of science and technology. These developments attempted to emulate reality and made it easier to study and understand diverse phenomena under controlled conditions. Laboratory prototypes have permitted to improve the quality of systems, reduce costs, increase aircraft safety, and even accelerate system development. Similarly, these prototypes have facilitated the development of approaches for creating models and control systems in real-time applications that govern the behavior of aerial vehicles.

The 3 degree of freedom (3-DOF) helicopter model is a complex mechanical and typical multi-input and multi-output system which contains the properties of strong multivariable coupling, strongly nonlinear characteristics and has open-loop unstable dynamics which make the control of such system a challenging task. The helicopter represents an underactuated mechanical system since only two control inputs are available for controlling three degrees of freedom. Many efforts have demonstrated that it is difficult to illustrate the dynamic characteristic of the helicopter, because of its extremely complex and particular flying state. Normally, its dynamic characteristics will be correspondingly varied with flying altitude and flying state, all of which are nonlinear and multivariable coupling. Consequently, it is probably impossible to achieve formulations of the helicopter as well as its accurate models [2].

The purpose of this project is to regulate desired pitch and travel positions using a linear quadratic regulator. Initially the nonlinear mathematical model of the helicopter is derived. The 3DOF helicopter system control involves linearization of the nonlinear dynamics about a given operating (equilibrium) point within the flight envelope. Then the LQR design methodology is discussed to deal with both performance and stability of the system. The design problem is then dealt with finding an LQR controller gain matrix, which gives a control solution. Finally, an approximation method is suggested for finding the design parameters for PID controller from the obtained LQR controller gain matrix.

CHAPTER 1
SYSTEM DESCRIPTION

1.1. Introduction

This introductory chapter is devoted to the review of the tandem helicopter concept as well as the system description and theoretical background behind the 3DOF helicopter.

1.2. System description and background

Most of the control system engineering studies have been restricted nowadays to simulations and mathematical demonstrations with occasional implementation of the real plant. In this context, laboratories are required more often to validate the suggested theories, data or mathematical models with a fully tested real-time implemented model. However, the implementation of some projects might consume huge financial resources; aircrafts are the perfect example due to their big size and expensive components. An optimal solution consists of replacing the main aircraft model desired to be studied with a practical prototype that uses the same aerodynamic principles in order to function properly. This project will handle the three degrees of freedom helicopter system which is the most accurate representation of the Chinook CH-47 helicopter.

The 3DOF helicopter is a model produced by Quanser to often simulate nonlinearities and design of a simpler representation of complex mechanical dynamics [3]. As depicted in Fig. 1, the helicopter basically consists of three hinge-mounted rigid body systems. The helicopter base, which can turn about the travel angle carrying the arm that can rotate about the pitch angle. A counterweight is attached to the second edge of the arm reducing the power requirements on the motors. The vertical rotation of the helicopter's main body around its center of mass is described by the roll angle [4]. The three state angles (pitch, roll and travel) are constantly measured using three rotary optical encoders.
. Two motors are attached to the two ends of the helicopter's body. A thrust force proportional to the voltage supplied to the motors is then generated using the propellers attached to them.

A computer equipped with a national instruments pci-6221 data acquisition board is used to control the helicopter and to enable the measurements from sensors and the voltages command to the electronic speed controllers (ESC) that drive the motors.

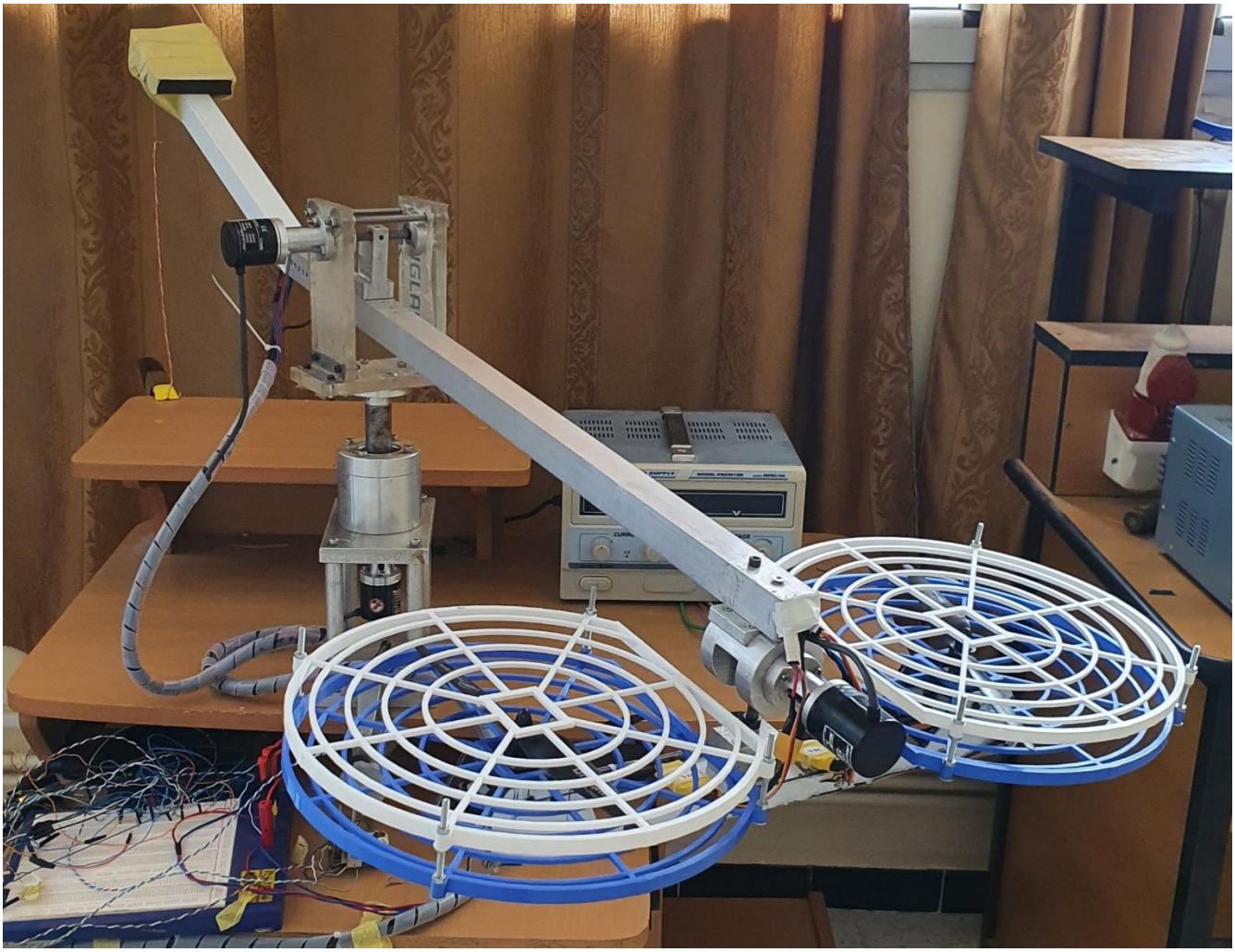

Figure 1.1 3DOF Helicopter

1.3. Tandem rotor concept

Unlike the majority of the helicopter models, Tandem rotor helicopter has two main rotor system mounted one in front of the other with no tail rotor attached to the back of the aircraft. Usually, the rear rotor is mounted at a higher position than the front rotor, the two are linked by a propulsion transmission that ensures the rotors are synchronized and do not hit each other, even during an engine failure. The first successful tandem rotor helicopter was built by Nicolas Florine in 1927. Tandem rotor helicopters use counter-rotating rotors, with each cancelling out the other's torque to neutralize the yawing movement Therefore, all of the power from the engines can be used for lift [5].

Figure 1.2 Tandem rotor CH-47 Chinook helicopter

Tandem rotor designs achieve yaw motion by applying opposite left and right cyclic to each rotor, effectively pulling both ends of the helicopter in opposite directions. To achieve pitch, opposite collective is applied to each rotor; decreasing the lift produced at one end, while increasing the lift at the opposite end, effectively tilting the helicopter forward or backward [6].

This configuration, which is mainly used for larger helicopters, has the advantage of being able to hold more weight with shorter blades, larger center of gravity range and good longitudinal stability. However, the rear rotor works in the aerodynamic shadow of the front rotor, which reduces its efficiency. This loss can be minimized by increasing the distance between the two rotor hubs, and by elevating one hub over the other [7].

For the 3DOF prototype laboratory introduced in this report, the rotor's blades available are fixed. Therefore, to achieve pitch, these two motors must apply two combined forces. To travel with a certain distance (degree), the helicopter must roll in either direction; this effect is produced when a difference between the thrusts of the motors is applied.

CHAPTER 2
SYSTEM MODELLING AND IDENTIFICATION

2.1. Introduction

As explained earlier, the 3DOF QUANSER is a physical model which behaves according to three main axes. Before proceeding to the control part, one must have a logical understanding on how the system is interacting with its physical parameters.

Throughout this chapter, a mathematical model of both the actuator (BLDC motors) and the plant (flight system) will be constructed using essential physics' principles such as: Newton's, Euler's equations and aerodynamic motion equations.

Once the mathematical expressions deduced match with the real system with a high degree of accuracy, different methods will be used on the same components to collect multiple data sets in order to decrease the risk of disturbance and noise presence during the experiments.

Finally, the very well-known software MATLAB is used to estimate the physical parameters using different toolboxes of the software.

2.2. System Modelling

A QUANSER is a 3 degrees of freedom aircraft (helicopter) attached to a support.

The motion of a 3DOF helicopter consists of moving around three main axes of rotations:

- Pitch: level of elevation of the helicopter from the ground which rotates around the support that holds the helicopter vertically which will be referred to as "p".
- Roll: the motion "R" of rotation of the helicopter around its center of mass when the body is unbalanced due to a torque difference.
- Travel: the rotational angle "T"crossed horizontally by the helicopter around the support holding the system.

2.2.1. Modelling of the Pitch-axis

The combined thrust forces of the two motors will lift the helicopter's body upward, going against the gravitational momentum of the total mass (counter weight + helicopter's mass + mass of the support).

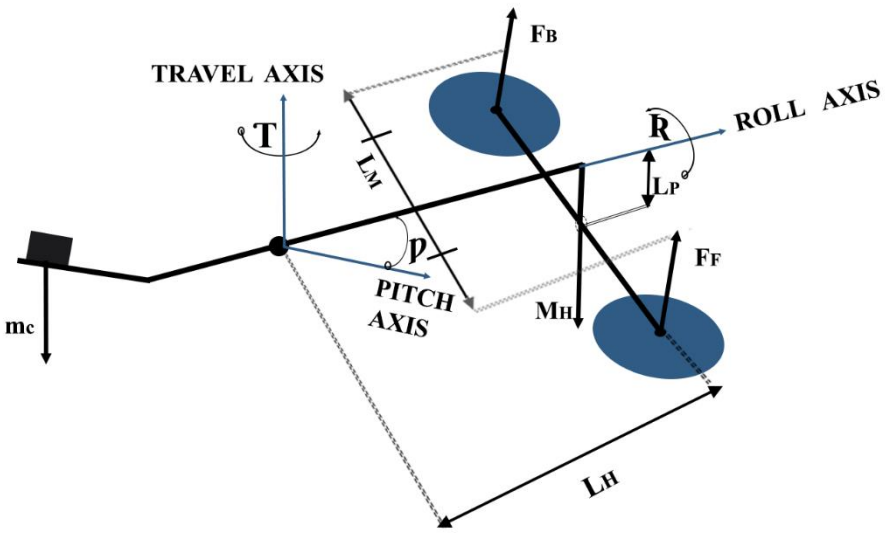

Figure 2.1: FBD of the pitch axis

- M_c: Mass of the counter weight.
- L_H: The distance between the center of pitch and the front end of the helicopter.
- M_H: The mass of the counter weight in the lower end of the rig.
- L_M: The distance between the center of the propellers and the middle of the rig that carries the motors.

The rotational motion of the helicopter can be represented using Euler's second law:

$$I_p.\ddot{p} = F_B.L_H.\cos R + F_F.L_H.\cos R - M_G(p) - \mu_p.\dot{p} \quad (2.1)$$

Setting $F_{sum} = F_B + F_F$

$$I_p.\ddot{p} = F_{sum}.L_H.\cos R - M_G(p) - \mu_p.\dot{p} \quad (2.2)$$

Where

I_p: The inertia about the pitch angle.

$M_G(p)$: Gravitational Momentum.

μ_p: The friction coefficient.

F_B: Backward Motor Force.

F_F: Forward Motor Force.

2.2.2. Modelling of the Roll-axis

A difference between the motors' forces will result into a vertical rotational motion around the helicopter's center of mass:

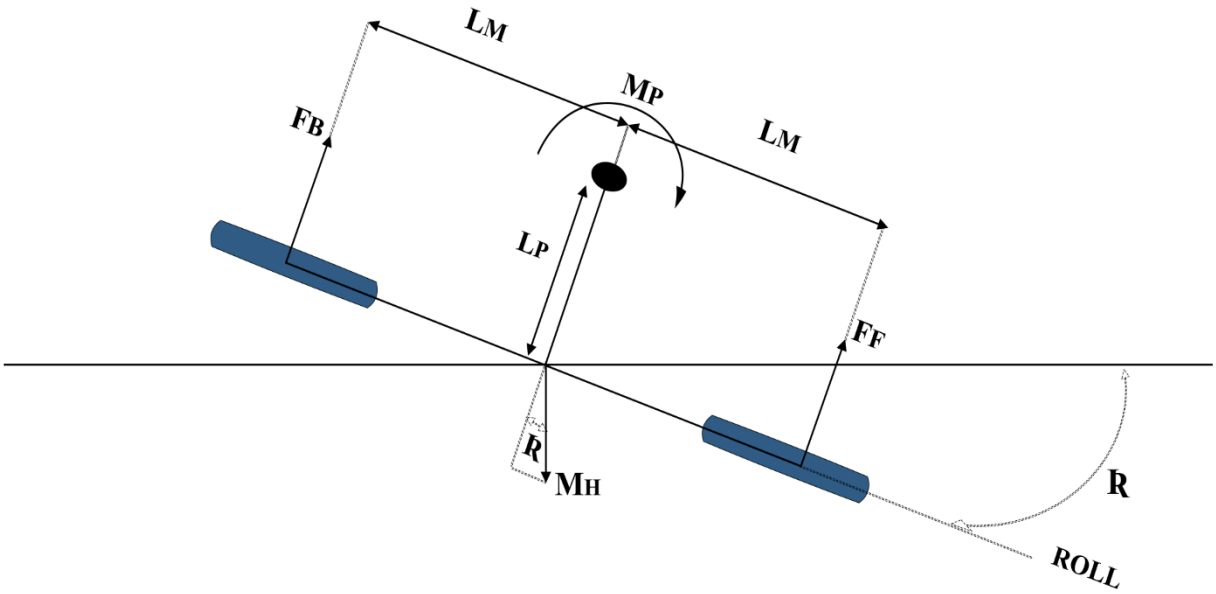

Figure 2.2: FBD of the Roll system

- M_H: Mass of the aircraft (helicopter's body + motors and propellers).
- Lp: The distance between the center of rotation of the pitch angle and the rig that carries the motors.

This rotation can also be represented using Euler's second law:

$$I_{\text{R}}.\ddot{\text{R}} = F_B.L_M - F_F.L_M - \mu_{\text{R}}.\dot{\text{R}} - M_H.g.L_P \sin \text{R} \cos \text{P} \quad (2.3)$$

Setting $F_{diff} = F_B - F_F$:

$$I_{\text{R}}.\ddot{\text{R}} = F_{diff}.L_M - \mu_{\text{R}}.\dot{\text{R}} - M_H.g.L_P \sin \text{R} \cos \text{P} \quad (2.4)$$

Where

I_R: The inertia about the Roll angle.

μ_R: The friction coefficient

2.2.3. Modelling of the Travel-axis

The travel axis is directly linked to the roll axis; A difference between the forces will carry the helicopter to move in a rotational shape horizontally around the support holding the system in either direction (forward or backward) depending on the sign of the force difference F_{diff}:

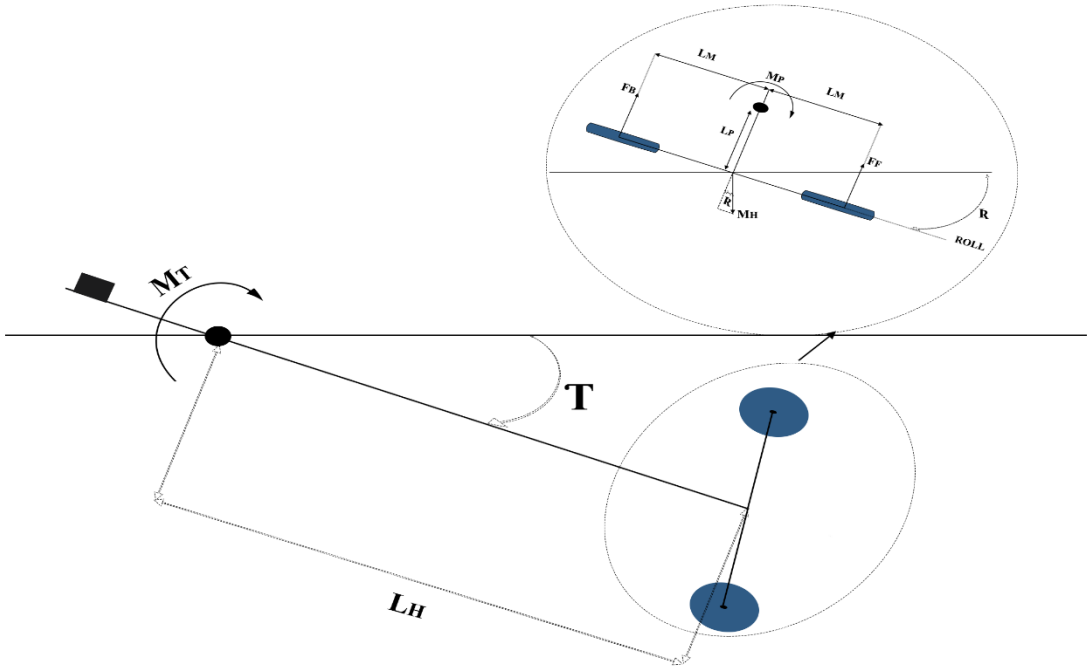

Figure 2.3: FBD of the Travel system

The equation of motion is given as:

$$I_T.\ddot{T} = F_B.L_H.\cos p \sin R + F_F.L_H.\cos p \sin R - \mu_T.\dot{T} \quad (2.5)$$

Summing up the forces as what have been done in equation (2.11):

$$I_T.\ddot{T} = F_{sum}.L_H.\cos p \sin R - \mu_T.\dot{T} \quad (2.6)$$

- The total dynamics of the system can be described by the following differential equations:

$$\begin{cases} I_{\mathrm{p}}.\ddot{\mathrm{p}} = F_B.L_H.\cos\mathrm{R} + F_F.L_H.\cos\mathrm{R} - M_G(\mathrm{p}) - \mu_{\mathrm{p}}.\dot{\mathrm{p}} \\ I_{\mathrm{R}}.\ddot{\mathrm{R}} = F_{diff}.L_M - \mu_{\mathrm{R}}.\dot{\mathrm{R}} - M_H.g.L_P\sin\mathrm{R}\cos\mathrm{p} \\ I_{\mathrm{T}}.\ddot{\mathrm{T}} = F_{sum}.L_H.\cos\mathrm{p}\sin\mathrm{R} - \mu_{\mathrm{T}}.\dot{\mathrm{T}} \end{cases} \quad (2.7)$$

2.3. BLDC Motor identification

Brushless DC motors are very high performing type of motors that use electronic principles to operate. A magnetic field will be created between the 3 poles of the BLDC motor when a PWM signal is fed.

Attaching a propeller on the top of the motor will generate a thrust effect when the motor's shaft starts spinning. However, the thrust equations of the propellers can't be expressed by a simple linear term, hence, indirect approaches will be followed instead to deduce the force of the motors which consist of using mass sensors.

2.3.1. Mass sensors

To identify the expression of the motor's thrust, two different sensors will be used to validate experimental data, one consists of a digital balance that converts mass into analog readings, while the other sensor consists of a load-cell which translates the mass readings into output voltages.

❖ Load-cell Parameter Identification

2.3.1.1. Load-cell To Instrumental Amplifier

The output range of the load-cell sensor is very limited (millivolts). This type of signals needs a very basic signal processing to expand the range of readings of the sensor by using an amplifier that converts the readings of a given sensor into larger intervals.

An instrumentation amplifier INA110 entry is connected to the load-cell with a gain of G=100:

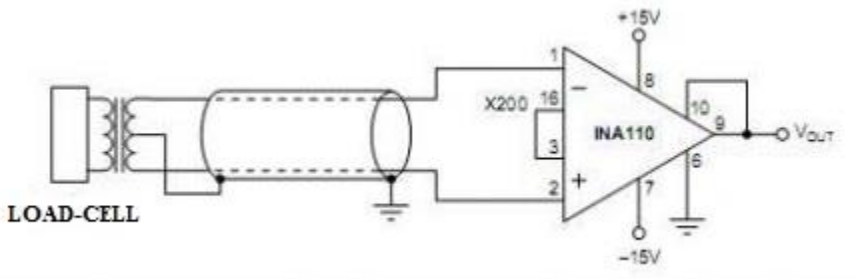

Figure 2.4: Load-cell To Instrumental Amplifier

If it is desired to interface the load-cell with an acquisition board instead of using a simple voltmeter (step or dynamic response), then the HX711's amplifier is a better linkage option:

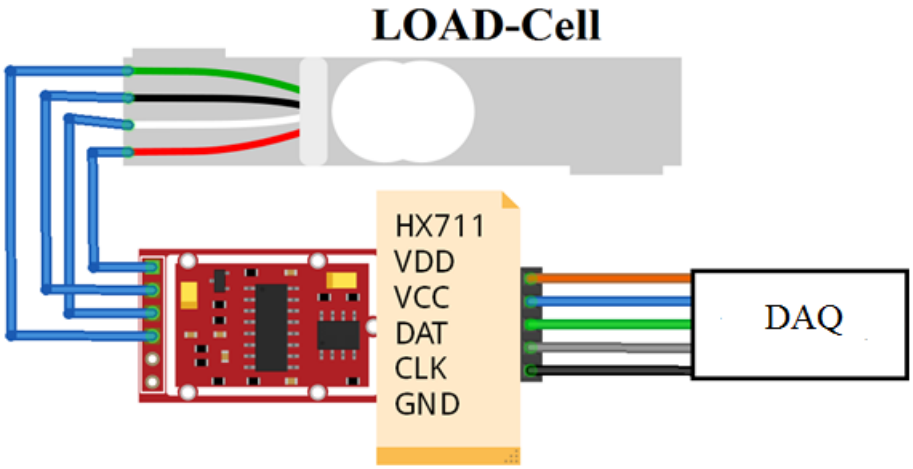

Figure 2.5: Load-cell To HX711

2.3.1.2. Load-cell: Mass versus Voltage

The load-cell is a 1st order linear transducer that converts masses/weights (gravitational force) into output voltages.

To collect the data, objects with known masses are fixed on the sensor's surface.

To determine coefficients of any given equation, MATLAB Curve fitting toolbox is used whenever real time data and its mathematical representation are available.

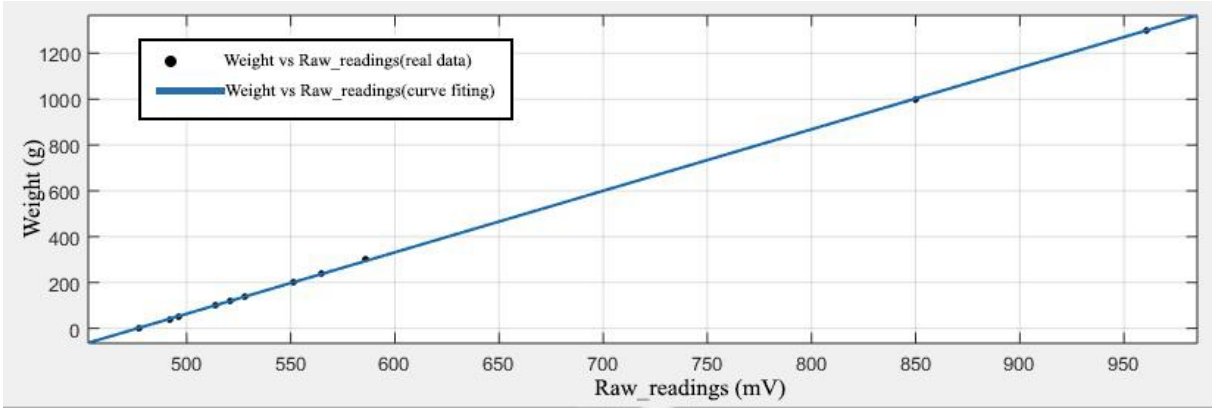

Figure 2.6: Weight(g) vs raw readings of voltmeter(mV)

$$mass(grams) = 2.682 * V_{out}(mV) - 1278 \quad (2.8)$$

2.3.2. Motor Rotational Speed (RPM) versus Input Voltage

Once again, the MATLAB Curve fitting toolbox is used to determine the relationship between the variable dc analog control voltage and the propeller speed. The controlled voltage is applied to the Arduino analog input and generates an appropriate PWM signal to control, through ESC, the propellers rotational speed measured using the tachometer:

Vin (volts)	0,5	0,75	1,25	1,5	1,6	1,8	2	2,2	2,5	2,7	2,8	3	3,25
Speed(RPM)	1362	1688	2225	2454	2544	2711	2900	3092	3318	3554	3684	3828	4080

Table1: Motor Rotational Speed (RPM) versus Input Voltage

The relation between the two variables can still be improved by approximating the data to a 2nd order polynomial, however, the R-square error is only improved from 0.9972 to 0.9976 (0.04% error) which is insignificant due first to measurement error, and also the fact that more nonlinearities errors will be introduced in the system. To simplify further computations, the equation (2.9) is maintained:

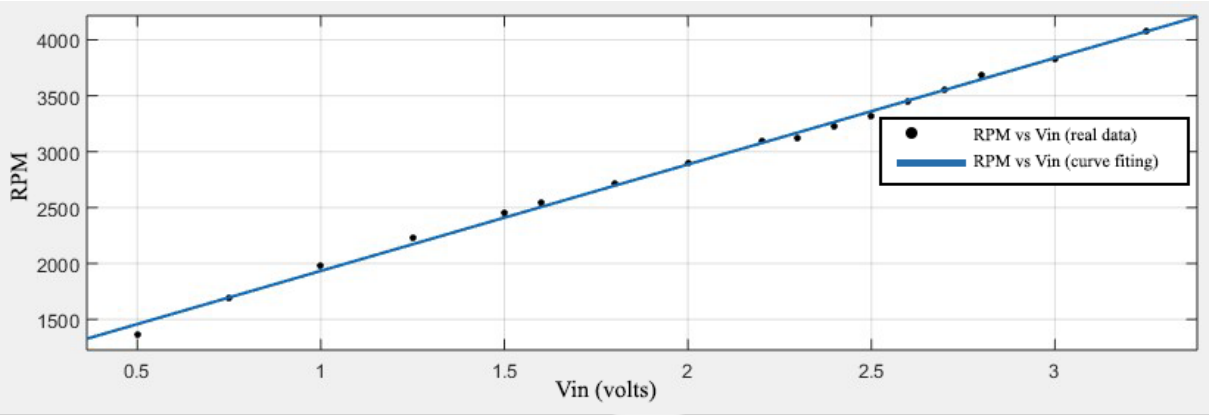

Figure 2.7: Motor Rotational Speed (RPM) versus Input Voltage

$$n(RPM) = 951.9 * v_{in} + 981.8 \quad (2.9)$$

2.3.3. Thrust versus Vin and RPM

2.3.3.1. Background Computation

The Newton's 2nd law $F = m.a$, can be expressed in terms of mass rate and velocity change:

$$F = \frac{dm}{dt}.\Delta v = \frac{dm}{dt}.(V_e - V_{ac}) \quad (2.10)$$

V_e: Static velocity.

V_{ac}: Aircraft velocity.

Equation (2.10) represents the air molecules moving around the propeller's surface with a certain mass rate.

Representing the same equation in terms of the propeller's cross section, [8]:

$$F = p.A.V_e^2 - p.A.V_e.V_{ac} \quad (2.11)$$

The equation (2.11) will lead ultimately to the following equation [8]

$$F = \frac{\rho.\pi.(0.0254.d)^2}{4}.\left(n.\frac{0.0254.p}{60}\right)\left(\left(n.\frac{0.0254.p}{60}\right) - V_0\right).(k_1.{}^{d}/_{p})^{k_2} \quad (2.12)$$

Where:

$\rho = 1.225\,\frac{Kg}{m^3}$: Air density constant at sea level.

p=12.5 cm: Propeller's pitch, this constant represents the distance crossed by the propeller's wing per one revolution.

d=28 cm: Propeller's diameter.

k_1 and k_2 are constants that vary with the pitch and diameter of propeller and its number of wings.

V_0: Wind speed.

The experience will be performed in a closed room where the wind speed is as close as possible to zero. Hence, the propeller's thrust equation is given by:

$$F = \frac{p.\pi.(0.0254.d)^2}{4}.\left(n.\frac{0.0254.p}{60}\right)^2.(k_1.{}^{d}/_{p})^{k_2} \quad (2.13)$$

2.3.3.2. Measurement Experiment

On one end of a stick, a BLDC motor with a propeller attached to it is fixed while the second end of the stick is fixed on the sensor's surface as shown on the following diagram:

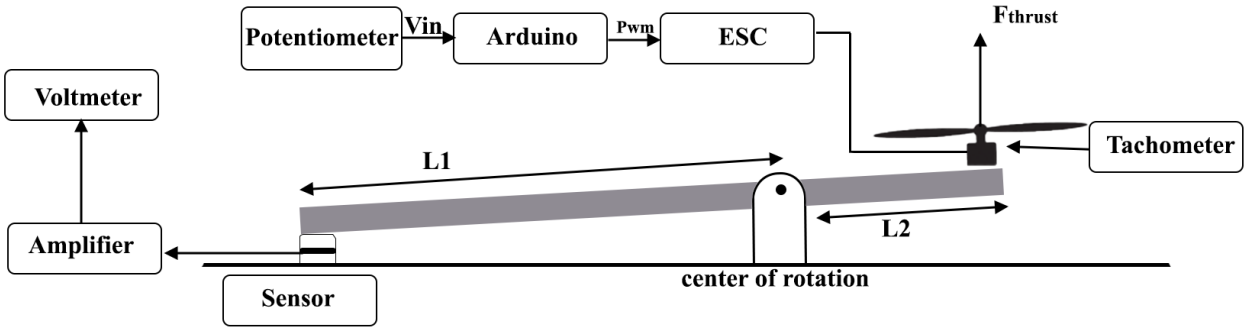

Figure 2.8: diagram for the identification of thrust

The motor's thrust can be deduced using torque's principle:

$$T_{thrust} = T_{sensor}$$

$$m.g.L1 = F * L2$$

$$F = \frac{m.g.L1}{L2} \qquad (2.14)$$

m: The reading of the sensor (kg).

g=9.81m²/s.

L1=50.5 cm.

L2=17.5 cm.

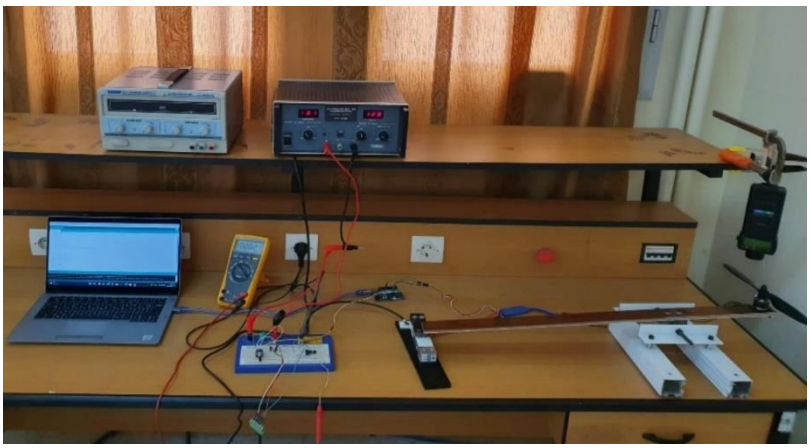

Figure 2.9: lab experiment for the identification of thrust force

As explained earlier, the thrust relation will be extracted using two different data sets, then, the final result will be computed using the average data of the two sensors combined:

weight converted from voltage (g)	Thrust using load cell (N)	Vin (V)
8,046	0,22754088	0,5
13,41	0,3792348	0,75
18,774	0,53092872	1
21,456	0,60677568	1,25
29,502	0,83431656	1,5
32,184	0,91016352	1,6
34,866	0,98601048	1,7
37,548	1,06185744	1,8
40,23	1,1377044	1,9
45,594	1,28939832	2
48,276	1,36524528	2,1
50,958	1,44109224	2,2
53,64	1,5169392	2,3
56,322	1,59278616	2,4
59,004	1,66863312	2,5
64,368	1,82032704	2,6
69,732	1,97202096	2,7
72,414	2,04786792	2,8
77,778	2,19956184	2,9
83,142	2,35125576	3
96,552	2,73049056	3,25

Table 2 The thrust and weight converted from voltage of the motor

The equation (2.13) Will be tested in the curve fitting toolbox to obtain the constants k_1,k_2:

$$k_1 = 11.87$$

$$k_2 = 4.946$$

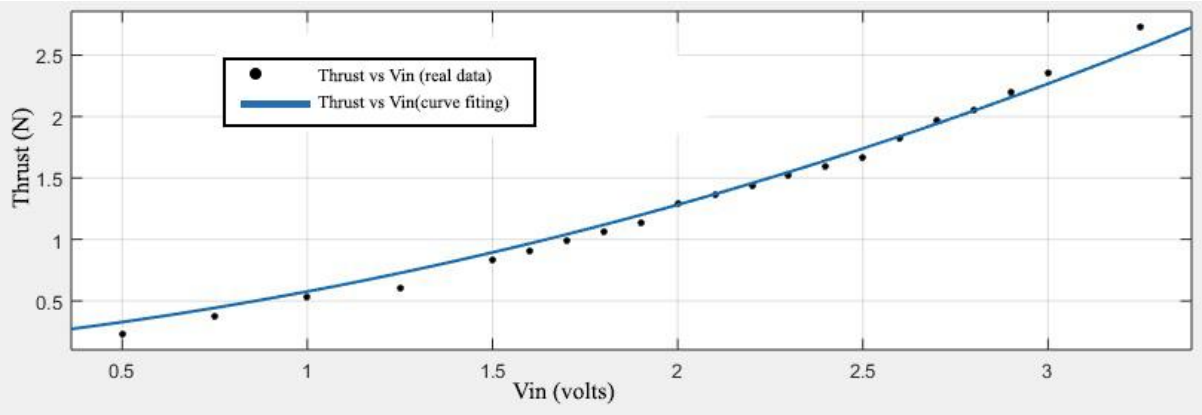

Figure 2.10: curve fitting thrust vs vin

Finally, the thrust expression vs rotational speed (RPM) can be represented as follows:

$$F = 1.54 \times 10^{-7}.n^2 \qquad (2.15)$$

2.4. Parameter Estimation

2.4.1. Pitch-axis Parameter Estimation & Optimization

The parameters of the pitch axis can be classified into two main categories: dynamic and static parameters described in eq (2.2).

- The static parameters are constants that grant the system its characteristic properties such as: L_H, μ_p and I_p.
- On the other hand, the gravitational momentum $M_G(p)$ is a representation of the gravitational force during an angular motion, meaning that $M_G(p)$ varies with respect to the elevation angle p.

2.4.1.1. Gravitational Momentum Identification

When the system enters its steady state at a given input force supplied by the BLDC motors, the helicopter will cease elevating meaning that both angular velocity and acceleration are converging to zero (d/dt terms are 0).

In this case, the momentum can be deduced using equation (2.16)

$$F_{sum}.L_H.\cos R = M_G(p) \qquad (2.16)$$

Considering that the thrusts F_B, F_F are equally distributed:

$$\frac{F_{sum}}{2} = F_B = F_F \qquad (2.17)$$

This relationship translates into a zero degree of roll $\mathrm{R} = 0$ when the forces are equal to each other and the system is balanced around the roll-axis resulting into the direct relation between the gravitational moment and the elevation angle at steady state. Then eq. (2.16) becomes:

$$F_{sum}.L_H = M_G(\mathrm{p}) \qquad (2.18)$$

Collecting pitch data at given voltage/thrust values will allow us to build up a relationship between the momentum and pitch angle:

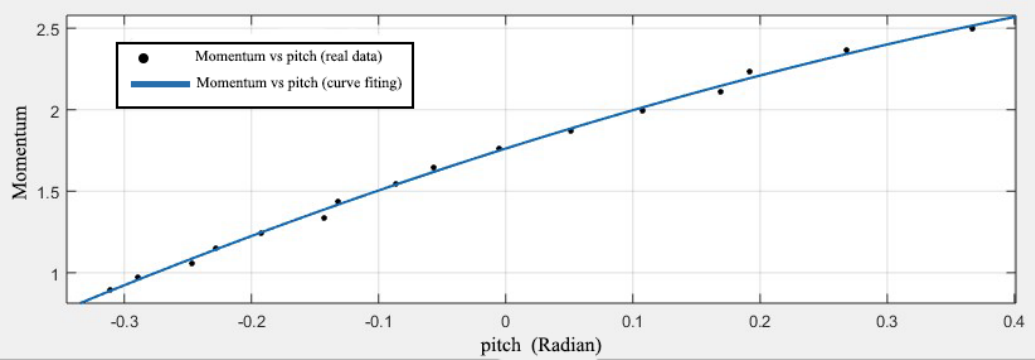

Figure 2.11: curve fitting the momentum vs pitch angle

The curve fitted graph can be represented by the 2nd order polynomial:

$$M_G(\mathrm{p}) = -1.111.\mathrm{p}^2 + 2.462.\mathrm{p} + 1.762 \qquad (2.19)$$

Notes:

The plant is exposed to many disturbance variables such as wind and defective electronic devices (ESC) which can make it difficult for the system to enter steady state. Computing the average value for pitch angles can deflect this problem.

The angles are used in terms of Radian, which will also be the case for the entirety of the project.

2.4.1.2. Physical Parameters Estimation

The most efficient way to estimate a parameter, is by applying a step response to the system. Since the characteristic differential equation is clearly a nonlinear one, it is preferable to use two different approaches to reach accurate results:

a. Using Simulink Toolbox

When dealing with nonlinearities, Simulink is a very powerful tool since it can simulate high orders of nonlinearities.

Characteristic equations can always be built using block diagrams in Simulink:

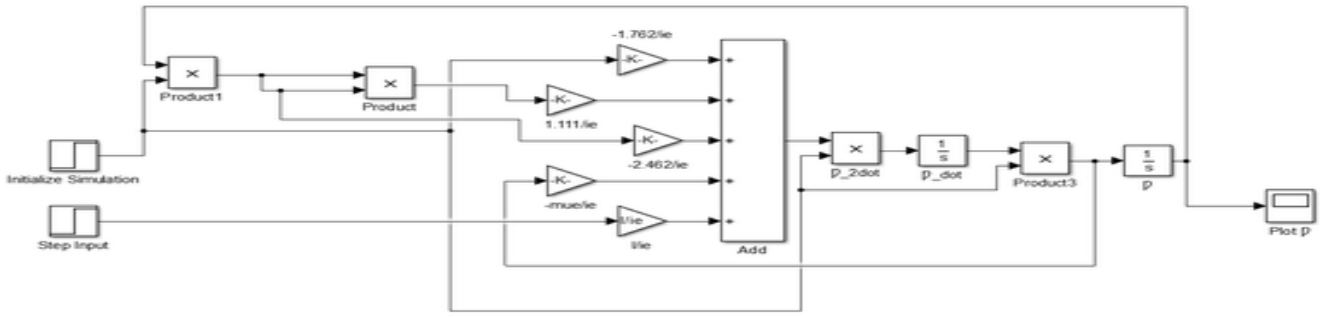

Figure 2.12: Characteristic equations of pitch system in SIMULINK

Keep in mind that both data (simulated and experimental) should have the same initial conditions before proceeding to the estimation procedure:

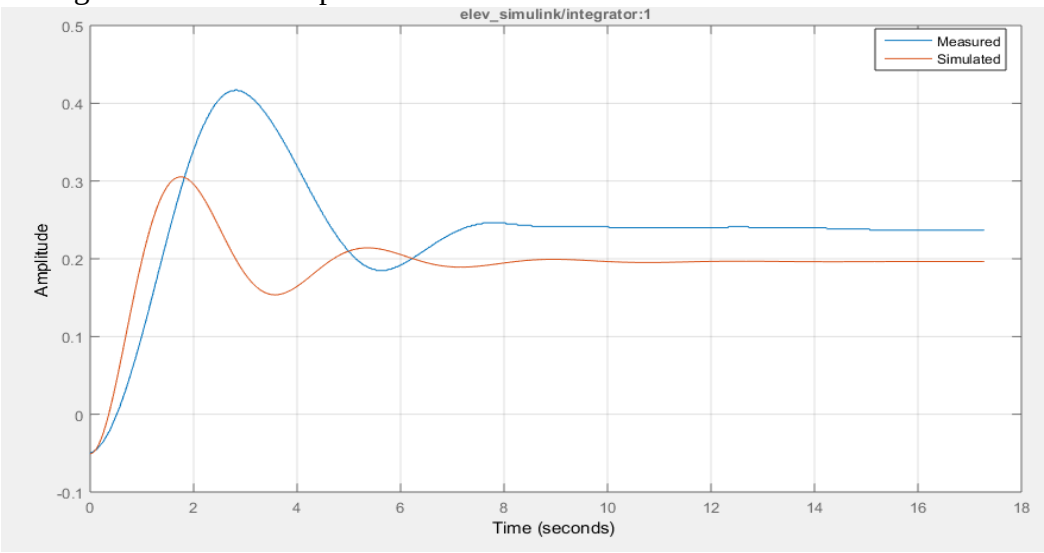

Figure 2.13: Simulated vs measured pitch(Radian) step response with initial guess

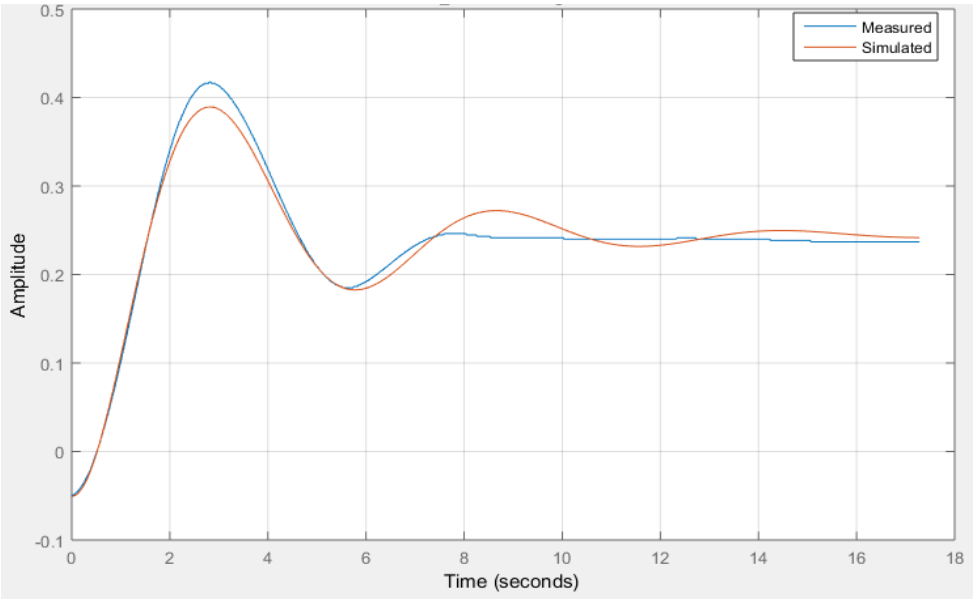

Figure 2.14: Simulated vs measured pitch (Radian) step response after parameter estimation

b. Using Optimization Toolbox:

The optimization toolbox is a more efficient toolbox than Simulink's parameter estimation toolbox, even though it takes more significant time to return the desired results.

The optimization toolbox requires an objective function and a nonlinear state space of the model desired to optimize formed using MATLAB's ode45 solver [9].

Objective function: the objective function is the term that is aimed to be minimized. In this section, the least square error was chosen to be minimized:

$$LSE = \sum_{i=1}^{n}(X_i - \widehat{X}_i)^2 \qquad (2.20)$$

n: Sample size of the experimental data.

X_i: Sample data.

$\widehat{X}_i$: Estimated data.

The nonlinear minimization constraints algorithm "fminunc" [10] searches for minimal least square error for the necessary optimal parameters.

In order to have parameters that fit for the widest possible range of input, two separate samples of data with two different inputs are used for the estimation process: the first experience was done for a step of an input voltage from 1.9 volts to 2.5 volts.

The second input response was given for a step of 1.3 volts to 1.9 volts.

Combining all the data obtained from the previous simulations, the final result of parameters that is going to be used for the rest of the project can finally be estimated:

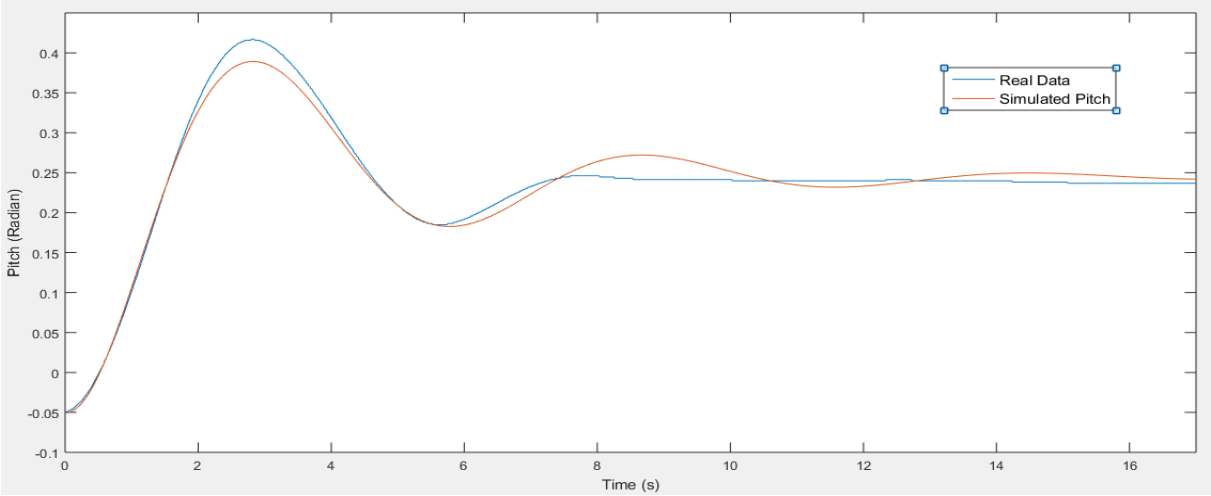

Figure 2.15: Simulated vs measured time response

Optimal parameters:

$I_p = 1.531\ Kg.m^3$

$\mu_p = 0.8531\ {}^{N.s.m}/_{Rad}$

$L_H = 0.6569\ m$

2.4.2. Roll-axis Parameter Estimation & Optimization:

The step response applied on the system for the identification of the roll-axis can be performed using two methods:

- The first experiment is done by attaching a mass to one of the ends of the helicopter (attach the mass under one of the BLDC motors), the data is then recorded the moment the mass is released.
- The second step response is applied by fixing the pitch-axis at a given angle (preferably at 0°), a force difference is applied by assigning different voltage values to the motors, and this will force the plant to rotate with a fixed roll angle until it reaches steady state.

The voltage difference is dropped to zero the moment the helicopter enters the steady state and data is then instantly recorded.

As expected, the system oscillates numerous times before going back to zero angle roll.

- Keep in mind that a step-down response was also performed for this experiment, this is simply done by raising the voltage difference at time t_0 and computing its corresponding thrust.

A Simulink model was also constructed to plot accurate response for the roll-axis:

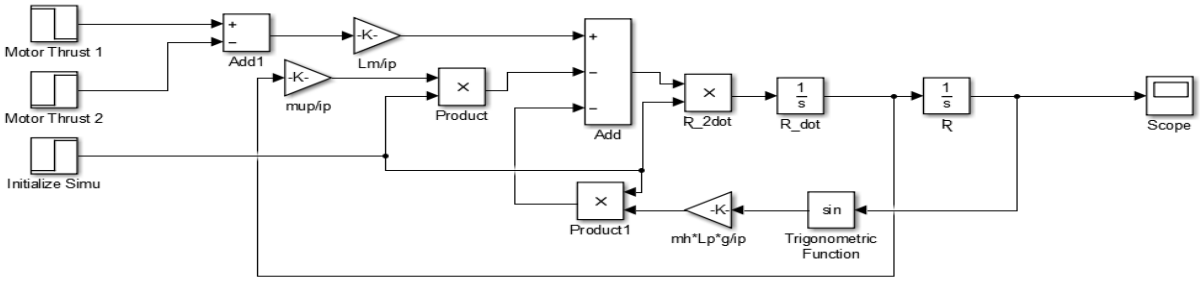

Figure 2.16: Roll's characteristic differential equation respresentation in SIMULINK

Combining the two toolboxes used in the previous section with the data set of 3 experiments;

- Step response by releasing a mass of 200 g equivalent to 1.962 N.
- Step-up by applying a voltage difference of 0.8 volts.
- Step-down by releasing force difference due to the 0.8 volts.

The optimal parameters are obtained for the following roll step response comparison between real versus simulated data:

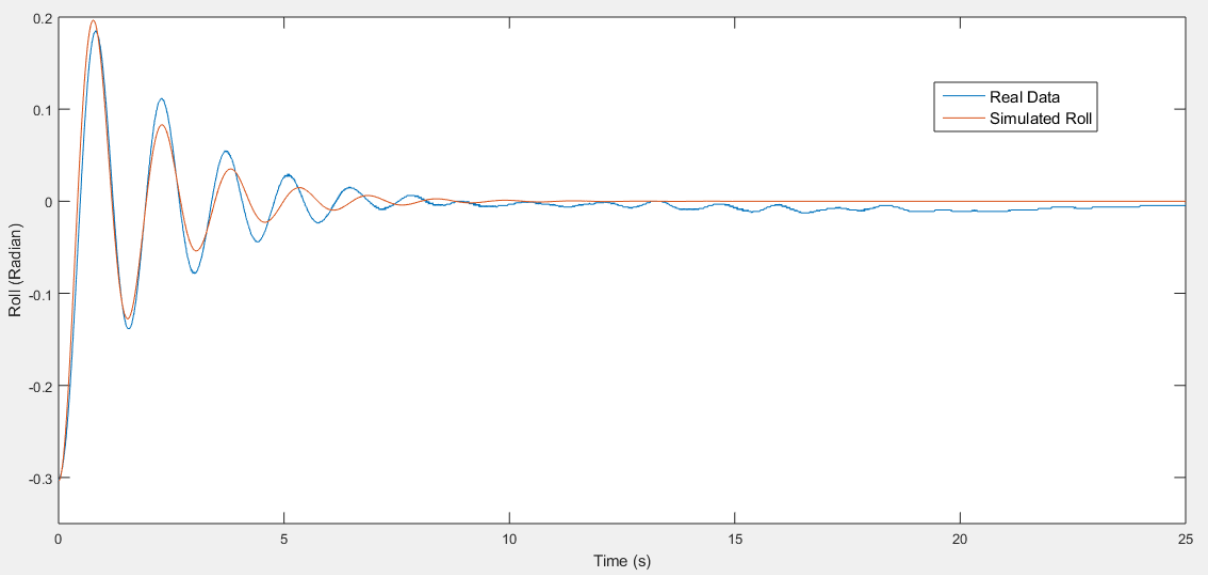

Figure 2.17: Real data vs simulated time response for the Roll system for optimal parameters

Optimal parameters:

$L_M = 0.175\ m$

$I_{\mathrm{R}} = 0.0452\ Kg.m^3$

$\mu_{\mathrm{R}} = 0.0511\ {}^{N.s.m}/_{Rad}$

$M_H = 3.2059\ Kg$

$L_P = 0.025\ m$

2.4.3. Travel-axis Parameter Estimation & Optimization:

The travel axis depends directly on the roll angle and the force difference between the motors.

To estimate the travel parameters, an input force difference is applied while the helicopter is held at a fixed position until the roll axis enters steady state, the body is then released and data is recorded for fixed pitch and roll angles to simplify the computations.

The estimation in this section is executed by repeating the exact same steps followed in previous sections:

- First ,a Simulink block for the travel axis is built:

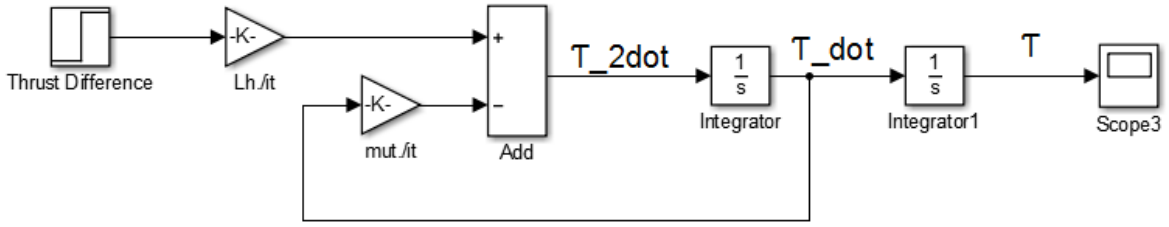

Figure 2.18: Characteristic equations of Travel system in SIMULINK

- The results of the Simulink estimation toolbox and MATLAB optimization toolbox are combined to deduce the following parameters:

$I_T = 1.918\ Kg.m^3$

$\mu_T = 0.446\ {}^{N.s.m}/_{Rad}$

For the following step response:

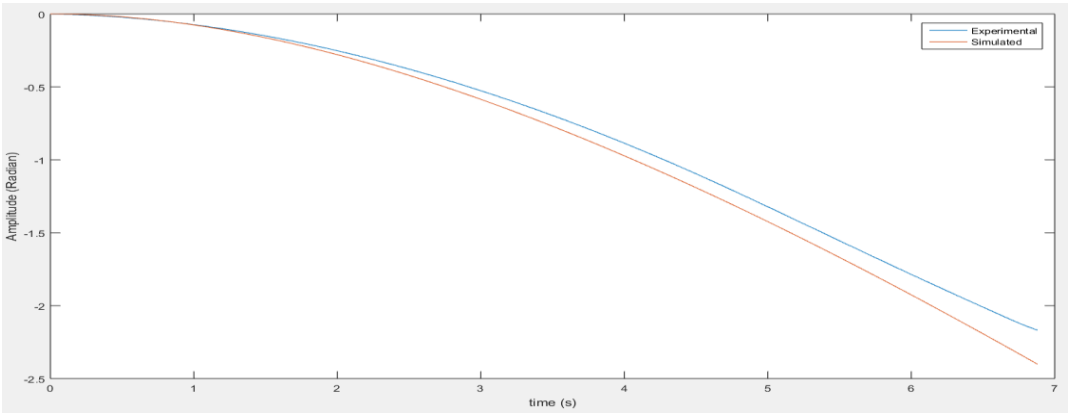

Figure 2.19: Real data vs simulated time response for the Travel system for optimal parmeters

It is important to mention that after fixing the angles of roll and pitch, the travel axis can be considered as a linear system. Therefore, the adjusted system can be identified using classical identification methods such as the MATLAB's ident toolbox or numerical methods such as the Mean Square Error.

2.5. Conclusion:

The identification of any given physical system depends mainly on the comprehension of its behavior and interaction with its components.

The process of identification went first through the analysis of the relationships responsible of controlling the input forces acting on the motors, where the main purpose is to link the controllable variable like the voltage to these forces whom the physical properties can't be determined using direct approaches.

The construction of an equivalent mathematical model was first done by expressing the three degrees of freedom into three main differential equations.

Even though the mathematical modelling was achieved by following essential physics principles, the estimation of the parameters defining this model was complicated due to its exposure to numerous nonlinearities which gave the opportunity to explore different optimization, estimation and identification techniques and toolboxes.

The results obtained through this chapter will be used for the rest of the project in the approximation to the real system, the construction of the controller and the implementation of the final design in the upcoming chapters.

CHAPTER 3

CONTROLLER DESIGN

3.1. Introduction

In the previous chapter, a mathematical model of the helicopter system was derived with all the necessary system parameters. But, every model contains errors that can be reduced using compensators and control theory But, before proceeding to the design of the controller, the nonlinear system must first be represented in its state space form in order to be linearized using the indirect Jacobian approach around the equilibrium point.

Since the system is underactuated which means that there are three degrees of freedom but only two inputs, it is hard to design one controller capable of keeping all the angels at the reference angle given by the user. A good solution is to decouple the system into two state spaces, and then, proceeding to the design of the two controllers; one for the pitch system and the second one for the travel, the roll will be automatically included in the control system of the travel axis.

For the controller design, there are several methods to choose from. If the real model (nonlinear) is properly approximated to a linear one, then the best method to control a linear system is by controlling the placement of the state poles using feedback control loop. For this reason, a linear quadratic controller fits perfectly in the implementation of this project since it is designed specifically for this type of requirements.

A full state feedback is required to develop the LQR controller. The only sensors used in the helicopter are the rotary encoders, which simply provide the acquisition system with the helicopter's angular coordinates. Because the system involves disturbances and some measurement noises, a state estimator will be developed to acquire the remaining states (angular velocities). The Kalman filter is an excellent choice for this project. After building the LQR controller and Kalman filter, the combination of the two will result into what it can be known as Linear Quadratic Gaussian.

3.2. State Space Representation

Following the identification of all system parameters. A state space model was developed for the helicopter dynamics. Since the system is highly nonlinear, the state space model will be in the form:

$$\begin{cases} \dot{X} = f(X, u, t) \\ y = g(X, t) \end{cases} \qquad (3.1)$$

Where X is the state vector.

y is the output vector.

u is the input vector and t is the time.

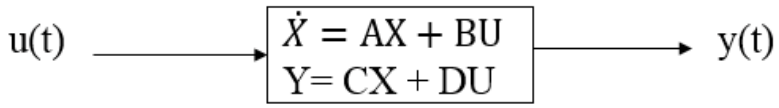

Figure 3.1 : State space representation

The state variables to be selected are the angles and their corresponding angular velocities as follow:

$$\begin{cases} x_1 = \text{p} \\ x_2 = \text{R} \\ x_3 = \text{T} \\ x_4 = \dot{\text{p}} \\ x_5 = \dot{\text{R}} \\ x_6 = \dot{\text{T}} \end{cases} \qquad (3.2)$$

The state space model according to (3.2), the variables become:

$$\begin{cases} \dot{x_1} = \dot{\text{p}} = x_4 \\ \dot{x_2} = \dot{\text{R}} = x_5 \\ \dot{x_3} = \dot{\text{T}} = x_6 \\ \dot{x_4} = \dfrac{-\mu_p x_4 + L_H \, \text{U}_{sum} cos x_1 - M_G(x_1)}{I_P} \\ \dot{x_5} = \dfrac{-M_H \, g \, L_R \sin x_2 \, cos x_1 - \mu_R x_5 + L_M \, \text{U}_{diff}}{I_R} \\ \dot{x_6} = \dfrac{\text{U}_{sum} \, L_H \sin x_2 \, cos x_1 - \mu_T \, x_6}{I_T} \end{cases} \qquad (3.3)$$

Where:

$$U_{sum} = F_{backward} + F_{forward} \qquad (3.4)$$

$$U_{diff} = F_{backward} - F_{forward} \qquad (3.5)$$

3.3. Linearization

To design the controller, the nonlinear state space model must first be linearized. The system will be linearized using a method known as "Jacobian linearization of a nonlinear system." This method is an indirect approach that uses the Taylor series expansion technique to linearize the system around a certain operating point known as an equilibrium point. A point $\bar{x} \in R_n$ is called an equilibrium point if there is a specific $\bar{u} \in R_m$ (called the equilibrium input) such that:

$$\text{f}(\bar{x}, \bar{u}) = 0 \qquad (3.6)$$

3.3.1. Equilibrium point

To get the equilibrium points and their corresponding inputs the equation (3.7) must be solved:

$$\begin{pmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \\ \dot{x_5} \\ \dot{x_6} \end{pmatrix} = \begin{pmatrix} \overline{x_4} \\ \overline{x_5} \\ \overline{x_6} \\ \frac{-\mu_p \overline{x_4} + L_H\ \bar{u}_{sum} cos x_1 - M_G(\overline{x_1})}{I_P} \\ \frac{-M_H\ g\ L_R \sin x_2\ cos x_1 - \mu_R \overline{x_5} + L_M\ \bar{u}_{diff}}{I_R} \\ \frac{-\bar{u}_{sum}\ L_H \sin \overline{x_2}\ cos \overline{x_1} - \mu_T\ \overline{x_6}}{I_T} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (3.7)$$

$$0 = \overline{x_4} \quad (3.8)$$

$$0 = \overline{x_5} \quad (3.9)$$

$$0 = \overline{x_6} \quad (3.10)$$

$$0 = \frac{-\mu_p \overline{x_4} + L_H\ \bar{u}_{sum} cos x_1 - M_G(\overline{x_1})}{I_P} \quad (3.11)$$

$$0 = \frac{-M_H\ g\ L_R \sin x_2\ cos x_1 - \mu_R \overline{x_5} + L_M\ \bar{u}_{diff}}{I_R} \quad (3.12)$$

$$0 = \frac{-\bar{u}_{sum}\ L_H \sin \overline{x_2}\ cos \overline{x_1} - \mu_T\ \overline{x_6}}{I_T} \quad (3.13)$$

Equation (3.8), (3.9) and (3.10) give: $\overline{x4} = \overline{x5} = \overline{x6} = 0$, so only three equations are left to be solved:

$$\begin{cases} \frac{L_H\ \bar{u}_{sum} cos x_1 - M_G(\overline{x_1})}{I_P} = 0 \\ \frac{-M_H\ g\ L_R \sin x_2\ cos x_1 + L_M\ \bar{u}_{difF}}{I_R} = 0 \\ \frac{-\bar{u}_{sum}\ L_H \sin \overline{x_2}\ cos \overline{x_1}}{I_T} = 0 \end{cases} \quad (3.14)$$

This is a system with three equations and four unknowns, implying that there is unlimited number of equilibrium points, as expected. One of the unknowns must be fixed in order to solve this system.

Setting $x_1 = 0$ and $x_2 = 0$, only two equations are remaining:

$$L_H \ \bar{u}_{sum} - M_G(\bar{0}) = 0 \qquad (3.15)$$

$$L_M \ \bar{u}_{diff} = 0 \qquad (3.16)$$

Equation 3.16 gives $\bar{u}_{diff} = 0$. equation 3.16 is important because with it the required thrust to keep a certain elevation angle can be deduced.

Solving for $\bar{u}_{sum}$ gives: $\bar{u}_{sum} = \frac{M_G(0)}{L_H}$

Finally, note that the state x5 (travel angle) does not appear in any of the equations above, implying that its value has no influence whatsoever on the obtained state space.

3.3.2. Linearized state-space model

It is possible to linearize the model using the indirect approach about the equilibrium points and inputs obtained previously:

$$\begin{cases} \dot{X} = \mathrm{AX} + \mathrm{Bu} \\ y = \mathrm{CX} + \mathrm{Du} \end{cases} \qquad (3.17)$$

The matrices A and B can be deduced by calculating the Jacobian of f(X, u) at the equilibrium points and inputs.

$$\mathrm{A}=\mathrm{J}_x f(X,u) = \frac{\partial \mathrm{f}}{\partial \mathrm{x}}(\bar{X},\bar{u}) = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{-\frac{dM_G(0)}{dx}}{I_P} & 0 & 0 & \frac{-\mu_p}{I_P} & 0 & 0 \\ 0 & \frac{-M_H\, g\, L_R}{I_R} & 0 & 0 & \frac{-\mu_R}{I_R} & 0 \\ 0 & \frac{\mathrm{U}_{sum}\, L_H}{I_T} & 0 & 0 & 0 & \frac{-\mu_T}{I_T} \end{pmatrix} \qquad (3.18)$$

$$\mathrm{B}=\mathrm{J}_\mathrm{u}\mathrm{f}(\mathrm{X},\mathrm{u})=\frac{\partial \mathrm{f}}{\partial \mathrm{u}}(\bar{\mathrm{X}},\bar{\mathrm{u}})=\begin{pmatrix} 0 & 0 \\ 0 & \frac{\mathrm{L_H}}{\mathrm{I_P}} \\ 0 & 0 \\ 0 & 0 \\ \frac{\mathrm{L_R}}{\mathrm{I_R}} & 0 \\ 0 & 0 \end{pmatrix} \qquad (3.19)$$

$$C=\begin{pmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\end{pmatrix} \tag{3.20}$$

The linearized state-space model now reads:

$$\begin{pmatrix}\dot{x_1}\\\dot{x_2}\\\dot{x_3}\\\dot{x_4}\\\dot{x_5}\\\dot{x_6}\end{pmatrix}=\begin{pmatrix}0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\\\frac{-\frac{dM_G(0)}{dx}}{I_P}&0&0&\frac{-\mu_p}{I_P}&0&0\\0&\frac{-M_H\ g\ L_R}{I_R}&0&0&\frac{-\mu_R}{I_R}&0\\0&\frac{U_{sum}\ L_H}{I_T}&0&0&0&\frac{-\mu_T}{I_T}\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\\x_4\\x_5\\x_6\end{pmatrix}+\begin{pmatrix}0&0\\0&\frac{L_H}{I_P}\\0&0\\0&0\\\frac{L_M}{I_R}&0\\0&0\end{pmatrix}\begin{pmatrix}U_{sum}\\U_{diff}\end{pmatrix}$$

$$y=\begin{pmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\end{pmatrix}\begin{pmatrix}x1\\x2\\x3\\x4\\x5\\x6\end{pmatrix}$$

3.4. Decoupling

The simplest way to design a controller for a Multiple Input Multiple Output (MIMO) system is to decouple it by matching its inputs and outputs to create two distinct systems with a Single Input Single Output (SISO) that work independently of each other.

From the linearized matrix A, the pitch system is fully independent from the roll and travel, with the pitch angle only reacting to the U_{sum}input and the roll angle only reacting to the U_{diff} input. Pairing the input and outputs as so gives two separate systems.

3.4.1. Pitch system

The decoupled system will be as follows:

$$\begin{pmatrix}\dot{x_1}\\\dot{x_4}\end{pmatrix}=\begin{pmatrix}0&1\\\frac{-\frac{dM_G(0)}{dx}}{I_P}&\frac{-\mu_p}{I_P}\end{pmatrix}\begin{pmatrix}x_1\\x_4\end{pmatrix}+\begin{pmatrix}0\\\frac{L_H}{I_P}\end{pmatrix}U_{sum} \tag{3.21}$$

$$y=p=x_1=(1\quad 0)\begin{pmatrix}x_1\\x_4\end{pmatrix} \tag{3.22}$$

The transfer function of the decoupled elevation system will be as follows:

$$G(s) = C(sI - A)^{-1}B = \frac{I_P}{I_P s^2 + \mu_p s + \frac{dM_G(0)}{dx}}$$

3.4.2. Travel and Roll system

The decoupled system will be as follows:

$$\begin{pmatrix} \dot{x_2} \\ \dot{x_3} \\ \dot{x_5} \\ \dot{x_6} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{-M_H\, g\, L_R}{I_R} & 0 & \frac{-\mu_R}{I_R} & 0 \\ \frac{U_{sum}\, L_H}{I_T} & 0 & 0 & \frac{-\mu_T}{I_T} \end{pmatrix} \begin{pmatrix} x_2 \\ x_3 \\ x_5 \\ x_6 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \frac{L_M}{I_R} \\ 0 \end{pmatrix} U_{diff}$$

$$y = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_2 \\ x_3 \\ x_5 \\ x_6 \end{pmatrix}$$

3.5. System properties

After linearizing the system now, a controller can be developed. However, before developing a controller, the evaluation of the fundamental properties of the system dynamics is important to see if it is controllable and fully observable.

3.5.1. Stability

Stability deals with system's behavior after a perturbation from a nominal state of operation. From basic control courses, the following statements can be set:

- For a system described by a state equation, stability (asymptotic) is determined by the location of the eigenvalues of A: stability if $\mathrm{Re}\{\lambda i\} < 0$ for all i, instability if at least one λi, such that $\mathrm{Re}\{\lambda i\} > 0$ and marginal stability if non-repeated eigenvalues on the jw axis. Instability otherwise.
- For a system described by a transfer function, stability (BIBO) is determined by the location of the poles in the same way.

3.5.1.1. Pitch system

The eigenvalues are calculated from the A matrix, for simplification only two significant figures are taken for all the matrices.

$$A=\begin{pmatrix} 0 & 1 \\ \frac{-dM_G(0}{dx} & \frac{-\mu_p}{I_P} \end{pmatrix}=\begin{pmatrix} 0 & 1 \\ -1.60 & -0.55 \end{pmatrix}$$

$$B=\begin{pmatrix} 0 \\ \frac{L_H}{I_P} \end{pmatrix}=\begin{pmatrix} 0 \\ 0.42 \end{pmatrix}$$

$$\lambda_{1,2} = -0.27 \pm 1.23j$$

Since the real part of λ_1 and λ_2 is negative the system is stable.

3.5.1.2. Roll and travel system

$$A=\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{-M_H\, g\, L_R}{I_R} & 0 & \frac{-\mu_R}{I_R} & 0 \\ \frac{U_{sum}\, L_H}{I_T} & 0 & 0 & \frac{-\mu_T}{I_T} \end{pmatrix}=\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -17.38 & 0 & -1.12 & 0 \\ 0.91 & 0 & 0 & -0.23 \end{pmatrix}$$

$$B=\begin{pmatrix} 0 \\ 0 \\ \frac{L_M}{I_R} \\ 0 \end{pmatrix}=\begin{pmatrix} 0 \\ 0 \\ 3.86 \\ 0 \end{pmatrix}$$

$$\lambda_1 = 0$$

$$\lambda_2 = -0.23$$

$$\lambda_{3,4} = -0.56 \pm 4.13j$$

Due to the existence of a pole at the origin, the system may be marginally stable while all the remaining poles are situated on the left hand plane.

3.5.2. Reachability

The reachability problem is to find the set of all the final states x(t_0) reachable starting from a given initial state x(t_0):

A state x(t_1) of a dynamic system is reachable from the state x(t_0) in the time interval [t_0, t_1] if it exists an input function u(t) ∈ U such that x(t_1) = ψ(t0, t_1, x(t_0), u(t)) [11]

In other meaning, reachability is the ability of changing the eigenvalues of the system from {λ1, λ2, · · ·, λn} into any arbitrary {$\overline{\lambda 1}, \overline{\lambda 2}, \cdots, \overline{\lambda n}$, } via a state feedback input:

$$u(t) = -KX(t).$$

Mathematically speaking, a system is described by $\begin{cases} \dot{X} = \mathrm{AX} + \mathrm{Bu} \\ y = \mathrm{CX} + \mathrm{Du} \end{cases}$ is reachable if and only if the reachability matrix: $W_r = (B\ AB\ \ldots . \mathrm{A}^{\mathrm{n}-1}\mathrm{B})$ is full rank.

3.5.2.1. Pitch system

$$W_r = (B\ AB) = \begin{pmatrix} 0 & 0.42 \\ 0.42 & -0.55 \end{pmatrix}$$

Since the reachability matrix W_r is full rank, the pitch system is fully reachable.

3.5.2.2. Roll and Travel system

$$W_r = (B\ AB\ A^2B\ A^3B\ ) = \begin{pmatrix} 0 & 3.86 & -4.37 & 62.29 \\ 0 & 0 & 0 & 3.55 \\ 3.86 & 4.37 & 62.29 & 146.35 \\ 0 & 0 & 3.55 & -4.84 \end{pmatrix}$$

Since the reachability matrix W_r is full rank, the pitch system is fully reachable

3.5.3. Controllability

The controllability problem is to find the set of all the initial states x(t_0) controllable to a given final state x(t_1). A state x(t_0) of a dynamic system is controllable to state x(t_1) in the time interval [t_0, t_1] if it exists an input function u(t) ∈ U such that x(t1) = ψ (t0, t_1, x(t_0), u(t)).

In other words, controllability is the ability of the control input to drive the state from any initial value X(0) to the origin X =0.

In this project, the reachability is the main concern in order to control the system properly.

3.5.4. Observability

The ability of the output y(t) to give enough information about the internal dynamics of the system to allow estimation of the state X(t) is referred to as observability. In this system, the only measured states are the pitch, Roll and travel angles, so it is very important to check if the system is fully observable so that the observation of the remaining states (angular velocities) can be possible.

Mathematically speaking, a system describes by $\begin{cases} \dot{X} = \mathrm{AX} + \mathrm{Bu} \\ y = \mathrm{CX} + \mathrm{Du} \end{cases}$ is observable if and only if the observability matrix $W_o = \begin{pmatrix} C \\ CA \\ \cdot \\ \cdot \\ CA^{n-1} \end{pmatrix}$ is full rank.

3.5.4.1. Pitch system

Calculating the observability matrix of the pitch system gives:

$$W_o = \begin{pmatrix} C \\ CA \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

The observability matrix W_0 is full rank, hence, the pitch system is fully observable.

3.5.4.2. Roll and travel system

Calculating the observability matrix of the pitch system gives:

$$W_o = \begin{pmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0.91 & 0 & 0 & -0.23 \\ 0 & 0 & 0.91 & 0.05 \end{pmatrix}$$

The observability matrix W_o is full rank, hence, the Travel system is also fully observable.

3.5.5. Canonical forms

Standard forms having certain unique and valuable qualities are known as canonical forms. Canonical forms are obtained using similarity transformations.

3.5.5.1. Controller canonical form

Given the system $\begin{cases} \dot{X} = \mathrm{AX} + \mathrm{Bu} \\ y = \mathrm{CX} + \mathrm{Du} \end{cases}$ the similarity transformation Tc can be deduced. Such that Xc= Tc

The new state space is defined as $\begin{cases} \dot{X} = \mathrm{Ac\,X} + \mathrm{Bc\,u} \\ y = \mathrm{Cc\,X} + \mathrm{Dc\,u} \end{cases}$ (3.23)

Where:

$A_C = T_C \mathrm{A} {T_C}^{-1}$

$B_C = T_C \mathrm{B}$

$C_C = C {T_C}^{-1}$

$D_C = \mathrm{D}$

a. Pitch system

Controller canonical form for the pitch system is as follows:

$$A_C = \begin{pmatrix} 0 & 1 \\ -1.60 & -0.55 \end{pmatrix}$$

$$B_C = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$C_C = (0.42 \quad 0)$$

$$D_C = D = 0$$

b. Travel system

Controller canonical form for the travel system is as follows:

$$A_C = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -4.04 & -17.64 & 1.36 \end{pmatrix}$$

$$B_C = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

$$C_C = (3.55 \quad 0 \quad 0 \quad 0)$$

$$D_C = 0$$

3.5.5.2. Observer canonical form

Given the system $\begin{cases} \dot{X} = \mathrm{AX} + \mathrm{Bu} \\ y = \mathrm{CX} + \mathrm{Du} \end{cases}$. Assuming that is fully observable, then a similarity transformation to convert the system from general form to observer canonical form can be set.

The similarity transformation to converts the original state Y into a new state X. according to

X = $T_o X_o$ .Leading to:

$$\mathrm{Xo} = T_o{}^{-1}(\mathrm{AX} + \mathrm{B\,u})$$

The new state space is define as $\begin{cases} \dot{X} = A_o \mathrm{X} + B_o \mathrm{u} \\ y = C_o \mathrm{X} + D_o \mathrm{u} \end{cases}$ (3.24)

Where:

$A_O = T_o{}^{-1} A T_o \;;\; B_O = T_o{}^{-1} B \;;\; C_O = C T_o \;;\; D_O = D$

a. Pitch system

Observe canonical form for the pitch system is as follows:

$$A_O = \begin{pmatrix} 0 & -1.60 \\ 1 & 0.55 \end{pmatrix}$$

$$B_O = \begin{pmatrix} 0.42 \\ 0 \end{pmatrix}$$

$$C_O = (0 \quad 1)$$

$$D_O = 0$$

b. Travel system

Observe canonical for the travel system is as follows:

$$A_O = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -4.04 \\ 0 & 1 & 0 & 17.64 \\ 0 & 0 & 1 & -1.36 \end{pmatrix}$$

$$B_O = \begin{pmatrix} 3.55 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$C_O = (0 \quad 0 \quad 0 \quad 1)$$

$$D_O = 0$$

3.6. Controller design

After decoupling the system, two separate controllers can be designed. One is for the pitch and the other is for the travel. Below is the schematic of the system with the controllers:

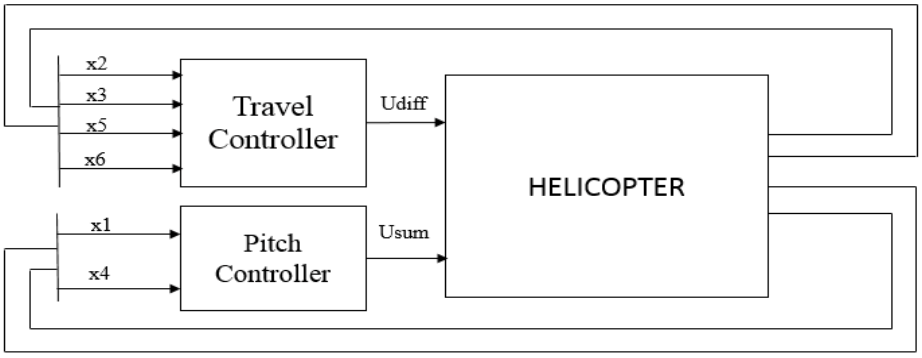

Figure 3.2 : Helicopter controllers diagram

3.6.1. LQR controller design

The main idea is to design a full state feedback, to reallocate the eigenvalues from {λ1, λ2, · · ·, λn} into desired {$\overline{\lambda 1}, \overline{\lambda 2}$, · · ·,$\overline{\lambda n}$, }. So, to determine the best eigenvalues for the best performance of the system, a good solution is to use the LQR controller in both the pitch and travel control.

LQR (Linear Quadratic Regulator) is a powerful optimal control theory based on selecting a feedback gain that minimizes the cost function J. The control gain K can be obtained by minimizing the performance cost function described as:

$$J = \int_0^{\infty} (X^T QX + U^T RU) dt \qquad (3.25)$$

Here, Q and R are weighting matrices.

Q is defined as symmetric positive semidefinite matrix $X^T QX \geq 0$, R is defined as symmetric positive definite $U^T RU \leq 0$. The gain matrix K is calculated using this formula:

$$\mathrm{K}=R^{-1}B^T S \qquad (3.26)$$

Where S is the solution of the algebraic Recatti equation:

$$\mathrm{A}S^T + \mathrm{S\ A} + \mathrm{S\ B}R^{-1}B^T\ \mathrm{S} + \mathrm{Q} = 0 \qquad (3.27)$$

After using, LQR the control law is going to be u = −KX.

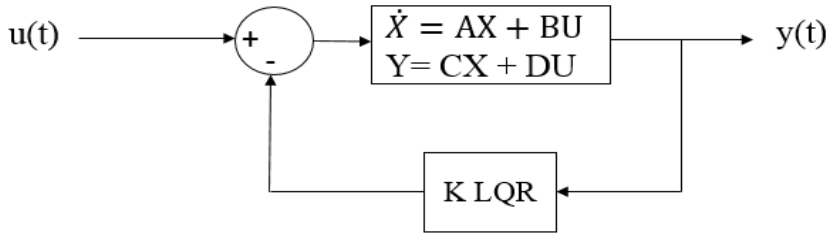

Figure 3.3 : Closed loop LQR controller

The state space is defined as:

$$\begin{cases} \dot{X} = (\mathrm{A} - \mathrm{BK})\mathrm{X} \\ y = (\mathrm{C} - \mathrm{DK})\mathrm{X} \end{cases} \qquad (3.28)$$

3.6.2. LQI controller design

For state-feedback control gain matrix, the LQR controller develops a systematical calculation model. In a control system, the integral effect reduces steady-state error, resulting into a more stable system response under changing conditions (changing reference, voltage, and load). Although the controller is doing a great job in settling the response rapidly in several circumstances, it is not achieving a steady-state error of zero. A good solution is using the Linear Quadratic Integrator controller [12].

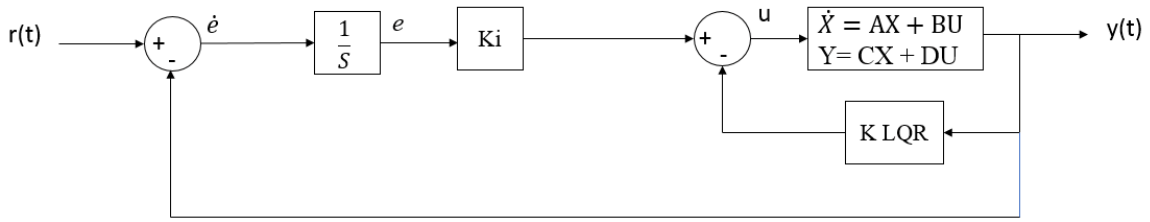

Figure 3.4: closed loop control design with LQI controller

The LQI controller is created by combining the LQR and integral effects. an extra state e(t) is added to the plant dynamics, which is the integral of the output error defined as:

$$\dot{e} = y(t) - r(t) \tag{3.29}$$

The control law for this augmented system will be:

$$u = -K_{Lqr}X - Ki \cdot e \tag{3.30}$$

Where K_{Lqr} is the controller gain and Ki is the integral gain.

The full augmented system will be

$$\dot{X} = AX + Bu \tag{3.31}$$

$$\dot{X} = AX + B\,(-K_{Lqr}X - Ki \cdot e) \tag{3.32}$$

$$\dot{X} = (A - B\,K_{Lqr})X - B\,Ki \cdot e \tag{3.33}$$

$$\begin{bmatrix} \dot{X} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} A - B\,K_{Lqr} & -B\,Ki \\ C & 0 \end{bmatrix} \begin{bmatrix} X \\ e \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix} r(t) \tag{3.34}$$

The integral gain can be deduced using two methods:

1- The MATLAB software generates automatically the integral term using the linear quadratic integrator command.

It is important to mention that the augmented state space is the system to be used in the LQI generator.

2- Estimating Ki using the LQR command from the MATLAB to get K and then it is automatically deduced using the following relation $Ki = (C\,(BK - A)^{-1}\,B)^{-1}$[13].

3.6.3. Controller Tuning

To get the controller gain matrix k, the weightining matrices Q and R are selected arbitary, where a MATLAB text can generate the corresponding gains K, the procedure is repeated several times until the desired simulated response is reached.

3.6.3.1. Pitch axis

To acquire the satisfying results with the optimal gain matrix K, the matrices Q and R are tuned until getting an optimal gain K that provides a good response.

$$Q=\begin{pmatrix}5 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 5\end{pmatrix}$$

$R=1$

$$K=[2.17 \quad 2.28 \quad -2.23]$$

3.6.3.2. Travel axis

Repeating the same process, the results are as follows:

$$Q=\begin{pmatrix}6 & 0 & 0 & 0 & 0\\ 0 & 10 & 0 & 0 & 0\\ 0 & 0 & 4 & 0 & 0\\ 0 & 0 & 0 & 4 & 0\\ 0 & 0 & 0 & 0 & 10\end{pmatrix}$$

$R= 30$

$$K=[0.49 \quad 2.56 \quad 0.39 \quad 4.14 \quad -0.57]$$

3.7. State Observer

The observer state is a dynamic system that is used to estimate the state of a system or some of the states of a system. A full-state observer is used to estimate all the states of the system. The state observer can be designed as either a continuous-time system or a discrete-time system [14]

In the real time system, three encoders are used to measure the angles of pitch, roll and travel however the angular velocities are not directly measurable, instead they have to be estimated using the Euler-forward method or estimated by the use of a state estimator/observer. It is chosen to work with Kalman filter as the state estimator because it showed good results with real time system.

3.7.1. Kalman Filter

The Kalman filter estimates the state $X \in \Re^n$ of a discrete-time controlled process that is governed by the linear stochastic difference equation:

$$x_k = A\,x_{k-1} + x_{k-1} + \mathrm{B}\,W\,x_{k-1} \tag{3.35}$$

$$y = Cx_k + V_k \tag{3.36}$$

The random variables represent the process and measurement noise (respectively). They are assumed to be independent (of each other), whom can be represented in form of white noise, and with normal probability distributions: $p(w) \sim N(0, Q)$; $p(v) \sim N(0, R)$

The Kalman filter equation can be described as:

$$\hat{x}_{K+1} = A\hat{x}_K + Bu_K + L(y_K - C(A\hat{x}_K + Bu_K) \tag{3.37}$$

The Kalman filter estimates a process by using a form of feedback control: the filter estimates the process state at some time and then obtains feedback in the form of (noisy) measurements. As such, the equations for the Kalman filter fall into two groups: time update equations and measurement update equations. The time update equations are responsible for projecting forward (in time) the current state and error covariance estimates to obtain the a priori estimates for the following sample response. The measurement update equations are responsible for the feedback—i.e. for incorporating a new measurement into the a priori estimate to obtain an improved a posteriori estimate.

The time update equations can also be thought of as predictor equations, while the measurement update equations can be thought of as corrector equations. Indeed, the final estimation algorithm resembles that of a predictor-corrector algorithm for solving numerical problems as shown below in Figure 3.5:

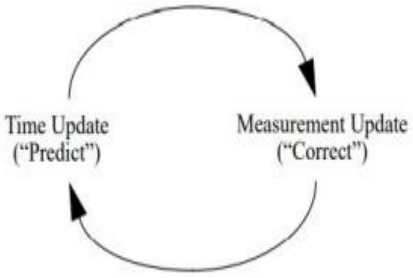

Figure 3.5: the ongoing discrete Kalman filter cycle

Discrete Kalman filter time update equations:

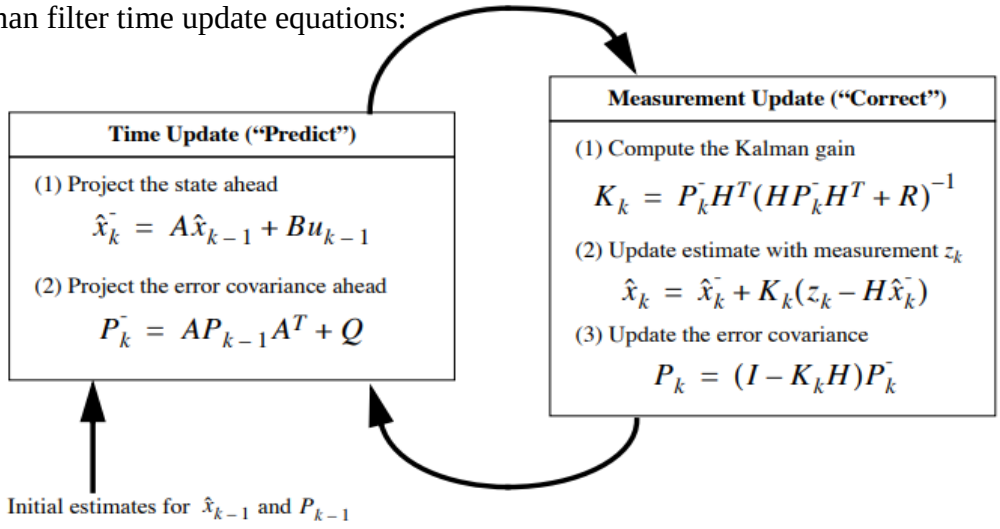

Figure 3.7 : Kalman equations

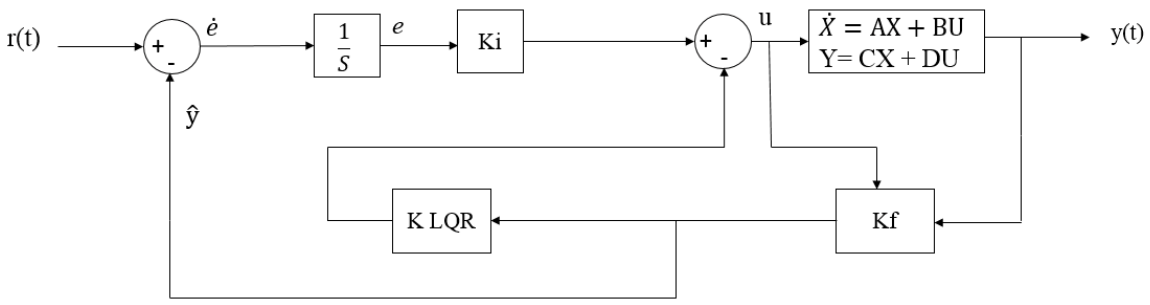

Figure 3.8: Controlled feedback loop with Kalman filter

$$\hat{x}_K = A\hat{x}_{K-1} + Bu_{K-1} \quad (3.38)$$

$$P_K = AP_{K-1}A^T + Q \quad (3.39)$$

3.7.2. Kalman Filter tuning

To tune the Kalman filter the noise covariance matrices Q and R are selected based on the real time measurements, then simply by using a MATLAB command deduce Kf the gain of the filter. Based on the simulation Q and R can be re-selected to suite the desired performance for optimal simulation.

3.7.2.1. Pitch system

Q and R are chosen as follows:

$$Q=\begin{pmatrix}1 & 0\\ 1 & 1\end{pmatrix}$$

R=1

The obtained Kalman coefficients:

$$Kf=\begin{pmatrix}0.87\\ -0.12\end{pmatrix}$$

3.7.2.2. Travel system

Due the high instability of the system when stepping it with a pitch angle. Q and R are chosen as follows:

$$Q=\begin{pmatrix}0.10 & 0 & 0 & 0\\ 0 & 0.10 & 0 & 0\\ 0 & 0 & 0.10 & 0\\ 0 & 0 & 0 & 0.10\end{pmatrix}$$

R=1

Kalman gains:

$$Kf=\begin{pmatrix}-0.0024\\ 0.6650\\ -0.0026\\ 0.1711\end{pmatrix}$$

3.8. LQG CONTROL

Previously the optimal state feedback (LQR) and the optimal observer (Kalman filter) were designed, combining these two gives the LQG controller.

Linear-quadratic-Gaussian (LQG) control is a modern state-space technique for designing optimal dynamic regulators and servo controllers with integral action (also known as setpoint trackers). This technique allows you to trade off regulation/tracker performance and control effort, and to consider process disturbances and measurement noise [15].

$$\dot{X} = AX - B\widehat{K} + W \qquad (3.43)$$

$\widehat{X}$ is defined as $\widehat{X} = X - (X - \widehat{X})$ and the estimation error $\epsilon = X - \widehat{X}$ gives:

$$\dot{X} = AX - BKX + BK(X - \widehat{X}) + W \qquad (3.44)$$

$$\dot{X} = (A - BK)X + BK\epsilon + W \qquad (3.45)$$

Taking the derivative of the error and rearrange the observer equation gives:

$$\dot{\epsilon} = (A - KfC)\,\epsilon \qquad (3.46)$$

Taking measurement noise and disruptions into consideration results in the following Equation:

$$\dot{\epsilon} = (A - Kf\,C)\,.\,\epsilon + W - Kf\,V \qquad (3.47)$$

LQG state space equation can be described as:

$$\begin{bmatrix}\dot{X}\\ \dot{\epsilon}\end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - KfC \end{bmatrix}\begin{bmatrix}X\\ \epsilon\end{bmatrix} + \begin{bmatrix} I & 0 \\ I & -Kf \end{bmatrix}\begin{bmatrix}W\\ V\end{bmatrix} \qquad (3.48)$$

Even if the system is coupled, the eigenvalues of X are still controlled by the LQR gain and the eigenvalues of the error are controlled by the Kalman filter.

3.9. Conclusion

In this chapter, the system was represented in its state space model in order to be linearized around the available equilibrium point. The system was then decoupled to design the different controllers for pitch and travel. The evaluation of fundamental properties such as stability, observability and controllability were discussed in order to design the LQR controller in combination with integral action which results into the LQI controller. Since the only observable outputs were given by the rotary encoders. the Kalman filter was designed to acquire the full state of the system .the combination of the LQR with Kalman filter results into the LQG controller .

CHAPTER 4

IMPLEMENTATION

4.1. Introduction

Fitting the identification experimental process that has been determined in the second chapter with the theoretical state space that has been built in the preceding chapter, the true purpose of this project can finally be translated into a healthy real-time functioning system.

After dissecting each component individually, this chapter will aim to build up a full detailed explanation about the transition from a theoretical model to a finalized project that functions properly.

This project was realized using two main software: the MATLAB software and the LABVIEW graphical design interface, hence explaining the different circuits adopted throughout this project is crucial for the validation of the parameters and controllers constructed in the previous sections.

Of course, the controller is expected to behave differently when the projected is implemented in real life, hence different applications of the 3DOF helicopter will be demonstrated by the end of this chapter allowing the analysis of the observed mechanical system in the final chapter.

Figure 4.1: Real-time 3DOF Helicopter system

1: computer.

2: Power supply.

3: Encoder.

4: DAQ-board.

5: Protection.;

6: Motor+Propeller.

7: ESC.

8: Counter-weight.

4.2. Hardware description

4.2.1. Brushless direct current motor (BLDC MOTOR):

A brushless DC electric motor is a synchronous motor that is driven by a direct current voltage source and commutated electronically using a controller to switch DC currents to the motor windings producing magnetic fields which effectively rotates in space and whom the permanent magnet rotor follows.

Figure 4.2 : BLDC motor

When a direct current motor is driven, a magnetic field is generated within the stator, attracting or repelling the magnets in the rotor this causes the rotor to start spinning. To control the speed and torque of the motor, the controller adjusts the phase and amplitude of the DC current pulses. This control method substitutes the mechanical commutator (brushes) used in many traditional electric motors. BLDC consists of an outer stator with permanent magnets or electromagnetic coils and an inner rotor with coil windings. The number of phases refers to the number of windings in a brushless motor.

Though brushless motors can be constructed with different number of phases, three phase brushless motors are the most common type. Each of the three phases of the brushless motor must be capable of

being driven to either the input supply voltage or the ground, hence, three "half bridge" driving circuits, each with two switches, are employed to accomplish this. Depending on the voltage and current required, the switches can be bipolar transistors, IGBTs, or MOSFETs [16].

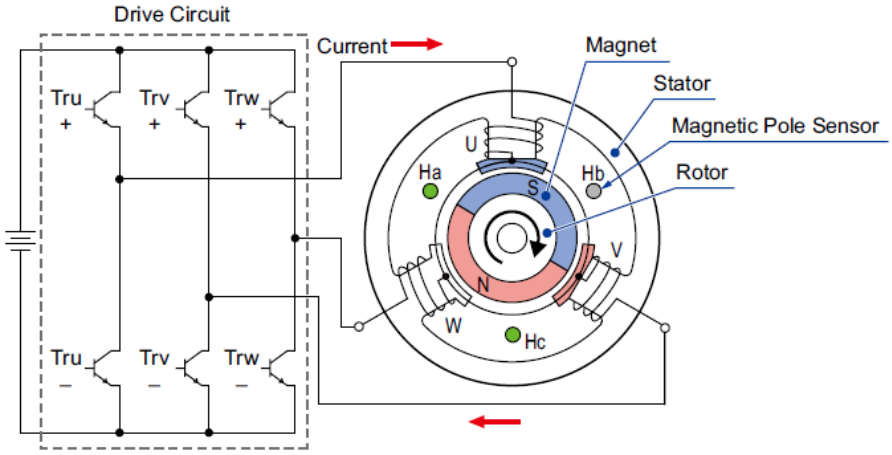

Figure 4.3 : BLDC model

This type of transistor based motor controllers are known as ESC (Electronic Speed Controller).The most important characteristic that defines the performance of BLDC motors is the revolution of the motor. The motor's revolution is a physical constant that relates the rotational speed of the motor's shaft to the input voltage supplied to it in the absence of a load:

$$n(v_{in}) = v_{in} \times k \qquad (4.1)$$

k: The motor's ideal revolution when no load is connected, unit: $^{RPM}/_{volts}$.

4.2.2. Electronic Speed Controllers (ESC)

Unlike a brushed DC motor, the commutation of a BLDC motor is controlled electronically using an electronic speed controller. The ESC uses FET-transistors to vary the speed of the motor either by changing the duty cycle of the input signal or by changing its frequency. The ESCs used in the control of the BLDC motors require periodic input signals in the form of a pulse width modulation [17]. Usually, Back EMF from the motor windings is often utilized to detect rotation, however there are also versions that employ separate magnetic (Hall effect) sensors. In this project, a 30 Amper ESC was used

Figure 4.4 : ESC

4.2.3. PROPELLER

The propeller is a tool used to produce thrust forces perpendicular to the rotational plane of the motor. Propellers feature two or more blades that are uniformly spaced around the hub and can have a fixed pitch or variable pitch.

In this project, the Graupner Propeller shown in Figure with size of 11 x 5 inch (diameter 28 cm and a pitch 12.5 cm) is used.

Figure 4.5 : Propeller

4.2.4. Arduino Board

The Arduino Uno is an open-source microcontroller board designed by Arduino.cc and based on the Microchip ATmega328P microcontroller [18].

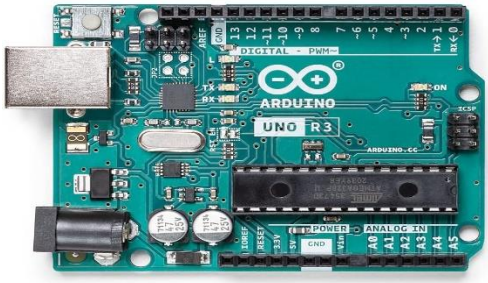

Figure 4.6 : Arduino Uno board

As explained earlier, BLDC motors require PWM signal in order to operate properly, this signal can be generated by loading a casual servomotor code to one of the Arduino's PWM pins [19]:

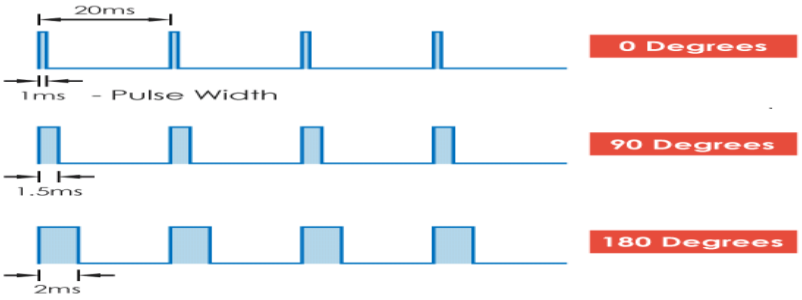

Figure 4.7: PWM Signal with its corresponding Servomotor angles

4.2.5. Acquisition Board

Using an Arduino board as a single processing unit used for the interface of all the parts of the project can be very expensive when it comes to the running time of the system. As a result, the usage of a National Instrument acquisition board optimizes the overall performance of the system. The NI-6221 acquisition board used in this project contains numerous numbers of analog/digital inputs and outputs to interpret the readings of sensors and counters used in implementations.

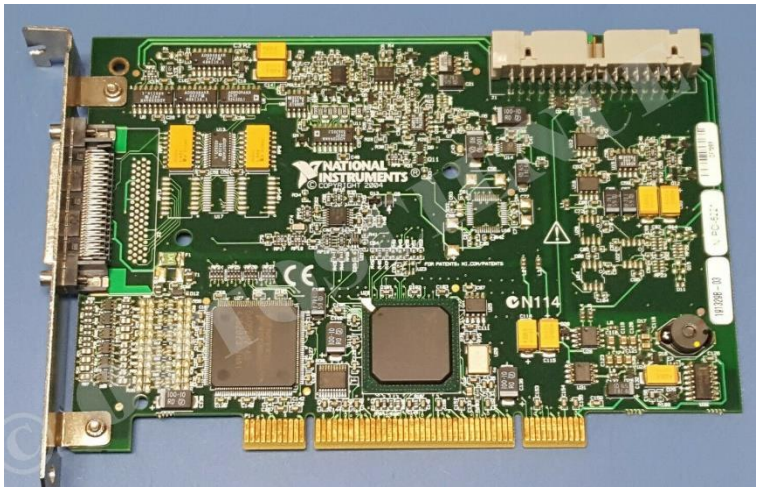

Figure 4.8 : NI-PCI 6221

4.2.6. Incremental Rotary Encoder

Incremental rotary encoder is a device that converts angular position of its shaft to digital output signal. The optical encoder Omron (E6B2-CWZ6C) was used to capture the rotation of the motion axis. A separate counter situated in the acquisition board detects the number of output pulses.

Figure 4.9 : Rotary encoder

The reading of incremental encoder can be done in different modes and methods. The X4 mode was used in the reading during the implementation; the pulse of the pin “A” indicates that a new pulse was

detected while the pulse "B" indicates the direction of the rotation so that the counter can either increment or decrement.

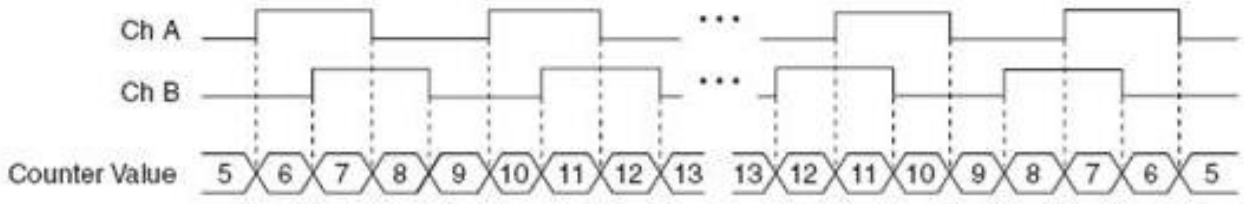

Figure 4.10 : Counter behavior during X4 mode

4.3. Design and Interface

4.3.1. Open-loop Design

Before generating the general state space that represents the overall behavior of the system, an open loop circuit had first to be implemented based only on the relations determined in the 2nd chapter:

As explained for the equation (2.15):

$$F_{sum}.L_H = M_G(\text{p})$$

A similar equation relating the roll angle to the force difference between the motors at a steady state can be derived by adopting the same logic used to determine the equation (2.15):

At steady state the angular acceleration and velocity of the roll-axis are equal to zero:

$$0 = F_{diff}.L_M - M_H.g.L_P \sin \text{R} \cos \text{p}$$

Hence the force difference required to roll the helicopter to a certain angle can be deduced as well:

$$F_{diff} = (M_H.g.L_P \sin \text{R} \cos \text{p})/L_M \qquad (4.2)$$

Taking into consideration that a voltage value must be entered the to motors, the input forces must be first converted into voltages using an equivalent relationship derived from (2.2) And (2.6):

$$v_{in} = \frac{\sqrt{\frac{T}{1.54}\times 10^7} - 981.8}{951.9} \qquad (4.3)$$

The following diagram can explain how the components are connected to each other for a basic control of the helicopter:

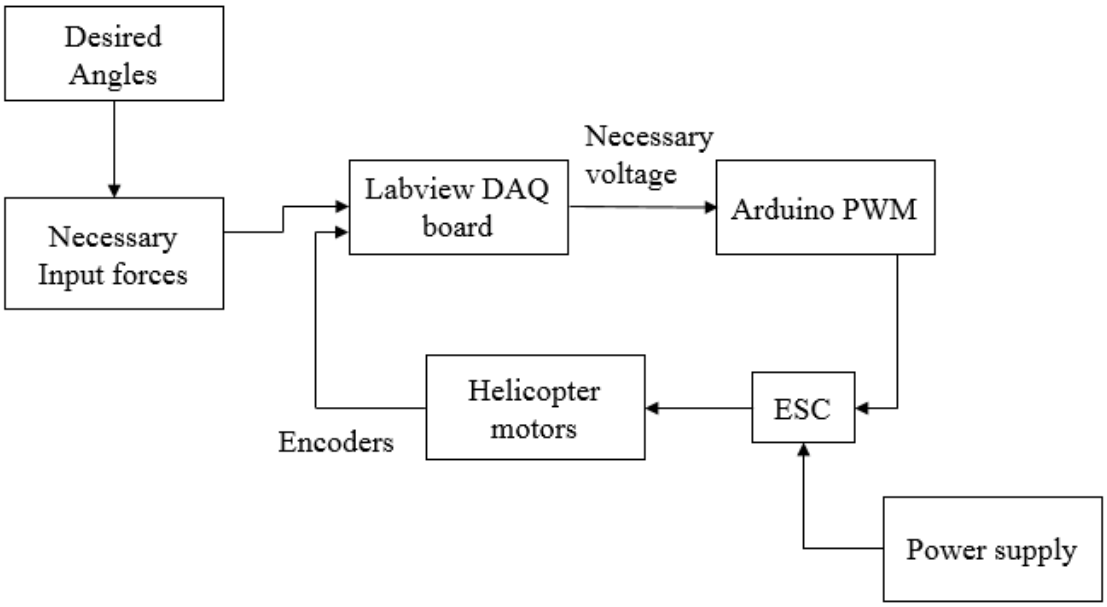

Figure 4.11 : Explicative open-loop block diagram

The above diagram can be built in the LABVIEW program using sub-Vis and the DAQ-assistant module, this module is the component responsible of controlling the signals leaving and entering the acquisition board, in this particular case, the DAQ-assistant is sending analog values to the Arduino's board which will enter the ESCs as a PWM signal.

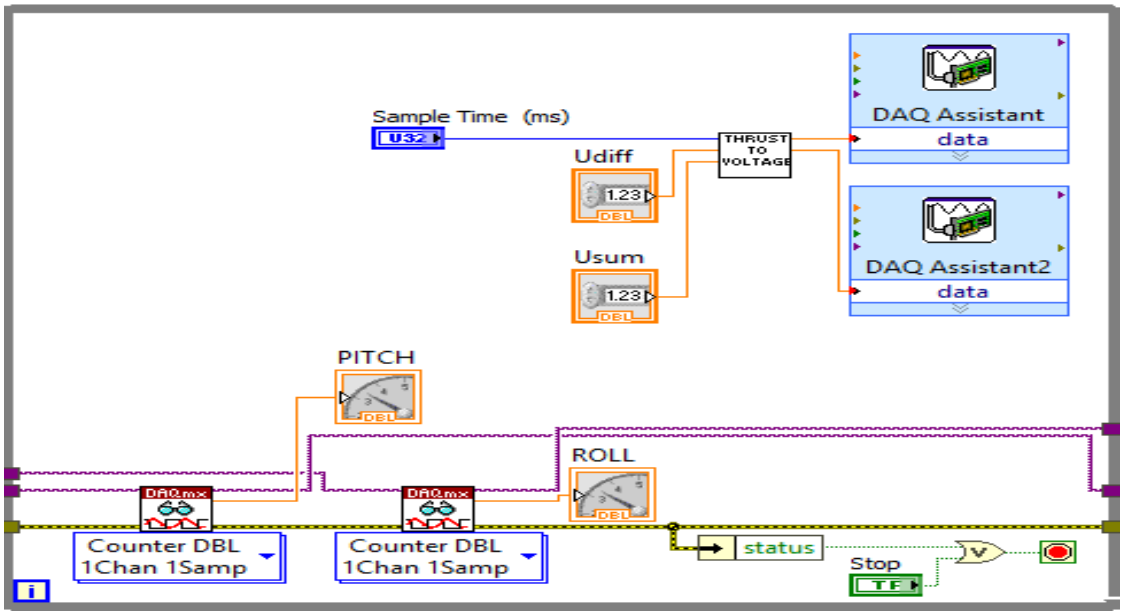

Figure 4.12 : Labview open-loop circuit

4.3.2. Controller Implementation

4.3.2.1. Controller's Interface

The purpose of designing a controller for a specific system is to force a physical model to move from its initial state to a reference state, and to be able to maintain this desired state despite the disturbances applied on the plant.

As explained in the previous (3.5.2), an additional state is added to the state space; the physical explanation of this error is the angle or the distance remaining for the helicopter to achieve the desired value, this error is obtained by computing the difference between the reference value and the real-time measurements given by the sensors.

To optimize the usage of the LABVIEW program, it is preferred to estimate the controller gains on the MATLAB software since it is customized for this type of tasks.

The computations done to obtain all the parameters for this part are specified for an equilibrium point of 0° for all the three angles. However, when the power is turned on, the helicopter starts floating from an elevation angle of -20°, this means that the system needs to be driven manually to its equilibrium point before the activation of the controller, a simple switch-case loop can alter between the two modes (manual pilot and controlled pilot). The following LABVIEW's while-loop represents the software part of the implementation during the control mode.

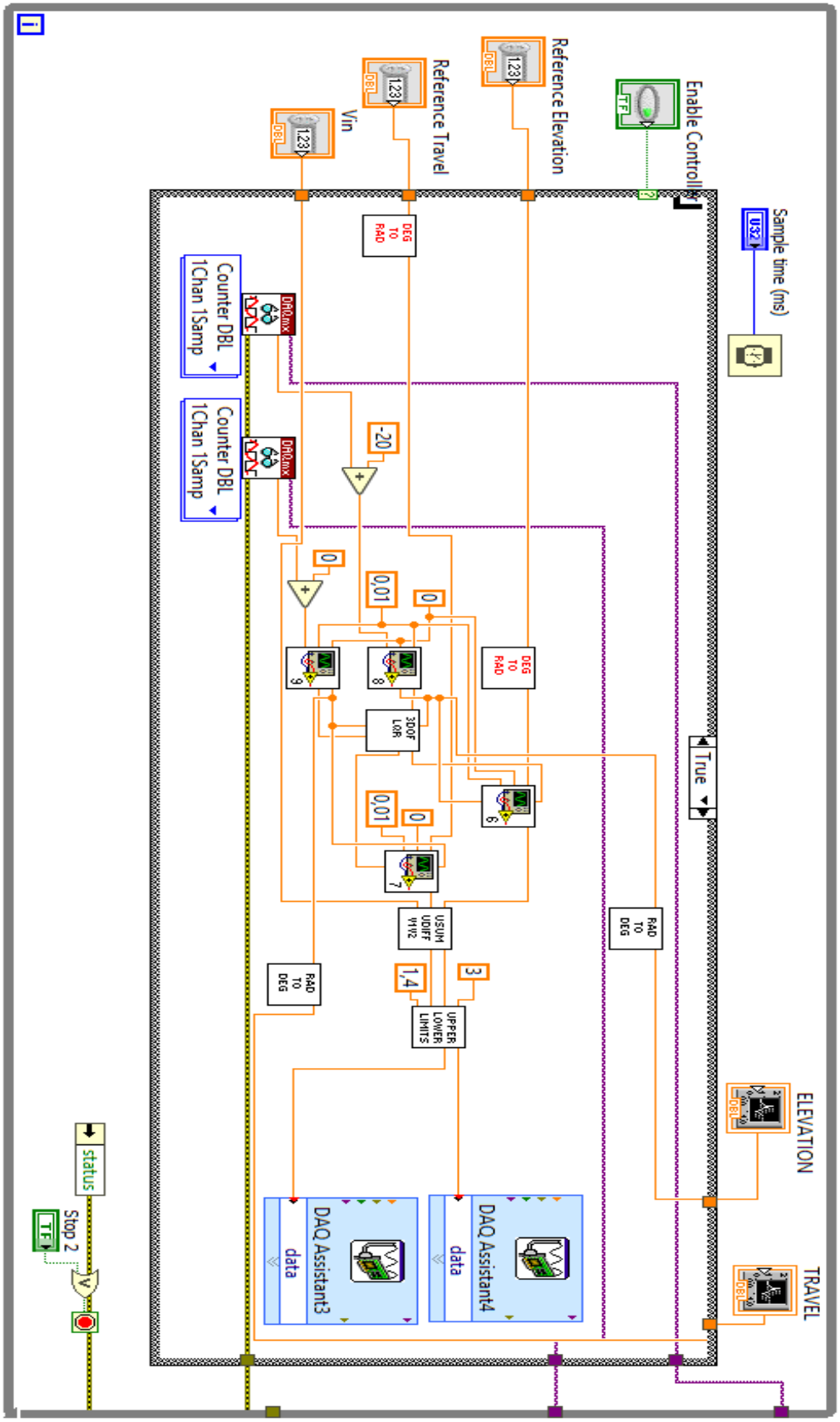

Figure 4.12 : Labview closed-loop LQR contoller circuit

To simulate this operation on the MATLAB software, the following Simulink model can be proposed for the pitch-axis (the same model can be proposed for the rest of the axis).

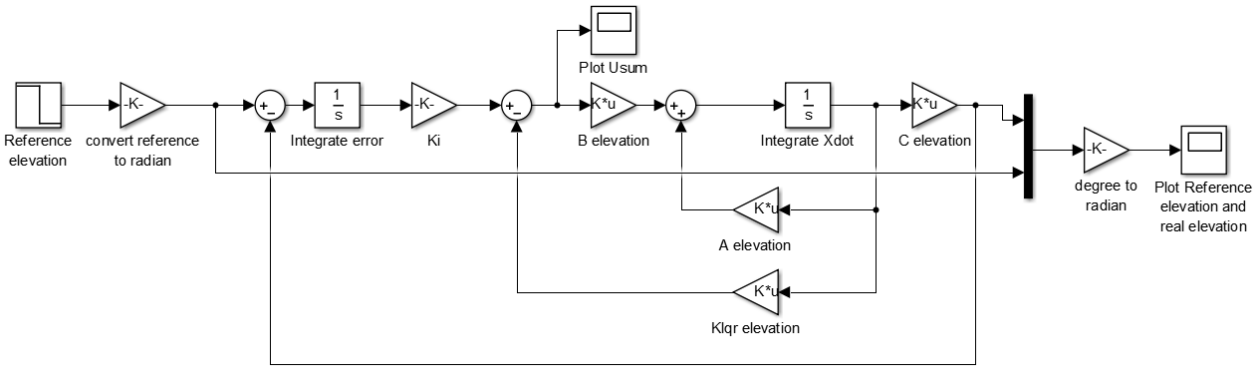

Figure 4.13 : SIMULINK closed-loop block diagram with a Kalman filter

4.3.2.2. LQR to PID conversion

As have been seen in the fig-3.8 the design of some controllers can be very complicated to achieve on the LABVIEW platform, as a solution, the very well-known PID controller can substitute the LQI controller.

The purpose of this operation is to simplify and optimize the software design of the system without affecting the overall performance of the program, this can only be achieved if the PID gains are equivalent to the pole placement gains.

Proportional Integral Derivative Controller: The proportional integral derivative controller or as often referred to as the PID controller is a compensator based on the closed-loop design [20]. The PID controller is composed of three terms: the proportional term which is responsible of augmenting the error with a certain gain K_P, the integral term K_I reduces the error to zero while the derivative term K_D works on the performance of the controller. The general expression of a PID compensator can be given as:

$$y(t) = x(t).K_P + \frac{dx}{dt} \times K_D + K_I \times \int x(t).dt \qquad (4.3)$$

$x(t)$: The input of the PID controller at instance t.

$y(t)$: The output of the PID controller at instance t.

From the feedback state space, an expression of u_{sum} and u_{diff} can be derived since:

$$u(t) = -K_{lqi}.x(t) + \frac{1}{s} \times \dot{e}(t).K_I \qquad (4.4)$$

For the Pitch axis:

$$u_{sum}(t) = \frac{1}{s} \times \dot{e}_{\mathrm{P}}(t).K_{Ip} - K_{p1}.x_1(t) - K_{p2}.\dot{x_1}(t) \qquad (4.5)$$

K_{p1}, K_{p2}: LQR gains.

The pitch angle can be written in its second term:

$$x_1(t) = r(t) - \dot{e}_{\mathrm{P}}(t) \qquad (3.29)$$

Replacing (3.29) in (4.5):

$$u_{sum}(t) = \frac{1}{s} \times \dot{e}_{\mathrm{P}}(t).K_{Ip} - K_{p1}.r(t) + K_{p1}.\dot{e}_{\mathrm{P}}(t) - K_{p2}.\dot{r(t)} + K_{p2}.\ddot{e}_{\mathrm{P}}(t)$$

The simplest method to deduce the gains, is by setting the reference input to zero:

$$u_{sum}(t) = \frac{1}{s} \times \dot{e}_{\mathrm{P}}(t).K_{Ip} + K_{p1}.\dot{e}_{\mathrm{P}}(t) + K_{p2}.\ddot{e}_{\mathrm{P}}(t) \qquad (4.6)$$

By comparing (4.3) to (4.6), the different PID gains can be obtained:

$$K_P = K_{p1}$$

$$K_I = K_{Ip}$$

$$K_D = K_{p2}$$

Following the same technique, the PID terms for the travel-axil control can also be computed:

$$u_{diff}(t) = \frac{1}{s} \times \dot{e}_{\mathrm{T}}(t).K_{I\mathrm{T}} - K_{\mathrm{T}1}.x_2(t) - K_{\mathrm{T}2}.x_3(t) - K_{\mathrm{T}3}.\dot{x_2}(t) - K_{\mathrm{T}4}.\dot{x_3}(t)$$

Writing $x_3(t)$ in terms of $r(t)$ and $\dot{e}_{\mathrm{T}}(t)$:

$$u_{diff}(t) = \frac{1}{s} \times \dot{e}_{\mathrm{T}}(t).K_{I\mathrm{T}} - K_{\mathrm{T}2}.r(t) + K_{\mathrm{T}2}.\dot{e}_{\mathrm{T}}(t) - K_{\mathrm{T}4}.\dot{r(t)} + K_{\mathrm{T}4}.\ddot{e}_{\mathrm{T}}(t) - K_{\mathrm{T}1}.x_2(t) - K_{\mathrm{T}3}.\dot{x_2}(t)$$

Setting $r(t)$ to zero:

$$u_{diff}(t) = \frac{1}{s} \times \dot{e}_{\mathrm{T}}(t).K_{I\mathrm{T}} + K_{\mathrm{T}2}.\dot{e}_{\mathrm{T}}(t) + K_{\mathrm{T}4}.\ddot{e}_{\mathrm{T}}(t) - K_{\mathrm{T}1}.x_2(t) - K_{\mathrm{T}3}.\dot{x_2}(t)$$

This relation means that $u_{diff}(t)$ can be constructed using two parallel PID controllers, one takes the travel error $\dot{e}_{\mathrm{T}}(t)$ as an input, while the second controller takes roll angle $x_2(t)$ as an input:

- 1st PID gains:

$$K_{P1} = K_{\mathrm{T}2}$$

$$K_{I1} = K_{I\mathrm{T}}$$

$$K_{D1} = K_{T4}$$

- 2nd PID gains:

$$K_{P2} = K_{T1}$$
$$K_{I2} = 0$$
$$K_{D2} = K_{T3}$$

When replacing the LQI gains with the PID controllers, the optimized system can be represented in the following block diagram.

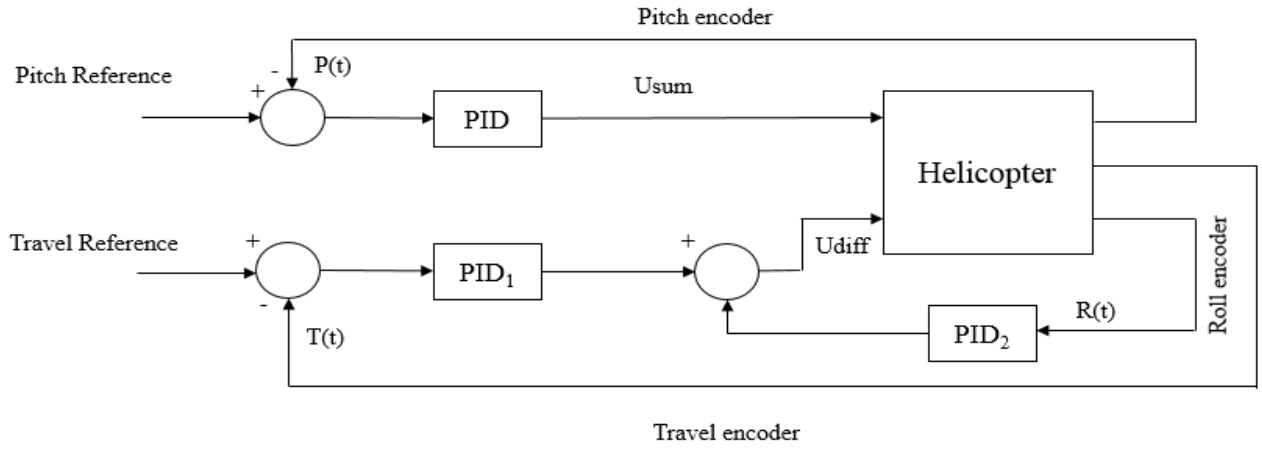

Figure 4.14 : PID Block diagram of the 3DOF

4.3.2.3. Kalman Filter Implementation

The Kalman filter combines the real output coming out from the encoders with the estimated output after taking into consideration the magnitude of the noise applied on the sensors and the magnitude of the disturbance applied on the system.

The estimated output generated from the Kalman filter is then processed through the LQI controller where the computed error is more accurate leading to an even better response from the plant.

The following Simulink block diagram explains how the Kalman filter gains are added up to the already established components of the control system.

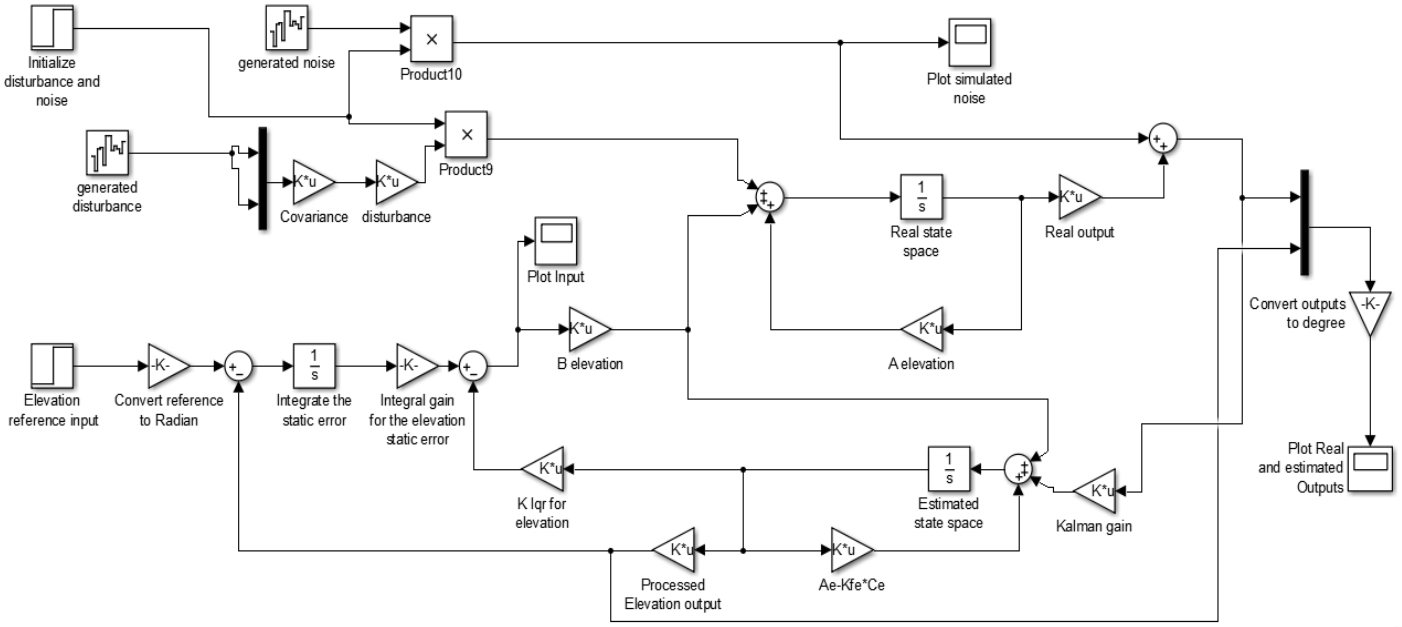

Figure 4.15 : Full LQG Block diagram

Another Simulink block for the travel-axis can be constructed using the same principle.

4.4. Applications

In this section, two applications of the control system will be explained including their realization:

4.4.1. Controlling the system using a joystick

By integrating the joystick into the control system, the two signals generated by the joystick will serve as a reference point, these signals will be captured by the acquisition board, processed through the controller to be finally converted into PWM signals generated by the Arduino.

As explained in the early chapters, the travel motion is directly linked to the roll angle, therefore, to control the movements of the 3DOF helicopter, it is enough to control the roll of the system and its elevation only. This means that the system can be considered in this case as a two degrees of freedom system since the travel angle can be observed by the user without the need of a sensor specified for this particular axis:

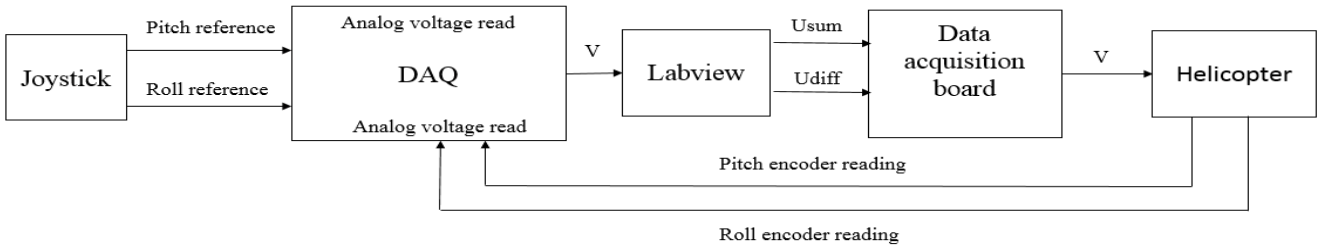

Figure 4.16 : Block diagram of 2DOF Control system using a joystick

4.4.2. Real-time 3D Model

One of the main goals when designing a flight control system, is to be able to display a real-time 3D model engine using correct coordinates that reflect the exact position of the system on the map.

The LABVIEW software's 3D helicopter package comes with a standard library for all the Quanser's 3DOF model, this library takes for input, the state space of the system and the angular readings which can be either generated using simulation or obtained by the encoders' readings.

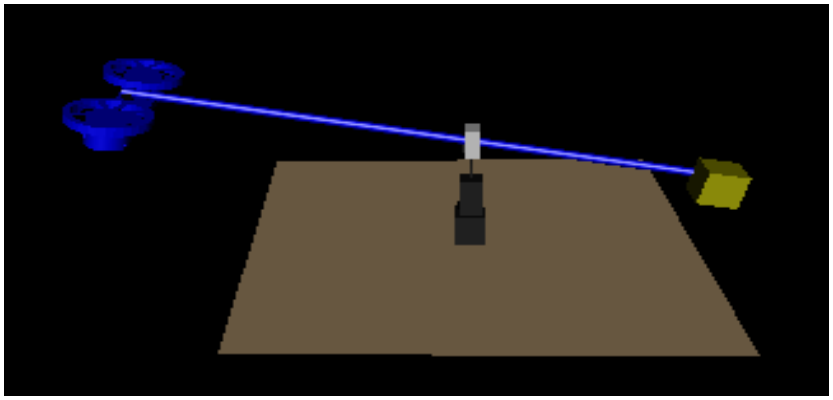

Figure 4.17 : 3DOF 3D Model

4.5. Conclusion

This chapter was the perfect background to understand the architecture behind the implementation of a controller from scratch.

Starting by designing simple input-output interface was a great opportunity to facilitate the understanding of the function of each component and put the derived equations into work.

This chapter dealt with the controller's fusion with the already established components and programs, while it also gave a degree of importance to certain design optimization adjustment by replacing the main controller by simpler control design.

In order to have better results, an observer design was put in place to reduce the effect of certain disturbances such as the wind effect and some of the physical irregularities in the symmetry of the design. Planting a Kalman filter allowed also the identification of some states such as the angular velocity that usually can't be observed in the absence of sensor.

Two example of how final control design of a 3DOF helicopter were proposed to validate the principles adopted in this chapter.

CHAPTER 5

Results and future plans

5.1. Introduction

Different estimation methods were adopted to approximate the real system to a healthy functioning mathematical model. As a result, it is critical to assess how well the developed model in the previous chapters is behaving in a similar way to the realistic model (nonlinear model).

Of course, a three degrees of freedom flight system is expected to be exposed to many physical constraints and disturbances due to the quality of the materials and the design adjustments. These difficulties will be explained by the end of this chapter where solutions and improvements will be suggested to deal with this type of obstacles.

5.2. Evaluation

5.2.1. Linear vs Nonlinear Open-loop

The Jacobian-linearized model is proven valid when operating around the equilibrium point, the following step-response is simulated to evaluate the linear pitch model when the system is operating at 10° further from the equilibrium point:

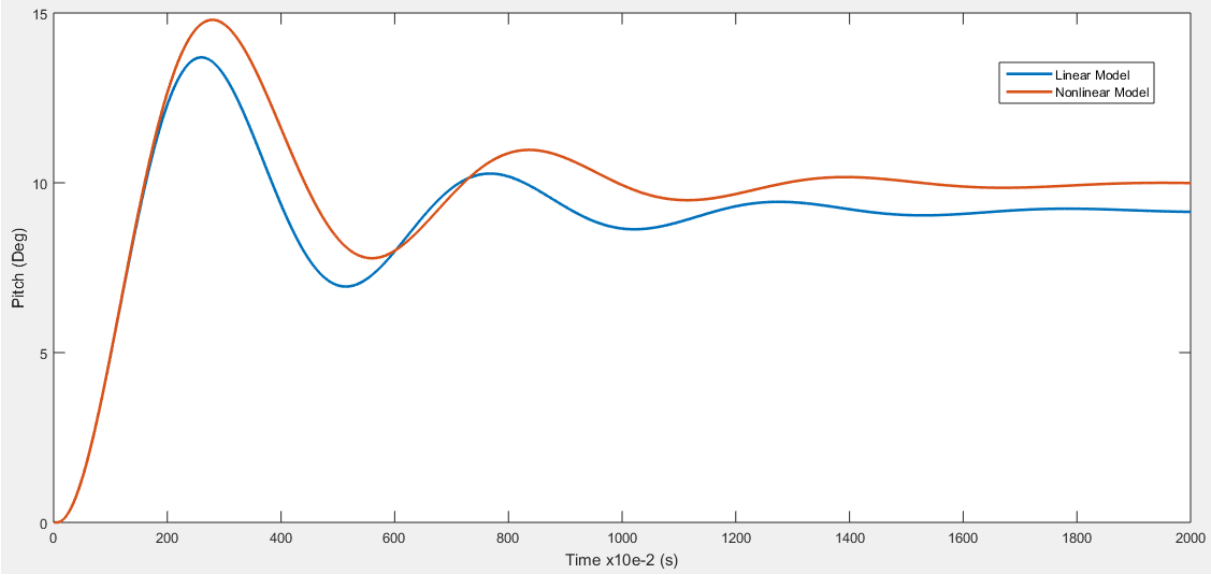

Figure5.1: Linear vs nonlinear time response for open-loop pitch system

The steady state error between the linear and nonlinear models can be calculated:

$$e = \frac{value_{real} - value_{estimated}}{value_{real}} \times 100\% \quad (5.1)$$

$$e = \frac{10 - 8.5}{10} \times 100\% = 15\%$$

This error was computed for a step response significantly large.

The linearized travel model can be tested using the same method, in the following simulation, the error of the estimated model doesn't exceed 5% of the nonlinear magnitude after travelling for a long period of time (20 seconds):

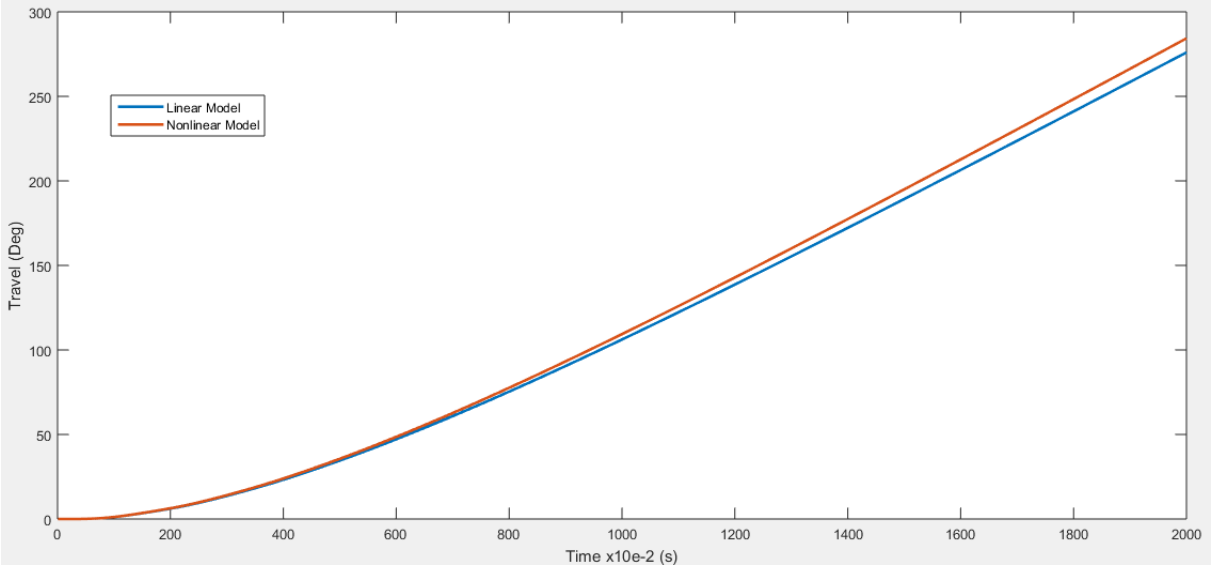

Figure5.2: Linear vs nonlinear time response for open-loop travel system

$$e = \frac{280 - 265}{280} \times 100\% = 5.35$$

5.2.2. Linear vs Nonlinear Closed-loop

The purpose of this part is to compare the effect of the controller on both the real and mathematical model and to check if whether or not the nonlinear model (real model) reaches the desired reference point.

A reference value of pitch=15° and travel=90° were given at instance time "t". The graphs shown on the next page display how the controller is affecting both models.

It can be seen that the real models follow the reference input and enter eventually the steady state successfully with small errors:

$$e_{Pitch} = \frac{15 - 13.6}{15} \times 100 = 9.33\%$$

$$e_{Travel} = \frac{90 - 86}{90} \times 100 = 4.44\%$$

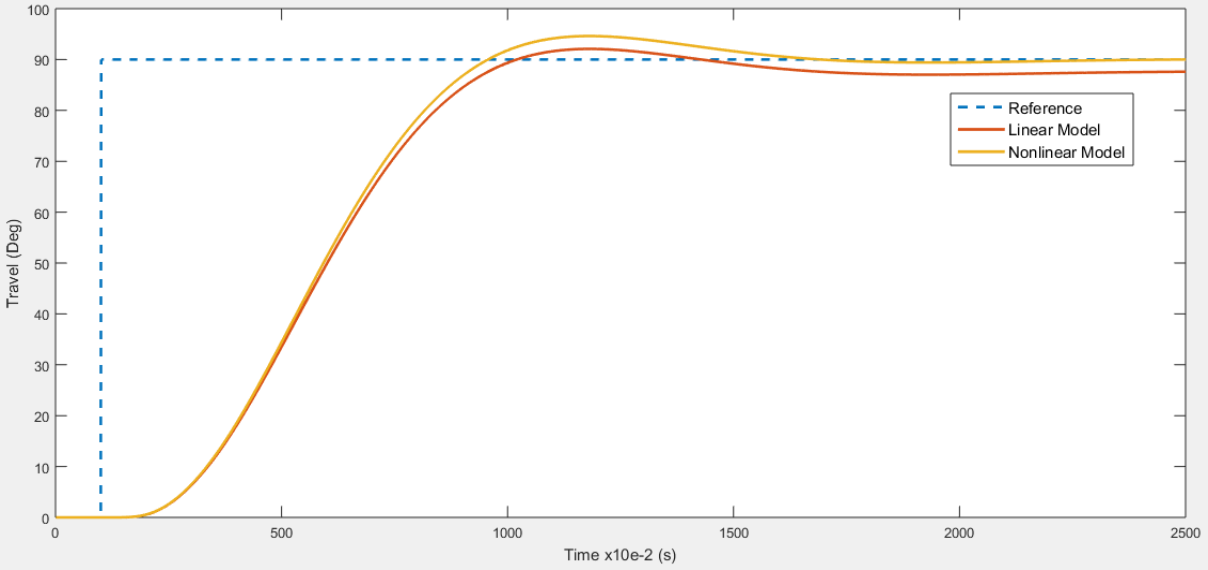

Figure5.4: Linear vs nonlinear with reference time response for closed loop Travel system

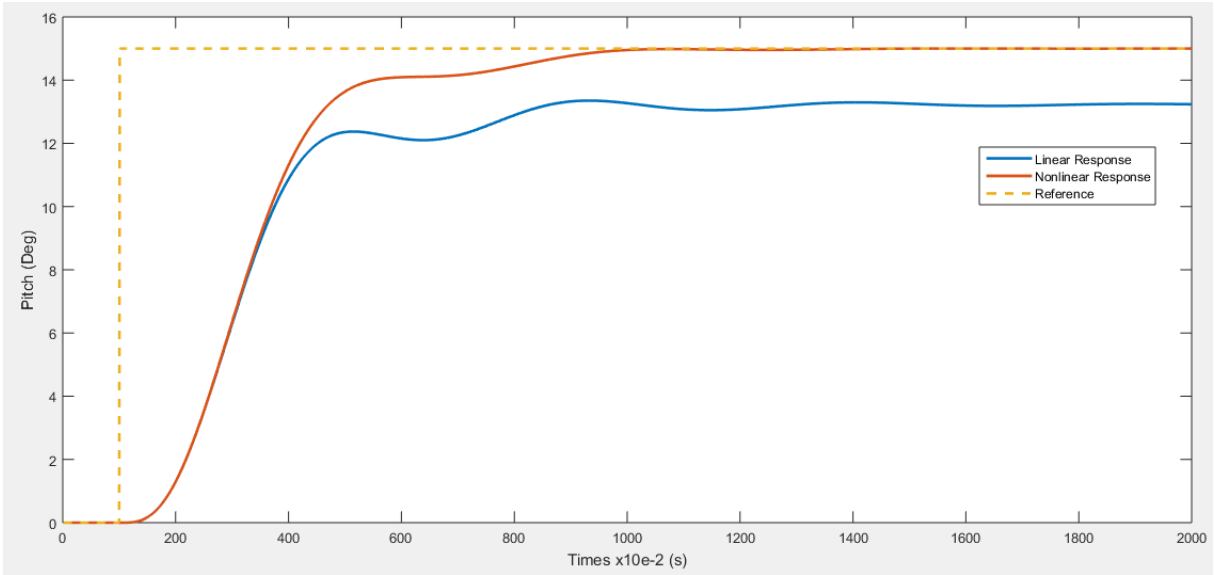

Figure5.3: Linear vs nonlinear with reference time response for closed loop pitch system

5.2.3. Real Time Response:

A real-time experiment has been done to test the validity of the controller:

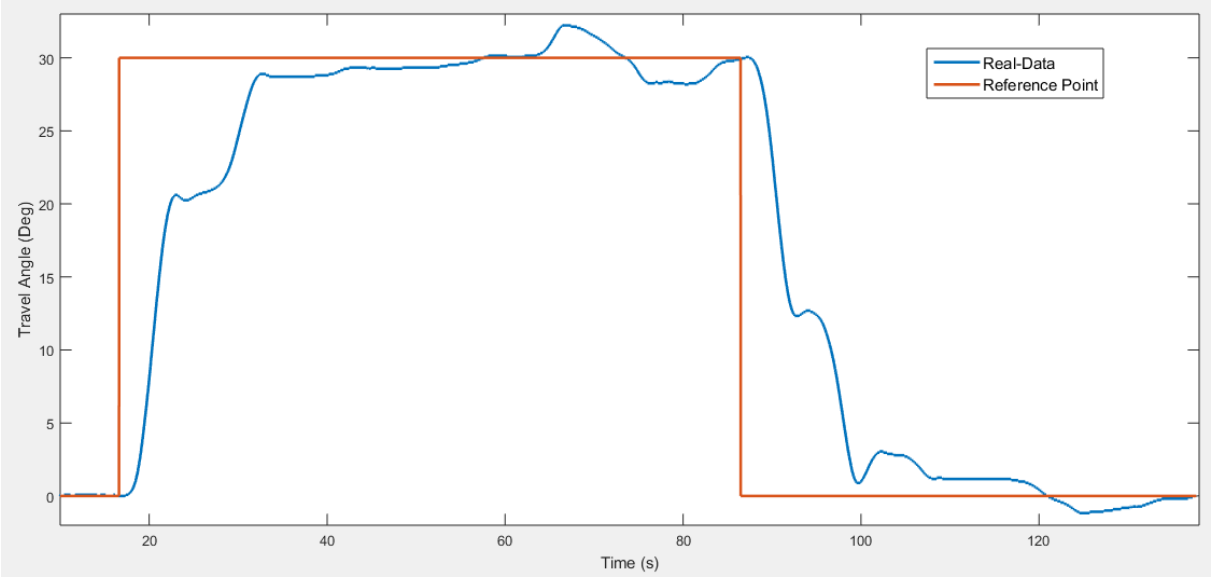

Figure5.5: Real vs reference time response for closed loop Travel system

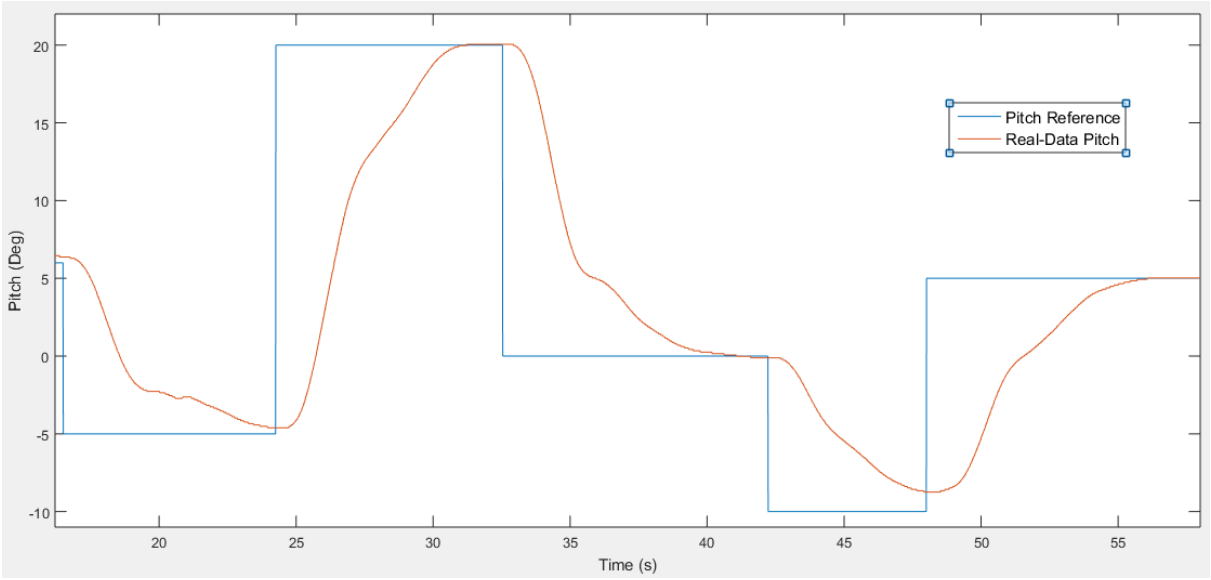

Figure5.6: Real vs reference time response for closed loop pitch system

As it can be seen, the plant tracks the reference points with high accuracy and with small overshoot amounts. This indicates that the controller can be put into work, however, it can be noticed that for the travel axis the plant not only takes a considerable time to reach the desired input, but it also doesn't follow a smooth path, the controller gains may be the reason behind such behavior.

5.2.4. Frequency Response of Linearized Model:

Now that the linearized mathematical model represents with high degree of accuracy the behavior of the nonlinear model, the state spaces obtained earlier with their respective frequency responses can be trusted for all operating ranges. The transfer function of the 3-axis can be deduced using the following relation:

$$G_{\mathrm{P}}(s) = \frac{0.4291}{s^2 + 0.5572\,s + 1.608}$$

$$G_{\mathrm{R}}(s) = \frac{3.868}{s^2 + 1.13\,s + 17.38}$$

$$G_{\mathrm{T}}(s) = \frac{3.553}{s^4 + 1.363\,s^3 + 17.64\,s^2 + 4.047\,s}$$

The behavior of these transfer functions can be determined by the famous bode plot responses:

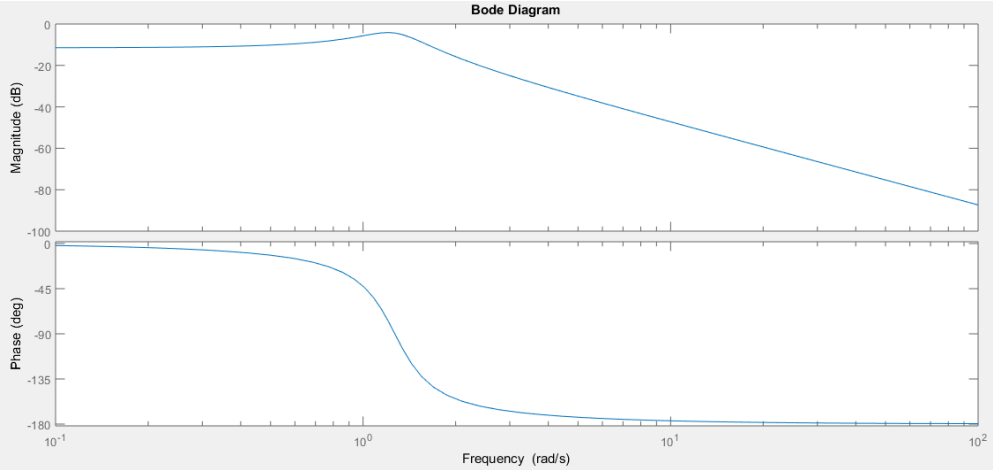

Figure5.7: Bode plot for pitch system

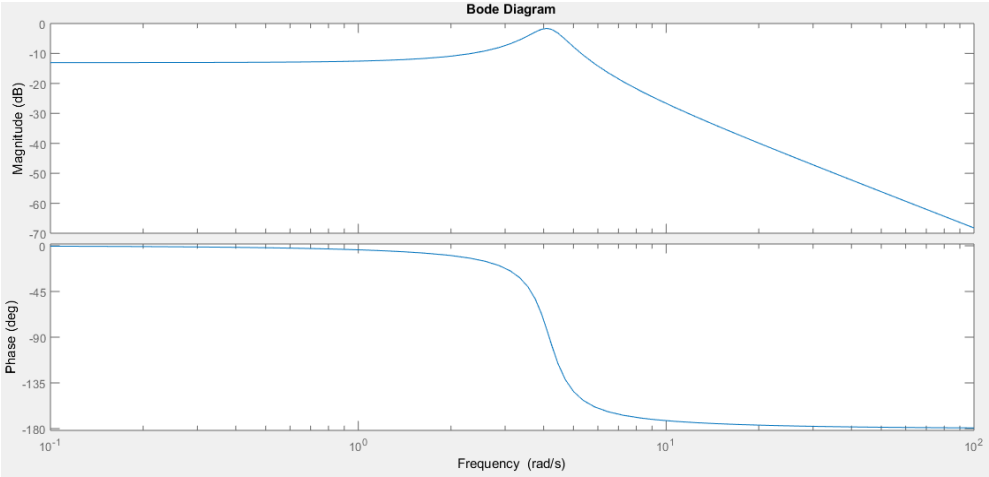

Figure5.8: Bode plot for Roll system

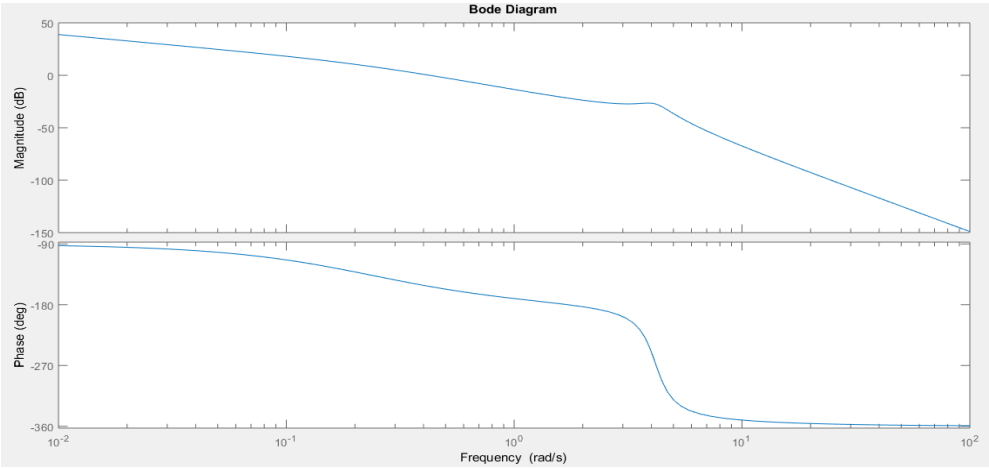

Figure5.9: Bode plot for Travel system

5.2.5. Kalman Filter Estimation vs Real Model:

The accuracy of the Kalman filter gains obtained can be tested for a step response joined with a white noise disturbance of a mean 0 and variance of 0.05N. The system is also coupled with noise applied on the sensors simulated by another white noise signal of a variance of 0.02 radian:

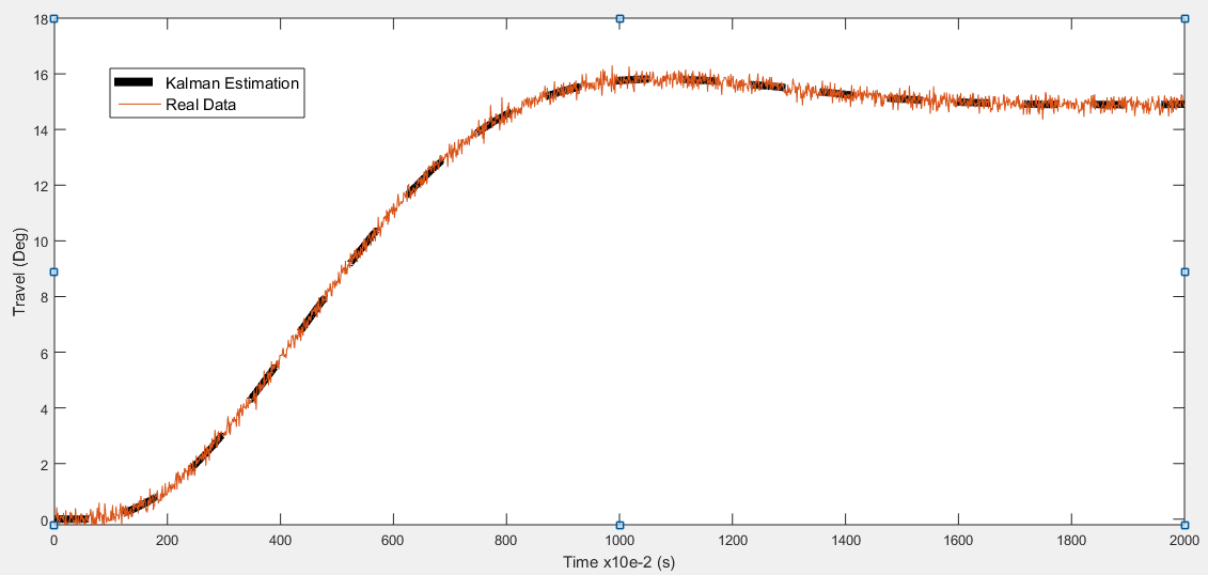

Figure5.10: Kalman estimation vs Real data for travel system

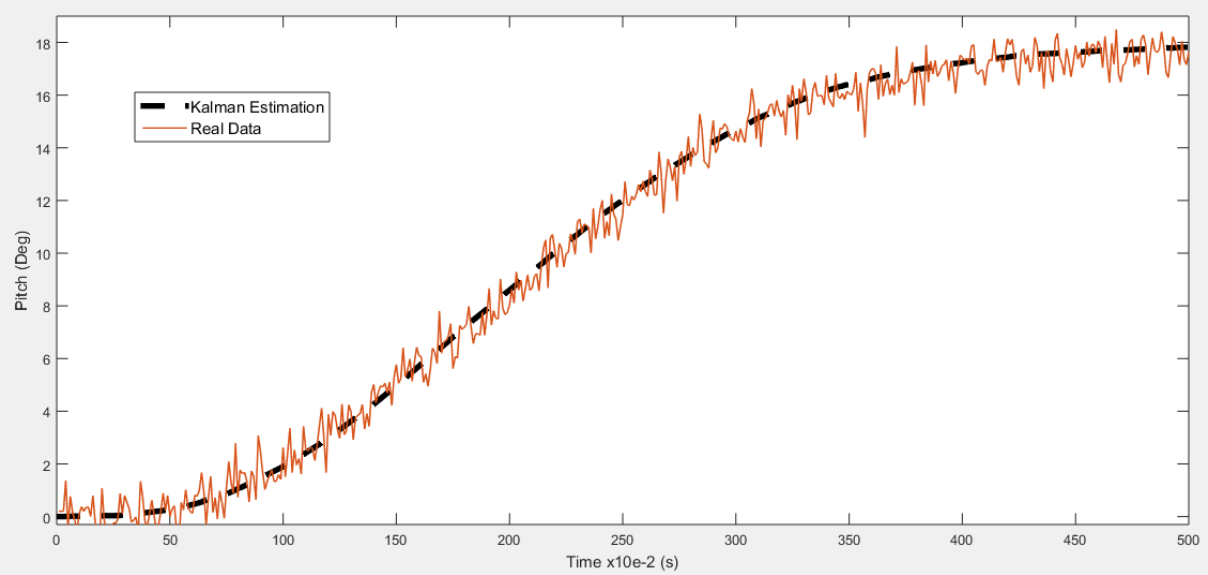

Figure5.11: Kalman estimation vs Real data for Pitch system

As expected, the Kalman filter regulates the experimental errors and returns a more realistic estimation of the data.

5.3. Difficulties and Improvements:

5.3.1. Physical constraints:

The 3DOF-helicopter used in this project is exposed to design limitations; this helicopter model needs a minimum voltage of 1.4 volts corresponding to 0.82 Newton to start floating in the air, any less voltage fed to the motors will result into the helicopter hitting the surface platform (the table) damaging the components.

The helicopter can't also exceed a pitch angle of 30° which corresponds to a motors' voltages of 2.8 volts or 2.27 Newton.

These two limitations can determine the operating range of the motors:

$$0.82 \leq u_{1,2} \leq 2.27$$

Where the thrust of each motor can be given by:

$$\begin{cases} u_1 = u_0 + \frac{u_{sum}}{2} + \frac{u_{diff}}{2} \\ u_2 = u_0 + \frac{u_{sum}}{2} - \frac{u_{diff}}{2} \end{cases} \quad (5.2)$$

Where: $u_0 = 1.28\ N$ is the necessary hovering voltage to reach the equilibrium point before activating the controller.

Applying the constraints on the thrust expressions:

$$\begin{cases} -(0.92 + u_{diff}) \leq u_{sum} \leq 2 - u_{diff} \\ u_{diff} - 0.92 \leq u_{sum} \leq 2 + u_{diff} \end{cases} \quad (5.3)$$

These two inequalities can be represented on the u_{sum} vs u_{diff} plane by the following region:

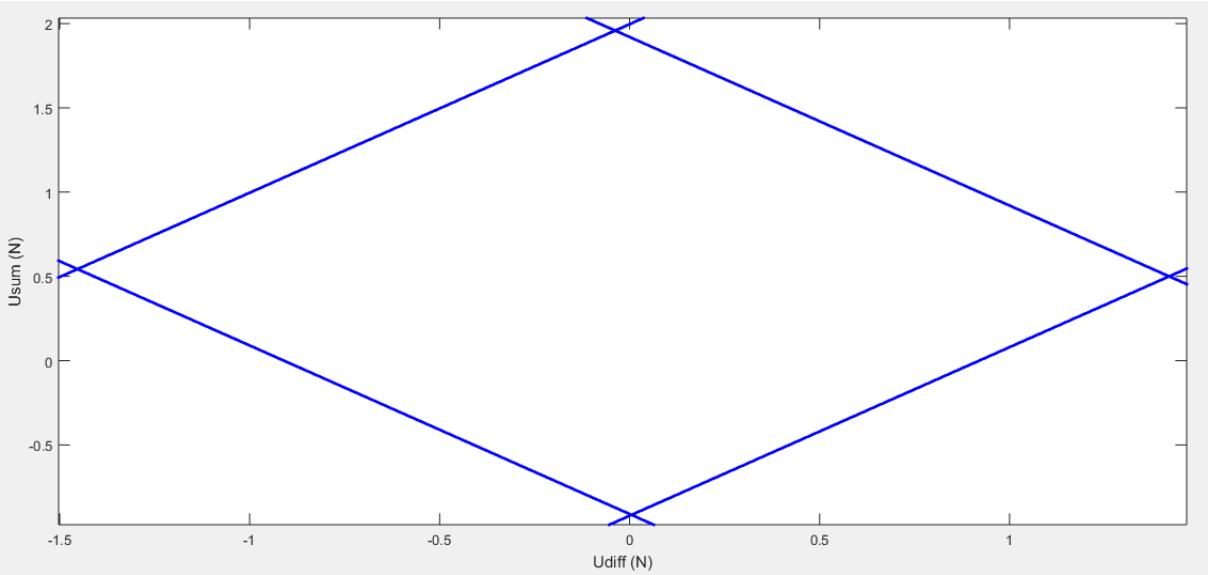

Figure5.12: Input limitations

The validity of the hypothesis can be tested for the following response:

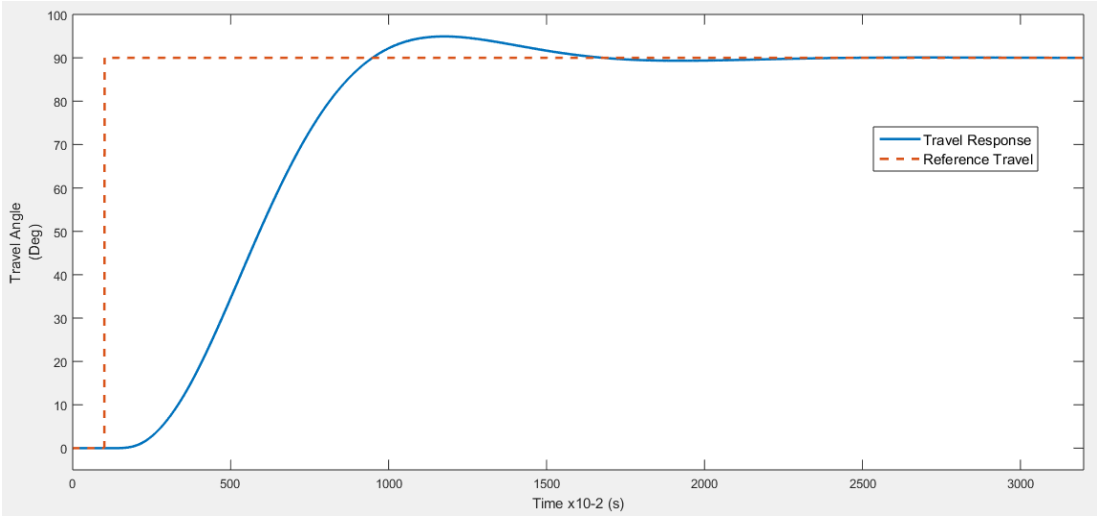

Figure5.13: Travel response vs Reference Travel

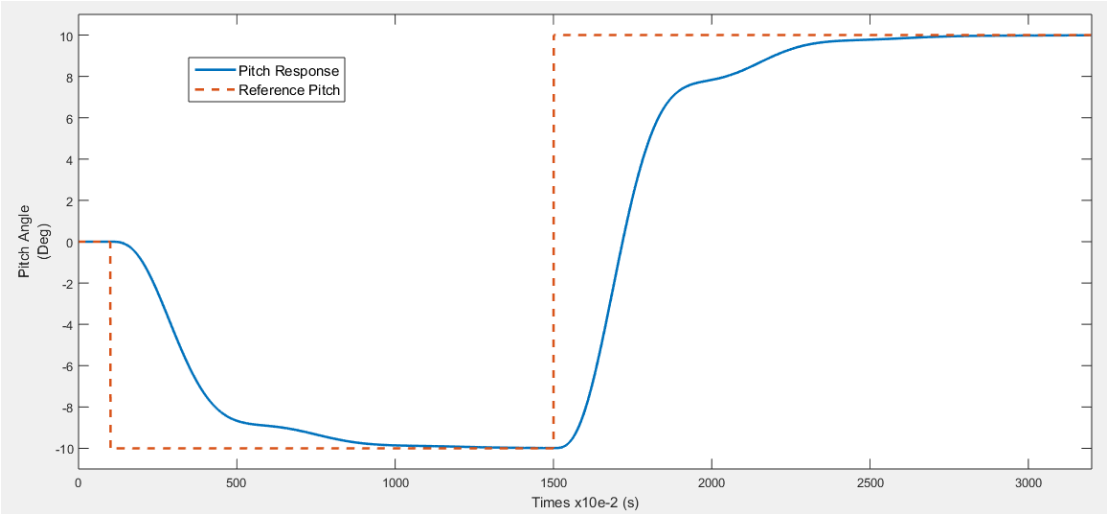

Figure5.14: Pitch response vs Reference Pitch

The plots of the BLDC motors' voltages indicate that the signals required to perform the simulation don't exceed the physical constraints:

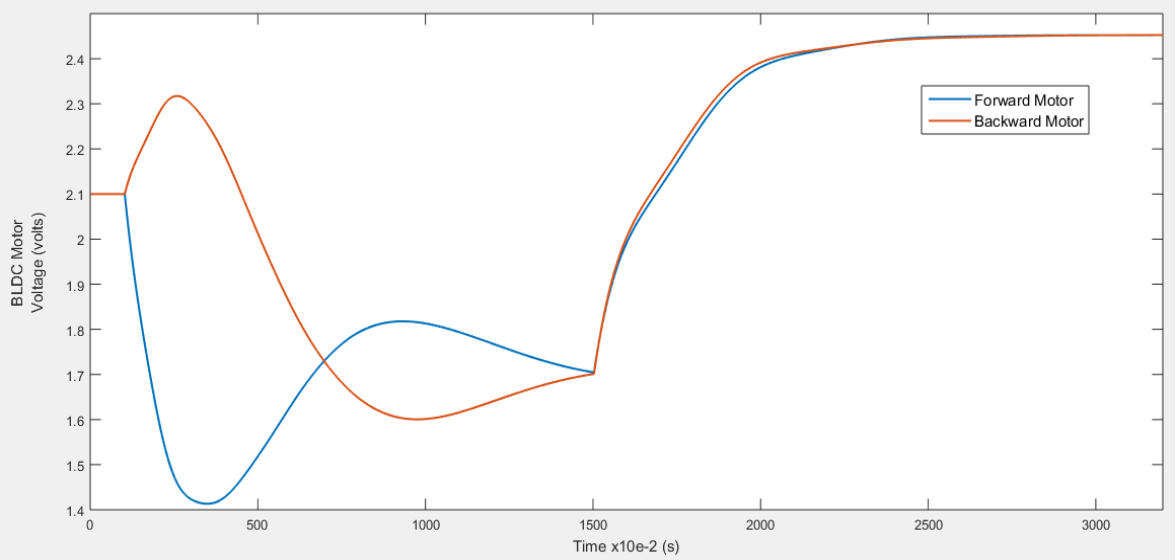

Figure5.15: Forward vs backward motor voltage

However, during the implementation, the controller kept turning off. When checking if the region of operation was respected, it can be seen that the controller applied more forces than allowed:

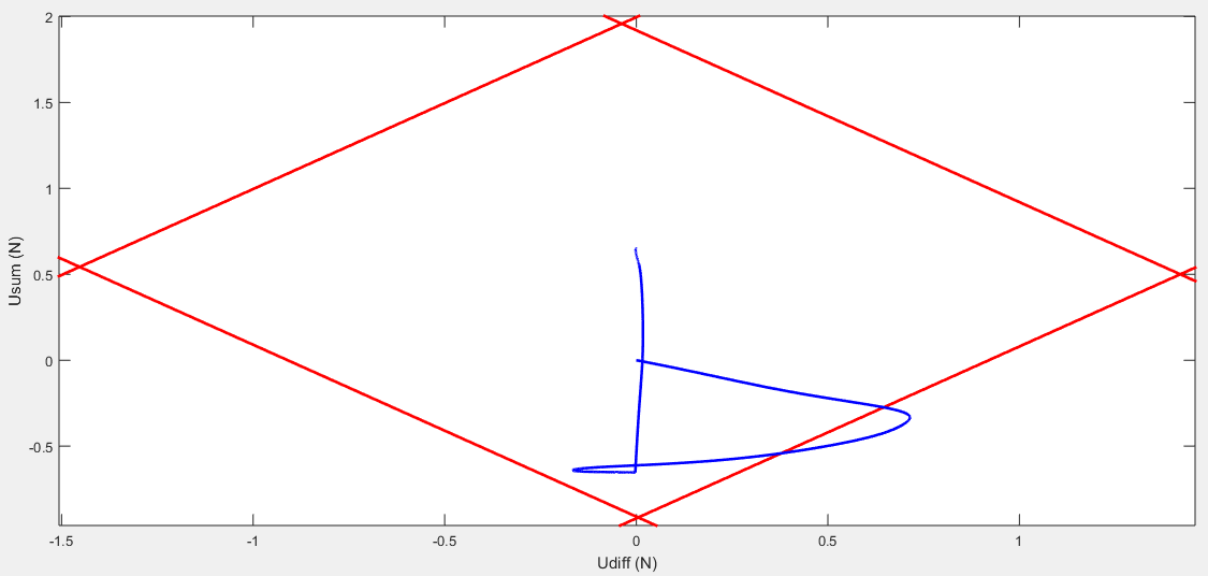

Figure5.15: Input limitations with respect to motor

5.3.2. Lack of Digital Counters

A counter is an essential part for the ideal functioning of an encoder. However, the data-acquisition board used in this project is supplied with 2 counters only, this can be a critical issue since the controller needs all the three states obtained by the three encoders. This means that the application of the axis-control of the helicopter can't be implemented in the absence of a 3rd counter in the used acquisition board (one can only connect two angular position encoders at once). This problem doesn't affect the course of the project since three solutions offer to the user:

a. Replacing the acquisition board with a different model that is supplied with three counters, however this could be very expensive financially and causing time management problems since all the programs need to be redesigned on a new setup.

b. A third solution can be suggested, if enough time is available, a customized counter can be designed to detect the pulses generated by the encoder using the digital input pins of the acquisition board.

c. Using the Arduino microcontroller as a 3rd counter: this will require an extra connections to interface the arduino with the Labview and the DAQ assistant, the VISA library is used on the Labview diagram to read the angular position directly from the Arduino's serial monitor:

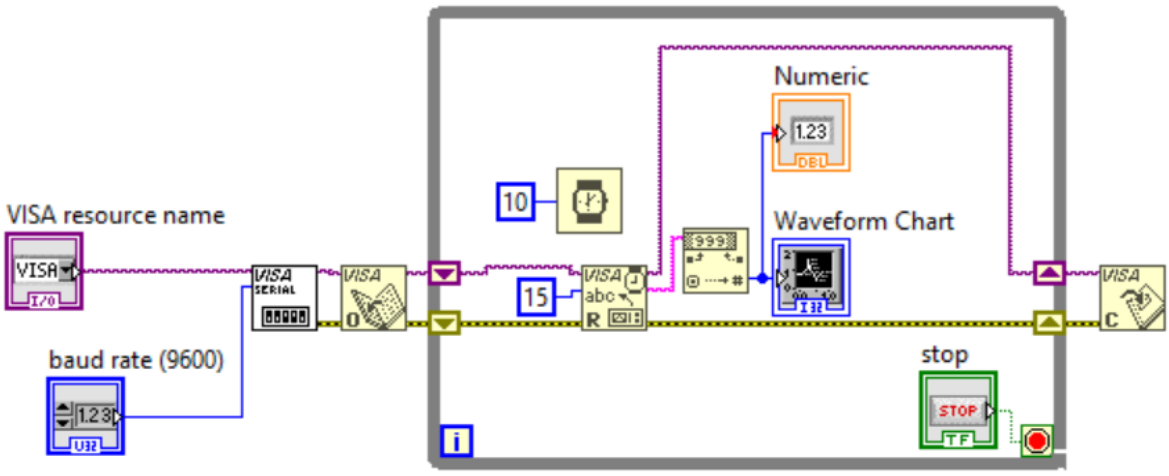

Figure5.16: counter implementation with LabVIEW

This method also has its inconvenient: the VISA program returns float data as integers which may decrease the precision of the controller, the following real time response describes perfectly this problem:

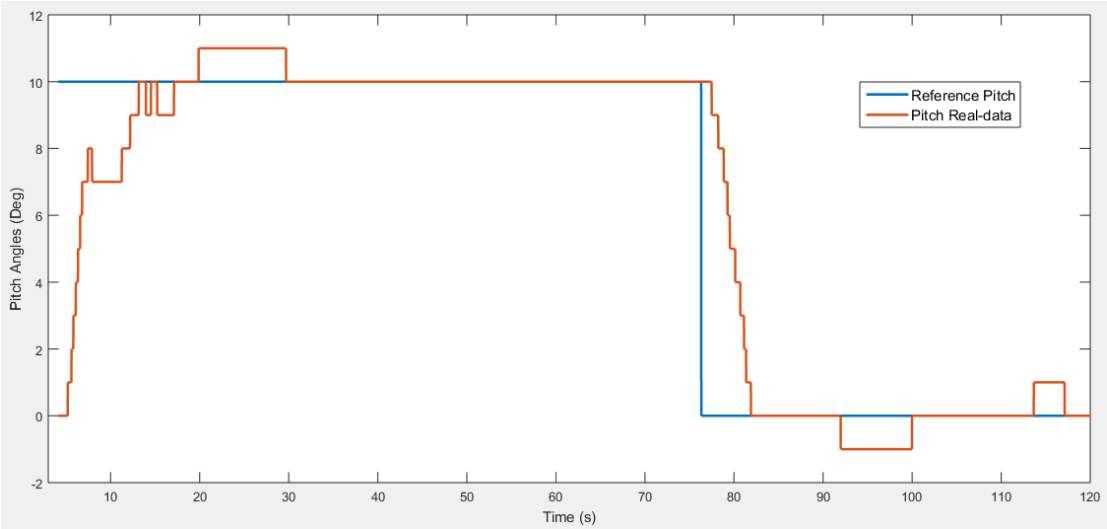

Figure5.17: Reference pitch vs pitch real data

Real-time systems should be as close as possible to the continuous data representation. Even though the physical system behaves exactly as desired, the data plot shows the limitations of this method.

Due to the time limitations faced in the implementation of this project, the Arduino using VISA method was selected in this realization for its simplicity.

Note: Working with pitch controller only, or the travel + roll controller at once doesn't cause a problem, since these two compensators are independent from each other.

5.3.3. Hardware protection

During the realization period of this project, multiple hardware problems have been encountered which resulted into damaging propellers and speed controllers. Designing a protection kit using the 3D software "SOLIDWORKS" helped in eliminating most of these technical problems:

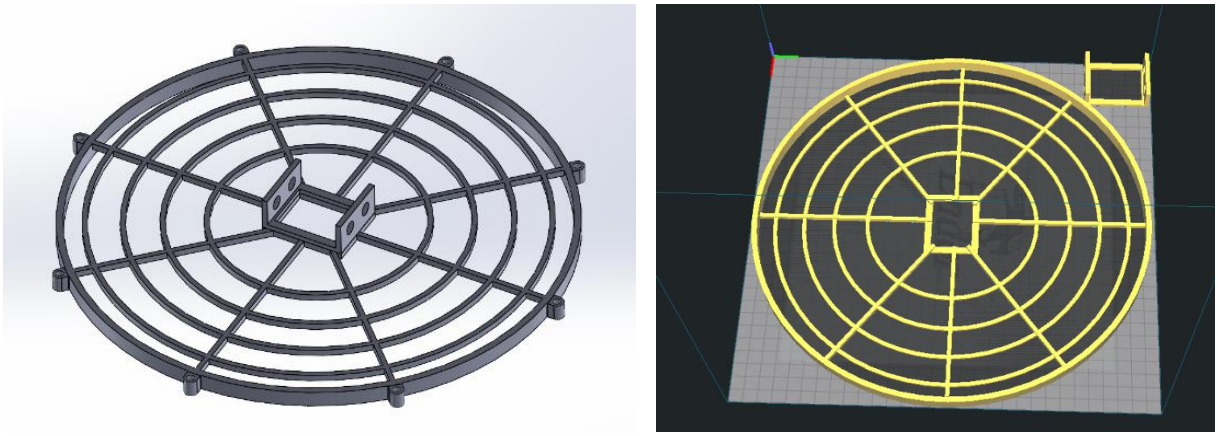

Figure5.18: Protection design with Solid works

Figure5.19: Protection implementation In lab

Keeping in mind that attaching the wings' protections will increase the total mass of the system and will block certain amount of air flow changing the overall characteristics of the system such as: the thrust relations of the motors and the inertia of all of the three axis.

5.4. Future Plans

- Observer state: the design of the Kalman filter in this project depends heavily on the estimation of the disturbance force and the magnitude of the noise in sensors. However, in real time, the flight system can face some unknown forces with high magnitudes that cannot be analyzed by the Kalman filter.
  The State observer on the other hand can estimate errors and disturbances for unknown inputs.
- Model Predictive Controller:

  The more the system is operating further from its equilibrium point, the less accurate is the input estimated by the controller gains.

  Another common problem faced when designing control systems is that it might be extremely difficult in certain cases to decouple the MIMO systems.
  The model predictive controller or the MPC[21] can process this type of difficulties. Since it requires usually very powerful microprocessors, the MPC can predict all the inputs of the system instantaneously to reach a certain reference using the future state trajectory of the system.

  Unlike the linear quadratic models, the MPC compensator always takes into consideration the imposed constraints on the system.

  It is clear now that the MPC controller can be a better design option for this flight system.

- 6DOF Helicopter: the goal behind this project so far is educative; the 3DOF helicopter can represent the perfect background for the students to implement the knowledge earned during their study years at the university into work, however, the 3DOF helicopter cannot be considered as a controlled flight system as long as it is not detached from the counterweight and the pillar that restraints its movement. Hence, the 6DOF helicopter project can offer more challenges and learning opportunities.
- Inertia measurement Unit:

  One of the main objectives of designing controllable flight systems is to track the aircraft and being able to visualize it on a 3D plane.

  The inertia measurement unit or the IMU sensor is a sensor capable of converting the applied forces into linear acceleration and angular velocity which can be used to estimate position of the object attached to IMU.
- Remote Control System: one of the main problems that occurred a lot during this project was the sudden shut down of the system when one of the wires is disconnected, this issue can be completely eliminated

if all the hardware interface is replaced with remote control signals that link the microprocessor (Arduino Board) to the computer device.

5.5. Conclusion

This chapter accorded a certain amount of accuracy and precision to the overall designed control system by comparing and simulating the performance of the estimated model to the real one. This step was followed by the test of the real time system allowing the diagnosis of the problems and issues that are disturbing the ideal performance of the system.

The second part of this chapter not only suggested solutions to these problems, but it also opened the door for future work-plan and ideas that can be implemented to the current state of the helicopter.

General Conclusion

Working on flight control systems represents a great field of opportunities due to the polyvalence of such projects by offering numerous challenges in the different backgrounds: theoretical automation and control background on one hand, and robotics and mechatronics hard skills on the other.

Entering the world of industrial machines, vehicles and aircrafts always starts with working for a long time on small prototypes in order to earn maximum experience in the concerned field of study. The 3DOF helicopter concept introduced by Quanser is the perfect representation of tandem aircrafts.

This project merged the world of theory that is often represented in the engineering domain through simulations with the world of practice and experiment by modelling the 3DOF helicopter from scratch and slowly building up its complete system.

Projects with the aim of implementation usually go through three main phases: identification, design and diagnosis whom were explained in details throughout the chapters.

The second chapter is a collection of different methods and techniques with the purpose of decorticating a specific system and identifying its behavior by visualizing a mathematical model and defining all the factors and variables acting on the system. The power and efficiency of the modern softwares can showcased in this chapter, by computing and solving nonlinear differential equations in short periods of time with high accuracy degrees.

In the third and fourth chapter, a fully detailed workplan of the adopted roadmap during this project has been explained by first deducing a linearized mathematical model using indirect approaches such as the Jaobian method. Then, a pole placement based model controller was designed to achieve the desired performance, however, the obtained results can still be improved implementing an estimation state. Among all the offered estimation designs, a Kalman filter was selected to work with due to its high capacity to deal with noise and disturbances, and its ability to generate accurate measurements.

All the designed and implemented concepts have been proven correct in the last chapter after evaluating the real-time situations with the simulated experiments performed earlier. However, since this project is a merge of simulation and implementation, it was expected to be facing more problems due to the hardware limitations which limited in some occasions, the from achieving its full potential.

Bibliography

[1] KieferT,GraichenK,KugiA(2010)Trajectorytrackingofa3DOF laboratory helicopter under input and state constraints.IEEETrans Control Syst Technol 18(4):944–952

[2] Xiaoxiao Zhao. The 3 Dof Helicopter Control System Based on PID Contmller. Joumal of Shandong Electric Power College. Vol.12 No.4, pp. 52-55.

[3] Trajectory Tracking of a 3DOF Laboratory Helicopter Under Input and State Constraints Thomas Kiefer, Knut Graichen, and Andreas Kugi, Member, IEEE

[4] LQR Based PID Controller Design for 3-DOF Helicopter System Santosh Kr. Choudhary

[5] “Tandem rotors,” [Online]. Available: https://en.wikipedia.org/wiki/Tandem_rotors

[6] "US Army FM 3-04.203 Fundamentals of Flight; page 1-33, para 1-81, 1-82"

[7] Principles of helicopter aerodynamics by J. Gordon Leishman p. 73.

[8] G. Staples, “Propeller static & dynamic thrust calculation,” Electric RC Aircraft Guy, 2013. [Online]. Available: https://www.electricrcaircraftguy.com/ 2013/09/propeller-static-dynamic-thrust-equation.html.

[9] “ODE45,” [Online]. Available: https://www.mathworks.com/help/matlab/ref/ode45.html?fbclid=IwAR1WbhTNdt1nmW1EOB-mZq3u4EcWMEGWasnn0lLSg1M-QIVA9dOIgTLwqT0

[10] “Fmincon,” [Online]. Available: https://www.mathworks.com/help/optim/ug/fmincon.html?fbclid=IwAR0LWOBtJvJXIqj3RhGeZJyfMSEFkwzf1P04kuFd83Xa6c4r8gJDZRY5lsg

[11] Zanasi Roberto - System Theory, Luc_TDS_2016.dvi. Available: unimo.it

[12] Model Reference Adaptive Control and LQR Control for Quadrotor with Parametric Uncertainties Alia Abdul Ghaffar, Tom Richardson

[13] Dynamic Performance Comparison of LQR and LQI Controllers on Buck Converter by O. Fatih Kececioglu

[14] Full-State Observer Notes and Example Modern Control Theory, 3rd Edition, William Brogan, Prentice Hall, 1991, pp. 532–533

[15] “Linear-Quadratic-Gaussian (LQG) Design” [Online]. Available: https://www.mathworks.com/help/control/getstart/linear-quadratic-gaussian-lqg-design.html.

[16] “How brushless motor works”, [Online]. Available: https://howtomechatronics.com/how-it-works/how-brushless-motor-and-esc-work/

[17] “How ESC works”, [Online]. Available: https://howtomechatronics.com/how-it-works/how-brushless-motor-and-esc-work/

[18] “Arduino,” [Online]. Available: https://www.arduino.cc/

[19] “Arduino brushless motor control”, [Online]. Available: https://howtomechatronics.com/tutorials/arduino/arduino-brushless-motor-control-tutorial-esc-bldc/

[20] Model Reference Adaptive Control and LQR Control for Quadrotor with Parametric Uncertainties Alia Abdul Ghaffar, Tom Richardson

[21] A Lecture on Model Predictive Control Jay