The Aeronautical Journal (2024), **128**, pp. 1767–1788
doi:10.1017/aer.2024.11

RESEARCH ARTICLE

Attitude modelling and real-time robust control of a 3-DoF quadcopter UAV test bench

S.M. Ahmad[1,2] and S. Fareed[3]

[1]Control & Instrumentation Engineering Department, King Fahd University of Petroleum and Minerals, Dhahran, Kingdom of Saudi Arabia, [2]Interdisciplinary Research Center for Intelligent Manufacturing and Robotics, King Fahd University of Petroleum and Minerals, Dhahran, Kingdom of Saudi Arabia and [3]Drone Robotics Pvt Ltd, The Catalyst GIK Incubator, Topi, Pakistan
Corresponding author: S.M. Ahmad; Email: sarvat.ahmad@kfupm.edu.sa

Received: 17 July 2023; **Revised:** 24 December 2023; **Accepted:** 23 January 2024

Keywords: quadcopter UAV; mathematical modelling; robust real-time control; system identification; LQR-FF

Abstract

In this work, a three degrees-of-freedom (3-DoF) static quadcopter unmanned aerial vehicle (UAV) test-rig of a pendulum-type configuration is custom-designed, developed, instrumented, and interfaced with a PC. The rig serves as a test bed to develop high-fidelity mathematical models as well as to investigate autopilot designs and real-time closed-loop controllers' performances. The Simulink Desktop Real-Time software is employed for the quadcopter's attitude signals acquisition and real-time implementation of closed-loop controllers on a target microcontroller hardware. The mathematical models for pitch, roll, and yaw axes are derived via the first principle and validated with the experimental linear system identification (SI) techniques. Subsequently, employing the multi-parameter root contour technique, the classical proportional integral derivative (PID) controllers are designed and implemented in real-time on the quadcopter UAV test rig. This served as a benchmark controller for comparing it with an integral-based linear quadratic regulator (LQR) controller. Further, to improve the transient response of the LQR controller, a novel robust integral-based LQR controller with a feedforward term (LQR-FF) is implemented, which shows much superior performance than the benchmark and basic LQR controller. This work thus will act as a precursor for a more complex 3-DoF autopilot design of an untethered quadcopter.

Nomenclature

Symbol	Name	Units
F	force	newton (N)
m	mass	kilogram (kg)
g	acceleration due to gravity	m/sec^2
τ	torque	N. m
I	moment of inertia	kg.m^2
b	angular damping coefficient	N.m.sec/rad
t	time	sec
θ	pitch angle	rad
ϕ	roll angle	rad
ψ	yaw angle	rad
k	angular spring coefficient	N.m/rad
r	distance b/w pivot and CG	meter
R	distance b/w CG and motor	meter

α	angle b/w arm and x-axis	rad
p, q, r	angular rates	rads/sec
L,M,N	sum of angular moments along roll, pitch and yaw axes, respectively	N. m

1.0 Introduction

Although underactuated i.e., six degrees-of-freedom (DoF) with only four actuators, quadcopters with their remarkable hovering capabilities have led to a spectacular spectrum of applications such as search and rescue [1], packet delivery [2], aerial photography and filmography, mapping [3], precision agriculture [4] patrolling [5], and military operations, just to name a few. Realisation of this versatile platform was in part made possible to the advancement in modern electronics, especially the Micro Electrical Mechanical System (MEMS) based Inertial Measurement Unit (IMU), microcontrollers, and efficient power storage technologies, resulting in the overall reduction of cost and size of on-board avionics of multirotor vehicles [6] and references therein. Despite advances made at the component level, developing a complex inherently unstable and multivariable quadcopter as a research platform is far from trivial.

Central to the developing prototype quadrotor UAV is the modelling, simulation, analysis, and stabilising control design [7–10]. An important first step in model-based control design is the development of high-fidelity dynamic system models. Controlling of quadcopters is a challenging task, as it exhibits nonlinear, underactuated, unstable, and coupled dynamics. To solve such highly complex quadrotor nonlinear modelling and control problems, several authors have adopted static test benches to varying degrees of success. In this approach, the multirotor is fixed on a static test rig that allows 3-DoF rotational movement but limits translation motion. For instance, Patel et al. [11] have utilised an off-the-shelf commercial 2-DoF helicopter for designing and testing an adaptive backstepping control scheme. While Rajappa et al. [12] have conducted their research on a representative helicopter modelling using a commercial static helicopter test rig. The estimation of the asymmetric inertia tensor of a static commercial type 3-DoF quadcopter UAV test-rig has been investigated by Dhaybi and Daher [13]. Drone-specific commercial test bench to check vehicle's reliability and operational robustness is also reported [14]. Bouabdallah et al. [7] and Mardan et al. [8] devised a 3-DoF static test rigs for evaluating Lyapunov-based angular rotation control and QFT control of a 2-DoF attitude control respectively. The latter [8] also utilised commercially available quadrotors made by 3D Robotics Company to perform experimental evaluations. To undertake altitude or hover control of a quadcopter UAV in a tunnel-like environment Vong et al. [9] conducted experiments utilising T-slot vertical aluminum extrusions test stand. A test platform similar to one developed by the authors is that of Beharie et al. [15], and the focus has been on estimating powerplant parameters alone. Indoor untethered testing of a UAV in a cluttered environment for accurate altitude determination of an object has been investigated in Ref. [33] and references therein. Untethered navigation of a micro aerial vehicle (MAV) for inspection of the dam's penstocks and pipelines utilising sensor fusion techniques was demonstrated by Özaslan et al. [34].

Numerous control techniques have been proposed to stabilise the unstable multirotor aircraft (or quadcopters mounted on static test benches) from the classical controller [21], modern state-space linear controllers [10] to more complex nonlinear controllers [22, 23], linear parameter varying controller [24] as well as computationally intensive fuzzy control [25]. Although these controllers are adequate for quadcopter stabilisation their performance evaluation is mainly reported through simulation. This is due to the difficulties and limitations of physically operating quadrotors in outdoor environments, coupled with authorisation required from the aviation authorities [21].

1.1 Research rationale and contributions

Evident from the above design, development, simulation, and testing of a flying quadcopter UAV is an iterative process that necessitates frequent modifications to mechanical/electronics hardware, (airframe,

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

test bench, sensors, and communication tether) access to the embedded control system as well as the freedom to incorporate digital filters, compensators, and custom-designed controller interface software. These changes are not always possible with the off-the-shelf commercial test benches or quadrotors, an approach routinely taken by researchers.

As opposed to the above, the work presented here relies on an in-house designed, instrumented, and fabricated 3-DoF quadcopter UAV test-rig in pendulum configuration. That enables with easy access to all the critical sub-systems such as the real-time data acquisition system, replete with a low-cost microcontroller board, on-board sensors and motor drivers. As well as easily modifiable Matlab-Simulink desktop real-time interface software. Moreover, these features ensure easy access to important signals necessary to undertake experimental modelling and real-time control operations. In contrast to Beharie et al. [15], in this work in addition to the estimation of powerplant (actuator) dynamic models, the *first principle-based* Euler-Newtonian approach [10, 16] is also employed to identify a typical quadcopter's model structure. However, unlike in [7, 10, 21] wherein model structure is fixed *a priori*, in this research the data-driven *SI* approach [6, 17–19] is exploited here to not only capture the quadrotor dynamics but also unmodelled dynamics, sensor dynamics as well as the gains of actuator and power amplifiers. Confidence is further instilled in identified models through time-domain cross-validation tests. The methodology provides reliable and high-fidelity dynamic models necessary for the ensuing model-based control design.

Further, this paper addresses closed-loop control of a 3-DoF quadrotor employing benchmark PID controllers as well as modern multi-input-multi-output (MIMO) state-space LQR controllers. To ensure the desired closed-loop performance specifications are met, a novel combination of an LQR controller with a feedforward gain is also investigated and presented herein. The test bench will serve as a stepping stone for ongoing research on an inverted pendulum quadrotor configuration, ultimately leading to an untethered flight vehicle demonstration.

The outline of the paper is as follows: The upcoming sections describe the experimental setup followed by the experimentation procedure. The next section presents the detailed mathematical modelling of the developed 3-DoF test rig along with SI-validated models. Subsequently, the design of classical controllers and their implementation are presented. Finally, classical and modern full-state feedback controllers are designed and implemented in real-time. The last section concludes the paper.

2.0 Experimental setup

The experimental setup consists of a static quadcopter UAV test bench interfaced with a personal computer (PC) as shown in Fig. 1. The test bench consists of a custom-designed metallic frame, 3-DoF aluminum gimbal joint, and the quadcopter UAV frame. The quadcopter UAV is mounted on the 3-DoF joint such that it can rotate freely in pitch, roll, and yaw axes.

The 3-DoF joint enables the quadcopter UAV to rotate freely about the x, y, and z-axes yielding rotation in pitch (θ), roll (ϕ), and yaw (ψ) axes (attitude) while restricting the translational motion in the x, y, and z-axes. The actual and wireframe view of the quadcopter UAV along with the earth and body frame axes are shown in Fig. 2.

2.1 Quadcopter UAV test-rig

The quadcopter UAV consists of four motors, propellers, frame, electronic speed controllers (ESCs), IMU sensor, and microcontroller. The motors with propellers generate the desired up-thrust where motors 1 and 3 rotate in the clockwise direction, and motors 2 and 4 rotate in the counterclockwise direction, as depicted in Fig. 2. The IMU provides the orientation angles of vehicle in pitch, roll, and yaw axes. It has a three-axis accelerometer and a three-axes gyroscope. Accelerometers tend to have high-frequency noise, and gyroscopes tend to have low-frequency drift. Since both of them are not desirable, a complementary filter [20] is designed to filter out the noise and remove the drift from the sensor output (refer Fig. 3).

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

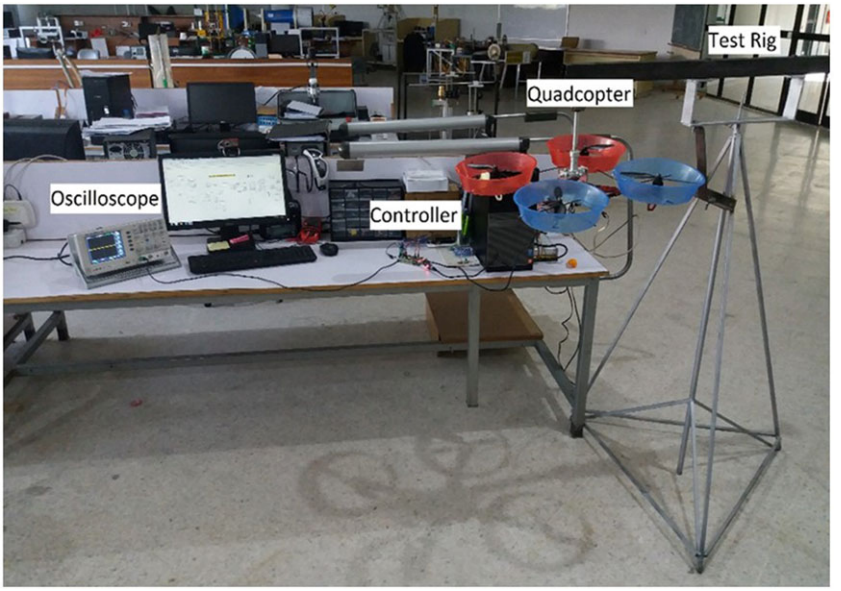

Figure 1. *Experimental setup of a 3-DoF quadcopter UAV test rig.*

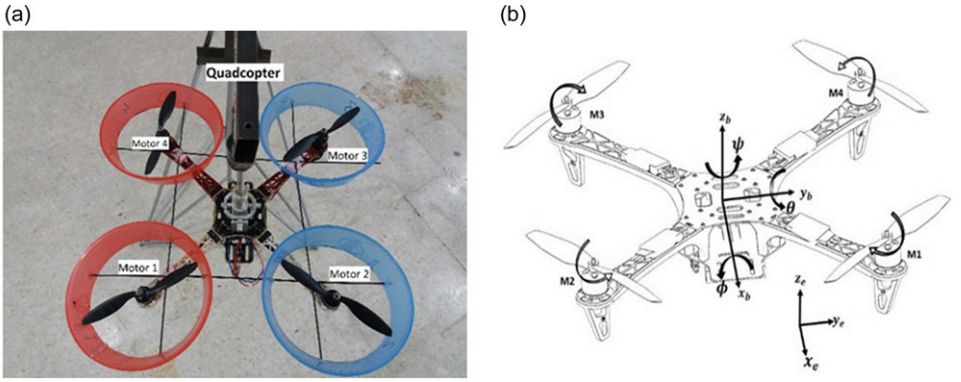

Figure 2. *Actual (a) and wireframe view (b) of quadcopter UAV including orientation axes.*

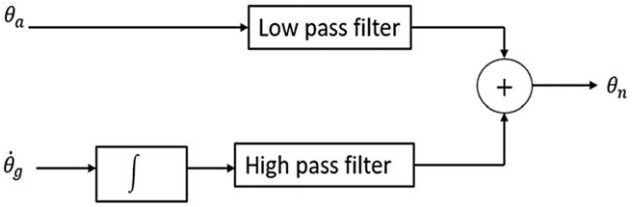

Figure 3. *Complementary filter architecture.*

By mixing data from the accelerometer and gyroscope, the expression for the complementary filter is obtained,

$$\theta_n = (1-\alpha)\left(\theta_{n-1} + \dot{\theta}_g dt\right) + (\alpha\theta_a) \tag{1}$$

$$\alpha = t_c/\left(t_c + 1/f_s\right) \tag{2}$$

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

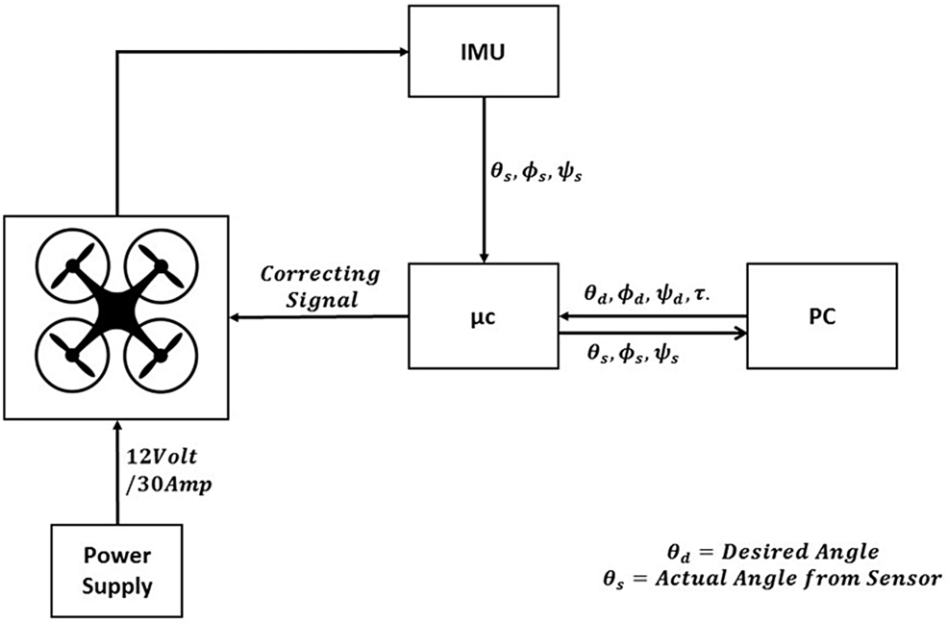

Figure 4. *Control architecture for a 3-DoF test rig.*

Where θ_n is the current output, θ_{n-1} is the previous output, $\dot{\theta}_g$ is the gyroscope data, θ_a is the accelerometer data, t_c is the time constant and f_s is the sampling frequency.

The microcontroller, which acts as the brain of the quadcopter, is the main data processing unit that processes the input data from the IMU sensor, input commands, and based on the embedded control algorithm, alters the speed of each motor accordingly. The microcontroller selected for this test rig is Atmel ARM cortex M3, 32-bit CPU mounted on Arduino Due. It has an 84MHz clock speed with 512KB of flash memory, giving it enough capabilities to execute the attitude control algorithm up to 100 times in a second. The data communication between the PC, microcontroller, and other components of the test rig is indicated in the control architecture as shown in Fig. 4.

2.2 Experimentation procedure

The test rig is interfaced with the PC via a USB communication cable which provides duplex data communication between the quadcopter UAV and the PC. Simulink Real-Time desktop software is utilised for deploying the control algorithms onto the microcontroller. For controlling the pitch, roll, and yaw angles (attitude), the speed of each motor needs to be controlled appropriately. For **X** or cross configuration of the quadcopter, as shown in Fig. 2, the pitch control can be achieved by increasing the speed of motors 1 and 2 (forward-facing) and decreasing the speed of motors 3 and 4 (back-facing) or vice versa. For roll control, the speed of motors 2 and 3 (left-facing) is increased and the speed of motors 1 and 4 (right-facing) is decreased or vice versa. The yaw control can be achieved by increasing the speed of one pair of diagonal motors and decreasing the speed of the other diagonal pair, simultaneously. The sampling frequency of 100Hz is selected for the control loop based on Nyquist frequency criteria to avoid aliasing phenomena, similar to that of Ref. [21]. As well as based on the desired closed-loop bandwidth of 3–5Hz, necessitating a sampling rate of about 20–30 times the bandwidth [30–32].

3.0 Mathematical modelling of a 3-DoF quadcopter

The multivariable quadcopter test rig exhibits a high level of nonlinearity and cross-coupling between pitch-roll-yaw degrees of freedom. The nonlinearity issue is addressed by employing small-angle

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

approximation and ignoring the higher terms thereby reducing the nonlinear coupled equation of motion (EOM) into linear second-order differential equation (refer Equation (3)). Since the test rig is pivoted at the centre of gravity (CG) point hence it has three angular degrees of freedom, with no translation degrees of freedom. Moreover, the CG and the body frame of reference coincide, resulting in three angular EOM for a body with three degrees of freedom.

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{pmatrix} \frac{I_y - I_z}{I_x} qr \\ \frac{I_z - I_x}{I_y} pr \\ \frac{I_x - I_y}{I_z} pq \end{pmatrix} + \begin{pmatrix} \frac{1}{I_x} L \\ \frac{1}{I_y} M \\ \frac{1}{I_z} N \end{pmatrix} \tag{3}$$

Where, p, q, r are the angular rates in body frame. Due to the geometric symmetry of the quadcopter in the roll and pitch planes, the dynamics of both axes can be assumed to be the same. That is, due to the axes of symmetry ($I_y = I_z$; $I_z = I_x$) of the quadcopter the cross-coupling inertia terms are neglected. Furthermore, justification for neglecting the terms pr and pq is that the terms q, r are not large and if p is small then their products can be neglected. Thus, Equation (3) reduces to three uncoupled EOM.

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{pmatrix} \frac{1}{I_x} L \\ \frac{1}{I_y} M \\ \frac{1}{I_z} N \end{pmatrix} \tag{4}$$

Where, L, M, and N are the summation of external torques acting on the quadcopter. Furthermore, when quadcopter undergoes small perturbation following relations holds [35]

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} \tag{5}$$

Where ϕ, θ, ψ are the quadcopter's roll, pitch, and yaw angles, respectively.

3.1 Pitch and roll math modelling

Due to symmetric pitch and roll axes in the quadcopter, the mathematical model for pitch and roll axes are same. Therefore, only pitch axis modelling is considered in this section; however, this is also true for the roll dynamics. The mathematical model of the pitch axis of the quadcopter UAV mounted on the test rig is extracted from Equations (4) and (5) (refer Fig. 5).

$$I_y \dot{q} = M \tag{6}$$

Where $M = \tau_\theta(t) - b_\theta \dot{\theta}(t) - k_\theta \theta(t)$ and replacing $\dot{q} = \ddot{\theta}$, Equation (6) becomes:

$$I_y \ddot{\theta}(t) + b_\theta \dot{\theta}(t) + k_\theta \theta(t) = \tau_\theta(t) \tag{7}$$

Where θ is the pitch angle, b_θ is the damping coefficient in pitch axis and k_θ is the restoring spring coefficient. The pitching torque about pivot is denoted with τ_θ by I_y representing the moment of inertia in pitch plane.

Taking Laplace transform of Equation (7) yields

$$T_\theta(s) = I_y s^2 \theta(s) + b_\theta s \theta(s) + k_\theta \theta(s) \tag{8}$$

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

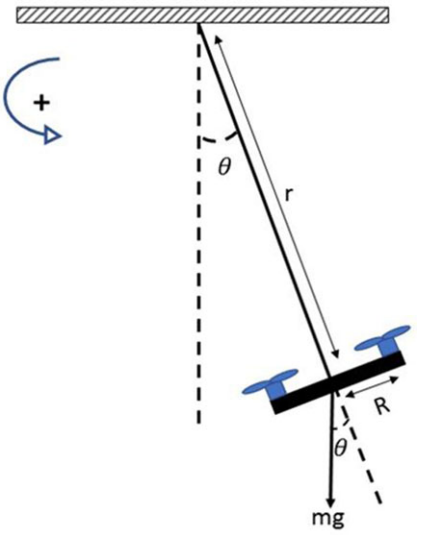

Figure 5. *Schematic of pitch axis of a 3-DoF test rig.*

The stiffness term 'k_θ' arising due to gravitational force, similar to a simple pendulum, which can be represented with the expression:

$$k_\theta = mgr\,sin(\theta)$$

The stiffness term is nonlinear due to the presence of trigonometric function i.e., $sin(\theta)$. Using the small-angle perturbation theory, the nonlinear term can be linearised around hover state at which $\theta = 0^o$, so

$$sin(\theta) \approx \theta$$

Then, Equation (8) becomes

$$T_\theta(s) = I_y s^2\theta(s) + b_\theta s\theta(s) + mgr\theta(s) \tag{9}$$

Rearranging Equation (9) to obtain transfer function:

$$\frac{\theta(s)}{T_\theta(s)} = \frac{1}{I_y s^2 + b_\theta s + mgr} \tag{10}$$

The pitching torque 'T_θ' is the combined torque exerted by of all four motors, which can be expressed as

$$T_\theta(s) = 4RFcos\,(\alpha) \tag{11}$$

Where R denotes the moment arm, which is the distance between motor and pivot point. And F represents the force/thrust produce by each motor mounted at an angle α with respect to x-axis. Substituting Equation (11) in Equation (10) gives pitch transfer function as

$$\frac{\theta(s)}{F(s)} = \frac{4Rcos\,(\alpha)}{I_y s^2 + b_\theta s + mgr} \tag{12}$$

Writing in standard second-order transfer function format gives

$$\frac{\theta(s)}{F(s)} = \frac{4Rcos\,(\alpha)\,/I_y}{s^2 + \frac{b_\theta}{I_y}s + \frac{mgr}{I_y}} \tag{13}$$

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

Table 1. *Parameters of 3-DoF test rig*

Parameter	Symbol	Value	Unit
Pitch moment of inertia	I_y	0.01	kg.m^2
Damping coefficient	b_θ	0.02658	Nm.sec
Mass	m	0.780	kg
Length	r	0.02	m
Moment arm	R	0.225	m
Arm angle	α	0.785	rad
Acc due to gravity	g	9.81	m/sec^2

The moment of inertia I_y was estimated using bifilar pendulum test and damping coefficient 'b_θ' is determined through the log decrement method. The remaining parameters of Equation (13) are dependent on the physical properties and configuration of the quadcopter, which are listed in Table 1.

Substituting parameters from Table 1 in Equation (13) gives

$$\frac{\theta(s)}{F(s)} = \frac{63.6}{s^2 + 2.66s + 15.3} \tag{14}$$

Equation (14) is the transfer function model of the pitch/roll axis of the quadcopter. The analytical model derived from the first principle can be validated by linear system identification technique and discussed in the next section.

3.1.1 System identification-pitch/roll axis

In this research, data driven linear system identification (SI) technique is employed to extract high fidelity dynamic models via experimentation. The SI modelling process involves three iterative steps, that is, (i) defining a model structure, (ii) estimating the model parameters, and (iii) validating the estimated model. The quadrotor's pitch axis is excited with an input signal rich in eigen-frequencies, thereby exciting the pertinent system modes. The corresponding output response in conjunction with the excitation signal forms an input-output pair that is fed to MATLAB System Identification Toolbox. The input-output data is pre-processed to remove the bias and outliers, before assigning the model order. Given *a prior* knowledge of the model order and structure from Equation (7) arriving at a transfer function domain was straightforward. The resulting pitch/roll axis identified model is of second-order with complex poles ($-1.6500 \pm 3.2981i$):

$$\frac{\theta(s)}{F(s)} = \frac{60}{s^2 + 3.3s + 13.6} \tag{15}$$

The model validation is carried out by employing signal not used for developing the model. As can be observed from Fig. 6 the predictive capability of the identified model is good, as it closely traces the quadrotor's output.

3.2 Yaw axis math modelling

The yaw dynamics of the quadcopter UAV mounted on the test rig can be modeled employing Equations (4) and (5). Thus, the EOM for the yaw axis can be written as:

$$I_z \dot{r} = N \tag{16}$$

Where $N = \tau_\psi(t) - b_\psi \dot{\psi}$ and replacing $\dot{r} = \ddot{\psi}$, Equation (16) becomes:

$$I_z \ddot{\psi} + b_\psi \dot{\psi} = \tau_\psi(t) \tag{17}$$

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

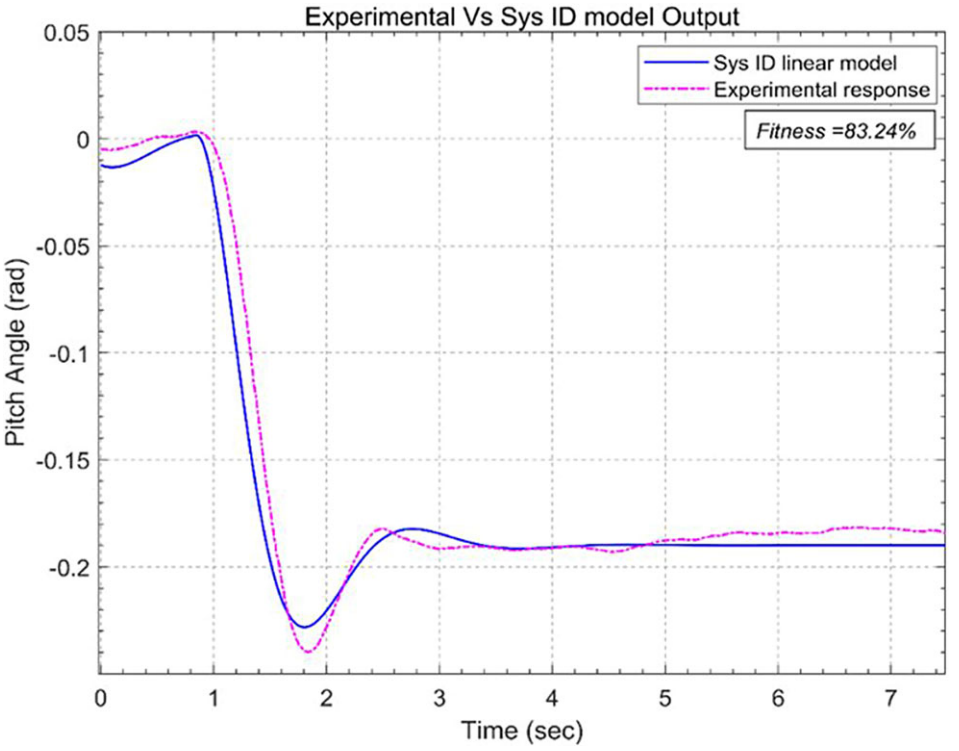

Figure 6. *Pitch/roll axis cross-validation.*

For quadcopter UAV mounted on the test rig, the variable of interest in the yaw axis is yaw rate r instead of yaw angle ψ. Thus, Equation (17) may be written as:

$$I_z \frac{dr}{dt} + b_\psi r_\psi = \tau_\psi(t) \tag{18}$$

$$G_\psi = \frac{r(s)}{\tau_\psi(s)} = \frac{1/I_z}{s + b_\psi/I_z} \tag{19}$$

Where 'b_ψ' is the damping coefficient in yaw axis and cannot be determined experimentally due to the physical limitation of the +test rig. However, the system identification technique can be utilised to identify the yaw transfer function based on experimental data. Therefore, a first-order linear model, with only one pole needs be identified utilising the open-loop step response data. This is discussed next.

3.2.1 System identification-Yaw axis

For system identification, the open-loop test is conducted in which the test signal applied to the quadcopter UAV is a pulse signal i.e., the speed of diagonal motors is changed, and the output yaw rate is measured. The input torque and output yaw rate are shown in Fig. 7.

The open-loop data is pre-processed and iteratively a first-order linear transfer function model with one pole and a gain is identified as:

$$G_\psi = \frac{306}{s + 1.196} \tag{20}$$

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

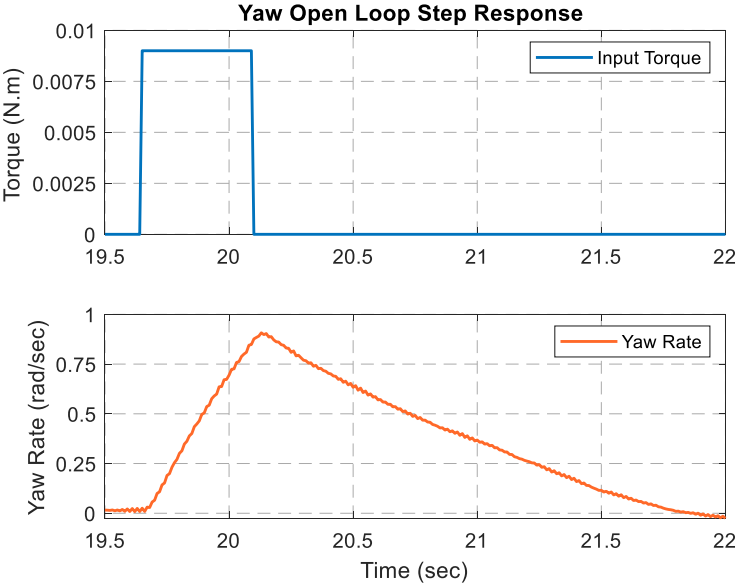

Figure 7. *Yaw axis open-loop pulse response.*

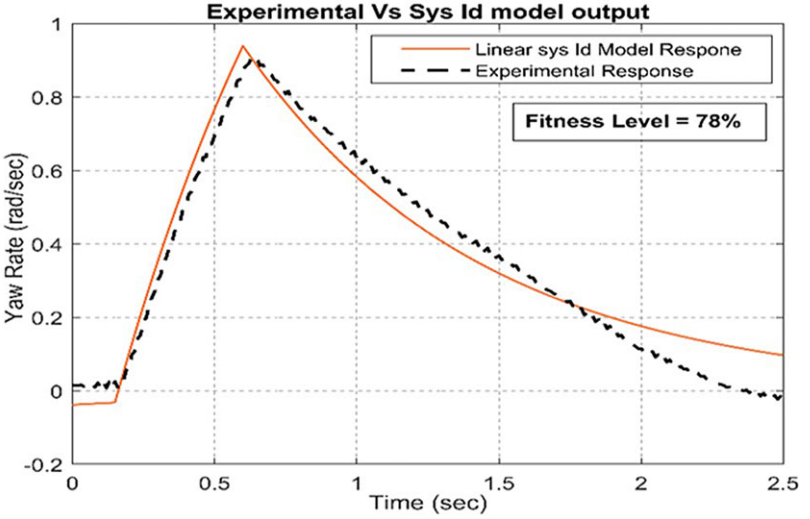

Figure 8. *Self validation test – yaw axis.*

The transfer function obtained via the system identification technique in Equation (20), needs to be validated. Figure 8 illustrates the system model predicted response for the training data and is observed that the model (solid line) tracks closely the system output (dashed line).

A fitness level of 78% is achieved between the SI model and the experimental response model fit. Nonetheless, the crucial part of SI is the time domain cross validation. In this test, data not used for the SI is utilised to evaluate the model's predicting capability. For obtaining open-loop data for cross-validation, a new test is conducted in which a multistep input is provided to the yaw axis and output response is recorded. The experimental yaw response of the test rig to multistep input is shown in Fig. 9.

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

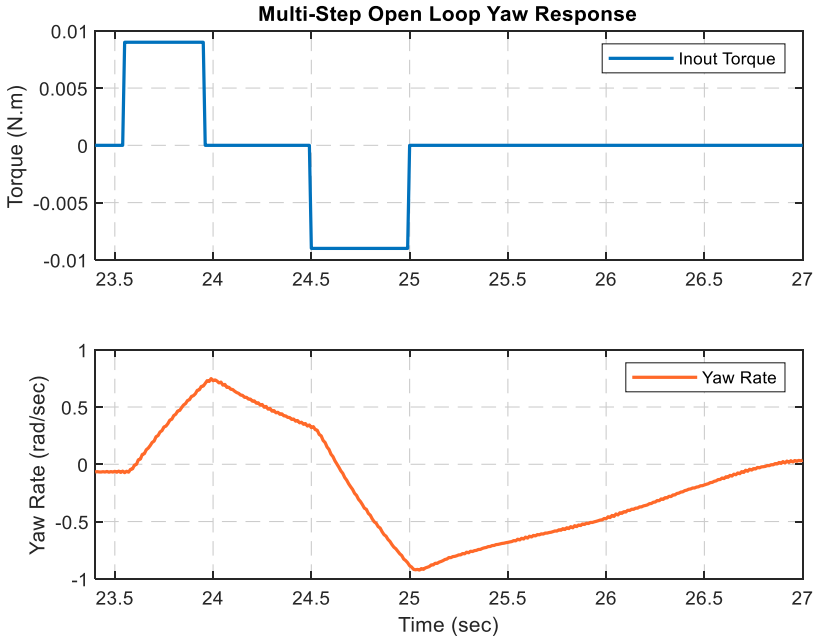

Figure 9. *Open-loop yaw response for a multi- step input.*

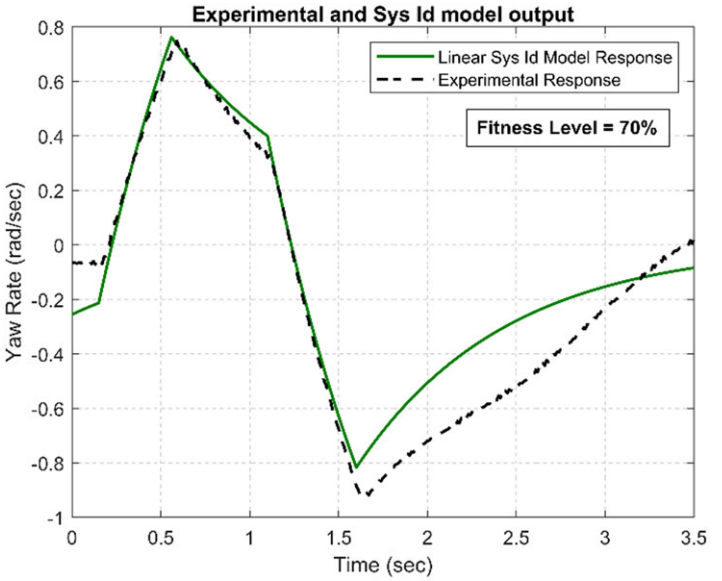

Figure 10. *Cross-validation test for yaw axis.*

The model predicted output is superimposed on the quadcopter UAV yaw response as depicted in Fig. 10.

The fitness level of 70% was achieved between the system identification model and multistep experimental data, which shows that identified model is rather a very good representation of the real system and the identified model can now be confidently used for designing the closed-loop feedback controller for the yaw axis.

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

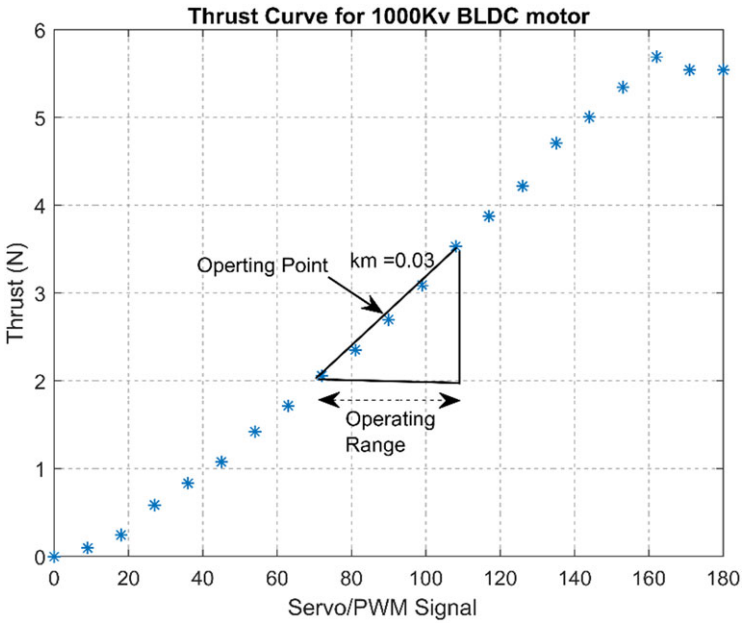

Figure 11. *Thrust curve for 1,000KV BLDC motor with 1,045 propeller.*

Actuator modelling

Actuator is an important component of any control system and needs to be accurately modelled to design a better controller for the plant. In quadcopters, the motor along with a propeller is regarded as the actuator. Each motor propeller combination can be modeled as a first-order transfer function, as discussed in the propulsor model given in Ref. [10]. The standard first-order transfer function is

$$G_{a,F} = \frac{F}{u_s} = \frac{k_m}{t_m s + 1} \tag{21}$$

Where u_s is the input servo PWM signal sent to the ESC which ranges between 0–180. And k_m is the motor gain with t_m representing the time constant of each motor/actuator. The motor gain 'k_m' can be determined by computing the slope of the thrust curve at normal operating (hover) condition, which is at half throttle. For this purpose, an experimental setup is developed with a load cell to measure the thrust. The motor is fitted with a propeller and the output thrust is measured with a load cell. The thrust curve is generated between the input servo pulse width modulated (PWM) signal and output motor thrust, which is shown in Fig. 11.

The motor gain 'k_m' is computed in the linear operating region around the hover point, as shown in Fig. 11. The slope of the thrust curve in the operating range is thus obtained as: $k_m = 0.03$. A similar thrust measuring setup is developed in Refs [10, 17] to measure the time constant 't_m' for the BLDC motor and propeller combination. While the time constant obtained in Ref. [10] is $t_m = 0.1$sec

Substituting values of 'k_m' and 't_m' in Equation (21) gives

$$G_{a,F} = \frac{F_N}{u_s} = \frac{0.03}{0.1s + 1} \tag{22}$$

The above expression is the actuator linear thrust model for pitch and roll axes with servo PWM signal as an input and thrust force as an output. Similarly, the reaction torque model for the yaw axis is obtained as

$$G_{a,R} = \frac{T_{N.m}}{u_s} = \frac{0.0006}{0.1s + 1} \tag{23}$$

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

All the important dynamics of the physical system including plant and actuators have been captured in the linear mathematical model(s). Next, the developed mathematical models are employed to design the feedback controllers which should augment the open-loop system in such a way that it meets the specified closed-loop design requirements.

4.0 Feedback controller design

In the classical control schemes, the most popular controller is the PID controller. About 80% of the industrial control is based on it, and nearly all the commercially available open-source autopilots use it. Therefore, the next section discusses the designing and implementation of a classic PID controller, which will be regarded as a benchmark controller. Its performance will be compared subsequently with the modern LQR controller. Nonetheless, plant augmented with designed controllers are expected to meet the following design requirements:

1. The rise time of less than 1.5 seconds in the pitch/roll axis.
2. Settling time of less than 2 seconds for pitch axis with response settling within ±5% of the final steady-value.
3. Maximum overshoot of 10% or less for pitch/roll axis.
4. Zero steady-state error for pitch, roll, and yaw axes.
5. Rise time of 5 sec and settling time of 8 sec for yaw axis.
6. Actuator should not saturate i.e.; control efforts should be in the range of ±50% pwm.

4.1 Benchmark PID controller

The PID controller provides the correcting signal to the plant based on the error which is the difference between the commanded setpoint and the actual output of the system. The mathematical expression of the PID controller is as follows.

$$u(t) = k_p e(t) + k_i \int e(t) + k_d \frac{de}{dt} \tag{24}$$

Where k_p represents the proportional gain, k_i is the integral gain and k_d is the derivative gain. 'u' represents the correcting/controller signal and 'e' represents the error which is the difference between the commanded input and actual output. Taking the Laplace transform of Equation (24) yields:

$$\frac{u(s)}{e(s)} = \frac{k_d s^2 + k_p s + ki}{s} \tag{25}$$

The process of designing the PID controller is the determination of coefficients of Equation (25) that is, k_p, $k_{i,}$ and k_d gains. Therefore, subsequent section describes the designing of pitch, roll, and yaw controllers via the multi-parameter root contour technique.

4.1.1 Multi-parameter root contour technique

Due to the exact symmetry between the pitch and roll planes, the mathematical model for the roll axis is the same as that of the pitch axis. Therefore, the controller designed for the pitch axis is expected to work well for the roll axis as well. Also, for designing the controller, the dynamics of the actuators can be assumed to be very fast compared to the plant dynamics. Therefore, the transfer function of actuators can be approximated by a constant,

$$G_a = \frac{0.03}{0.1s + 1} \cong 0.03 \tag{26}$$

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

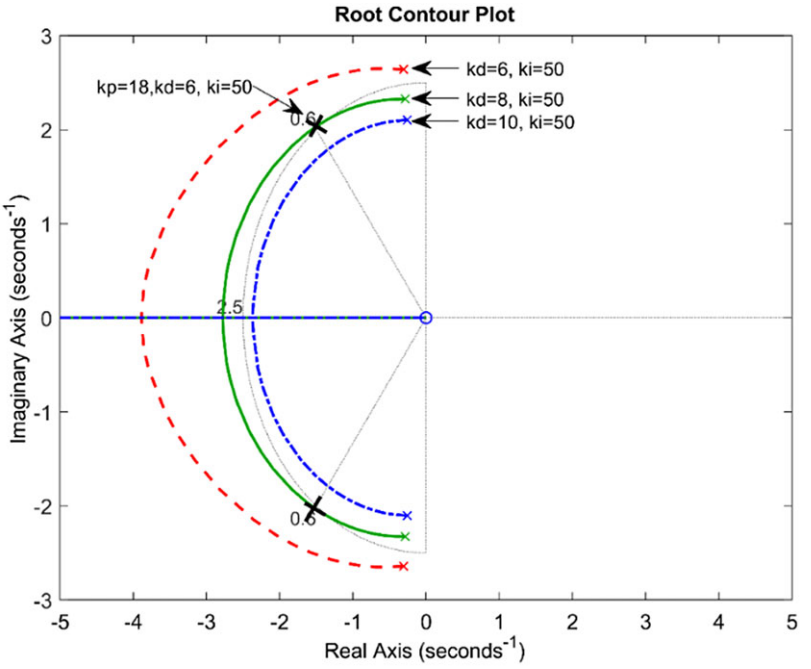

Figure 12. *Root contour plot of pitch controller.*

The closed-loop transfer function of the pitch axis is

$$G_{cl} = \frac{1.9\left(k_d s^2 + k_p s + ki\right)}{s^3 + (2.66 + 1.9k_d)s^2 + \left(15.3 + 1.9k_p\right)s + 1.9ki} \tag{27}$$

The controller design is based on the characteristic equation of closed-loop transfer function, which is the denominator expression of Equation (27):

$$s^3 + (2.66 + 1.9k_d)s^2 + \left(15.3 + 1.9k_p\right)s + 1.9ki \tag{28}$$

For designing the controller, multi-parameter root contour plot was generated using the characteristics equation. The multi-parameter root contour plot yields the location of closed-loop poles for several combinations of k_i and k_d gains with varying k_p gain [28]. The gains at which system closed-loop poles reached the desired poles location are the required controller gains. Assuming dominant second-order system characteristics, the pitch/roll design requirements of 2 *sec* settling time and 5% maximum overshoot corresponds to the desired poles locations as:

$$s = 1.5 \pm 2i \tag{29}$$

The root contour plot for Equation (28) is generated via MATLAB, which is shown in Fig. 12. The location of the desired closed-loop poles is indicated by the cross marks in Fig. 12.

From the root contour plot, the poles of the closed-loop pitch transfer function reach the desired poles location at gain values shown in Table 2. As the pitch and roll axes are symmetric, hence roll controller is the same as that of the pitch axis. Similar, to the pitch controller design, the yaw axis proportional-integral (PI) controller is also designed using the multi-parameter root contour plot and the resulting yaw controller parameters are also shown in Table 2. All the controllers were evaluated and fine-tuned in simulations before implementing on physical test rig.

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

Table 2. *Pitch PID controller gains obtained by root contour method*

Axis	k_p	k_i	k_d
Pitch	18	50	8
Roll	18	50	8
Yaw	10	50	–

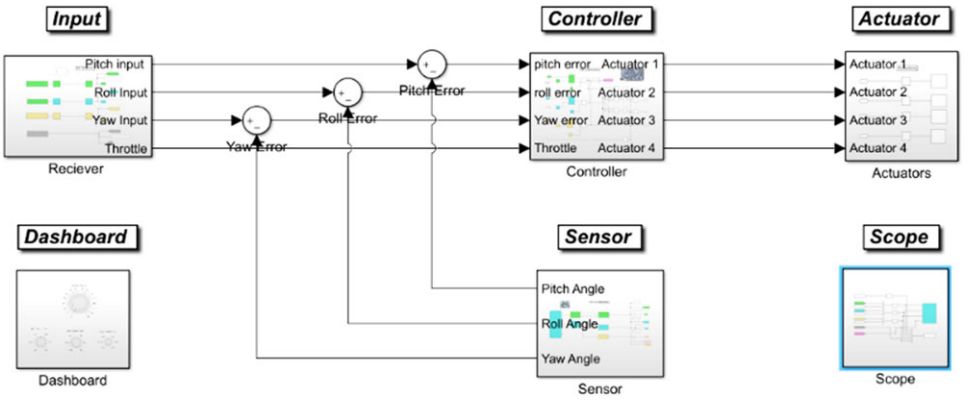

Figure 13. *Simulink GUI for 3-DoF PID controller implementation.*

4.2 Three DoF-coupled PID controller implementation

The 3-DoF-coupled closed-loop controllers are implemented on the quadcopter UAV test bench employing Simulink Desktop Real-Time software. The Simulink GUI is also developed to implement the 3-DoF control on the test rig for data acquisition and its real-time visualisation on a desktop PC. The Simulink-based GUI is shown in Fig. 13.

The step input commands are provided in pitch, roll, and yaw axes at different time intervals, and the corresponding output responses are recorded, which are also shown in Fig. 14. To avoid the saturation of actuators, the controllers' effort is also monitored during testing, which is also shown in the fourth subplot of Fig. 14

As evident from Fig. 14, pitch and roll responses have settling time of greater than 3 seconds with overshoot greater than 10%. Therefore, the PID controller does not meet the pitch/roll design requirements. Usually, this type of response is commonly observed with the PID controllers as the control action is based on the error signal alone hence making it unable to respond appropriately due to lack of detailed information about the system states.

On the other hand, a full-state feedback controller utilises all the important states of the system to take an appropriate control action, which normally results in a better response while classical control design just relies on output feedback. Nonetheless, in a quadcopter, since all the relevant states (attitude, rates) are available, modern state space-based full state feedback control design is an attractive candidate for designing a better pitch/roll controller that can meet all the design requirements. Moreover, the yaw response for PI controller meets all the desired performance requirement. Hence, there is no need to design the modern full state feedback controller for yaw axis. Also, during the yaw testing, it was observed that BLDC motors were sometimes approaching the saturation limits due to integral windup. Therefore, yaw controller was deactivated during future testing to prevent any damage to actuators or test rig.

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

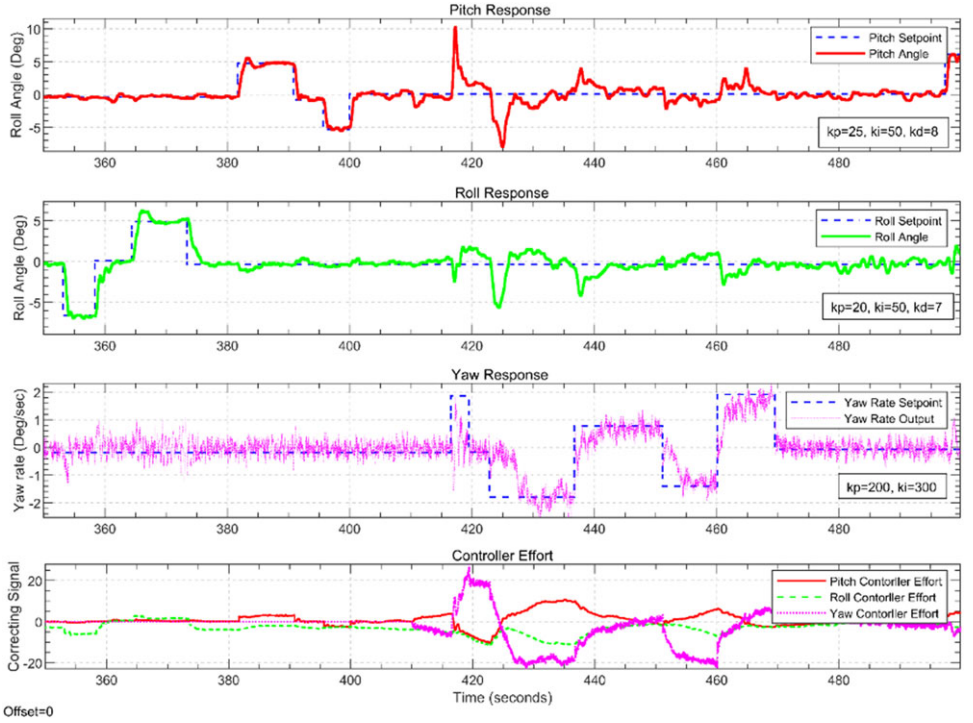

Figure 14. *Quadcopter closed-loop responses with PID controller implementation.*

4.3 LQR controller

LQR is an optimal control design technique that determines the controller gain matrix '$\boldsymbol{K}$' of the optimal control vector as

$$\boldsymbol{u}(t) = -\boldsymbol{K}\boldsymbol{x}(t)$$

to minimise the performance index J

$$J = \int_0^\infty (\boldsymbol{x}^*\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}^*\boldsymbol{R}\boldsymbol{u})\, dt$$

Where $\boldsymbol{Q}$ is a positive-definite Hermitian or real symmetric matrix and $\boldsymbol{R}$ is a positive definite Hermitian or real symmetric matrix. The $\boldsymbol{Q}$ and $\boldsymbol{R}$ matrices account for the relative importance of state error and input, respectively [20]. The matrix $\boldsymbol{Q}$ is called as a state penalisation matrix and $\boldsymbol{R}$ is called as input penalisation matrix. The LQR controller determines the gain '$\boldsymbol{K}$' matric based on the provided penalisation matrices. The penalisation matrices can be selected intuitively based on the experience of the control systems engineer or can be systematically determined based on some performance index like integral of absolute error (IAE), integral of time and absolute error (ITAE), etc. But before designing the pitch/roll LQR controller, the transfer function of the pitch/roll axis needs to be represented in state-space form as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -2.66 & -15.3 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$y = [1 \quad 0]\,\boldsymbol{x} + [\,0\,]\,u \tag{30}$$

The pitch axis of quadcopter UAV exhibits dominant second-order system characteristics with no inherent integral action, which means the open-loop plant will have a certain steady-state error for a step input. Therefore, the full-state feedback controller needs to augment the plant in such a way that

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

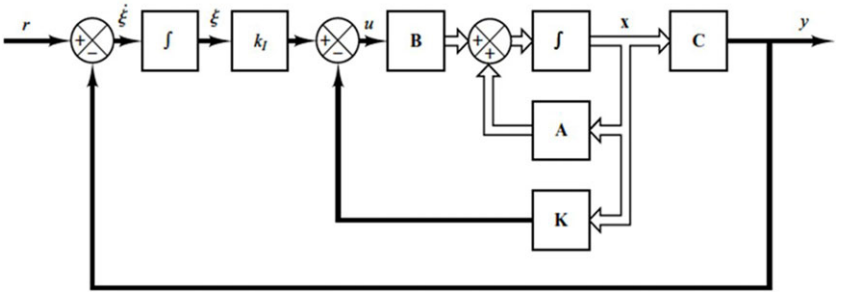

Figure 15. *Block diagram representation of type 1 servo system.*

it exhibits no steady-state error. This can be achieved by taking an integral action based on the output error. The integral action will eliminate the steady-state error from the tracking response. Hence, this integral type of control architecture is selected, the block diagram representation of the type 1 servo system is shown in Fig. 15 [29, 30].

Where 'ξ' represents the output of integer and 'r' represents the reference input. And K denotes the feedback gain with 'k_I' representing the integral gain. From the block diagram, we obtain

$$\dot{x} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u$$

$$y = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}u$$

$$u = -\boldsymbol{K}\boldsymbol{x} - k_I\xi \quad \text{and}$$

$$\dot{\xi} = r - y = r - \boldsymbol{C}\boldsymbol{x}$$

The above integral-based servo control problem needs be first converted to a regulator problem for designing the LQR controller. For brevity, the complete mathematical formulation of this conversion is not shown in this paper. However, the reader is referred to Ref. [29] for a complete mathematical background of type 1 servo systems. After the conversion of servo problem into regulator problem, the final form of state-space representation is given by:

$$\begin{bmatrix} \dot{\boldsymbol{x}}_e(t) \\ \dot{\xi}_e(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & 0 \\ -\boldsymbol{C} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_e(t) \\ \xi_e(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B} \\ 0 \end{bmatrix} u_e(t) \tag{31}$$

$$u_e(t) = -\boldsymbol{K}\boldsymbol{x}_e(t) + k_I\xi_e(t) \tag{32}$$

Defining a new (n+1)$^{\text{th}}$ order error vector $e(t)$ as

$$e(t) = \begin{bmatrix} \boldsymbol{x}_e(t) \\ \xi_e(t) \end{bmatrix}$$

Then, Equation (26) becomes

$$\dot{\boldsymbol{e}} = \tilde{\boldsymbol{A}}\boldsymbol{e} + \tilde{\boldsymbol{B}}u_e \tag{33}$$

Where,

$$\tilde{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{A} & 0 \\ -\boldsymbol{C} & 0 \end{bmatrix}, \quad \tilde{\boldsymbol{B}} = \begin{bmatrix} \boldsymbol{B} \\ 0 \end{bmatrix}$$

So, Equation (27) can be written as

$$u_e = -\tilde{\boldsymbol{K}}\boldsymbol{e} \tag{34}$$

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

Where

$$\tilde{\boldsymbol{K}} = \begin{bmatrix} \boldsymbol{K} & \vdots & -k_I \end{bmatrix}$$

The error state equation can be obtained by substituting Equation (29) into Equation (28) as

$$\dot{\boldsymbol{e}} = \left(\tilde{\boldsymbol{A}} - \tilde{\boldsymbol{B}}\tilde{\boldsymbol{K}}\right) e \tag{35}$$

The process of designing the full state feedback controller is the determination of '$\tilde{\boldsymbol{K}}$' matrix such that the closed-loop poles of the system lie at the desired location, provided that the system is state controllable.

4.3.1 Pitch/Roll LQR controller design

For designing the integral-based LQR controller, MATLAB control system toolbox is employed. Each element of the penalisation matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ is selected intuitively based on their relative importance. Such as, for pitch control, the pitch angle has greater importance than the pitch rate. Also, the steady-state error is penalised more to ensure a fast-tracking response with zero steady-state error. While, the input is penalised relatively less as the actuator can provide a higher actuation force. Therefore, based on these weighting criteria, the penalisation matrixes are selected and the LQR controller is designed and simulated iteratively in Simulink. The combination of penalisation matrixes that provided the best-simulated response is given by:

$$\boldsymbol{Q} = \begin{bmatrix} 70 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 250 \end{bmatrix}$$

$$\boldsymbol{R} = [\,0.1\,]$$

The feedback gains and integral gain are computed using MATLAB as

$$\boldsymbol{K} = [\,33.1;\ 7.9\,] \quad k_I = [\,50\,]$$

The LQR controller designed for the pitch axis is also valid for the roll axis due to geometric symmetry.

4.3.2 LQR controller implementation

The designed LQR controller is implemented on the pitch and roll axes utilising Simulink desktop real-time software and the vehicle is subjected to multistep inputs in both axes. The pitch and roll responses of the vehicle is measured by an IMU sensor, and shown in Fig. 16.

As evident from Fig. 16, integral-based LQR controller yielded a good reference tracking response with zero steady-state error. But the overall pitch and roll responses remained sluggish with rise time of 3 seconds, which does not meet the design requirement. This is due to the fact the rise time of response is mostly dependent on the integral gain. So, as to achieve a faster response, the integral gain should be increased monotonically. This may cause the system to go unstable and even the integral wind-up can drive the system towards instability. To improve it, a proportional action along with integral action is proposed and it is anticipated that it will exhibit faster response. The control scheme referred as robust integral LQR control with the feedforward term [30] is investigated next, which as per the authors' knowledge is a novel implementation on quadcopters.

4.4 Robust LQR controller with a feed-forward term

The integral-based LQR controller along with the feedforward term (LQR-FF) is proposed to improve the transient response of the quadcopter UAV and its robustness to the changes in system parameters and

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

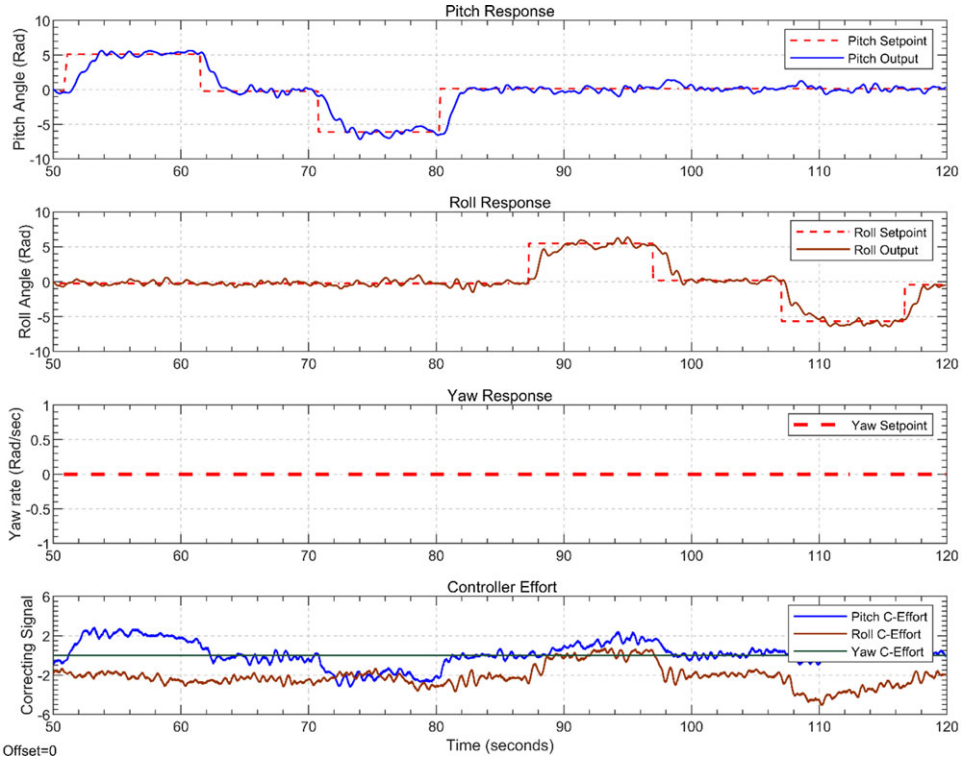

Figure 16. *Experimental response of 3-DoF closed-loop quadcopter with LQR controller implementation.*

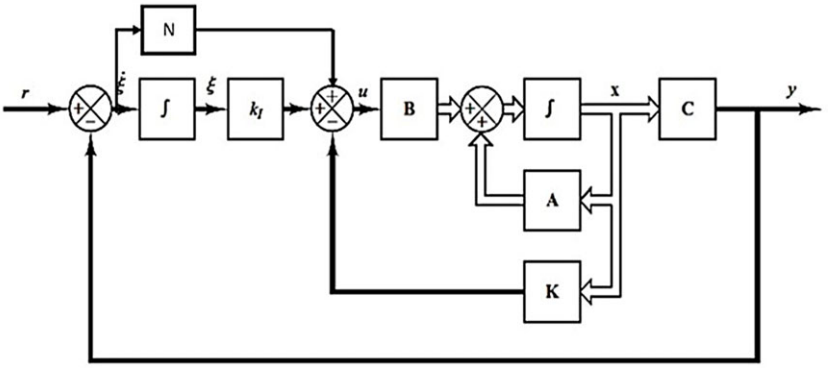

Figure 17. *Block diagram representation of robust integral LQR controller with feedforward term.*

external disturbances. The proportional/feedforward term is introduced in conjunction with an integral term such that the output is the sum of proportional and integral action, similar to a PI controller, which is shown in Fig. 17.

Although, some techniques are available to systematically determine the value of feedforward gain, but for simplicity, the feedforward gain is manually tuned until the satisfactory performance is achieved. The step responses of the quadcopter UAV in pitch and roll employing robust integral LQR controller with feedforward term, is shown in Fig. 18. The control efforts are also shown which is well within the actuator operational range of $\pm$ 50 servo PWM signal.

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

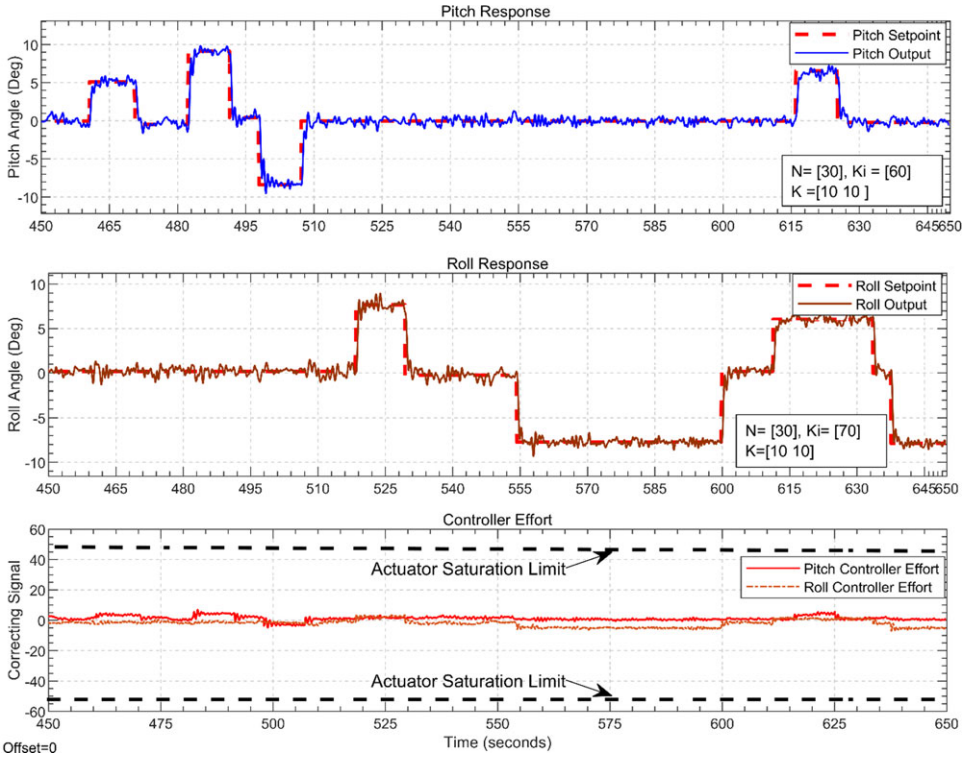

Figure 18. *Experimental step response of robust integral LQR controller with feedforward term.*

4.5 Results and discussion

As evident from the experimental step response plot of robust integral LQR controller with a feedforward term, as illustrated in Fig. 18, that the pitch and roll responses have improved remarkably with a much faster rise time of 0.2 seconds and settling time of less than 1 second with almost negligible overshoot, hence surpassing all the system design requirements. For comparative analysis, the performance characteristics of all the designed PID, LQR, and LQR-FF controller are summarised in Table 3, which clearly indicates that only LQR-FF can provide the acceptable responses and able to meet the stringent pitch and roll design requirements. However, in yaw axis, the PI controller which was able to meet the design requirements was deemed adequate.

The designed and instrumented test-rig was found to be well suited for the intended research objectives. For a more realistic test, the rig is next to be modified to allow an inverted 3-DoF quadcopter configuration similar to real unstable dynamics. For extreme angle manoeuvers the work is being compared with nonlinear sliding mode control technique and will be reported separately.

5.0 Conclusion

A 3-DoF static quadcopter UAV test rig has been developed replete with instrumentation and interfaced with a desktop PC for real-time implementation of classical and modern control techniques. The objective of developed test rig is to investigate the performance of classical and modern controllers on a quadcopter UAV in a lab environment. The focus of this work is on the model-based design of a 3-DoF autopilot for quadcopter. Thus, the mathematical models were derived utilising the Newton Euler method and then validated via the system identification technique. The benchmark PID controllers were subsequently designed using the multi-parameter root contour technique and implemented in real-time on

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

Table 3. *Performance analysis of classical and modern controllers*

Pitch response				
Controller	**Rise time**	**Settling time**	**% Overshoot**	**Acceptable**
PID	2sec	3sec	17%	× No
LQR	2.5sec	3sec	2%	× No
LQR + FF	0.3sec	1sec	2%	✓ Yes
Roll response				
Controller	**Rise time**	**Settling time**	**% Overshoot**	**Acceptable**
PID	2sec	3sec	5%	× No
LQR	3.5sec	4sec	2%	× No
LQR + FF	0.2sec	1sec	5%	✓ Yes
Yaw response				
Controller	**Rise time**	**Settling time**	**% Overshoot**	**Acceptable**
PI	4sec	8sec	5%	✓ Yes

the static test rig. The experimental results revealed poor performance of the PID controllers since they exhibited high overshoot. Therefore, a modern full state feedback control technique was investigated and LQR controllers were designed. The implementation of the integral-based LQR controller showed an improved but sluggish response. Thus, to improve upon it, a novel robust integral-based LQR controller with a proportional/feedforward term was implemented which showed an excellent response, fulfilling all the design requirements. This work thus paves the way for ongoing research on a much more complex, unstable and untethered 3-DoF micro-quadcopter UAV attitude autopilot design, ultimately leading to a stable atmospheric flight.

Funding. No funding to report.

Competing interests. The authors have no relevant financial or non-financial interests to disclose.

References

[1] Silvagni, M., Tonoli, A., Zenerino, E. and Chiaberge, M. Multipurpose UAV for search and rescue operations in mountain avalanche events, *Geomatics Nat. Hazards Risk*, 2017, **8**, (17).
[2] Rizwan, R., Naeem Shehzad, M. and Naeem Awais, M. Quadcopter-based rapid response first-aid unit with live video monitoring, *Drones*, 2019, **3**, (2), 37.
[3] Navia, J., Mondragon, I., Patino, D. and Colorado, J. Multispectral mapping in agriculture: terrain mosaic using an autonomous quadcopter UAV, The 6th International Conference on Unmanned Aircraft Systems (ICUAS), Arlington, VA USA, 2016.
[4] Rahman, M.F.F., Fan, S., Zhang, Y. and Chen, L. A comparative study on application of unmanned aerial vehicle systems in agriculture, *Agriculture (Switzerland)*, 2021, **11**, (1), pp 1–26.
[5] Xiang, H., Han, Y., Pan, N., Zhang, M. and Wang, Z. Study on multi-UAV cooperative Path planning for complex patrol tasks in large cities, *Drones*, 2023, **7**, (6), 367.
[6] Chang, J., Cieslak, J., Dávila, J., Zhou, J., Zolghadri, A. and Guo, Z. A two-step approach for an enhanced quadrotor attitude estimation via IMU data, *IEEE Trans. Control Syst. Technol.*, 2018, **26**, (3), pp 1140–1148.
[7] Bouabdallah, S., Murrieri, P. and Siegwart, R. Design and control of an indoor micro quadrotor, Proceedings of the 2004 IEEE, International Conference on Robotics and Automation, New Orleans, LA, USA, 2004.
[8] Mardan, M., Esfandiari, M. and Sepheri, N. Attitude and position controller design and implementation for a quadrotor, *Int. J. Adv. Rob.*, 2017, **14**, (3), pp 1–11.
[9] Vong, C.H., Ryan, K. and Chung, H. Trajectory tracking control of quadcopters under tunnel effects, *Mechatronics*, 2021, **78**, p. 102628.
[10] Quan, Q. Introduction to Multicopter Design and Control, Springer, 2017, Singapore.

https://doi.org/10.1017/aer.2024.11 Published online by Cambridge University Press

[11] Patel, R., Deb, D., Modi, H. and Shah, S. Adaptive backstepping control scheme with integral action for Quanser 2-dof helicopter, International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2017.
[12] Rajappa, S., Chriette, A., Chandra, R. and Khalil, W. Modelling and dynamic identification of 3 DOF Quanser Helicopter, 16th International Conference on Advanced Robotics (ICAR), 2013.
[13] Dhaybi, M. and Daher, N. Real-time estimation of the inertia tensor elements of a quadcopter hover platform, Proceedings of the 2019 IEEE/ASME, International Conference on Advanced Intelligent Mechatronics, Hong Kong, China, July 8-12, 2019.
[14] Daponte, P., Vito, L.D., Lamonaca, F., Picariello, F., Riccio, M., Rapuano, S., Pompetti, L. and Pompetti, M. DronesBench: an innovative bench to test drones, *IEEE Instrum. Meas. Mag.*, 2017, **20**, (6), pp. 8–15.
[15] Beharie, M.M., Pedro, J.O. and Dala, L. Design of a quadrotor flight test stand for system identification, *Aeronaut. J.*, 2015, **119**, (1214).
[16] Powers, C., Mellinger, D. and Kumar, V. Quadrotor kinematics and dynamics, in Handbook of Unmanned Aerial Vehicles, Springer-Verlag, 2015, New York, NY, USA, pp 307–328.
[17] Resendiz, V.M.A. and Araiza, E.R. System identification of a quadrotor in X configuration from experimental data, *Res. Comput. Sci.*, 2016, **118**, pp 77–86.
[18] Gremillion, G. and Humber, J.S. System identification of a quadrotor micro air vehicle, AIAA Atmospheric Flight Mechanics Conference, Toronto, Ontario, Canada, 2010.
[19] Sarioglu, A. and Kural, A. Modeling and ARX identification of a quadrotor Mini-UAV, 2015 9th International Conference on Electrical and Electronics Engineering (ELECO), Bursa, Turkey, IEEE, 2015, pp 1196–1200.
[20] Brzozowski, B., Rochala, Z., Wojtowicz, K. and Wieczorek, P., Measurement data fusion with cascaded Kalman and complementary filter in the flight parameter indicator for hang-glider and paraglider, *Measurement*, 2018, **123**, pp 94–103.
[21] Fields, T.D., Ellisy, L.M. and Kingz, G.W. PID quadrotor 6-DOF HIL simulation and verification using a 6-axis load cell, AIAA Atmospheric Flight Mechanics Conference, 5-9 January, Kissimmee, Florida, USA, 2015.
[22] Xuan-Mung, N. and Hong, S.-K. Improved altitude control algorithm for quadcopter unmanned aerial vehicles, *Appl. Sci.*, 2019, **9**, 2122.
[23] Yuksel, B., Secchi, C., Bulthoff, H.H. and Franchi, A. Aerial physical interaction via IDA-PBC, *Int. J. Rob. Res.*, 2019, **38**, pp 403–421.
[24] Estrada, F.R.L., Ponsart, J.C., Theilliol, D. and Zhang, Y. LPV model-based tracking control and robust sensor fault diagnosis for a quadrotor UAV, *J. Intell. Rob. Syst.*, 2016, **84**, pp 163–177.
[25] Liu, H., Shi, P. and Chadli, M. Finite-time stability and stabilization for a class of nonlinear systems with time-varying delay, *Int. J. Syst. Sci.*, 2016, **47**, pp 1433–1444.
[26] Ahmad, S.M., Chipperfield, A.J. and Tokhi, M.O. Dynamic modelling and linear quadratic Gaussian control of a twin-rotor multi-input multi-output system, *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.*, 2003, **217**, (3), pp 203–227.
[27] Ljung, L. System Identification: Theory for the User, Prentice Hall, 1999, Englewood Cliffs (N.J.), USA.
[28] Kuo, B.C. and Golnaraghi, F. Automatic Control Systems, 8th ed, 2002, Wiley, India.
[29] Ogata, K. Modern Control Engineering, 5th ed, Pearson, 2011, New Jersey, USA.
[30] Franklin, G., Powell, J. and Emami-Naeini, A. Feedback Control of Dynamic Systems, 7th ed, Pearson, 2015, UK.
[31] Ahmad, S.M. Flight dynamics, parametric modelling and real-time control of a 1-DOF Tailplane, *Math. Comput. Model Dyn. Syst.* 2013, **19**, 220–237.
[32] Khan, Y.M., Ahmad, S.M., Ali, M. and Khan, M. Flight dynamics and parametric modelling of a 2-DOF lab aircraft, *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.,* 2019, **233**, pp 2923–2931.
[33] Özaslan, T., Loianno, G., Keller, J., Taylor, C.J., Kumar, V., Wozencraft, J.M. and Hood, T. Autonomous navigation and mapping for inspection of penstocks and tunnels with MAVs, *IEEE Robot. Autom. Lett.,* 2017, **2**, pp 1740–1747.
[34] Bavle, H., Sanchez-Lopez, J.L., de la Puente, P., Rodriguez-Ramos, A., Sampedro, C. and Campoy, P. Fast and robust flight altitude estimation of multirotor UAVs in dynamic unstructured environments using 3D point cloud sensors, *Aerospace,* 2018, **5**, p 94.
[35] Cook, M.V. *Flight Dynamics Principles*, Butterworth Heinemann, 2004, Oxford, UK.

Cite this article: Ahmad S.M. and Fareed S. (2024). Attitude modelling and real-time robust control of a 3-DoF quadcopter UAV test bench. *The Aeronautical Journal*, **128**, 1767–1788. https://doi.org/10.1017/aer.2024.11