MANCHESTER 1824
The University of Manchester

1-DOF and 2-DOF Drone Control

Third Year Individual Project – Final Report

April 2023

Mahmoud Shanan

10614473

Supervisor: Dr Long Zhang

Contents

Word count: 8739

Abstract

This paper uses a hardware-in-loop platform (Quanser Aero 2) used in research and development for designing control systems of UAVs. The aim of this paper is to design a cascade PID controller to control the pitch and yaw angles of both configurations of the Aero 2, the 1-DOF bi-copter and 2-DOF helicopter, and evaluate the feasibility of developing a new 2-DOF quadcopter configuration. Unlike the helicopter configuration, it was not possible to control both pitch and yaw simultaneously in the bi-copter configuration. Pitch control in both configurations was almost identical, with fast time response, low steady-state error, and small oscillations. For the bi-copter yaw control, time response was thirty times slower than the helicopter as both rotors were horizontal compared to the helicopter where one rotor was vertical. Finally, mathematical and mechanical approaches were proposed to attempt controlling the system as a quadcopter, yet they were both unsuccessful. This implied that the Quanser Aero 2 could not be controlled as a quadcopter.

Declaration of originality

I hereby confirm that this dissertation is my own original work unless referenced clearly to the contrary, and that no portion of the work referred to in the dissertation has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Intellectual property statement

i. The author of this thesis (including any appendices and/or schedules to this thesis) owns certain copyright or related rights in it (the "Copyright") and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.

ii. Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made **only** in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.

iii. The ownership of certain Copyright, patents, designs, trademarks and other intellectual property (the "Intellectual Property") and any reproductions of copyright works in the thesis, for example graphs and tables ("Reproductions"), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.

iv. Further information on the conditions under which disclosure, publication and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see http://documents.manchester.ac.uk/DocuInfo.aspx?DocID=24420), in any relevant Dissertation restriction declarations deposited in the University Library, and The University Library's regulations (see http://www.library.manchester.ac.uk/about/regulations/_files/Library-regulations.pdf).

1 Introduction

1.1 Background and Motivation

Unmanned aerial vehicles (UAVs), commonly known as drones, are aircrafts that operate without the presence of a human pilot. They are often controlled by a person on the ground or autonomously following a planned path, and are fitted with numerous sensors and cameras such as gyroscopes, magnetometers, accelerometers, and GPS, allowing them to navigate in their environment and collect data.

The development of drones dates back to 1849, when the Austrian army sent several uncontrolled hot air balloons filled with explosives over Venice [1]. Today, balloons equipped with navigation systems are widely used by countries like China and the United States for spying and meteorological purposes [2]. With the rapid research and development of UAV technology in the 20th century, the first drones were allowed to come to market, primarily for military purposes [3].

Nowadays, drones are nearly used in all the industries including cinematography, agriculture, climate monitoring, industrial inspection, wildlife monitoring and soon, product delivery. This has been possible to achieve due to the incredible breakthrough in microprocessors, sensors, motors and LiPo batteries. Most importantly, developing robust flight control systems has ensured the stability, manoeuvrability, and performance of UAVs when operating in complex and dynamic environments.

Generally, the flight control system of a drone processes the data captured by the sensors such as orientation, speed, and altitude. Then, it generates control signals to adjust the different parameters of the UAV accordingly. It should be able to handle unexpected circumstances and nonlinear behaviour, such as turbulence, sudden wind blowing and changes in orientation.

While numerous types of UAVs exist today, there is still a need for developing new designs to improve efficiency and functionality. New designs can improve stability, payload capacity and flight times. They can also allow UAVs to fly in narrow spaces, making them suitable for more applications.

Finally, plants and power stations often have automated processes that require constant monitoring and examination. Some faults can sometimes occur in these systems resulting in failures interrupting the production processes. Thus, finding and developing new efficient and safe

methods for fault inspection is essential. It could be dangerous and expensive to deploy human workforce to perform these types of tasks, which makes UAVs a good alternative [4].

1.2 Aims and Objectives

The main aim of the project is to model and control a 2-DOF quadcopter using two 1-DOF bi-copters. Three objectives were set.

The first objective is to study the dynamics of the 1-DOF Quanser Aero 2 system, and design a PID controller to control pitch, and yaw of the system. A mathematical model would be created and validated according to the real system. The system performance of the simulation and hardware would be compared while also looking at the effect of different control parameters.

The second objective is to use the 2-DOF configuration of the platform to test the performance of the PID controller in controlling pitch and yaw simultaneously. A mathematical model would also be derived from the system dynamics and compared to the hardware. The coupling between both rotors of the system would be analysed to understand how each rotor influenced the other.

Having understood the dynamics of the Aero 2, the last objective is to use the mathematical model of the 1-DOF bi-copter to attempt modelling a 2-DOF quadcopter using two Aero 2 platforms. The feasibility of controlling both pitch and yaw of the system simultaneously using PID control would then be evaluated.

2 Literature Review

2.1 Popular Drones

There are currently two main types of drones; multi-rotor and fixed-wing. Each type is used in different situations and industries. As each design has its own limitations, hybrid UAVs or fixed-wing VTOL UAVs are also currently under research and development in order to maximise the performance of the drone and increase its functionality.

Multi-rotor drones are available in different sizes and configurations. Relatively small ones are mainly used by civilians for aerial photography and videography, while the larger are more used for industrial purposes, such as inspection of buildings and wind turbines, and monitoring of crops. The main advantage of multi-rotor drones is their ability to take-off/land vertically almost anywhere, allowing them to reach dangerous areas. They can also hover (standstill in the air), and

fly in any direction, giving them precise positioning and manoeuvrability. However, they have limited flight time due to high power consumption [5].

Quadcopters are the most widely used drones due to their low cost, high stability, and simple control system. They consist of four rotors spaced equally by 90 degrees. They could be arranged in either a plus (+) or cross (x) configuration, yet the latter is more popular and used in many applications due to its superior stability [6].

Tricopters are another type of VTOL UAVs, composed of three rotors equally spaced by 120 degrees from each other. Their main advantage is their small size, lightweight, and long battery life. They are occasionally used for cinematography, but their stability is relatively poor and could be dangerous in case of a rotor failure [6].

Other types, such as hexacopters and octocopters, have six and eight rotors, which provides them with even higher stability than quadcopters, and can carry heavier payloads. Nevertheless, they are more complex to develop, larger in size, and require larger batteries due to their high-power consumption [6].

Fixed-wing drones are mainly used in military applications due to their long flight range, high speed, ability to carry heavy loads and large areas coverage. Nevertheless, they require a runway for take-off/landing and additional skills to control. They could also be used in aerial surveillance and cargo delivery [5].

Hybrid UAVs aim to combine the advantages of both types of drones, limiting the drawbacks of each type. They can take-off/land vertically like multi-rotor drones, as well as fly for long distances, and carry heavy payloads like fixed-wing UAVs. Yet, it is more complex and expensive to develop this design [5]. A current example of a hybrid UAV is the Bell Boeing V-22. Tiltrotor technology allows this military UAV to rotate its rotors either vertically (VTOL) or horizontally for high-altitude flight [7].

2.2 Control Methods

Drones can be controlled using different approaches, such as manually, semi-autonomously and autonomously. Most UAVs have a flight control algorithm stored in a microcontroller allowing it to read captured data from sensors and cameras, and send control signals to the motors to perform the required manoeuvres.

2.2.1 Open-Loop Control

Open-loop control is a non-feedback controller where the system input does not depend on the output, the control action is determined by the human manually operating the system. The implementation of open-loop control is simple and cheap, and could be successful in basic applications. However, it requires experience and significant skill as a pilot to fly drones manually, as fast changes could occur in the environment, which requires instant reaction [8].

Semi-autonomous systems are also open-loop systems operated manually, but instead some parameters are controlled autonomously to facilitate control. The drone operator still controls the system and gives it certain commands, but the onboard control algorithm ensures that the system is behaving as expected and following the correct path. For instance, in offshore applications like wind turbine inspection, keeping the drone at a fixed altitude and distance to the blades is important to capture high quality data. Due to the nature of this environment, some disturbances are created by heavy wind blowing, rain and birds, which will be difficult to overcome manually. Thus, semi-autonomous systems could ensure constant speed, altitude, distance to blades, pitch, yaw, etc, and shift the focus of the operator to the actual task [8].

2.2.2 Closed-Loop Control

Closed-loop control is a feedback controller where the control input to the system is determined by the current state. The sensors onboard provide real time data on the current position and orientation of the drone, then it is compared to a reference or required state. Depending on the error and controller gains, the input control signal is sent to the motors.

PID or Proportional-integral-derivative control is the most used control algorithm of today's systems, due to its simple structure and implementation. It is a closed-loop control system using feedback, which is the error between a desired target and the measured value, and performs an adjustment using the proportional, integral, and derivative gains (k_P, k_i, k_d). The general mathematical expression for a PID controller is:

$$u(t) = K_p e(t) + K_i \int_{-\infty}^{t} e(\tau)\, d\tau + K_d \frac{de}{dx}(t) \qquad (\mathbf{1})$$

$u(t)$: Control Signal $\quad e(t)$: Error

When the proportional gain K_p is increased, the overshoot of the system response is increased while the rise time and steady state error are decreased. It is related to the current error and gives

a faster system response, but if too large, overshoot will occur and the system will become unstable.

When the integral gain K_i is increased, the overshoot is also increased, and the system response is slowed down, but the main advantage is that the steady state error could be eliminated. However, if it is too high, the system could become underdamped and oscillatory.

The differential gain K_d is generally used to decrease the system overshoot and settling time, making the system response more stable. It could also decrease the steady state error and make the system less oscillatory. Nevertheless, it is easily affected by noise and is not ideal for fast response systems. Also, It often requires a filter such as Kalman filter to get a smooth output [9].

It was observed that PID control compared to other control methods such as Neural Network (NN) or direct inverse control neural network (DIC-ANN) does not perform well in non-linear systems. For example, for drone altitude control, PID control seemed to have more steady state error at hovering and had slower settling time than the DIC-ANN [10].

Fuzzy logic is also another popular control method, which consists in modelling logical reasoning where the truth of a statement is not binary. It is generally used in non-linear and complex systems which their mathematical model is not easy to derive and express explicitly, unlike popular control methods. Fuzzy logic allows the design of a fuzzy interference system, a form of artificial intelligence, which is less based on mathematical models but more on human-interpretable rules that guide the mapping of a set of inputs to outputs. A fuzzy logic controller is formed through three main parts: Fuzzifier, Rule base and Defuzzifier.

The fuzzifier is the first step in the process which consists of transforming the numerical inputs into fuzzy variables or membership functions. The rule base contains a set of If-statements that relate between the input and output. Finally, the defuzzifier will convert the result back into a numerical control output signal [11].

2.3 UAV Control Review

A research paper conducted at the Islamic University Malaysia used a 3-DOF bench-top helicopter to compare between the performance of fuzzy logic and PID controllers in UAVs [11]. The controllers were compared according to their performance in controlling the elevation angle of the system. The closed-loop system was designed and simulated using MATLAB/Simulink. Although it was found that both controllers had zero steady-state error, fuzzy logic controller had

faster settling-times and smaller overshoot percentages, which makes the fuzzy logic controller more robust. However, the paper only compared the ability of both controllers to track three degrees of elevation (4°, 5°, 6°), which was a relatively limited range, and did not show how the controllers behave in larger or smaller degrees of elevation. Moreover, the dynamics and mathematical model of the system were not shown.

Another research paper published at Beihang University used a bi-copter to design a PID altitude controller [12]. Unlike a typical 1-DOF bi-copter where only pitch or roll (depending on choice of axis) is controlled [13], this bi-copter had 6 degrees of freedom with only 4 input controls. While the difference in speed between both rotors controlled lift, yaw and roll, the axes on which rotors were fitted were actuated by two actuators, allowing the rotation of the propellors and pitch control. It was shown that the PID controller was able to track pitch and yaw angles, yet the system showed overshoot, slight oscillations, and a relatively small steady state error. Nevertheless, the performance of the controller in controlling roll and altitude were not mentioned.

Finally, the reviewed research papers showed how bi-copters and helicopters performed with PID and fuzzy logic controllers. It was found that fuzzy logic controllers performed better than PID controllers for elevation tracking in helicopters. Despite some limitations, pitch and yaw control were also successful with PID control in bi-copters. In this paper, a configurable UAV platform (Quanser Aero 2) would be used to compare between different UAV configurations (bi-copter and helicopter) in terms of PID control performance, and whether this platform could simulate a quadcopter.

3 Closed-Loop Control for Quanser Aero 2

This section introduces the hardware and software structure of the Quanser Aero 2 platform and its different configurations. It then analyses the dynamics of each configuration starting from the simplest configuration to the most developed, and explained the controller design.

3.1 System Introduction

3.1.1 Hardware Structure

The Quanser Aero 2 is a fixed platform used in research and development of control systems for UAVs. It consists of two brushless DC motors and rotors generating thrust allowing the system to

tilt up and down (pitch), or turn right and left (yaw). It is also equipped with four encoders measuring angular velocities of both motors, and pitch and yaw movements. For orientation tracking, a triaxial gyroscope, accelerometer and magnetometer are also fitted inside the platform. The interface between the Aero 2 and a host computer running the Quarc software is then managed by the data acquisition system, allowing the system to send real time data, and receive control signals.

The Aero 2 has two known configurations: half-quadrotor (1-DOF VTOL) and 2-DOF helicopter. In the bi-copter configuration, both propellers were facing upwards as shown in Fig. 1. Due to the dynamics of this layout, any speed difference between both rotors created a change in both pitch and yaw angles, thus the pitch for instance could not be adjusted without impacting the yaw and vice versa. Nevertheless, as the system is reconfigurable, the pitch or yaw axis could be locked in order to control one parameter individually (1-DOF). In the helicopter configuration, one propeller is rotated by ninety-degrees such that pitch and yaw movements could be controlled simultaneously (2-DOF).

The possibility of a third arrangement was also evaluated: 2-DOF quadcopter. By putting two Aero 2 platforms together in the bi-copter layout, it could be possible to model the system as a cross-configuration quadcopter. Several solutions have been studied and tested (section 4.4).

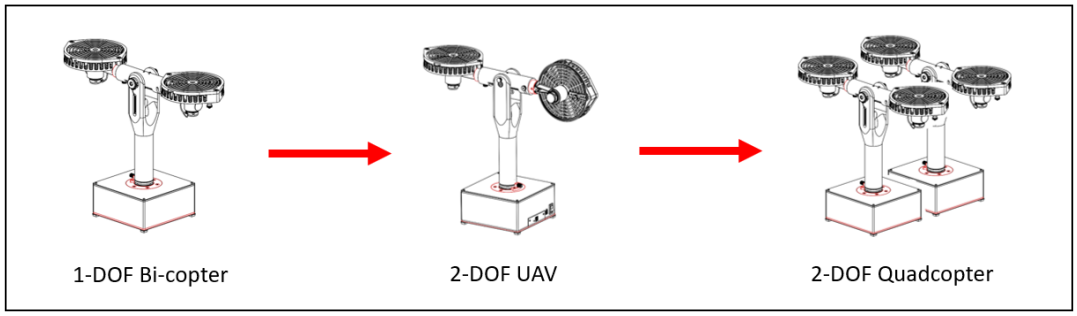

Fig. 1. Quanser Aero 2 configurations [14]

Finally, all physical parameters of the Quanser Aero 2 used in modelling and simulation were found in the datasheet and user manual, and were presented in the Appendices.

3.1.2. Software Structure

The Quanser Aero 2 uses hardware-in-the-loop simulation (HIL) allowing real-time processing and simulation. Quarc software was used to manage the interface between the platform and the host computer. For modelling and simulation, MATLAB/Simulink were used. Quarc added additional

functions designed specifically for the Aero 2 in Simulink. The main three blocks used were HIL Initialise, HIL read and HIL write. The initialise block was used to choose the correct board and determine some parameters like clock frequencies. The read block was used to get real-time data from the different sensors on board such as pitch and yaw angles, rotor speeds, and motor voltages/currents. The write block allows you to send an input voltage to the motors as well as the LEDs [15].

Finally, the simulation and control system were then created using the common blocks provided by Simulink, such as step functions, transfer functions, scopes, sum/difference, gain, and saturation [15]. Fig. 2 summarises how the overall system functions.

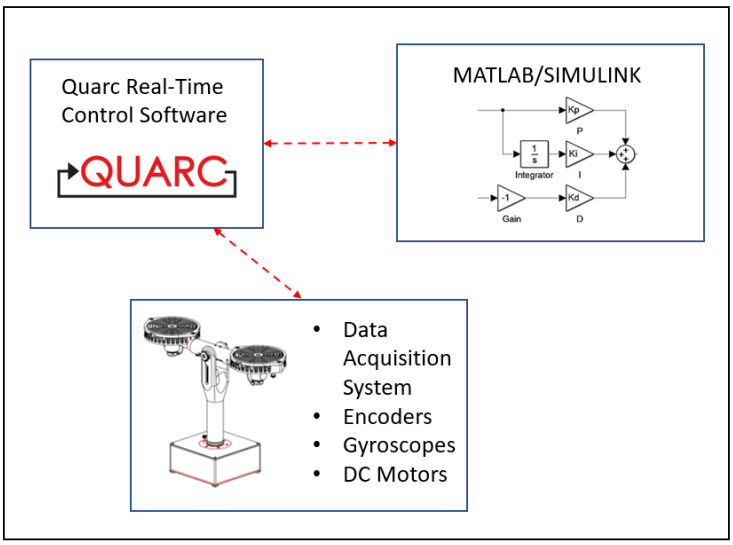

Fig. 2. System Overview

3.2 1-DOF Bi-coper

3.2.1 System Dynamics

To begin with, a mathematical model was created for the bi-copter system to fully understand the dynamics of the system. As a starting point, a free body diagram was made to represent all the forces acting on the system and the different variables of the system, to analyse the different forces. As shown in Fig. 3, the only two forces acting on the system are f_0 and f_1, which are created by the thrust of the two rotors. The pitch and yaw angles were represented by θ and ψ respectively, which are the result of moments created by the speed difference between both rotors on the y and z-axis. The thrust displacement or the distance between the centre of the rotor and the rotation axis was indicated by D_t.

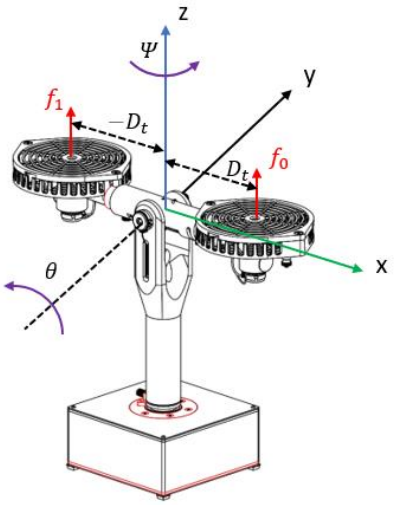

Fig. 3. 1-DOF Configuration System Diagram

To derive the expressions of the moments acting on the y and z-axis τ_θ and τ_ψ, the following approximations were used: $f_0 = k_{yy}\omega_0$ and $f_1 = k_{yy}\omega_1$ with k_{yy} as pitch thrust gain, D_t as thrust displacement and k_{zz} as drag constant.

$$\tau_\theta = k_{yy}D_t(\omega_0 - \omega_1) \tag{2}$$

$$\tau_\psi = k_{zz}(\omega_0 - \omega_1) \tag{3}$$

Combining Eq. (2) and Eq. (3), the total moments acting on the system were:

$$\begin{bmatrix}\tau_\theta \\ \tau_\psi\end{bmatrix} = \begin{bmatrix}k_{yy}D_t(\omega_0 - \omega_1) \\ k_{zz}(\omega_0 - \omega_1)\end{bmatrix} \tag{4}$$

$$\begin{bmatrix}\tau_\theta \\ \tau_\psi\end{bmatrix} = \begin{bmatrix}k_{yy}D_t & -k_{yy}D_t \\ k_{zz} & -k_{zz}\end{bmatrix}\begin{bmatrix}\omega_0 \\ \omega_1\end{bmatrix} \tag{5}$$

$$\begin{bmatrix}\tau_\theta \\ \tau_\psi\end{bmatrix} = M\begin{bmatrix}\omega_0 \\ \omega_1\end{bmatrix} \tag{6}$$

To evaluate the resulting angular positions or pitch and yaw angles (θ, ψ), the kinematics of the systems were considered [16]. The angular velocities of the system can be expressed as follows:

$$\begin{bmatrix}\ddot{\theta} \\ \ddot{\psi}\end{bmatrix} = \begin{bmatrix}\frac{1}{J_y}\tau_\theta \\ \frac{1}{J_z}\tau_\psi\end{bmatrix} - \begin{bmatrix}\frac{D_y}{J_y}\dot{\theta} \\ \frac{D_z}{J_z}\dot{\psi}\end{bmatrix} - \begin{bmatrix}\frac{K_{sp}}{J_y}\theta \\ 0\end{bmatrix} \tag{7}$$

J_y, J_z: moments of inertia of pitch and yaw axes

D_y, D_z: damping coefficients of pitch and yaw axes

In the 1-DOF configuration, pitch and yaw angles could not be controlled simultaneously using both rotors. Thus, they were tested separately by locking the axis of the unwanted parameter and using only one rotor.

For pitch control only, the transfer function was derived from Eq. (7) using Laplace transform:

$$J_y\ddot{\theta} + D_y\dot{\theta} + K_{sp}\theta = \tau_\theta = K_{yy}D_t\omega_m \tag{8}$$

$$J_y s^2\Theta(s) + D_y s\Theta(s) + K_{sp}\Theta(s) = K_{yy}D_t\Omega_m(s) \tag{9}$$

$$s^2\Theta(s) + \frac{D_y}{J_y}s\Theta(s) + \frac{K_{sp}}{J_y}\Theta(s) = \frac{D_t K_{yy}}{J_y}\Omega_m(s) \tag{10}$$

$$P(s) = \frac{\Theta(s)}{\Omega_m(s)} = \frac{\frac{D_t K_{yy}}{J_y}}{s^2 + \frac{D_y}{J_y}s + \frac{K_{sp}}{J_y}} \tag{11}$$

To achieve a positive pitch angle, the speed of the motor was increased clockwise. For a negative pitch angle, the speed of the motor was also increased but counter-clockwise.

For yaw control only, the transfer function was also derived from Eq. (7) using the Laplace transform:

$$J_z\ddot{\psi} + D_z\dot{\psi} = \tau_\psi = k_{zz}\omega_m \tag{12}$$

$$J_z s^2\Psi(s) + D_z s\Psi(s) = k_{zz}\Omega_m(s) \tag{13}$$

$$P(s) = \frac{\Psi(s)}{\Omega_m(s)} = \frac{k_{zz}}{J_z s^2 + D_z s} \tag{14}$$

The rotor rotated counter-clockwise for positive yaw angle, and clockwise for negative yaw angle.

As values for the thrust gain K_{pp} and pitch stiffness K_{sp} were not given in the user manual, they had to be derived using experimentation.

For the pitch parameter estimation procedure, the pitch transfer function derived earlier (Eq. (11)) was approximated to a second-order system transfer function,

$$\frac{Y(s)}{U(s)} = \frac{K_{ss}\omega_n^2}{s^2 + 2\zeta\omega_n + \omega_n^2} \tag{15}$$

where K_{ss} is the steady-state value when a step $1/s$ is applied, ω_n is the natural frequency, and ζ is the damping ratio.

After applying a step input voltage to one motor for 60 seconds, the natural frequency ω_n of the free-oscillation response, i.e., the response after the step is no longer applied (60 s to 180 s) and the damping ratio ζ were calculated using the following equations with ω_d as natural damped frequency and subsidence ratio δ :

$$\omega_d = \frac{2\pi}{T_{osc}}, \qquad \omega_n = \frac{\omega_d}{\sqrt{1-\zeta^2}}, \qquad \zeta = \frac{1}{\sqrt{1+\left(\frac{2\pi}{\delta}\right)^2}}, \qquad \delta = \frac{1}{n}\ln\frac{O_1}{O_n} \tag{16}$$

The stiffness, damping coefficient and thrust gain were then calculated using the following equations with θ_{ss} as steady-state pitch angle:

$$K_{sp} = J_y\omega_n^2, \qquad D_y = 2J_y\zeta\omega_n, \qquad K_{yy} = \frac{\theta_{ss}K_{sp}}{D_t\omega_0} \tag{17}$$

Considering the yaw transfer function (Eq. (14)), thrust gain k_{zz} and damping coefficient D_z were also derived experimentally using a similar approach.

3.2.2 Cascade Control Design

To control the pitch or yaw angles of the 1 DOF bi-copter, a cascade control system was designed. Cascade control consists of two controllers where one controller sets the reference point of the second controller. In this configuration, pitch or yaw angles were controlled by varying the speed of one rotor, which means that outer loop controller compared between a reference angle and angle feedback to compute the control input speed, which will be the set point to the inner loop speed controller. This approach was chosen to ensure the stability, robustness, and efficiency of the system.

To begin with, a PI controller was designed for the inner loop to control the speed of the DC motors. First, the motors were modelled as a first order system (Eq. (18)). A voltage step function was applied to both DC motors, and the step response was analysed to get the steady state gain k and time constant τ. This system identification method was used due to its simplicity and rapidity.

$$P(s) = \frac{k}{\tau s + 1} \tag{18}$$

The transfer function derived was then simulated in Simulink and compared to the actual motors to validate the model (section 4.1). After validation of the model, the PI controller was designed first in simulation to compare a set reference speed with the current speed of the motor to send a control signal voltage to the plant.

$$C(s) = k_P + \frac{k_i}{s} \tag{19}$$

Using the PI controller and voltage to speed transfer functions (Eq. (18) and Eq. (19)), the closed-loop transfer function of the rotor was obtained:

$$G(s) = \frac{\Omega_m(s)}{V_m(s)} = \frac{\frac{Kk_i s}{\tau}}{s^2 + \frac{1 + Kk_p}{\tau}s + \frac{Kk_i}{\tau}} \tag{20}$$

It was then compared to a second-order system transfer function to calculate k_P and k_i.

$$\frac{Y(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n + \omega_n^2} \tag{21}$$

$$k_p = \frac{2\zeta\omega_n\tau - 1}{K}, \qquad k_i = \frac{\omega_n^2\tau}{K} \tag{22}$$

The calculated values for proportional gain k_P and integral gain k_i were used to tune the controller and were adjusted until the system output was smooth and stable.

After tuning the controller gains in the simulation, limiting overshoot, and ensuring an appropriate rise time, the controller was integrated with the actual motors and tuned again.

The process of designing the outer loop PID controller was similar to the inner loop design. In the 1-DOF configuration, only one parameter could be controlled at a time, thus two outer loop controllers were designed separately.

For pitch control only, the pitch transfer function derived earlier (Eq. (11)) was first simulated and compared to the real system to validate the model. After validating the model, the PID controller was designed first in simulation to compare a set reference pitch angle with the current pitch angle of the system to set a reference speed to the speed controller. The controller gains k_P, k_i, k_d were then tuned to limit overshoot, ensure appropriate rise and settling times with limited steady state error.

$$C(s) = k_P + \frac{k_i}{s} + k_d s \tag{23}$$

The closed-loop transfer function of the system was derived using the PID controller and pitch model transfer functions, and values for k_P, k_i and k_d were calculated.

$$G_{\theta,d}(s) = \frac{\Theta(s)}{\Theta_d(s)} = \frac{K_t D_t\left(k_p s + k_i\right)}{J_p s^3 + \left(D_p + D_t K_t k_d\right)s^2 + \left(K_{sp} + D_t K_t k_p\right)s + D_t K_t k_i} \tag{24}$$

To tune the controller, the heuristic tuning approach was chosen. This method consisted of testing different values of the controller gains until a stable response is achieved, depending on the understanding of how each PID controller gain affect the system (section 2.2.2). While this method was relatively simple and did not require much knowledge about the system, it was time consuming and did not always lead to robust responses. The controller was then integrated with the actual system and tuned again, forming a cascade control system.

Finally, for yaw control only, the same process of designing the PID controller of the pitch control was repeated. The transfer function derived (Eq. (14)) was simulated and compared to the real system, then a PID controller was designed and tuned, comparing a set reference yaw angle with the current yaw angle of the system to set a reference speed to the speed controller.

To run the Quanser Aero 2 in the 1 DOF configuration and controlling only pitch or yaw, the unwanted parameter controller was disabled. Fig. 4 shows the block diagram of the simulation of pitch and yaw control.

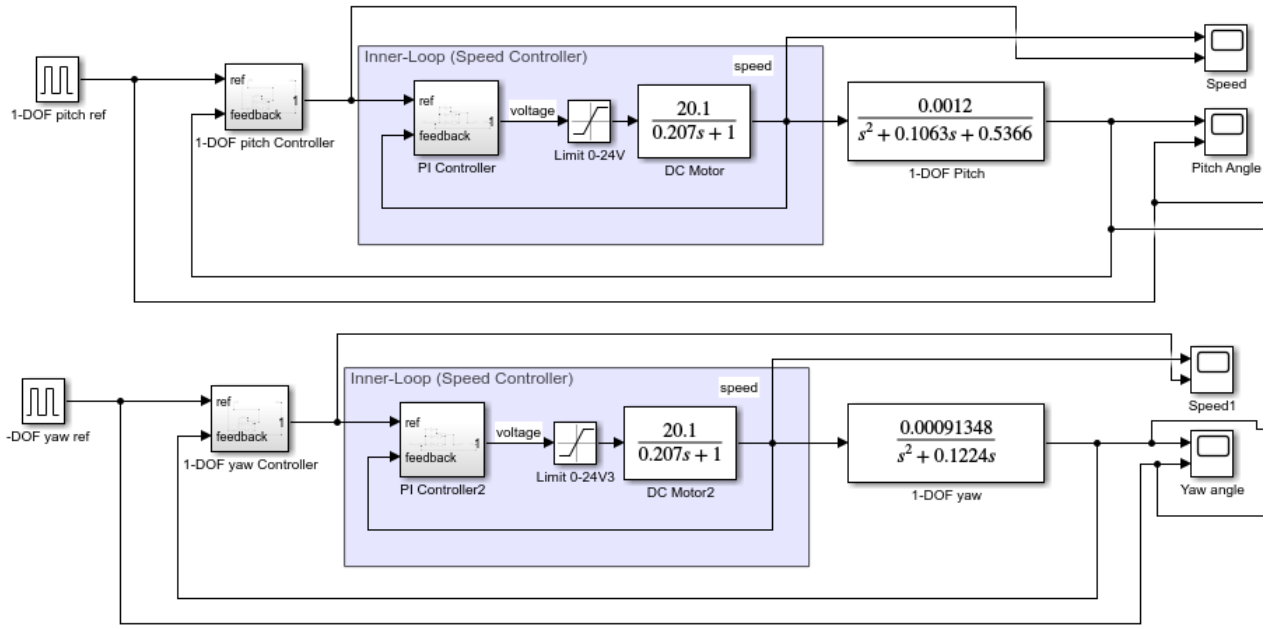

Fig. 4. PID Cascade control simulation block diagram

3.3 2-DOF UAV

3.3.1 System Dynamics

As shown in the previous section, in a bi-copter where the two rotors are horizontal or parallel to the ground, it was not possible to control both pitch and yaw simultaneously. Nevertheless, if one

rotor is rotated by ninety degrees to keep the rotor perpendicular to the ground, it could be possible to achieve a 2 DOF system (Fig. 5). In order to test this concept, the 2 DOF configuration of the Quanser Aero 2 was used.

Again, as seen on the system diagram, there are two forces f_0 and f_1 acting on the system, which were created by both rotors. In this case, the pitch and yaw angles of the system were controlled by motor 0 and motor 1, respectively. The pitch movement was mostly affected by rotor 0 as it is horizontal, while yaw movement was governed by perpendicular rotor 1. Although, rotor 0 (pitch motor) still produced a small amount of torque on the z-axis affecting the yaw movement, while rotor 1 (yaw motor) also produced a small amount of torque on the y-axis, affecting the pitch movement.

All parameter notations on the diagram are the same as the ones in section 3.2**.** To represent the effects of each rotor on the other, two variables were introduced: k_{yz} (yaw thrust gain of pitch motor) and k_{zy} (pitch thrust gain of yaw motor).

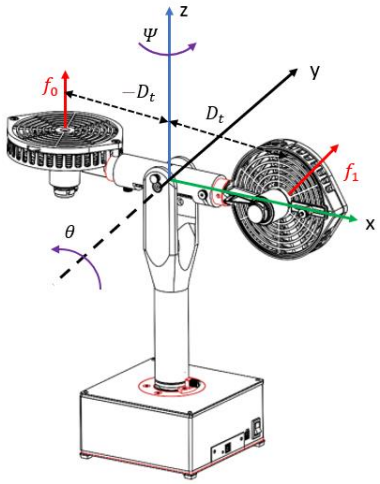

Fig. 5. 2-DOF Configuration System Diagram

The torques produced on the y-axis were evaluated first.

$$\tau_\theta = J_y\ddot{\theta} + D_y\dot{\theta} + K_{sp}\theta = k_{yy}D_t\omega_0(t) + k_{yz}D_t\omega_1(t) \tag{25}$$

Laplace transform was then used to derive the transfer function of the pitch angle.

$$J_y s^2\Theta(s) + D_y s\Theta(s) + K_{sp}\Theta(s) = K_{yy}D_t\Omega_0(s) + K_{yz}D_t\Omega_1(s) \tag{26}$$

$$\Theta(s) = \frac{\frac{K_{yy}D_t}{J_y}}{s^2 + \frac{D_y}{J_y}s + \frac{K_{sp}}{J_y}}\Omega_0(s) + \frac{\frac{K_{yz}D_t}{J_y}}{s^2 + \frac{D_y}{J_y}s + \frac{K_{sp}}{J_y}}\Omega_1(s) \tag{27}$$

The torques produced on the z-axis were then analysed.

$$\tau_{\psi} = J_z\ddot{\psi} + D_z\dot{\psi} = k_{zz}D_t\omega_0(t) + k_{zy}D_t\omega_1(t) \tag{28}$$

Again, Laplace transform was used to derive the transfer function of the yaw angle.

$$J_z s^2\Psi(s) + D_z s\Psi(s) = k_{zy}D_t\Omega_0(s) + k_{zz}D_t\Omega_1(s) \tag{29}$$

$$\Psi(s) = \frac{\frac{K_{zy}D_t}{J_z}}{s^2 + \frac{D_z}{J_z}s}\Omega_0(s) + \frac{\frac{K_{zz}D_t}{J_z}}{s^2 + \frac{D_z}{J_z}s}\Omega_1(s) \tag{30}$$

Thus, the system could be described by the pitch and yaw transfer functions (Eq. (27) and Eq. (30)). To verify that the transfer functions derived accurately described the system, they were simulated in Simulink and compared with the real system (section 4.3). Moreover, the two new variables k_{yz} and k_{zy} were estimated experimentally using the same method in section 3.2.1.

3.3.2 PID Controller Design

To control the pitch and yaw movements of the system, two PID controllers were designed. As shown in Fig. 6, the first controller compared feedback of pitch angle of the system to a reference angle and provided the reference speed to the inner loop speed controller of motor 0 (pitch motor). The second controller compared feedback of yaw angle of the system to a reference angle and provided the reference speed to the other inner loop speed controller of motor 1 (yaw motor) which would be converted a voltage input.

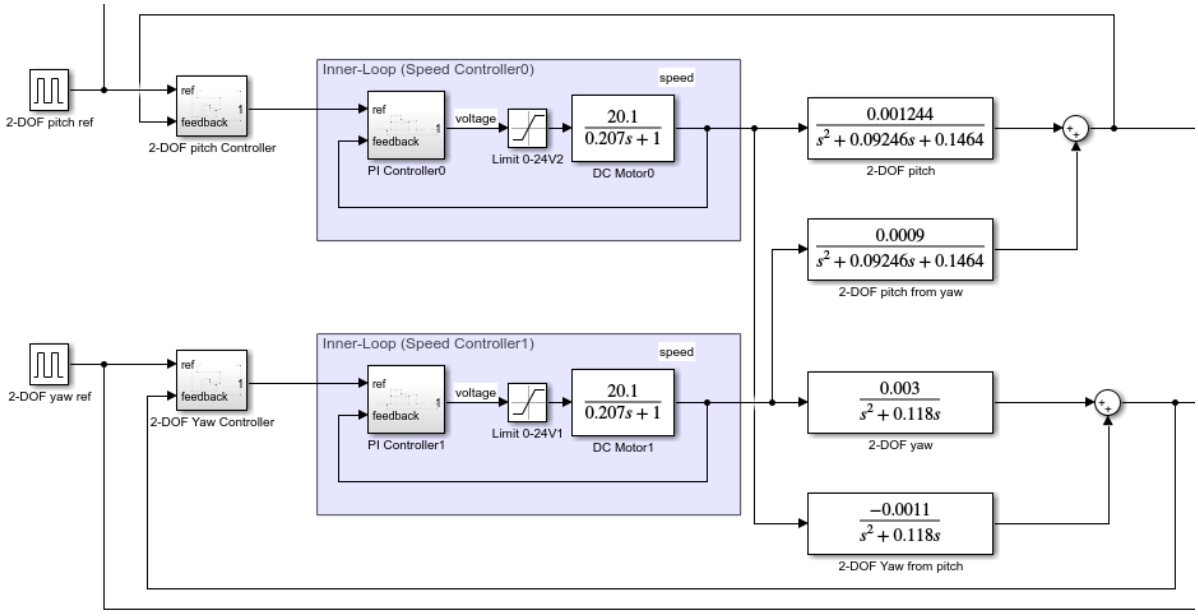

Fig. 6. 2-DOF cascade PID Control Block Diagram

4 Results and Discussion

This section shows the results of the cascade PID controller implemented in each configuration of the Aero 2 (bi-copter and helicopter), and discusses the viability of a possible third configuration (quadcopter) while comparing between the three UAV models.

4.1 1-DOF Bi-copter Control

4.1.1 Inner-Loop Speed Control

As demonstrated in section 3.2.2, a model was developed for the DC motors fitted in the Aero 2 to design the speed controller (inner-loop controller). After analysing the step response of the motors, the steady state gain k and time constant τ were calculated as follows:

$$k = \frac{\Delta y}{\Delta u} = \frac{100.5}{5} = 20.1$$

$$\tau = t_1 - t_0 = 1.207 - 1 = 0.207\ s \tag{31}$$

where t_1 was the time the system output takes to reach 63.2 % of its steady state output, and t_0 is the time at which the step voltage was applied. Δy was the steady state output of the system, and Δu was the step voltage value. The derived transfer function of the DC motors was calculated:

$$\Omega(s) = \frac{20.1}{0.207s + 1} V(s) \tag{32}$$

Fig. 7 compared between the estimated motor model and the actual motors. It was observed that the model is almost identical to the hardware, having the same steady state gain and time response.

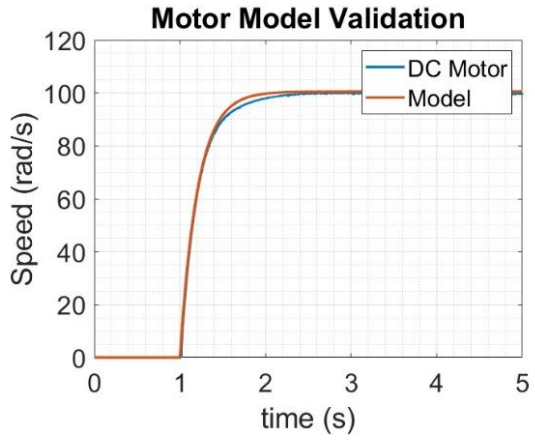

Fig. 7. Step response comparison of DC motor and model

In Fig. 8**,** the response of the controlled motor speed was recorded and compared to the reference speed given to the controller to evaluate the performance of the controller. The reference speed was a square wave with amplitude of 100 rad/s, time period of 20 seconds and 50 % duty cycle ratio. The tuned controller gains used were $k_P = 0.1$ and $k_i = 0.5$.

It was observed that the PI controller was able to track the speed reference almost perfectly. Small oscillations were present in the system output; however, they were negligible as the system was not affected. Other speed references were also tested starting from 20 rad/s to 200 rad/s. The system response was almost identical at all speeds.

Thus, with a PI speed controller, the user could choose a specific required speed for the rotors which would be then converted to a voltage input by the controller. A saturation block was also used to ensure that the input voltage applied to the system would not be higher than 24 V to avoid damaging the device.

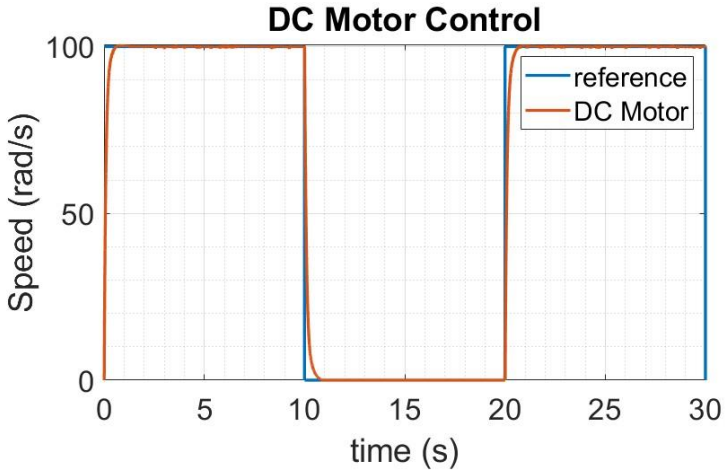

Fig. 8. DC motor closed-loop reference tracking

The designed speed controller was used as the inner-loop controller in all the following experiments and simulations.

4.1.2 Pitch PID Control

First, the pitch parameters of the system, pitch thrust gain K_{yy}, pitch stiffness K_{sp} and damping coefficient D_y were estimated through the experimental procedure explained in section 3.2.1, and were presented in Table 1.

Table 1. Pitch Parameter Estimation Results

Symbol	Value	Unit
K_{yy}	0.00016149	N/m
K_{sp}	0.0124	Nm/V
D_y	0.0025	$N/m/(rad/s)$

Plugging the values for K_{yy}, K_{sp} and D_y in Eq. (11), the pitch transfer function relating rotor speed Ω_m and pitch angle Θ was calculated.

$$\Theta(s) = \frac{0.0012}{s^2 + 0.1063s + 0.5366}\Omega_m(s) \tag{33}$$

Second, the model was compared to the real hardware by giving both the model and the plant the same input, and the output pitch angle was recorded (Fig. 9).

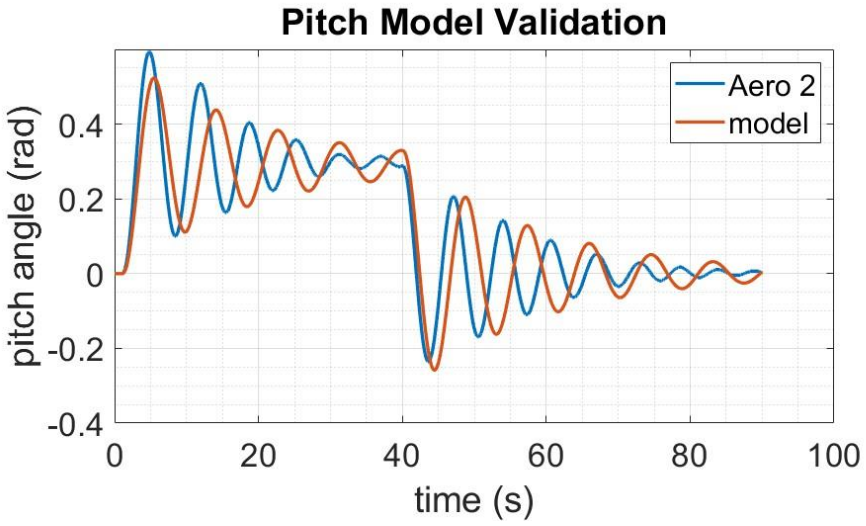

Fig. 9. 1-DOF pitch model validation

The input given to the system was a step voltage of 3 V lasting 40 seconds, creating forced oscillations. Oscillations recorded after 40 seconds were the natural oscillations of the system. It was observed that the response of the model was not identical to the hardware. The response of the hardware was slightly more damped and oscillations were smaller for the first 40 seconds. When the system was oscillating at natural frequency (40 – 90 s), oscillations were larger in the hardware than the model, yet damping was almost identical. The model was expected to have a slight error as the parameters were estimated experimentally and the system was unstable.

Nevertheless, the model was acceptable as the output of the model had a similar trend to the Aero 2.

The model was then tested with chosen PID gains according to required overshoot and rise time values. The calculated values were $k_P = 500$, $k_i = 300$ and $k_d = 700$ for an overshoot of 5% and rise time of 5 seconds. Although the calculated controller gains were used as initial safe parameters to observe how the system behaved, it was expected that the system would need more tuning.

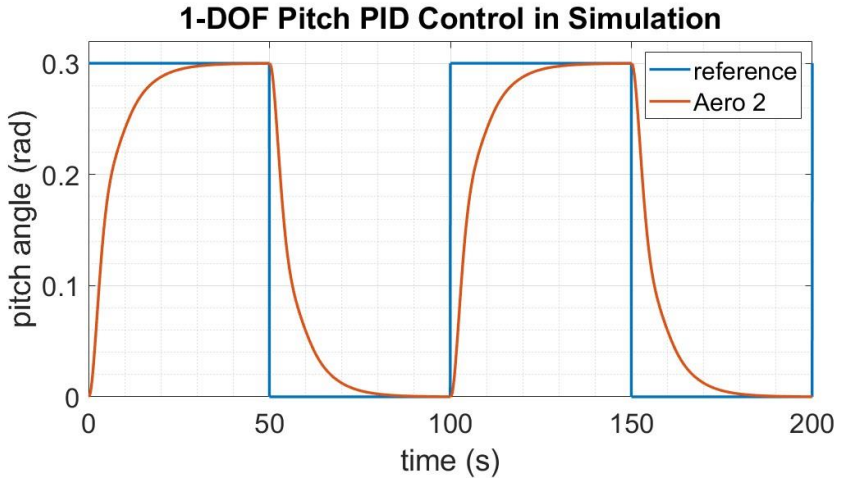

Fig. 10. 1-DOF closed-loop pitch tracking (Simulation)

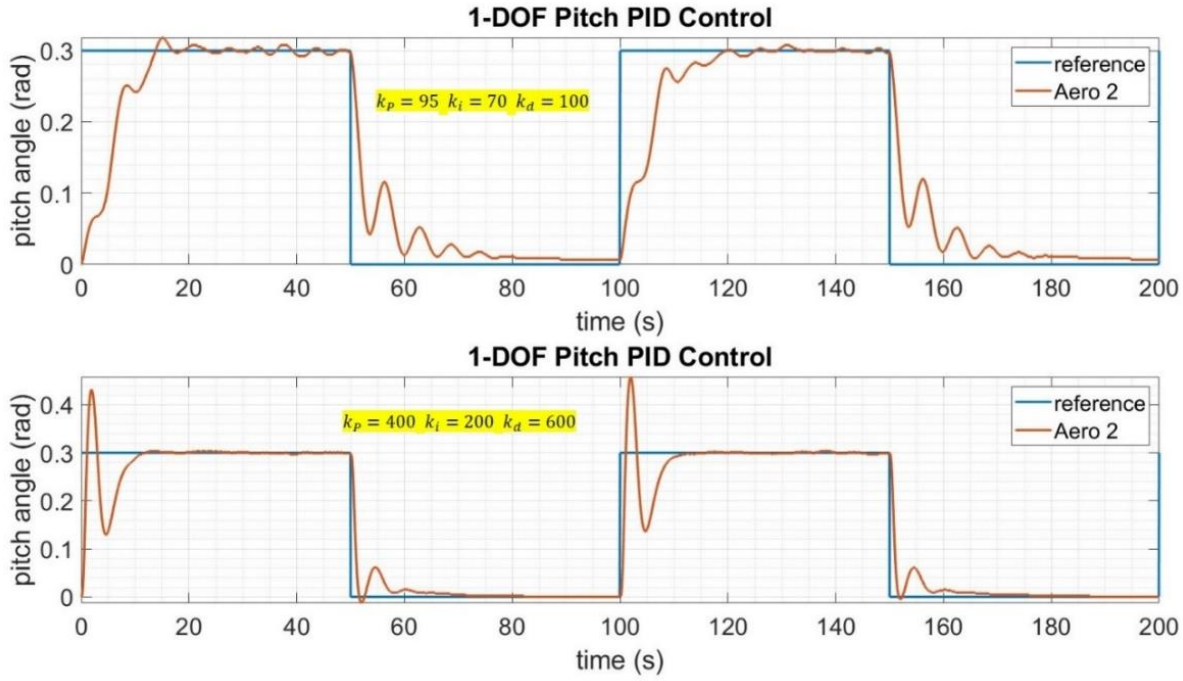

Fig. 11. 1-DOF closed-loop pitch tracking (Aero 2)

Table 2. 1-DOF System performance comparison with PID control

PID controller gains	Peak Overshoot (%)	Steady-state error (rads)	Time Response (seconds)	5 % Settling time (seconds)
$k_P = 95, k_i = 70, k_d = 100$	10.82	0.0066	30	18.12
$k_P = 140, k_i = 100, k_d = 200$	19.74	0.0032	25	14.85
$k_P = 400, k_i = 200, k_d = 600$	45.36	0.0022	20	9.16

After testing the PID controller on the model simulation (Fig. 10), it was implemented on the physical Aero 2 (Fig. 11). Three different sets of gains were tested and compared in terms of system performance (Table 2).

As shown in the presented data, using smaller controller gains eliminated the overshoot, yet the system was oscillatory, especially when a zero-pitch angle was demanded. With larger gains, a high overshoot was manifested, however the response had very little oscillations and almost zero steady state error. Moreover, the larger the gains, the faster the time response was. In a system such as the Aero 2, oscillations and overshoot were not critical since the system did not fly and did not pose any danger. However, if the system did fly, the larger gain results would not have been accepted as the system will be at a higher risk of crashing. Bi-copters have only two rotors aligned rotating in opposite directions, which makes controlling them challenging. In this test, only one rotor was used to control the pitch angle, which made it more unstable. This could explain why the system response was oscillatory and had overshoot. To add, using both rotors was also attempted but little change in the performance was noticed.

Moreover, the controller performance was also evaluated on the complete pitch angle range of the Aero 2 (-0.8 to 0.8 rads). For small pitch angles ($\pm 0.3, \pm 0.4, \pm 0.5$ rads), the controller was able to track the reference as shown in Fig. 11. For large pitch angles (± 0.8), the system still tracked the reference angle yet it became more unstable and had increased oscillations. In fact, when the system reached such large angles, the propellors physically collided with the main body of the system, which made it unstable.

Thus, in the bi-copter configuration, it was possible to control the pitch angle individually when the yaw axis was locked. It was shown that it was difficult to get a smooth response when using PID control. Either the system had no overshoot and more oscillations with small control gains, or larger gains leading to high overshoot but little oscillations and fast settling time. This behaviour was caused by the unstable dynamics of the system. It was also observed that the controller performed better at smaller pitch angles.

4.1.3 Yaw PD Control

Table 3 presents the values for the estimated yaw thrust gain K_{zz} and yaw damping D_z parameters.

Table 3. Yaw Parameter Estimation Results

Symbol	Value	Unit
K_{zz}	0.00013019	N/m
D_z	0.0029	$N/m/(rad/s)$

Using the yaw model derived in Eq. (14), the yaw transfer function of the bi-copter system was expressed as

$$\Psi(s) = \frac{0.00091348}{s^2 + 0.122s} \tag{34}$$

The model was then given the same input as the physical plant and both responses were compared. A step voltage of 6 V was given to both the model and Aero 2 for thirty seconds.

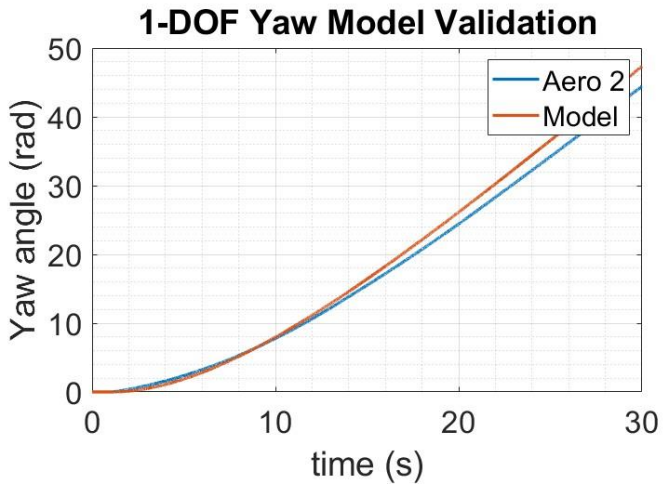

Fig. 12. 1-DOF Yaw model validation

Looking at Fig. 12, the derived model was almost identical to the Aero 2 during the first fifteen seconds. At this instance, the yaw angle reached almost 19 rads, which means that the system rotated almost three times around the z-axis. Then, the model response drifted slightly from the actual response. This constant rotation led to more energy loss in the system which the model might have not taken into consideration, which explains the fact that the yaw angle of the model is larger than the Aero 2 one after several rotations. It was also expected to observe some error in the model since the parameters of the model were estimated. However, it was sufficiently accurate and could be used to design the control system as the error increased only at high frequencies.

After testing several controller gains, PD control was the most appropriate control method as the minimum amount of integral gain added in the control would immediately make the system unstable. As shown in Fig.13, it was possible to achieve a smooth step response with no oscillations or overshoot, and the controller managed to track the demanded yaw angle in simulation. However, the time response of the system was approximately 65 seconds, which was extremely slow.

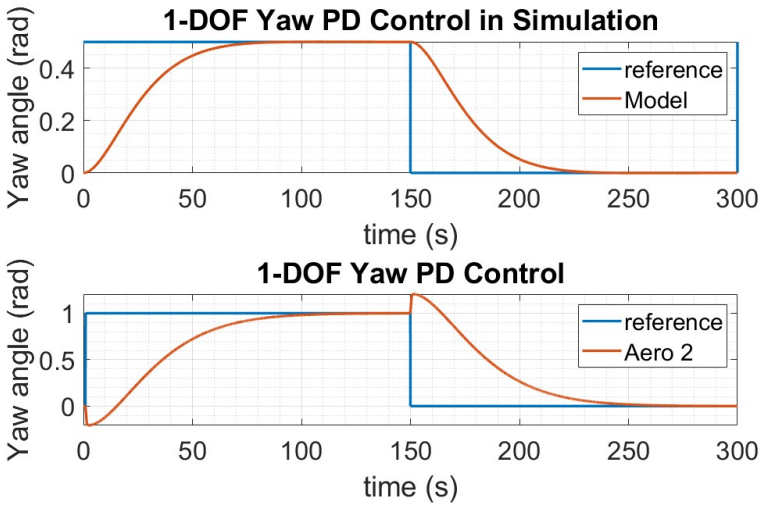

Fig. 13. 1-DOF closed-loop yaw reference tracking (simulation/Aero 2)

On the hardware, the response was slower than the simulation, reaching almost 100 seconds, and an overshoot of 2 % was observed. Although controlling the yaw angle of the 1-DOF bi-copter was possible, it required a relatively long time compared to controlling the pitch angle. Generally, in a PID controller, the rise time of the system response is reduced by increasing the proportional and integral gains. In this case, the integral gain was equal to zero as it made the system unstable and overshoot. Increasing the proportional gain also meant increasing overshoot. As seen on the Aero

2 response, by trying to reduce the time response, an overshoot was present. Furthermore, overshoot is theoretically decreased by increasing the derivative gain. In fact, the values of the gains used in this simulation were $k_P = 4$ and $k_d = 45$. Increasing k_P to 6 for instance increased the overshoot to almost 5% while the time response only decreased by two seconds. Considering the derivative gain, increasing it did not reduce overshoot and made the time response slower [9].

In fact, the observed overshoot was mainly caused by the counter-clockwise torque created by the rotor. As explained previously in section 3.2.1, a clockwise movement was produced by rotating the rotor counter-clockwise, and vice-versa. In this specific simulation, the yaw angle demanded was positive, meaning that the rotor had to rotate counter-clockwise. After 150 seconds, the demanded yaw angle was zero, meaning that the rotor would have to rotate clockwise to reach its initial position. Thus, by suddenly changing the direction of rotation of the rotor over a small period, an overshoot was manifested. This phenomenon was also amplified by increasing the proportional gain. Finally, the controller showed the same behaviour with all the yaw angles withing the range of the system (2π rads).

4.2 2-DOF UAV PID Control

Pitch and yaw parameters of this configuration were calculated experimentally using similar procedures as the previous configurations. The estimated parameters could be found in the Appendices (Table 6) and the calculated transfer functions were presented in the following equations.

$$\Theta(s) = \frac{0.001244}{s^2 + 0.09246s + 0.1464}\Omega_y(s) + \frac{0.0009}{s^2 + 0.09246s + 0.1464}\Omega_z(s) \quad (35)$$

$$\Psi(s) = \frac{-0.0011}{s^2 + 0.118s}\Omega_y(s) + \frac{0.003}{s^2 + 0.118s}\Omega_z(s) \quad (36)$$

It was noticed that the yaw models were more accurate than the pitch ones. The yaw models were almost identical to the hardware, and started deviating from the real response at high speeds and large yaw angles. The errors at large angles were insignificant as the required yaw angle in the simulation did not exceed 1.5 rads. The fact that the accuracy of the model decreased with larger values was explained by the energy loss in the system generated after acceleration and several turns, almost 11 turns.

Considering the pitch models, it was more challenging to get accurate models. For the pitch movement created by the pitch motor, the natural frequency of the model was correct as the

oscillations had the same time period, despite that the magnitude of oscillations was 20 % smaller. For the pitch caused by the yaw motor, both natural frequencies and damping coefficients were only accurate by 50 %. The inaccuracy in the pitch models was caused by the fact that the system was extremely damped, and that during the parameter estimation procedures, the system could not generate enough oscillations to complete the system identification. Therefore, it was not possible to calculate initial PID controller gains and the task of tuning the controller required more effort. The model validation figures could be found in the Appendices (Fig. 20).

Fig. 14 represents the step response of both pitch and yaw angles with the most suitable controller performance found. The pitch controller gains were $k_P = 350$, $k_i = 150$ and $k_d = 400$, and yaw controller gains $k_P = 165$, and $k_d = 190$. Looking at the step response of the pitch angle, it was found that the system saw a slight overshoot (28%) followed by an oscillation, then a steady state was reached. In theory, this performance was accepted as it showed a relatively fast step response (20 s) with a small overshoot, which made the system more aggressive and faster in changing directions. However, if the system were to be a real UAV, this performance would have been too aggressive as it would endanger it, unless it was used for military purposes. The overshoot was caused by the proportional gain which was relatively large and the oscillation was amplified by the integral gain. Moreover, the yaw controller also induced the oscillations in pitch due to the coupling effect.

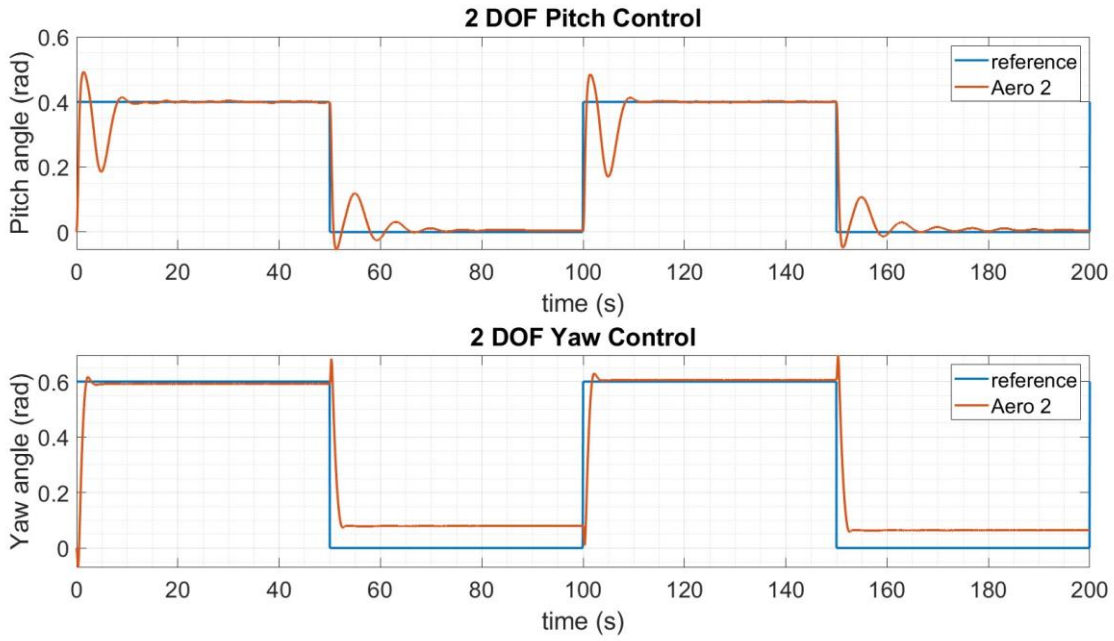

Fig. 14. 2-DOF closed-loop pitch/yaw reference tracking

For the yaw step response, overshoot was minimal (1 %) and step response was relatively fast (3 s). When turning from a zero-degree yaw to 30 degrees, steady-state error was approximately zero, yet when going back to zero degrees, a larger steady-state error (0.03) was present. This was explained by influence of the pitch controller on the yaw angle as both movements were happening simultaneously. As the system was fixed, this error caused no problem and was almost negligible. Nevertheless, if the system was flying for long distances, this would cause the system not reach its destination accurately. Often when tuning controllers, a trade-off must be made. If the time response was to fast, an overshoot was manifested. If the system was slower, less overshoot and oscillations would be present.

Other combinations of controller gains were tested and the results could be found in the Appendices (Tables 7 and 8).

4.3 2-DOF Quadcopter Implementation

It was shown in the previous two sections that both pitch and yaw angles could be controlled individually in the bi-copter configuration, and simultaneously in the helicopter configuration. This section tests the feasibility of controlling both pitch and yaw simultaneously in the bi-copter setup without locking the axes. The possibility of modelling and controlling a quadcopter using the Quanser Aero 2 would depend on the outcome of this simulation.

Theoretically, if it was possible to transform the 1-DOF bi-copter into a 2-DOF one, then by controlling two Aero 2's simultaneously, quadcopter control could be simulated. Looking at Eq. (5) representing the total moments acting on the system, it was observed that the matrix M was not invertible as its determinant was zero. In theory, this meant that both pitch and yaw angles could not be controlled simultaneously in the bi-copter configuration.

A suggested solution was to consider the thrust force generated by the rotors in the z-direction in the equation such that the matrix M is invertible (Eq. (37)). In a real bi-copter, thrust would be the force allowing it to fly, yet the Quanser Aero 2 is a fixed platform that is not able to fly, which makes the thrust force $f_z = 0$. By taking the pseudo-inverse of the matrix M, it was possible to get a mixer matrix which had torque as input and motor speed as output. The motor speed determined by the mixer would then be the reference for the inner loop speed controller. Motor 0 was configured to control pitch while motor 1 controlled yaw.

$$\begin{bmatrix} f_z \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} k_{dd} & k_{dd} \\ k_{dd}D_t & -k_{dd}D_t \\ k_M & -k_M \end{bmatrix} \begin{bmatrix} \omega_0 \\ \omega_1 \end{bmatrix} = M \begin{bmatrix} \omega_0 \\ \omega_1 \end{bmatrix}$$

$$\begin{bmatrix} \omega_0 \\ \omega_1 \end{bmatrix} = M^{-1} \begin{bmatrix} f_z \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \quad (37)$$

The mixer matrix was then implemented in the virtual system (for safety reasons) with two PID controllers (Fig. 15). According to the transfer function of system, the outer-loop controllers had the error in pitch or yaw angle as input and motor speed as output (control signal). Nevertheless, the inputs of the mixer matrix are torque, thus the control signal coming from the controller was converted from speed to torque.

The 1-DOF bi-copter PID controller gains used to control each parameter individually were first tested in this simulation to observe how the system behaved. As the control signal of the controllers in this configuration was converted to torque, it became almost 10^4 times smaller for pitch and 10 times for yaw. Thus, the gains were all multiplied by a factor of 10 and 10^4.

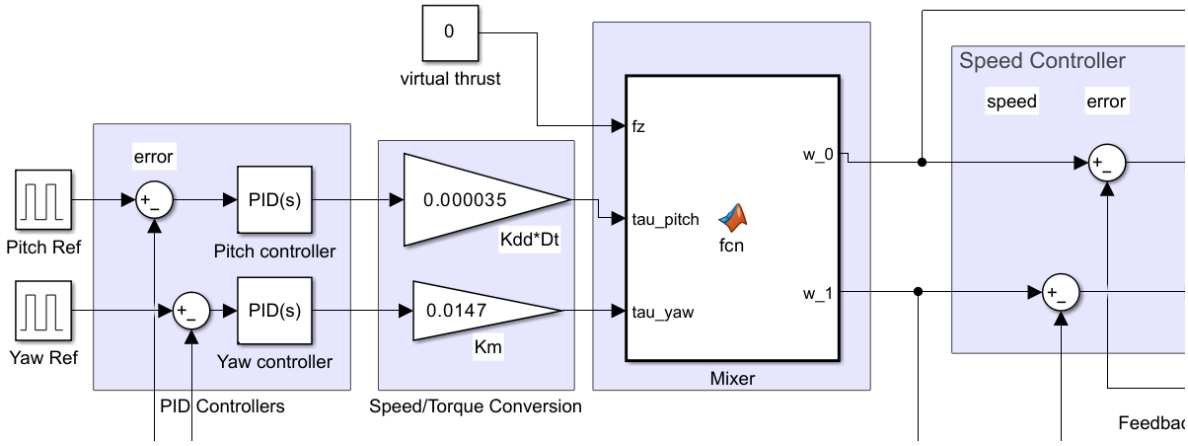

Fig. 15. System Block Diagram

First, each parameter was tested separately with the mixer matrix to check that the system behaved as expected. For instance, Fig. 16 showed that the pitch angle of the system could be controlled while disabling the yaw controller and leaving the yaw axis unlocked. The pitch response was identical to the 1-DOF simulation as expected. Considering the yaw angle, it was not being controlled and was mainly affected by the pitch motor due to the coupling effects of the system. The same process was repeated for the yaw parameter and the response was also identical to the 1-DOF simulation.

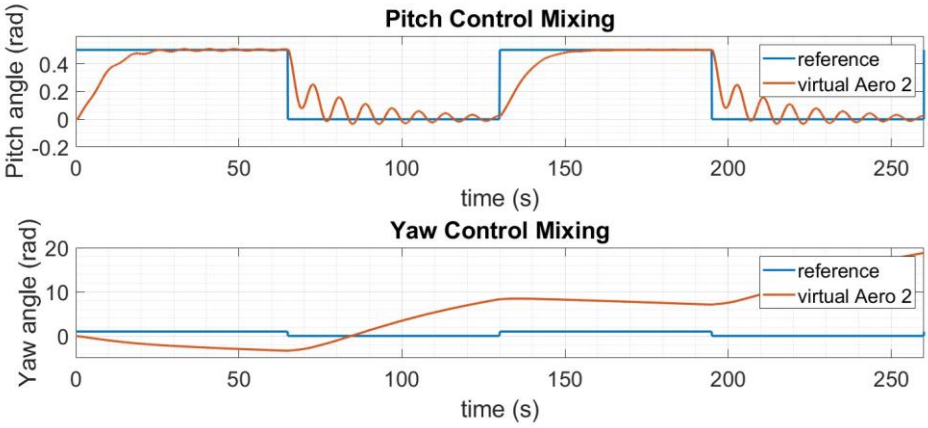

Fig. 16. 2-DOF bi-copter pitch/yaw tracking simulation

Then, both controllers were enabled such that pitch and yaw are controlled simultaneously. After many attempts in tuning the controllers in order to get an acceptable step response, it was not possible control both parameters simultaneously in the bi-copter configuration. This was explained by the dynamics of the system, as both pitch and yaw are dynamically coupled. To create a pitch or yaw movement, the speed of both rotors must be different to create torque. The problem was that both parameters were varied using the same method, which made it impossible to separate them. Adding the thrust force f_z in the system made the mixer matrix invertible indeed, yet it was set to zero as the system cannot fly, which means that eventually, nothing has been changed in the system. Thus, the suggested mixer matrix was not able to decouple the parameters of the system.

A second and different approach to solving this problem was then evaluated. The suggested solution was to physically connect two platforms together by a 3D printed part (Fig. 17), such that the full system is controlled as a quadcopter and that pitch and yaw movements are not coupled.

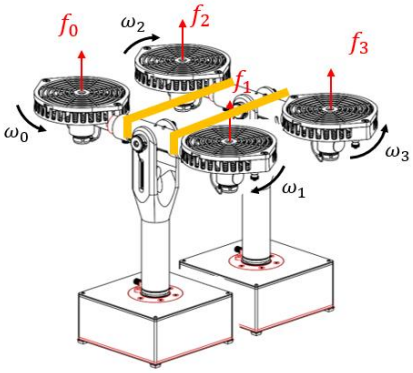

Fig. 17. 2-DOF Quadcopter System Diagram

In a quadcopter (cross configuration), two pairs of motors were used: clockwise (ω_0, ω_3) and anticlockwise (ω_1, ω_2). As shown in the system diagram, each pair is aligned diagonally, which according to Newton's third Law, the reaction torques acting on the stators of each motor will cancel out and achieve a zero net torque which ensures stability of the body of the drone. To control the drone and perform different manoeuvres, the speed of each motor is simply varied appropriately creating different combinations (Table 4).

To pitch the system forward, the front propellers are spun at a lower speed and the back ones at a higher speed, creating a difference between the lift forces at the front and back, and thus a moment on the y-axis. To pitch the drone backwards, the same principle is used but at the opposite side.

To create a yaw moment, it is sufficient to reduce the speed of rotors of one pair relative to the other depending on the direction of the rotation. For clockwise rotation, the speed of the counter-clockwise motors is increased, as for counter-clockwise rotation, the speed of the clockwise motors is increased [17].

Table 4. Motor Speeds Configurations

	Positive/clockwise	Negative/counter-clockwise
Pitch	$\omega_0, \omega_2 > \omega_1, \omega_3$	$\omega_0, \omega_2 < \omega_1, \omega_3$
Yaw	$\omega_0, \omega_3 > \omega_1, \omega_2$	$\omega_0, \omega_3 < \omega_1, \omega_2$

First, it was clear that pitch would be possible to control without affecting yaw. To control pitch, an imbalance in forces between the front and back rotors was created. Without the connecting parts, this imbalance would also create a yaw movement. However, the connecting parts would eliminate the reaction torques acting on the yaw axis, thus stopping the yaw movement.

Second, to control yaw, the speed of one diagonal pair had to be varied compared to the second pair to increase the reaction torque acting on the yaw axis. Essentially, each platform was controlled individually, therefore by trying to change the yaw angle of the whole system, each individual platform will also try to create a pitch movement in the opposite direction due to the coupling effect. Yet, the connecting parts will stop this phenomenon.

Unlike a classic quadcopter, each Aero 2 platform is rotating on a different yaw axis, which makes the total system impossible to rotate if the 3D printed parts are completely rigid. To solve this problem, a gear or belt system could be designed for the connecting parts allowing a certain degree of movement, still, the yaw angle would be limited to 45 degrees (Appendix A, Fig. 18). Although the Quanser Aero 2 could be theoretically controlled as 2-DOF quadcopter using this approach, it is an impractical and limited solution.

4.4 Different Configuration Comparison

In the previous sections, three different configurations of UAVs were presented: 1-DOF bi-copter, 2-DOF helicopter and quadcopter.

Comparing the first two configurations, unlike the 2-DOF helicopter, pitch and yaw angles of the 1-DOF bi-copter could not be controlled together. Although pitch control for both setups was almost identical in terms of overshoot, steady-state error and time response, tuning the controller of the helicopter required more effort as the effect of the yaw motor on the pitch had to be considered. It can be also seen in Table 5 that the pitch thrust gain of the motor in both configurations were almost the same, which explains the similarity in the pitch response.

Considering yaw, the time response of the bi-copter was almost 30 times slower than the helicopter, even though they both used PD controllers. For the helicopter, the vertical rotor (perpendicular to the ground) acted directly on the yaw with a large thrust gain. In contrast, the bi-copter used one horizontal rotor (parallel to the ground) to control yaw, which generated a relatively small yaw thrust gain (4 times smaller).

Table 5. Thrust Gains Comparison

Parameter		1-DOF Bi-copter	2-DOF Helicopter
Pitch Thrust Gain K_{yy}		0.00016149	0.00017729
Yaw Thrust Gain K_{zz}		0.00013019	0.00042543
Pitch	Overshoot	45.36 %	28 %
	Time response	20 s	20 s
Yaw	Steady-state error	0	0.03
	Time response	100 s	3 s

In comparison with the research paper discussing PID control for bi-copter pitch control (section 2.3), the time response of the bi-copter configuration of the Aero 2 was slower (almost 3 times slower). However, the specific platform used in this project was highly damped, which explains the slow response. Moreover, the research paper used an accelerometer and gyroscope to communicate the pitch feedback, while an encoder was used in this case. The quality and type of the feedback signal might have also affected the system performance [13].

Comparing the Aero 2 helicopter configuration to the 3-DOF helicopter discussed in (section 2.3), the Aero 2 showed a more oscillatory and slower response for elevation (pitch). Nevertheless, the dynamics of the 3-DOF helicopter were completely different as both rotors were placed at the back, and a mass was placed at the front as counter weight [11]. It was also not shown how the pitch and yaw movements affected each other, nor if the yaw was controlled.

Finally, the suggested solutions to developing a 2-DOF quadcopter configuration for the Aero 2 were not completely successful. In fact, both solutions looked different, yet they both relied on the same concept, decoupling pitch and yaw of a bi-copter. Mathematically, by making the matrix M invertible, pitch and yaw should be controllable. Unless the platform was modified such that it flew, adding the thrust force in the system matrix contradicts reality as the physical system did not change, which led to the second solution. After evaluating the dynamics of a classic 6-DOF drone [18], the four rotors of a drone are physically connected by a rigid body, cancelling out reaction torque. Applying the same concept to two Aero 2 bi-copters seemed logical. However, the problem in this case was that the two platforms did not rotate on the same axis, making this solution very limited.

5 Conclusion and Future Work

5.1 Conclusion

This paper attempted to use the hardware-in-loop system Quanser Aero 2 to model and control a 2-DOF quadcopter using two bi-copters. The Aero 2 was a powerful and flexible research platform providing a foundation for designing and testing control systems for different UAV configurations.

The 1-DOF configuration showed how bi-copters were unstable, slow and impractical as the pitch and yaw could not be controlled simultaneously. In simulation, the controllers of both parameters showed excellent performance and reference tracking with no oscillations or overshoot. With the hardware-in-loop, three sets of PID controller gains were compared. Larger gains provided the

best performance with small oscillations, overshoot, and steady-state error. However, the time response was relatively small.

The 2-DOF configuration demonstrated how helicopters were more effective compared to bi-copters, where it was possible to control the pitch and yaw at the same time while keeping the system stable. Despite the fact that cascade PID control required considerable tuning, it was successful. The system was able to perform reference tracking with fast time responses, small overshoot, and steady-state error.

Finally, it was concluded that the Aero 2 platform was not suitable for quadcopter modelling and control. Two approaches were discussed and compared. The mathematical approach provided an invertible matrix that decouples pitch and yaw angles (theoretically), yet it contradicted the real dynamics of the system as the system did not fly. Regarding the mechanical approach, the system could be viable but the yaw angle would be too constrained, resulting in a 1-DOF system, which did not meet the required behaviour. Nevertheless, it was interesting to test the limits of this system.

5.2 Future Work

The experiments and simulations described in this paper could be developed and evaluated further, considering that the proposed solutions were not completely successful.

First, the mathematical models for each configuration could be derived again using other system identification methods, such as the MATLAB system identification toolbox, to get more accurate models, thus designing better and accurately tuned PID controllers. Other control systems such as Fuzzy logic and multivariable control could also be tested, and compared to determine the best control approach.

Second, the geared system proposed to control the Aero 2 as a quadcopter could also be designed and built to evaluate the real limits of this solution.

Finally, an actual 6-DOF drone could be used and a PID control system could be implemented on a flight controller using C programming language. After designing the flight control system, a vision system could be then integrated with the system, and tested with wind turbine inspection.

References

[1] in H. | September 7th and 2021 L. a Comment, “The First Air Raid Happened When Austria Dropped Bombs on Venice from Pilotless Hot-Air Balloons (1849) | Open Culture.” https://www.openculture.com/2021/09/the-first-air-raid-in-history.html

[2] J. Borger and J. B. W. affairs editor, “US shoots down suspected Chinese spy balloon over east coast,” *The Observer*, Feb. 04, 2023. Available: https://www.theguardian.com/us-news/2023/feb/04/chinese-spy-balloon-shot-down-us

[3] “A Brief History of Drones,” *Imperial War Museums*. https://www.iwm.org.uk/history/a-brief-history-of-drones#:~:text=The%20first%20pilotless%20vehicles%20were

[4] J. P. Duffy *et al.*, “Location, location, location: considerations when using lightweight drones in challenging environments,” *Remote Sensing in Ecology and Conservation*, vol. 4, no. 1, pp. 7–19, Aug. 2017, doi: https://doi.org/10.1002/rse2.58.

[5] A. S. Saeed, A. B. Younes, S. Islam, J. Dias, L. Seneviratne and G. Cai, "A review on the platform design, dynamic modelling and control of hybrid UAVs," *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, Denver, CO, USA, 2015, pp. 806-815, doi: 10.1109/ICUAS.2015.7152365.

[6] M. Idrissi, M. Salami, and F. Annaz, “A Review of Quadrotor Unmanned Aerial Vehicles: Applications, Architectural Design and Control Algorithms,” *Journal of Intelligent & Robotic Systems*, vol. 104, no. 2, Jan. 2022, doi: https://doi.org/10.1007/s10846-021-01527-7.

[7] “V-22 Osprey,” Boeing Defense, Space and Security

[8] O. McAree, J. M. Aitken and S. M. Veres, "A model-based design framework for safety verification of a semi-autonomous inspection drone," *2016 UKACC 11th International Conference on Control (CONTROL)*, Belfast, UK, 2016, pp. 1-6, doi: 10.1109/CONTROL.2016.7737551.

[9] “The PID Controller & Theory Explained,” https://www.ni.com/en-gb/shop/labview/pid-theory-explained.html

[10] V. Kangunde, R. S. Jamisola, and E. K. Theophilus, “A review on drones controlled in real-time,” *International Journal of Dynamics and Control*, Jan. 2021, doi: https://doi.org/10.1007/s40435-020-00737-5.

[11] M. H. Jafri *et al.* (Aug. 2017). Development of Fuzzy Logic Controller for Quanser Bench-Top Helicopter. Presented at IOP Conf. Ser.: Mater. Sci. Eng. 260 012015. [Online]. Available: https://iopscience.iop.org/article/10.1088/1757-899X/260/1/012015/pdf

[12] Zhang, Z. Liu, J. Zhao, and S. Zhang, "Modeling and attitude control of Bi-copter," *2016 IEEE International Conference on Aircraft Utility Systems (AUS)*, Beijing, China, 2016, pp. 172-176, doi: 10.1109/AUS.2016.7748042.

[13] Fahmizal, M. Arrofiq, E. Apriaskar and A. Mayub, "Rigorous Modelling Steps on Roll Movement of Balancing Bicopter using Multi-level Periodic Perturbation Signals," *2019 6th International Conference on Instrumentation, Control, and Automation (ICA)*, Bandung, Indonesia, 2019, pp. 52-57, doi: 10.1109/ICA.2019.8916755.

[14] Quanser Inc, "Quanser Aero 2 User Manual", 2022. [Online]. Available: https://www.quanser.com/products/aero-2/

[15] Quanser Inc, "Quick Start Guide: Quanser Aero 2", 2022. [Online]. Available: https://www.quanser.com/products/aero-2/

[16] N. Ferry, “Quadcopter Plant Model and Control System Development With MATLAB/Simulink Implementation”, M.S. Thesis, Dept. Elect. Eng., Rochester Tech. Univ., Rochester, NY, USA, 2017. [Online]. Available: http://www.ritravvenlab.com/uploads/1/1/8/4/118484574/ferry.pdf

[17] R. Beard, "Quadrotor Dynamics and Control Rev 0.1", Brigham Young University, Feb. 19, 2008. Faculty Publications. 1325, [Online]. Available: https://scholarsarchive.byu.edu/cgi/viewcontent.cgi?article=2324&context=facpub

[18] E. Saif and İ. Eminoğlu, “Modelling of quad-rotor dynamics and Hardware-in-the-Loop simulation,” *The Journal of Engineering*, vol. 2022, no. 10, pp. 937–950, Aug. 2022, doi: https://doi.org/10.1049/tje2.12152.

[19] Quanser Inc, "Aero 2 Data Sheet", 2022. [Online]. Available: https://www.quanser.com/products/aero-2/

Appendices

A 2-DOF UAV Control

Fig. 18. 2-DOF Quadcopter Configuration

Fig.19 shows the physical parameters of the Quanser Aero 2 available in the datasheet.

Symbol	Description	Value	Units
	DC Motor		
V_{nom}	Nominal input voltage	18.0	Volts
τ_{nom}	Nominal torque	22.0	$\mathrm{mN \cdot m}$
$\dot{\omega}_{\mathrm{nom}}$	Nominal angular velocity	3050	RPM
I_{nom}	Nominal current	0.540	Amps
R_{m}	Terminal resistance	8.4	Ω
k_t	Torque constant	0.042	$\mathrm{N \cdot m/A}$
k_m	Motor back-EMF constant	0.042	$\mathrm{V/(rad/s)}$
J_m	Rotor inertia	4.0×10^{-6}	$\mathrm{kg \cdot m^2}$
L_m	Rotor inductance	1.16	mH
	Aero Body		
M_b	Mass of body	1.07	kg
D_m	Displacement of Center of Mass (z-axis)	−2.42	mm
J_p	Pitch inertia (y-axis)	2.32×10^{-2}	$\mathrm{kg \cdot m^2}$
J_y	Yaw inertia (z-axis)	2.38×10^{-2}	$\mathrm{kg \cdot m^2}$
D_t	Thrust displacement	0.168	m

Fig. 19. Quanser Aero 2 parameters [19]

The 2 DOF system was represented as a state space model using Eq. (27) and Eq. (30).

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases}$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{K_{sp}}{J_y} & 0 & -\frac{D_y}{J_y} & 0 \\ 0 & 0 & 0 & -\frac{D_z}{J_z} \end{bmatrix}, B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{K_{yy}D_t}{J_y} & \frac{K_{yz}D_t}{J_y} \\ \frac{K_{zy}D_t}{J_z} & \frac{K_{zz}D_t}{J_z} \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$x = \begin{bmatrix} \theta(t) \\ \psi(t) \\ \dot{\theta}(t) \\ \dot{\psi}(t) \end{bmatrix}, u = \begin{bmatrix} \omega_y(t) \\ \omega_z(t) \end{bmatrix}, y = \begin{bmatrix} \theta(t) \\ \psi(t) \end{bmatrix} \quad (38)$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -0.539 & 0 & -0.142 & 0 \\ 0 & 0 & 0 & -0.118 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.00191 & 0.0216 \\ -0.0189 & 0.0505 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad (39)$$

Table 6 grouped the estimated parameters of the 2-DOF helicopter configuration from experimentation.

Table 6. 2 DOF Parameter Estimation Results

Symbol	Value	Unit
K_{yy}	0.00017729	N/V
K_{sp}	0.0035	Nm/V
K_{yz}	0.00012826	N/V
D_y	0.000119	$N/m/(rad/s)$
K_{zz}	0.00042543	Nm/V
K_{zy}	-0.00016	N/m
D_z	0.0028	$N/m/(rad/s)$

Fig. 20 compared the 2-DOF pitch and yaw models to the Aero 2 helicopter configuration.

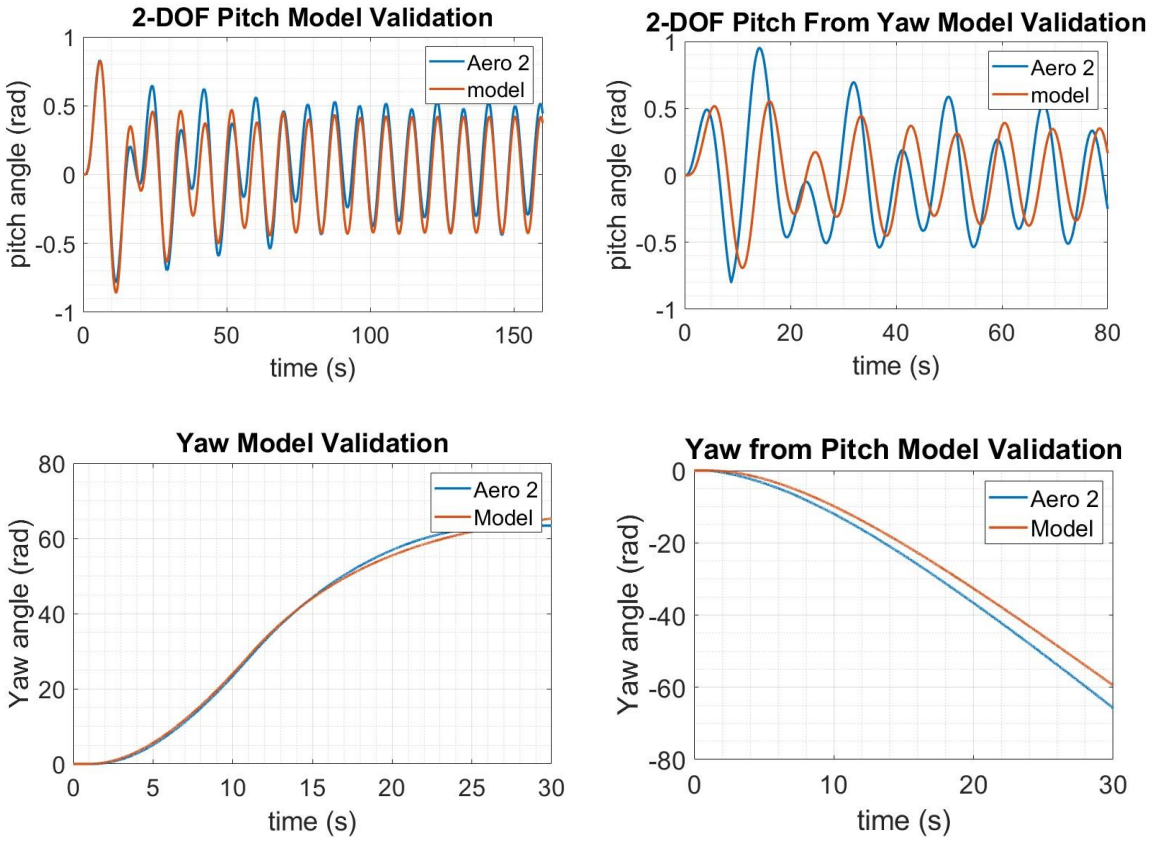

Fig. 20. 2-DOF Helicopter Model Validation

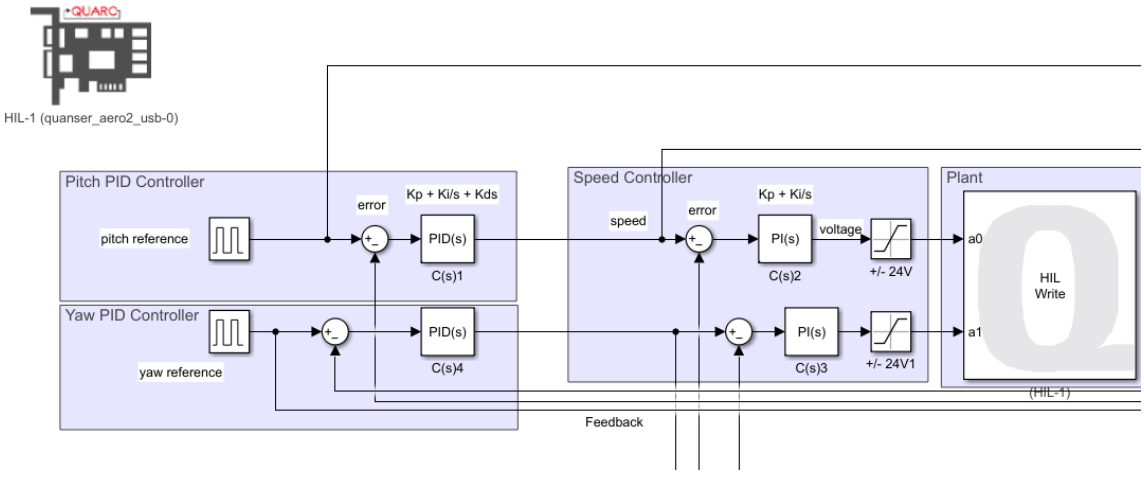

Fig. 21. 2-DOF PID controller Block Diagram

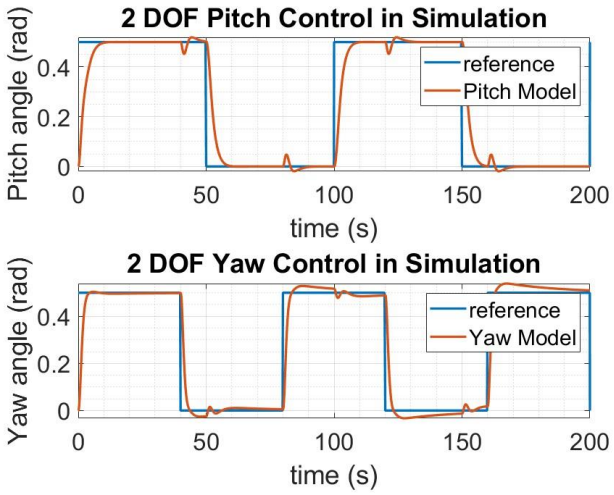

Fig. 22. 2-DOF Helicopter Control and tracking (Simulation)

Table 7. 2-DOF PID Pitch Control and System Performance

Yaw PID Controller Gains	Pitch controller gains	Peak Overshoot (%)	Steady-state error (rads)	Time Response (seconds)	5 % Settling time (seconds)
$k_P = 165, k_i = 0, k_d = 190$	$k_P = 350, k_i = 150, k_d = 400$	27.63	0.00436	25	7.76
	$k_P = 250, k_i = 90, k_d = 300$	19.47	0.0129	28	11.83

Table 8. 2-DOF PID Yaw Control and System Performance

Pitch PID Controller Gains	Yaw PID controller gains	Peak Overshoot (%)	Steady-state error (rads)	Time Response (seconds)	5 % Settling time (seconds)
$k_P = 350, k_i = 150, k_d = 400$	$k_P = 165, k_i = 0, k_d = 190$	1.02	0.027	3	1.74
	$k_P = 120, k_i = 0, k_d = 150$	0.61	0.078	3.6	1.98

B Risk Assessment

MANCHESTER 1824

The University of Manchester

General Risk Assessment Form

Date: (1)	Assessed by: (2)	Checked / Validated* by: (3)	Location: (4)	Assessment ref no (5)	Review date: (6)
14/10/2022	Mahmoud Shanan		Engineering Building (MECD)		

Task / premises: (7) **Drone control testing for wind turbine inspection**

The work will include a lot programming which will be performed from a pc on campus or at home. It will also involve data acquiring from sensors and a microcontroller.
Moreover, it will require flying a drone for control testing.

Activity (8)	Hazard (9)	Who might be harmed and how (10)	Existing measures to control risk (11)	Risk rating (12)	Result (13)
Use of display screen equipment (DSE)	Poor posture, repetitive movements, long periods looking at DSE (display screen equipment)	Student working on PC Repetitive strain injuries, neck and back pain, eye strain and/or fatigue	Provision of an adjustable chair, adjustable screen height, suitable and sufficient lighting is maintained in each area. Taking breaks to avoid eye strain and fatigue	Low	A
Computer use	Electric shock, burns, fires, electrocution	Student working on PC Electric shock, electrocution and/or burns	Avoid interfering with plugs, cables or any device, especially when any equipment is connected to the power supply. Report defective items to supervisor in the first instance. Avoid eating or drinking while working to minimise the risk of spillage onto electrical equipment.	Low	A

Result : T = trivial, A = adequately controlled, N = not adequately controlled, action required, U = unknown risk

Activity (8)	Hazard (9)	Who might be harmed and how (10)	Existing measures to control risk (11)	Risk rating (12)	Result (13)
Working from home	Lone working, electrical hardware failure	Home working student Anxiety, stress, electric shock, burns and fire	Remaining in contact with flatmates, friends or family Avoid working late at night and define working hours Visual checks to make sure electrical equipment and cables are free from defects Avoid daisy-chaining	Low	A
Flight testing of drone control (quadcopter)	Failure of propellers, uncontrolled behaviour on flight	Anyone present within the drone testing area Drone collision with person's body, sharp plastic parts scattering and damaging eyes	Drone is equipped with a 3D plastic mechanical frame including propeller guards. Testing in controlled room without human intervention	Low	A
Testing of drone control (quadcopter)	Electrical failure of batteries/microcontroller/sensors/motors	Anyone present within the drone testing area Electric shock, burns and fire	Visual and electrical checks before use to make sure equipment and wires are free from defects Testing in controlled room without human intervention Avoid overusing equipment	Low	A
Continuous testing of drone control (quadcopter)	Overheating	Anyone present within the drone testing area Skin Burn or fire	Limiting the time of testing per day Taking breaks between testing to let motors cool down	Low	A

Result : T = trivial, A = adequately controlled, N = not adequately controlled, action required, U = unknown risk

University risk assessment form and guidance notes.
Revised Aug07

Fig. 23. Project Risk Assessment