Actuator Fault Detection and Diagnosis for Multirotor Unmanned Aerial Vehicles

A Nonlinear-Observer-Based Approach

Márk Tamás Melczer

Master of Science Thesis

TU Delft Delft University of Technology

Section Control and Simulation
Department Control and Operations
Faculty of Aerospace Engineering

Actuator Fault Detection and Diagnosis for Multirotor Unmanned Aerial Vehicles

A Nonlinear-Observer-Based Approach

Master of Science Thesis

For the degree of Master of Science in Aerospace Engineering at Delft University of Technology

Márk Tamás Melczer

October 13, 2019

Faculty of Aerospace Engineering · Delft University of Technology

Delft University Of Technology
Department Control and Operations

Undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance the thesis entitled **"Actuator Fault Detection and Diagnosis for Multirotor Unmanned Aerial Vehicles"** by **Márk Tamás Melczer** in partial fulfillment of the requirements for the degree of **Master of Science**.

Date: October 13, 2019

Chair: _____
Dr. Q.P. Chu

Responsible supervisor: _____
Dr. Coen C. de Visser

Daily supervisor: _____
Ir. Pepijn van den Bos

External member: _____

Abstract

This thesis project aims to increase the safety of multirotor Unmanned Aerial Vehicle (UAV) systems by researching Fault Detection and Diagnosis (FDD) modules to fit into an Active Fault-Tolerant Control (AFTC) system designed to recover a damaged multicopter, hence avoiding crashes. In order to meet the ever-increasing safety standards set by authorities for these vehicles, the research immerses itself in the analysis of various FDD methods implemented on multicopters, focusing on Loss of Effectiveness (LoE) estimation.

The end-product of the research project is anticipated to be a fast and reliable FDD framework which ensures rapid response by the AFTC system saving the multirotor UAV from crashing after a fault is detected. The proposed framework is aimed to be flexible, meaning that its implementation on various multicopter platforms is straightforward and requires little effort, thus making it available for widespread use.

Ensuring that multirotor UAVs are able to detect their faults and update their control system accordingly will propel the development of AFTC system for drones, leading to faster certification of such systems by authorities, enabling large scale and more flexible utilization of autonomous flying.

Keywords: FDD, AFTC, LoE, adaptive observer

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Research background and motivation

As Unmanned Aerial Vehicles (UAVs) are ubiquitously used in an increasing number of fields, such as monitoring, delivery, maintenance, and recreational purposes, the safety issues they impose are getting in the focus of research. Since these unmanned systems are reliant on delicate control algorithms to perform their tasks, a damaged, faulty actuator (e.g. ailerons, propellers), responsible to steer the vehicle safely to its destination, may lead to losing the vehicle in a crash. Needless to say, UAVs falling from the sky endanger property and most importantly, lives. Thus, authorities around the world are issuing rigorous regulations concerning the flight of these vehicles, bolstering the research on safer and more robust control algorithms for these systems.

To increase the safety of UAV applications, Active Fault-Tolerant Control (AFTC) solutions are being developed aiming to keep UAVs safely airborne, even with a considerable loss in actuator effectiveness [1, 2]. Generally, an AFTC system may be divided into two main branches, Fault Detection and Diagnosis (FDD) and control reconfiguration, as shown in Figure 1.1. Fault detection algorithms monitor the condition of the UAV based on online measurements [3]. When a fault occurs, causing anomalies in the monitored signals, the algorithm detects these indicators, identifies the problem, evaluates its size and severity, and isolates the location of the fault. Thereafter, the control reconfiguration algorithm, based on the information sent by the fault detection module, updates the flight control system accordingly, in order to avoid entering an uncontrollable flight scenario. In contrast to rigid-wing UAVs, multirotor systems mostly rely solely on propellers to generate lift and translate through air. Therefore, losing effectiveness of even one actuator—for example due to a damaged propeller—greatly affects the stability and agility of the vehicle. Hence, it is crucial to detect the faults associated with the actuators and update the control reconfiguration module in a fast and reliable way, such that the flight-safety of multirotor UAVs is greatly increased.

Figure 1.1 serves to give an overview of the main concepts of this research project. Since the ultimate goal is to enhance the safety of multicopters, AFTC systems are being developed which has two main submodules, namely the control system and the FDD module. The scope of the research project lies somewhere between the two submodules, as the FDD module is designed for Loss of Effectiveness (LoE) estimation of the multirotor

actuators. Thus, it is less important to identify the fault such that it is known what exactly happened (identification) within a faulty actuator. It is more crucial to see which actuator is malfunctioning (isolation) and how its fault affects the whole system (evaluation).

Then, the control system has to be reconfigured, based on the FDD module output. The reconfiguration may be passively embedded in the system by designing an inherently robust control system which does not change its structure [4], or an active controller can be applied which adapts itself according to a specifically designed update law [5, 6]. Since the research project presented in this document focuses on the development of a FDD module aimed to be implemented within an AFTC system on multicopters, flexibility of the system is a key design requirement as it has to be fitted into existing modules. Furthermore, the system has to be fast and preferably have profound stability analysis.

In order to outline the future steps to be taken throughout the research in light of the reviewed literature, the ultimate, highest level objective of the research project is stated here. The main goal of this research is

to increase the reliability and safety of multirotor UAVs by enhancing their integrity using a FDD framework for LoE estimation

According to [3], the integrity of a system means its ability to detect its own faults and inform the operator about the system's operational availability. In terms of this research project this means that the multirotor UAV is able to detect its propulsion system losing effectiveness due to various faults and update the control system (its operator) using an estimation of the size and severity of the fault. Thus the research focuses on an important building block of a AFTC system: the FDD module. To formulate a well-defined research framework, some research questions are defined addressing the knowledge required to achieve the research objective.

The core research question focuses on the performance evaluation of the proposed FDD framework:

- To what extent can a FDD framework enhance the integrity of multirotor UAVs?
  - What methods are suitable to detect the faults specific to a multirotor UAV propulsion system?
  - Is fault isolation ensured by decentralized signal analysis on the actuator level?
  - To what extent do decentralized signals facilitate fault diagnosis and actuator effectiveness estimation?
  - What methods are suitable to interpret fault diagnosis results for control reconfiguration?

As seen from the subquestions, all aspects of FDD indicated in Figure 1.1 is covered, along with the interpretation of its results. The answers to these questions are indicators of how powerful the end-product of the research project proves to be, in light of the performance requirements for the FDD module. The speed and reliability of the FDD system is evaluated to check whether the ability of the multicopter to detect faults and reconfigure the control system is enhanced. However, in order to get to this result, suitable methods for FDD have to be reviewed first. Then, the role of actuator-level (decentralized) signal measurements has to be discussed, i.e., can these signals be utilized in order to make the FDD more accurate and ensure fault isolation. Finally, a method has to be developed to

relay the FDD output to the control system such that the multicopter is able to safely avoid crashes.

1.2 Outline of the document

The structure of this document is as follows. Firstly, Part I presents a comprehensive literature survey on FDD methods. This part starts with Chapter 2 providing a brief overview on models used for FDD, i.e., the nonlinear equations describing the dynamics of a multicopter, along with an actuator model. Thereafter, in Chapter 3, behaviour-based FDD applications focusing on low-level signals—such as current and vibrations [7], or even temperature [8]—of the actuators are discussed. These methods utilize signal monitoring and signal analysis in order to detect anomalies. Following, Chapter 4 covers some model based adaptive estimation methodologies, aiming to estimate faults based on model matching, and online process identification. Last but not least, in Chapter 5, several observer-based FDD frameworks are introduced, most of them relying on Thau observers or Kalman filters [9, 10, 11].

Subsequent to the literature survey, Part II introduces the end-product of the thesis project in the form of a scientific paper. The findings of the paper are based on the literature survey in Part I and discuss the performance of the proposed nonlinear observer for LoE estimation. The results of simulation and Hardware-in-the-loop (HIL) tests are evaluated in light of the research objective.

The document ends with a conclusion in Chapter 6 discussing the findings regarding FDD literature and the performance of the project's end-product, and projecting a possible future path for the research. Additionally, appendices are included on the concept of the Schur complement and multibody dynamics in Appendix A and Appendix B respectively.

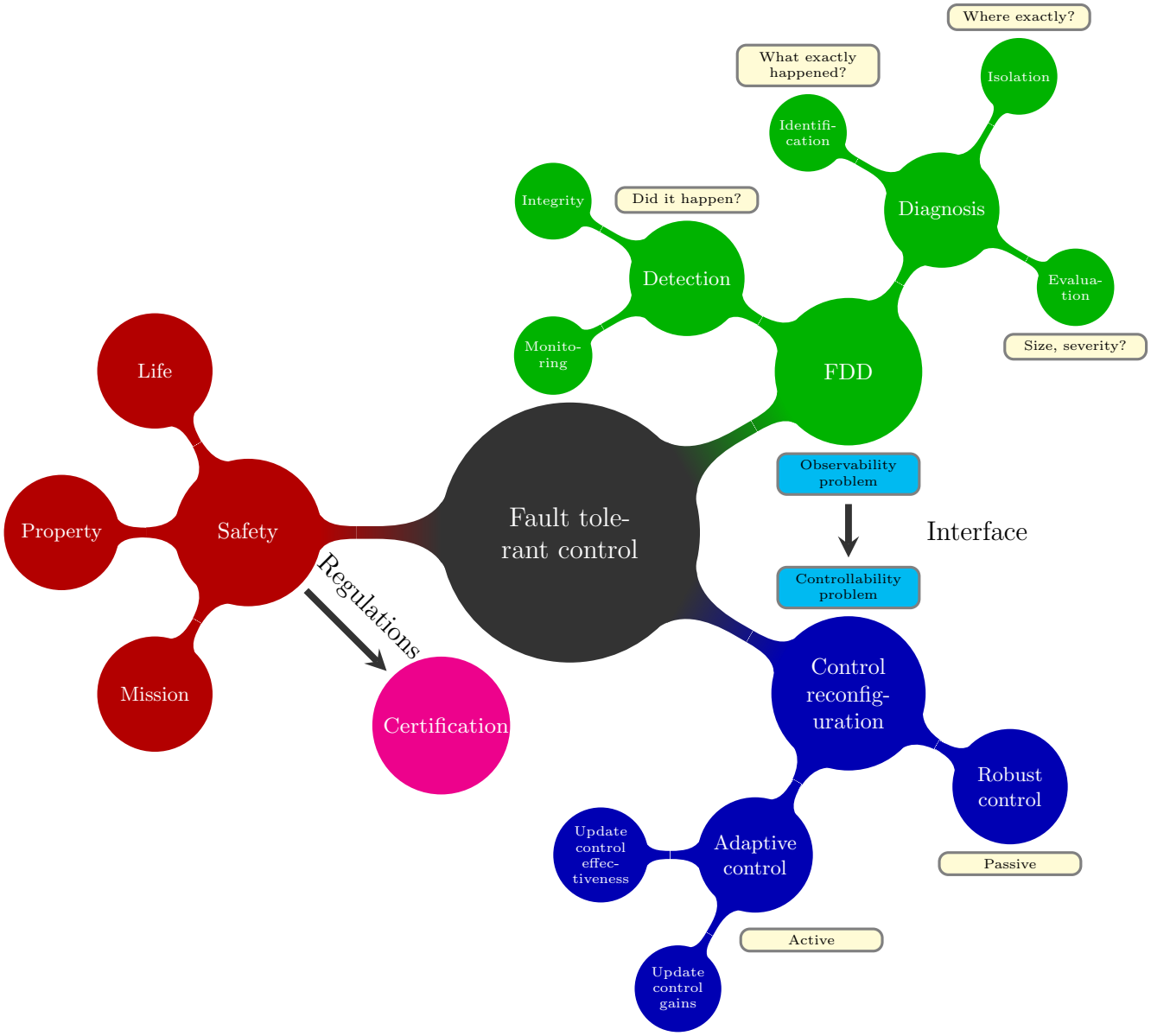

Figure 1.1: Mind map of the main concepts pertinent to the project

Part I

Literature survey

Chapter 2

Model formulation

In this chapter, a few modeling considerations are reviewed regarding a multicopter and its actuators. The actuators are assumed to be driven by electric Brushless DC (BLDC) motors. Thus, the chapter starts with the description of an underactuated multirotor Unmanned Aerial Vehicle (UAV) rigid-body kinematics and dynamics. Underactuated means that attitude and position control is closely coupled, as the drone is only able exert thrust along a single axis. For translation through air, first the attitude has to be changed to introduce thrust force components in the direction of the desired translation. In the general case, loss of actuators lead to unstable attitude and therefore unstable translation dynamics, eventually resulting crash.

Some of the most relevant fault cases of a multicopter propulsion unit is shown in Table 2.1. It is seen that most component failures result in certain amount of Loss of Effectiveness (LoE), i.e., the actuators cannot exert their nominal output to steer the multicopter. Therefore, the model fidelity should be detailed only to the level suitable for LoE estimation.

Table 2.1: Faults of the propulsion unit

Unit	Fault type	Indicators	Effect
Propeller	asymmetric blade damage	resonance due to eccentricity	LoE, vibration
	symmetric blade damage	peak in rotor speed	LoE
	lost/free-running blade	sudden peak in rotor speed	complete LoE
	stuck blade	motor current saturates	overheating, complete LoE
ESC	timing errors	oscillations in rotor speed	vibration
	short ciruit	motor stuck at constant speed	constant thrust
Motor	bearing damage	increase in current	vibration, overheating
	cracked stator	anomaly in current	LoE
	short circuit	anomaly in current	LoE
	overheating	motor temperature rises	loss of motor
Battery	battery damage	loss of electric power	complete LoE

After the introduction of the multicopter dynamics, the propeller aerodynamics and BLDC motor dynamics will be briefly introduced. The aim of this chapter is to see which

signals are relevant in the components of the rigid-body and the actuators when Fault Detection and Diagnosis (FDD) is considered.

2.1 Multicopter dynamics

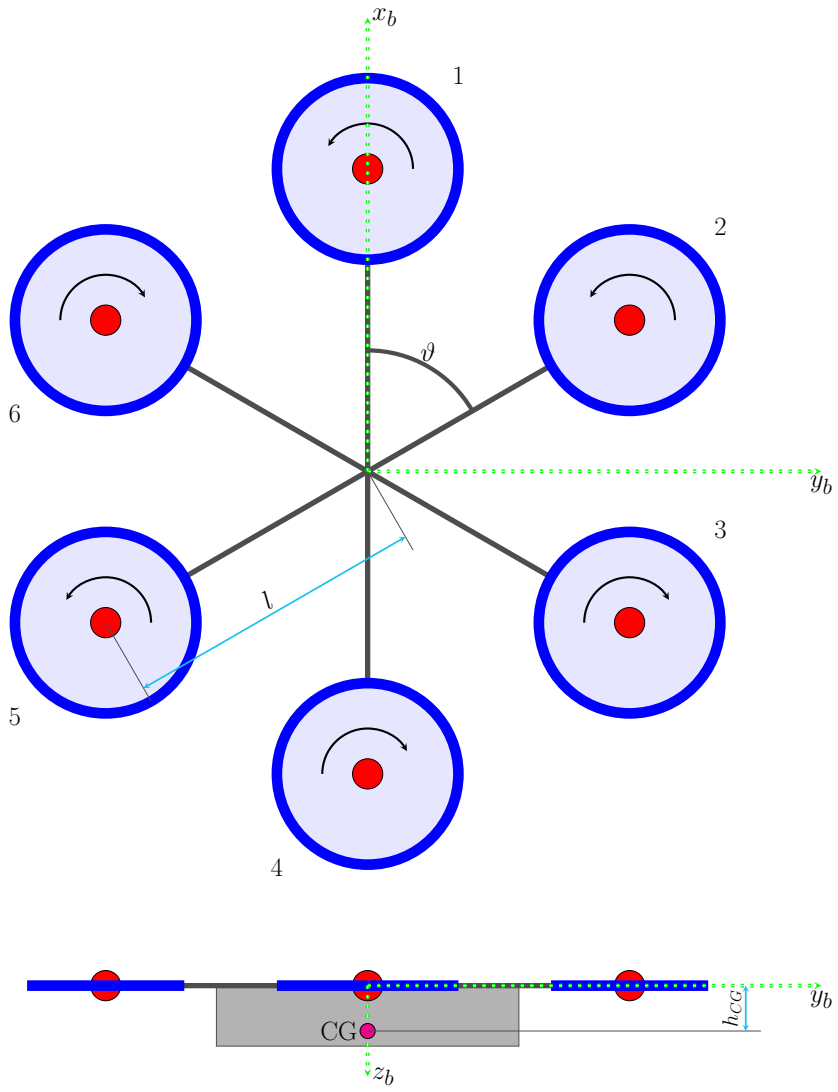

Figure 2.1: Geometry of a hexacopter

In order to design a FDD module for a generic multicopter, the mathematical description of the vehicle is needed to be constructed. As an example, Figure 2.1 shows an actual multicopter with six rotors. Two frames are used throughout the report to provide reference to the configuration (position and rotation) of the hexacopter. The first frame is the inertial frame $\mathcal{A}$ which is a global reference frame with its origin pinned in inertial space. Its axes are either in East-North-Up (ENU) or North-East-Down (NED) direction depending on the chosen convention. The second frame is the body-frame $\mathcal{B}$ whose origin is located in the Center-of-Gravity (CG) of the multicopter with axes fixed to its body,

see Figure 2.1.

Since a complete fault in an actuator may result in the hexacopter's attitude angles exceeding 90 degrees, an important criterion for the model is to avoid singularities imposed by the use of Euler-angles to describe the rotation of the hexacopter in the inertial frame. Instead, rotation matrices and quaternions are used for attitude representation [12]. Throughout this report, the rotation matrix and the respective quaternion will always denote a rotation from body to inertial frame, i.e., the columns of the rotation matrix are the unit body axes expressed with inertial coordinates.

Equations of motion of a general multicopter are derived as shown in (2.1).

$$\begin{aligned} \dot{v} &= ge_3 + \frac{1}{m}\big(Re_3\mathrm{T} + \Delta_\mathfrak{f}\big) \\ \dot{r} &= v \\ \dot{\Omega} &= J^{-1}\big(-\hat{\Omega}J\Omega + \tau + \Delta_\mathfrak{m}\big) \\ \dot{R} &= R\hat{\Omega} \end{aligned} \tag{2.1}$$

The first two equations in (2.1) describe the translational dynamics of the vehicle as $v \in \mathbb{R}^3$ denotes the inertial velocity vector and $r \in \mathbb{R}^3$ is the position coordinate vector. Vehicle mass is denoted by m and g is the value of the gravitational acceleration coefficient. $e_3 = [0, 0, 1]^T$ is the unit axis in inertial Up/Down direction, depending on the choice of coordinate frame (ENU or NED) $\mathrm{T} \in \mathbb{R}$ is the total thrust generated in the body frame, always pointing in e_3 direction. This means that the system is underactuated, as the input force may only act along a single body axis. $\Delta_\mathfrak{f} \in \mathbb{R}^3$ denotes external, unmodeled disturbance forces such as drag while $\Delta_\mathfrak{m} \in \mathbb{R}^3$ stands for disturbance moments.

As for the term Re_3, it stands for the last column of the rotation matrix $R \in SO(3)$ which is the body z_b axis expressed in inertial frame. $SO(3)$ denotes the special orthogonal group, hence if $\mathcal{I}$ is the 3-by-3 identity matrix, one has the following properties for a rotation matrix of the special orthogonal group

$$SO(3) = \{R \in \mathbb{R}^{3\times 3} \mid RR^T = \mathcal{I},\ \det(R) = +1\} \tag{2.2}$$

as described in [13]. Group theory is introduced here as it provides insight into geometric control and filter design—fundamental to the Active Fault-Tolerant Control (AFTC) control system which is augmented by a FDD module.

In (2.1) Ω is the angular velocity vector expressed in the body frame, J is the inertia tensor of the vehicle, and $\tau \in \mathbb{R}^3$ is the input torque acting on the hexacopter. The *hat* ($\hat{\bullet} : \mathbb{R}^n \to \mathfrak{so}(3)$) operator in (2.1) maps an n dimensional vector to the n-by-n dimensional space of skew-symmetric matrices ($\mathfrak{so}(3)$). (2.3) lists a few important properties of three-dimensional skew-symmetric matrices used in the report.

$$\begin{aligned} x \times y &= \hat{x}y \qquad && x, y \in \mathbb{R}^3 \text{ and } \hat{x} \in \mathfrak{so}(3) \\ \hat{x}^T &= -\hat{x} && \forall \hat{x} \in \mathfrak{so}(3) \end{aligned} \tag{2.3}$$

2.1.1 Control effectiveness matrix

In order to implement the controller, the control effectiveness matrix needs to be constructed as a mapping from the rotor speeds to the wrench exerted on the rigid body. A wrench in three-dimensional space is a vector containing the inertial forces and torques

acting on the rigid body. However, since the thrust force is generated by non-tilted rotors in the body frame—ignoring the aerodynamic effects—thrust will always point in body UP direction. Hence, the hexacopter is an underactuated system. The generated thrust will then be computed as (2.4)

$$\mathrm{T} = -(\kappa_\mathrm{T} \cdot \omega_{p_1}^2 + \ldots + \kappa_\mathrm{T} \cdot \omega_{p_6}^2) \tag{2.4}$$

where κ_T is the rotor thrust coefficient. Note the negative sign in the front due to the thrust pointing in the negative body z_b (UP) direction.

As for the generated torques, the following expression is derived based on the geometry seen in Figure 2.1.

$$\tau = \begin{bmatrix} -\kappa_\mathrm{T} l \sin(\vartheta_1) & \cdots & \cdots & -\kappa_\mathrm{T} l \sin(\vartheta_6) \\ \kappa_\mathrm{T} l \cos(\vartheta_1) & \cdots & \cdots & \kappa_\mathrm{T} l \cos(\vartheta_6) \\ -\kappa_\tau & \kappa_\tau & -\kappa_\tau & \cdots \end{bmatrix} \begin{bmatrix} \omega_{p_1}^2 \\ \vdots \\ \omega_{p_6}^2 \end{bmatrix} \tag{2.5}$$

with l denoting the rotor displacement from the CG in the body $x_b - y_b$ plane (in this case same for all rotors), κ_τ the rotor torque coefficient and ϑ the angle of the rotor's arm from body x_b axis.

Combining (2.4) and (2.5) the underactuated system's control effectiveness (for a hexacopter) is found to be

$$G = \begin{bmatrix} -\kappa_\mathrm{T} l \sin(\vartheta_1) & \cdots & \cdots & -\kappa_\mathrm{T} l \sin(\vartheta_6) \\ \kappa_\mathrm{T} l \cos(\vartheta_1) & \cdots & \cdots & \kappa_\mathrm{T} l \cos(\vartheta_6) \\ -\kappa_\tau & \kappa_\tau & -\kappa_\tau & \cdots \\ -\kappa_\mathrm{T} & \cdots & \cdots & -\kappa_\mathrm{T} \end{bmatrix} \tag{2.6}$$

which depends only on geometrical parameters that are assumed to be constant during flight, and the thrust and torque coefficients which are depending on aerodynamic effects.

2.2 Rotor aerodynamics

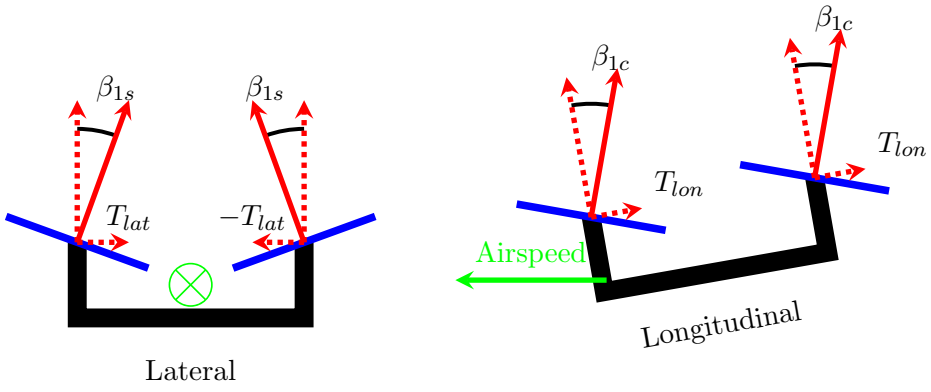

Figure 2.2: Blade flapping in translational flight

Aerodynamics of the propeller blades are hard to model accurately, but the movement of air around the propellers generate additional thrust and drag which can be neglected

in hover conditions, but not in translational flight. Various drag terms are discussed in detail in [14] stating that blade flapping induces an important damping in translational flight. The phenomenon of blade flapping is derived in [15] for helicopters and its effect is visualized in Figure 2.2.

A detailed aerodynamic model is derived in [16], which found that the thrust coefficient κ_{T} is varying with the rotor inflow ratio λ_i and the advance ratio μ. The rotor inflow ratio relates the inflow velocity to the rotor blade tip velocity as

$$\lambda_i = \frac{\nu_i - v_z^{\mathcal{B}}}{\omega_p r}$$

where $v_z^{\mathcal{B}}$ is the airspeed in the body z_b direction, ω_p is the rotational speed of the propeller, and r is the rotor blade length. The inflow velocity ν_i is computed via

$$\nu_i = \sqrt{-\frac{\left(v_{xy}^{\mathcal{B}}\right)^2}{2} + \sqrt{\left(\frac{\left(v_{xy}^{\mathcal{B}}\right)^2}{2}\right)^2 + \left(\frac{mg}{8\rho A}\right)^2}}$$

with $v_{xy}^{\mathcal{B}}$ being the planar airspeed $(x_b - y_b)$ of the multicopter in the body frame. ρ is the density of the air and A is the disk area of the rotor.

The advance ratio is the ratio between the planar airspeed and the blade tip velocity:

$$\mu = \frac{v_{xy}^{\mathcal{B}}}{\omega_p r}$$

It is shown in the paper that the actual thrust coefficient, affected by aerodynamic effects, can be modeled as seen in (2.7).

$$\kappa_{\mathrm{T}}(\lambda_i, \mu) = c_0 + c_1\lambda_i + c_2\mu^2 \tag{2.7}$$

The coefficients ($c_1,\ c_2,\ c_3$) of the model shown in (2.7) are very specific to a certain multicopter, furthermore, the reliability of the identification of these parameters can be questionable, as unmodeled gusts can cause high inaccuracy.

A more recent paper describes a meticulous gray-box model for a quadcopter, identified using data gathered from high speed wind tunnel tests [17]. The paper sets up multiple candidate model structures to find the best model describing the quadcopter in high speed flight, and great emphasis was given to model the aerodynamic effects accurately. According to the paper, when the flight speed of the quadcopter is high, the wake generated by the front rotors significantly degrade the aerodynamic effectiveness of the rear ones, meaning that in order to maintain the orientation of the quadcopter, the rear rotors need to rotate faster. Furthermore, the rotors generate additional yawing moment when flying at a nonzero sideslip angle.

To account for these effects, a detailed thrust model is used to obtain the individual propulsion force generated by each propeller. Similarly to the previous paper, the propeller-induced velocity is a crucial part of the model which depends on the advance ratio. System identification, i.e., the estimation model parameters was performed via a simple but powerful Ordinary Least Squares (OLS) procedure, thus the main challenge was to choose the best model structure to describe the nonlinear dynamics and aerodynamic effects of a quadcopter.

The novelty of the paper lies in these proposed candidate models capturing forces and moments containing various drag terms and aerodynamic interactions between the propellers in the form of polynomial functions. Thus, OLS can be applied to estimate the coefficients of these polynomials. The results of the paper are very promising, and since most observers designed for FDD purposes are model based—as will be shown later—the state-of-the-art methodology used in the paper can be utilized to obtain crucial model parameters in wind tunnel tests; this way, the FDD module will be more accurate and reliable.

2.3 Motor dynamics

After the high-level drone model has been introduced, the dynamical description of the propulsion unit is discussed here. First, the motor dynamics is modeled by merging the electrical and mechanical properties of the actuator. Thereafter, some insights are given on the aerodynamic effects of the propeller blades.

The input voltage (u_m) to the motor depends on the battery (source) voltage u_b and the Pulse Width Modulation (PWM) from the Electronic Speed Control (ESC) unit. According to [18], this relationship is described as

$$PWM = \left(\frac{u_m}{u_b}\right)^2 \tag{2.8}$$

where u_b and the PWM can be measured. However, the expression in (2.8) applies only when the source current is alternating and the effective current is used to determine the DC equivalent current at the motor. Since the source current from the battery is DC, the average voltage at the motor is computed via a linear expression as shown in (2.9).

$$u_m = u_b \cdot PWM \tag{2.9}$$

Knowing the input voltage to the BLDC motor, the electrical description of the system is written as

$$u_m = Ri + L\frac{di}{dt} + \kappa_e \omega_m \tag{2.10}$$

where R denotes resistance, i is the motor current (also measured), L is the motor inductance, κ_e is the counter-electromotive force of the motor, and ω_m denotes the rotational speed of the motor. Assuming that the motor inductance is negligible, and the torque generated by the electromagnetic forces is written as the function of the armature current

$$\tau_e = \kappa_q i \tag{2.11}$$

equations (2.10) and (2.11) can be merged by expressing the armature current as

$$\tau_e = \frac{\kappa_q}{R}\left(u_m - \kappa_e \omega_m\right) \tag{2.12}$$

The torque in (2.12) equals the load torque on the motor which emanates from the mechanical properties of the motor and the propeller. The motor friction and the drag generated by the propeller blades provide damping to the system [14]. Thus, the torque equation describing the drive train is written as

$$\tau_e = J_m \dot{\omega}_m + d_m \omega_m + J_p \dot{\omega}_p + d_p \omega_p^2 \tag{2.13}$$

where J_m and J_p are the moments of inertia of the motor and propeller respectively, with respect to their revolution axis. d_m emanates from the friction (κ_f) and back-emf (κ_e) of the BLDC motor while d_p depends on aerodynamic effects, such as aerodynamic drag ($\kappa_\tau \omega_p^2$), acting on the propeller.

Assuming that the propeller is rigidly mounted on the motor, i.e. $\omega_m = \omega_p$, (2.13) can be written as

$$\frac{R}{\kappa_q}\left(J_m + J_p\right)\dot{\omega}_m + \left(\frac{R}{\kappa_q}\kappa_f + \kappa_e\right)\omega_m + \frac{R}{\kappa_q} d_p \omega_m^2 = u_m \tag{2.14}$$

Rewriting (2.14) in a nonlinear state-space form results in

$$\dot{\omega}_m = a\omega_m + bu_m + h\omega_m^2 = f(\omega_m, u_m) \tag{2.15}$$

where the coefficients $a,\ b,\ h$ can be identified from test measurements. Furthermore, according to [19], the system can be linearized around hover conditions, i.e., at the motor speed (ω_{m0}) and the respective motor voltage (u_{m0}) required for hovering. Thus, one obtains the following linearized state-space model

$$\dot{\omega}_m = \left.\frac{\partial f(\omega_m, u_m)}{\partial \omega_m}\right|_{\substack{\omega_m=\omega_{m0}\\ u_m=u_{m0}}} (\omega_m - \omega_{m0}) + \left.\frac{\partial f(\omega_m, u_m)}{\partial u_m}\right|_{\substack{\omega_m=\omega_{m0}\\ u_m=u_{m0}}} (u_m - u_{m0}) \tag{2.16}$$

where the partial derivatives, evaluated at hover conditions, are constants.

The overall resistance R and back-emf coefficient κ_e can be obtained from (2.11) and (2.12) via linear regression methods. By measuring u_m, i, and ω_m without attaching the propeller to the motor, (2.17) can be used to find the resistance and back-emf coefficient [19].

$$\frac{u_m}{i} = R + \kappa_e \frac{\omega_m}{i} \tag{2.17}$$

2.4 Thrust model

Assuming that the propeller-generated thrust (T) and load torque on the motor (τ_l) can be modeled (in hover) as

$$\mathrm{T} = \kappa_\mathrm{T} \omega_p^2$$
$$\tau_l = \kappa_q i$$

it is found that the relationship between the generated thrust and torque is

$$\frac{\tau_l}{\mathrm{T}} = \frac{\kappa_q i}{\kappa_\mathrm{T} \omega_p^2} \tag{2.18}$$

From the power balance perspective, the electrical power has to be equal to the mechanical power and the dissipated power due to resistance, such that

$$u_b i - Ri^2 = \tau_l \omega_m$$

assuming that $\omega_m = \omega_p$ and noting that $u_m \leq u$. Using the relationship in (2.18), thrust can be related to power consumption and the motor speed as

$$\mathrm{T} = \frac{\kappa_\mathrm{T}}{\kappa_q}(u - Ri)\omega_m \tag{2.19}$$

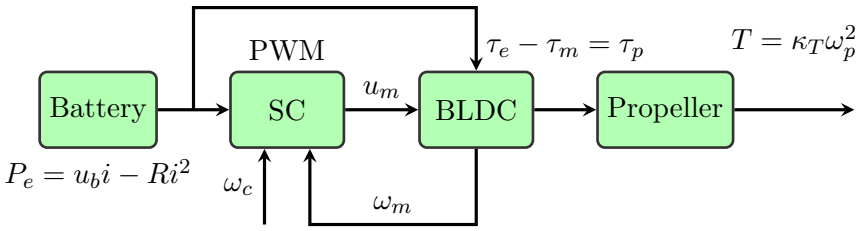

Figure 2.3: Components of the drive train

The interaction between drive train components is shown in Figure 2.3. The battery provides the source power to the system which is then channeled to the BLDC motor by the ESC via a PWM signal. The motor speed controller (SC) usually contains a Proportional-Integral-Differential (PID) controller which adjusts the PWM based on the error between the commanded motor speed ω_c and the measured motor speed ω_m. The motor then converts electrical power to mechanical power, such that the torque on the propeller τ_p will be given by the difference between the electrical torque (2.11) and the motor torque τ_m due to friction and other damping effects. Then, the propeller exerts the thrust force on the rigid body.

2.5 Chapter summary

In this chapter, the model of a generic multicopter was derived with the introduction of the control effectiveness matrix and its components. It was shown that propeller aerodynamics impose high level of uncertainty to the model. Moreover, due to propeller geometry being fundamental in thrust generation, modeling the aerodynamic effects is rather inaccurate and very specific to a certain system.

Apart from the multicopter, the dynamics of a generic BLDC motor were introduced as well. Since the motor is a less complicated system, its parameters can be identified more accurately especially if there is no propeller attached to it. However, these parameters are also specific to a certain type of motor.

All in all, this chapter provided a brief overview of models describing the rigid-body and actuator dynamics. It will be crucial in the subsequent chapters that the system model is observable, otherwise designing FDD observers will not be feasible.

Chapter 3

Condition monitoring

One of the fundamental ideas behind Fault Detection and Diagnosis (FDD) is to constantly monitor signals of a process (plant), and decide whether the plant works properly based on the behaviour of the monitored signals. This way, a fault can be detected if, for example, a signal surpasses a given threshold. Taking one step further, the statistical properties (mean, variance, etc.) of the monitored signals can be used as indicators of a fault as well, especially if the signal is periodic. Furthermore, if an estimate of a signal is available online, for example, via an observer, then the difference (residual) between the estimate and the actual measurement can be monitored [3] and compared to fault detection thresholds. These methods are based on feature generation, i.e., certain periodic, statistical, spectral features of the monitored signals are extracted for further evaluation. Changes in these features are good indicators of faults and anomalies within the system.

Since the propulsion unit of a multicopter generally consists of a propeller and a Brushless DC (BLDC) motor, signal analysis on the actuator level (low-level signals) is a popular choice for FDD purposes. Based on the behaviour of signal features, it is straightforward to detect drive-train faults, furthermore, the Power Spectral Density (PSD) or current signature can be used to classify faults, thus estimating their severity. As for fault isolation, since this research focuses on Loss of Effectiveness (LoE) estimation, it is not necessarily important to have a good idea of what exactly went wrong inside the actuator. Thus fault isolation is only required on a higher–less detailed–level, in other words, it is enough to see which actuator is faulty on a multicopter, but knowing the actual reason behind the fault is secondary.

This chapter discusses some low-level approaches for FDD on multicopters, such as vibration and current signature analysis. Vibration (acceleration) signals can be obtained via the central Inertial Measurement Unit (IMU) installed on the multicopter, or accelerometers can be installed individually below each propulsion unit to better see what is happening locally at an actuator. Temperature monitoring is also an interesting method for FDD, as overheating, due to bearing damage or short circuit, is a critical issue of BLDC motors. As it is shown, Neural Networks (NNs) can be trained to detect critical overheating, based on the statistical parameters of the measured temperature signal. Finally, adaptive thresholds are briefly introduced which help to add active robustness to fault detection. Adaptive thresholds of monitored signals are discussed in greater detail in Chapter 5.

3.1 Signal analysis

As suggested by [3], one of the most effective way to monitor the processes of rotating machinery, such as the propulsion system of a multirotor Unmanned Aerial Vehicle (UAV), is signal analysis—excelling at vibration-based fault detection. Harmonic oscillations appearing in the measured signals are good indicators of a damaged blade or motor bearings which may occur due to motor imbalance, eccentricity, and broken parts. Additionally broken propellers also cause vibrations if the blades are damaged in an asymmetric way.

3.1.1 Vibration analysis

When a rotor blade gets damaged, it is safe to assume that the cause was some kind of collision which entails a sudden peak in accelerations measured by the IMU. Thus, due to the resulting eccentricity of the damaged blade, harmonic vibrations can be detected by the on-board accelerometer. Processing these measurements, combined with measuring voltage, current, and rotational speed of the motor, greatly contributes to the development of FDD framework.

A review paper [20] on gearbox anomaly detection provides good insight what methods are available to detect faults in a rotating machinery. After obtaining a discrete measurement signal in the time domain, various statistical features can be used as indicators of a fault. For example, the peak value of a discrete signal $y[n]$ is written as

$$p = \frac{\max\left(y[n]\right) - \min\left(y[n]\right)}{2}$$

which shows immediately if vibration levels are above a certain threshold. The statistics of the measurement signal provide information on less apparent faults, such as bearing faults. The mean value of $y[n]$ from samples k to N is given as

$$\mu_y = \frac{\sum_{j=k}^{N} y[j]}{N-k+1}$$

whereas the variance of the same series of samples is defined as

$$\sigma_y = \frac{\sum_{j=k}^{N} (\mu_y - y[j])^2}{N-k+1}$$

Due to the rotational motion of the propulsion units, spectral (frequency-domain) analysis is also widely used for FDD. The most popular approaches are Fast Fourier Transform (FFT) and Wavelet Packet Decomposition (WPD) of the measured signal. Since these transformations are computationally expensive, a moving-window is applied on the measurement signal to transform only the samples within the window length [21]. Thus, after obtaining N sample long discrete series, the FFT is computed from the Discrete Fourier Transform (DFT) which is defined via the fundamental frequency

$$\omega_0 = \frac{2\pi}{N}$$

as a sequence of complex numbers

$$Y[k] = \sum_{n=0}^{N=1} y[n] e^{-ikn\omega_0} \qquad \text{for } k = 0, 1, \ldots, N-1$$

which contains the same information for positive and negative frequencies, due to the periodicity of the exponential term representing a unit phasor [3]. Therefore, according to the Cooley-Tukey FFT algorithm, the computation time can be reduced by computing only one side of the DFT on two $N/2$ long series.

Since it was assumed that the fundamental frequency ω_0 is constant while performing the FFT, other methods have to be used when the propulsion unit is operating under nonstationary conditions. In order to account for the appearing transients in the signals of a nonstationary BLDC, Short-Time Fourier Transform (STFT) and WPD can be applied on the signal. STFT is, in essence, a windowed DFT written as

$$Y(m,\omega) = \sum_{n=-\infty}^{\infty} y[n]w[n-m]e^{-i\omega n}$$

which is continuous due to ω, however, it can be discretized by evaluating the function at discrete multiples of $\frac{2\pi}{N}$. w is a window function whose length determines the frequency resolution of the frequency spectrum. For example, if a long window is applied, the obtained spectrum will have small frequency resolution and vice versa [3].

WPD is a transformation that decomposes the measured signal into two sets, namely a high and a low frequency set. Then, each set can be further split into another two set, based on the same principle. The more decomposition levels are in the WPD tree, the better the frequency resolution will be at the cost of more computations. According to [22], continuous signal $y(t)$ can be written as a sum of WPD functions ψ and their respective coefficients c as shown in (3.1).

$$y(t) = \sum_{i=1}^{2^j} y_j^i(t) = \sum_{i=1}^{2^j} \sum_{k=-\infty}^{\infty} c_{j,k}^i(t)\psi_{j,k}^i(t) \tag{3.1}$$

The WPD coefficients in (3.1) are obtained as

$$c_{j,k}^i = \int_{-\infty}^{\infty} y(t)\psi_{j,k}^i(t)dt$$

provided that the WPD functions are orthogonal, that is, $\psi_{j,k}^p \psi_{j,k}^q = 0 \;\forall\; p \neq q$.

Another approach utilizes FFT and WPD to obtain IMU acceleration signal features and train a neural network offline to classify faults related to rotor blade damage on a multicopter [21]. Although efficient for fault detection and classification, isolation of the faults is not ensured, due to using only the central IMU signals, which does not necessarily yield information on the location of the fault, i.e., which rotor's blade is damaged.

Similar applications had been developed in [22] and [7]. The authors of both papers used vibration data to train a NN for fault detection, based on the signal features such as wavelet coefficients. Although NNs are not in the scope of this research, it is valuable to study the methods used for feature generation to train the NN. Assumptions are made in both papers that different blade damage types result in different, well-distinguishable vibration features. The former paper applied WPD to interpret acceleration data of the IMU into feature vectors used for training the NN. The latter paper placed multiple accelerometers below each motor, arguing that these decentralized accelerometers can help isolate actuator faults. However, having multiple accelerometers installed on a single body means that each accelerometer will detect vibration signals of every actuator (to a

certain degree), which may trigger false alarms. Thus, feature generation and detection thresholds have to be chosen carefully.

Since many similarities between the drive train of a multicopter propulsion unit and a wind turbine can be found, investigating FDD methods applied in wind turbine practice broadens the apprehension of FDD working principles. However, it should be noted that detection time of a fault in a wind turbine is much less crucial than in case of a multicopter, furthermore, having a computer on a wind turbine with large computational power enables using algorithms more complex than those suitable for a multicopter. [23] describes a PSD-analysis-based wind turbine condition monitoring system using multiple accelerometers to measure vibrations of the turbine due to eccentricity of the blades. Applying Fourier-transformations real time, however, might be too slow and computationally expensive for the performance requirements of a multicopter FDD module.

3.1.2 Current signature analysis

If the propulsion unit of a multicopter is electrical, then motor current can provide valuable information on the actuator health. Using motor current to detect anomalies within the drive-train appears as Current Signature Analysis (CSA) in literature. Previously introduced transformations, such as WPD, can be used to detect mechanical faults related to an induction motor, as shown in [24]. Indicators of such faults are the accented harmonics appearing in the current's PSD functions and WPD coefficients.

CSA operates with feature extraction, meaning that the faults are diagnosed based on their respective features occurring in the PSD functions. It is also stated, that the STFT, which is applicable in real-time, cannot be effectively used for feature extraction, as the length of the sliding window imposes a trade-off between time and frequency resolution. Furthermore, [24] considers only induction motors which have different working principle than BLDC motors used on multicopters.

To overcome this issue, Quadratic Time-Frequency Representation (TFR) is applied on BLDC motors for anomaly detection in [25]. Quadratic TFR does not depend on the window size chosen, contrary to the STFT. Furthermore, it is suitable to apply in nonstationary operations of BLDC motors. More advanced methods utilized for fault detection in a BLDC motor are found in [26].

3.1.3 Temperature monitoring

Since overheating of the BLDC motor poses a frequent and serious issue in propulsion unit failures, a FDD system, based on motor temperature monitoring, was developed in [8]. A temperature sensor was placed below each motor to measure their temperature, then, based on previous temperature measurements, a threshold temperature was constantly updated during flight. Furthermore, variance of the temperature signal was also computed at every step.

An abnormal rise in the motor temperature is said to be detected when it surpasses a threshold temperature and its variance is increasing. In that case, the overheating motor is commanded to decelerate until the it is cooled off. If the temperature is not falling, the drone is instructed to land. Even though the threshold temperature is updated by an algorithm based on reinforcement learning, the method can surely be adapted in a suitable form to introduce additional safety to the drone.

3.1.4 Battery state monitoring

A few papers, [27] and [28], state that the battery State of Charge (SoC) should be accounted for when a propulsion system power output is estimated. Since the voltage of the battery cells deplete approximately linearly with respect to the SoC (between 20% and 80% of charge) this effect should not be complicated to correct. The Electronic Speed Control (ESC) can be commanded to allocate a surplus of power to the motor as a function of the SoC. Although, requiring the battery to maintain the output power as it is depleting, would result in a faster discharge rate and ultimately a shorter mission time.

For example, [29] found that the propulsion unit can be described accurately by a transfer function (between motor speed and motor voltage) of the form

$$H = \frac{k(s + z_b)}{(s + p_m)(s + p_b)}$$

where k is a constant gain, z_b and p_b are the zero and pole of the battery model respectively, and p_m is the pole of the motor driving the propeller. It is found, that the pole associated with the battery is much slower than the motor dynamics represented by a faster pole p_m. This means that over a relatively long period of time, due to the battery discharge, the rotor speed would slowly decrease.

However, if the rotor speed is in a feedback loop, then the integrator term in the ESC unit will account for maintaining the rotor speed of the propeller which will thus generate the same thrust (T), assuming that

$$\mathrm{T} = \kappa_{\mathrm{T}} \omega_p^2$$

where κ_{T} is the thrust coefficient and ω_p is the rotational speed of the propeller.

Nevertheless, if the battery is damaged then all electric power is lost, meaning that if all individual propulsion units are connected to a single battery, the system cannot be recovered. However, by increasing the redundancy of batteries, i.e., connecting the propulsion units to smaller but separate batteries, can significantly increase the safety of the multirotor UAV.

3.2 Adaptive robust thresholds

According to [30], the FDD module has to be more reliable than the system, therefore, active and/or passive robustness has to be incorporated into the FDD framework. Active robustness indicates that residuals serving FDD purposes are generated in a robust way, meaning that the residual generator (e.g., an observer) is synthesized via robust design methodologies. Passive robustness, on the other hand means that only the decision making is designed to be robust, i.e., the threshold to which the residuals are compared is continuously changed in an adaptive manner—thus avoiding false alarms and missed fault detection.

The use of robust, adaptive thresholds is known to be used in NN applications as well. For example, Lu [8] trained a NN to adaptively modify the fault detection threshold of the temperature measured at the motors of a quadcopter. The update is based on the mean and variance of measured temperature series, meaning that as the temperature of the motors increase naturally during the mission, the threshold adaptively increases with

it. This way, false alarms are avoided, and only abrupt changes in the temperature are detected as faults.

Although adaptive thresholds are effective to filter false alarms, they can also mask minor faults, especially when the UAV is exposed to strong disturbance. Therefore, their application should be weighed carefully. Nevertheless, when passive robustness is not an option to include in the design, robust thresholds are powerful tools to apply on a small number of monitored signals.

3.3 Chapter summary

Signal analysis in itself is a complete field within science and this chapter just scratched its surface in order to provide an overview of its application in FDD for multicopters. It was shown that condition monitoring and signal analysis is widely used for the detection of anomalies within the propulsion unit, due to the fact that the actuator is a rotating machine with periodic signals thereof.

Signal analysis and feature generation methods are popular among NN-related research, as most papers introduced were training NNs to obtain an effective online tool which is able to provide diagnosis on the multicopter's condition, based on vibration signal and current signature features. These features can be generated via STFT and WPD. Both methods can be used in nonstationary conditions, contrary to other Fourier-transformation methods.

It was shown that adaptive thresholds can be used to provide active robustness to the signal monitoring module, which can be crucial when monitoring temperature. Considering a BLDC motor on a multicopter, as its temperature is increasing to the operational level, overheat thresholds have to be adjusted adaptively in order to avoid detecting the inherent buildup of motor temperature.

Since this research project aims to isolate faults on the actuator level, and not within the individual actuators, the following chapters shift the focus onto more high-level FDD approaches, with the emphasis on LoE estimation.

Chapter 4

Adaptive control methods

The main goal of Fault Detection and Diagnosis (FDD)—with respect to this research project—is to estimate the Loss of Effectiveness (LoE) parameters of the actuators, in order to update and reconfigure the high-level Active Fault-Tolerant Control (AFTC) system. Therefore, parameter-identification-based FDD methods are discussed in this chapter, aiming mostly to continuously adapt the control system with respect to LoE parameters. These approaches operate with detailed models of the system, which are either identified offline, or continuously online.

There is abundant literature concerning AFTC of fixed-wing aircraft which researchers are gradually shaping to be applicable to multicopters. First, Direct Adaptive Control (DAC) and Indirect Adaptive Control (IAC) will be introduced serving as a basis for the adaptive control methodologies. Online parameter estimation and model-matching algorithms will provide the core of the chapter which are more frequently being used for multicopter FDD lately.

In general, adaptive control methods aim to update the control system online (real time), by estimating unknown model/control parameters. To achieve this, there are several approaches available in the literature, such as DAC and IAC methods [31]. The main distinction between the two approaches lies in the relationship between the control adaptation algorithm and the parameter estimator module. In a DAC system the parameter estimation is incorporated into the adaptation algorithm, i.e., these processes are closely coupled determining overall stability. These systems are usually designed via Lyapunov-based control laws.

IAC approaches separate control system adaptation and parameter estimation, allowing for the individual design of these modules. This means that when designing the control system, it can be assumed that parameter estimation is accurately provided at all times, and vice versa; when designing the estimator, it is assumed that the system is stabilized by the controller.

Although the focus of this research project is FDD for multicopters, in order to see how a FDD system can be incorporated into a complete flight control system, it is useful to have a short overview of adaptive control methods fundamental for AFTC. Therefore, this chapter will begin with a brief comparison of direct and indirect adaptive control methods followed by a few examples of IAC methods for FDD on multicopters, such as the two-step method and Least Mean Squares (LMS) estimation of LoE parameters.

Finally, Multiple Model Adaptive Estimation (MMAE) approaches are discussed briefly, with respect to FDD for aerial vehicles.

4.1 Direct Adaptive Control

4.1.1 Model Reference Adaptive Control

As mentioned before, DAC methods incorporate parameter estimation in the control algorithm. For example, Model Reference Adaptive Control (MRAC) (one of the most popular approaches) uses a reference model to generate an error between observed states. Then, the gains and parameters of the control law are constantly updated via update laws based on the this model output error with the intent of steering the model error and the tracking error asymptotically to zero, simultaneously.

An application of MRAC is introduced in [32] for a quadcopter. The reference model is assumed to be linearized, having the form of

$$\dot{x}_r = A_r x_r + Br$$

where the subscript $(\bullet)_r$ pertains to the reference model, and r denotes the reference signal. Then it is proposed that the control law

$$u = \hat{k}_1^T x + \hat{k}_2^T r + \hat{\theta}^T \phi(x)$$

adapted by gain (k_i) and parameter (θ) update laws

$$\begin{aligned} \dot{\hat{k}}_1 &= -\Gamma_1 x e^T P B \\ \dot{\hat{k}}_2 &= -\Gamma_2 r e^T P B \\ \dot{\hat{\theta}} &= -\Gamma_3 \phi(x) e^T P B \end{aligned} \tag{4.1}$$

is proposed to yield a stable system even in the presence of unknown parameters. In (4.1) Γ_i denotes a respective adaptation weight matrix (symmetric, positive definite), $e = x - x_r$ is the error between the actual and the reference model states, and P is a symmetric, positive definite Lyapunov gain matrix synthesized via

$$A_r^T P + P A_r = -Q$$

with Q being symmetric, positive definite as well. $f(x)$ represents nonlinearities in the system.

The proof of this adaptive control law rendering the uncertain, nonlinear system asymptotically stable is omitted, this example just serves to show how entangled the controller adaptation and parameter estimation designs are. It is seen that if a fault term can be expressed as an unknown parameter of the model, its value can be adaptively estimated online, indicating that a MRAC architecture is suitable for AFTC design. However, when a complete LoE occurs in one of the actuators, controllability of the system is reduced which requires the restructuring of the control system. Therefore, MRAC is only applicable for less severe uncertainties/faults [31], and other methods, capable of dealing with such large scale faults, should be considered for FDD and AFTC.

4.1.2 Adaptive backstepping

Adaptive Backstepping control is another DAC method utilized for FDD purposes in the literature. A method for LoE estimation is introduced in [6] for a quadcopter. It is assumed that the LoE (l_i) of the i^{th} is modeled as additive fault such that

$$\omega = \eta\omega_c$$

where ω_c is the commanded rotor speed and the LoE factor is denoted by η. It is assumed throughout the paper that η is always greater than zero, since $\eta = 0$ would mean a full fault which the designed control system is not able to handle. Then, introducing

$$\vartheta = 1 - \eta^2$$

and assuming the fault occurring as a step function $\delta(t - t_f)$ at an unknown time t_f, the following thrust model is established

$$\begin{aligned} \mathrm{T} &= (1 - \delta(t - t_f)\vartheta)\,\kappa_\mathrm{T}\omega^2 \\ &= \kappa_\mathrm{T}\omega^2 - \delta(t - t_f)(\kappa_\mathrm{T}\omega^2 - \eta^2\kappa_\mathrm{T}\omega^2) \end{aligned}$$

which, if the fault has occurred, i.e. $t > t_f \rightarrow \delta(t - t_f) \equiv 1$, equals

$$\mathrm{T} = \kappa_\mathrm{T}\eta^2\omega^2$$

Having the LoE fault parameters included in the thrust model—according to adaptive backstepping—the nonlinear system states are divided into two set of states; x_1 denote the autonomous states where the input does not appear in the derivatives, while x_2 are states whose dynamics are explicitly steered by the system input. Therefore, the system dynamics is written as

$$\begin{aligned} \dot{x}_1 &= f_1(x) \\ \dot{x}_2 &= f_2(x) + g(x, \vartheta)u \end{aligned}$$

Then, after introducing the error between the actual state and the reference as $e = x_1 - x_{1r}$ the following coordinate transformation can be applied

$$\begin{aligned} z_1 &= k_p e + k_i \int_0^t e d\tau \\ z_2 &= x_2 - \nu \end{aligned}$$

where the integral term is required for eliminating steady state tracking errors [33]. Then an adaptive control law can be designed based on the Lyapunov function

$$V = \frac{1}{2}z_1^T z_1 + \frac{1}{2}z_2^T z_2 + \sum_{i=1}^{4} \frac{1}{2\sigma} e_{\vartheta_i}^2$$

where $\sigma > 0$ is the adaptive learning rate of the LoE parameter estimation and $e_{\vartheta_i} = \hat{\vartheta}_i - \vartheta_i$ is the LoE estimation error. The derivation of the adaptive control law is found in [6], however, it is seen that adaptive backstepping laws are as complex to design as MRAC, due to being a DAC method.

Therefore, seeing these two examples, one can conclude that DAC approaches, although yielding a proven asymptotic convergence of the tracking and parameter estimation error to zero, the design is specific to a certain system and cannot handle complete faults in the actuators [31, 6].

4.2 Indirect Adaptive Control

IAC, compared to DAC, is more flexible to design, as the unknown parameters in the system are estimated separately, i.e., the control law does not contain the unknown parameters directly. Therefore, IAC separates the design of the estimator and the controller which gives greater freedom in the synthesis of these subsystem. One of the fundamental works in IAC [34] introduces the structure of an IAC system for an aircraft, similar to what is shown in Figure 4.1. It is seen that flight control system, consisting of the on-board model and the controller is updated by a separate state and parameter estimation module utilizing the two-step method. The two-step method, shown in Figure 4.2, reconstructs the system states (or flight path) via a state estimator, like a Kalman filter, which provides the states for the process (parameter) identification block estimating unknown model parameters such as LoE in case of a multicopter, as shown in Section 4.2.1. These parameters are then used to update/reconfigure the flight control system, making it capable of dealing with unknown model parameters and even faults. Thus, IAC implementations are popular in the FDD literature for both aircraft and multicopters.

This modular structure of IAC systems, therefore, enable the designer to separate control and estimator design. The interaction between blocks, as shown in [34], can also be manipulated meticulously. Furthermore, as will be shown later in this chapter, the estimator module can utilize various approaches to obtain the parameter estimates, for example, LMS estimators, multiple observers, and even neural nets.

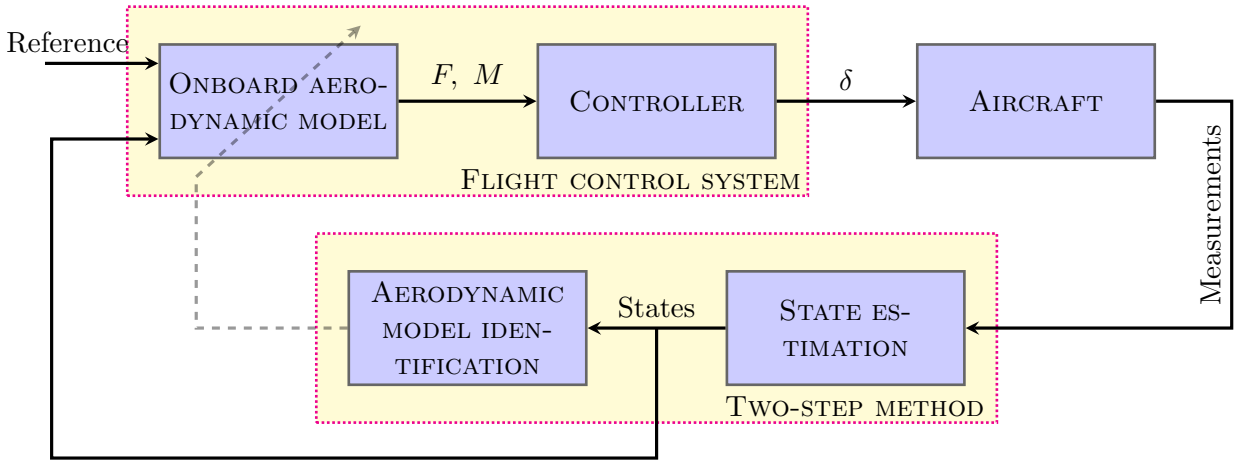

Figure 4.1: Indirect Adaptive Control

4.2.1 Online Loss of Effectiveness estimation

Some researchers [16, 35, 36, 37] are proposing an online, model-identification-based method to estimate the LoE factors ($\eta_1, \ldots, \eta_4$) of the actuators on a quadcopter.

$$u_a = (1 - \eta)u \tag{4.2}$$

Assuming additive fault structure (5.36), these LoE factors are zero, in case the actuators are healthy (actual input u_a equals the desired input u), however, if their value exceeds a certain threshold, full actuator failure is assumed. Denoting the fault-free con-

trol effectiveness matrix by G, see (2.6), the model of the quadcopter can be written as

$$\begin{bmatrix} \mathrm{T} \\ \tau \end{bmatrix} = G \cdot \mathrm{diag}(\omega_1^2, \ldots, \omega_4^2) \begin{bmatrix} 1-\eta_1 \\ 1-\eta_2 \\ 1-\eta_3 \\ 1-\eta_4 \end{bmatrix} \tag{4.3}$$

The wrench (forces T and moments τ) acting on the rigid body is estimated online, based on the model of the multicopter and measurements of the Inertial Measurement Unit (IMU), such as angular velocities and accelerations (4.4).

$$\begin{bmatrix} \hat{\mathrm{T}} \\ \hat{\tau} \end{bmatrix} = \begin{bmatrix} e_3^T R^T m(\dot{v} + g e_3) \\ J\dot{\Omega} + \Omega \times J\Omega \end{bmatrix} \tag{4.4}$$

Then, based on the wrench estimate in (4.4) and the measured rotor speeds (ω) of the propulsion units from (4.3), an estimate for LoE is computed via (4.5).

$$\begin{bmatrix} \hat{\eta}_1 \\ \vdots \\ \hat{\eta}_4 \end{bmatrix} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} - \mathrm{diag}(\omega_1^2, \ldots, \omega_4^2)^{-1} G^{-1} \begin{bmatrix} \hat{\mathrm{T}} \\ \hat{\tau} \end{bmatrix} \tag{4.5}$$

This approach is called the two-step method for online parameter estimation, as shown in Figure 4.2. First, the system states (x), such as translational velocity (v) and angular velocity (Ω) are reconstructed via a Kalman filter. In this figure, the Kalman filter is sensor-based, as it is driven solely by the sensor measurements, which has the advantage of minimal dependency on the multicopter model. The reconstructed states are used to estimate the forces and moments (wrench) acting on the quadcopter. Secondly, a parameter estimate ($\hat{\theta}$) is obtained utilizing, for example, a LMS algorithm. These estimates are then used to convey information about the effectiveness of the actuators.

Thus, the dynamics shown in (4.3) are inverted as if the wrench was driving the system (4.5). Measuring the rotor speed and using the control effectiveness matrix ($G \in \mathbb{R}^{4\times4}$), the LoE factors are obtained.

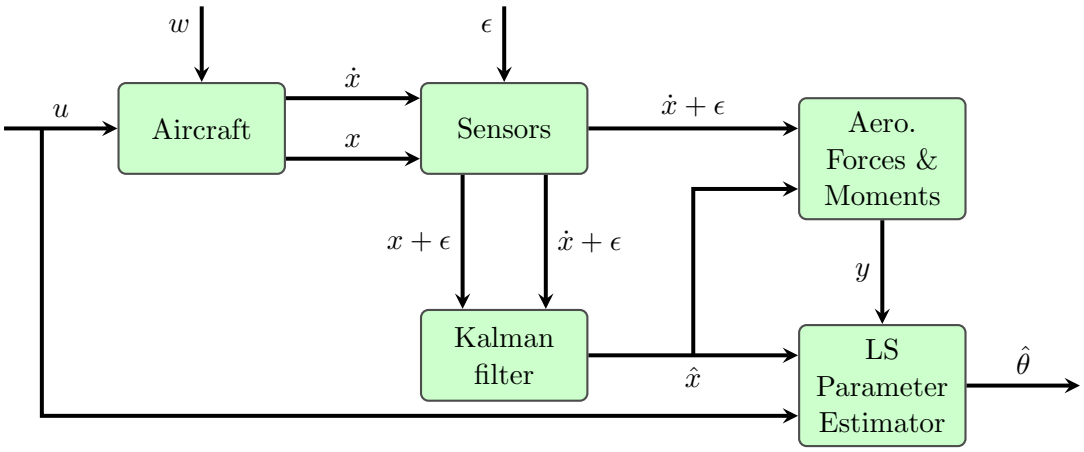

Figure 4.2: The two-step method

The papers present a very detailed description of the actuator aerodynamics which makes it suitable for non-hover applications, however, it also makes this application very specific to a certain quadcopter, as actuator parameters are needed to be identified offline (see Chapter 2).

The drawback of this method is that the estimate of the wrench (forces and moments) acting on the rigid body is rather inaccurate, as derivatives of the angular velocities are required for their computation, thus the system is sensitive to disturbances. Furthermore, the algorithm has only been implemented in simulation with the assumption of all external disturbances being zero.

In addition, the method cannot be generalized for multirotor Unmanned Aerial Vehicles (UAVs) with more than four propulsion units, as the system is underactuated, and fault isolation is not ensured due to having only estimates for thrust and three torque magnitudes. For example, the control effectiveness matrix G of a hexacopter would be a 4-by-6 matrix and by inverting it using the pseudoinverse, information is lost and fault isolation is not trivial. In other words, the row-space of the control effectiveness matrix ($\mathrm{im}(G^T)$) spans $\mathbb{R}^4$ at maximum, as the rank of G and G^T is maximum 4. However, if a multicopter is equipped with six propulsion units, the input to the system will be $[\omega_1^2, \ldots, \omega_6^2]^T \in \mathbb{R}^6$ which is not in the span of the columns of G^T, making the rotor speed unreachable from the four dimensional (underactuated) wrench estimate.

4.2.2 Online control effectiveness matrix estimation

A more recent paper [5] utilizes LMS adaptive filters to estimate the complete control effectiveness matrix including the changes in thrust and torque coefficients. The discrete update law is shown in (4.6).

$$G(k) = G(k-1) - \mu_2 \left(G(k-1) \begin{bmatrix} \Delta\omega \\ \Delta\dot{\omega} \end{bmatrix} - \Delta\dot{\Omega} \right) \begin{bmatrix} \Delta\omega \\ \Delta\dot{\omega} \end{bmatrix}^T \mu_1 \tag{4.6}$$

In (4.6), the finite difference (denoted by Δ) of filtered rotor speeds (ω) and their derivatives ($\dot{\omega}$) are used along with the change in the angular velocity ($\Delta\dot{\Omega}$) of the quadcopter. μ_1 and μ_2 are adaptation constants of the LMS filter, determining stability and convergence. If a fault is detected, then the control reconfiguration module switches to a reduced controller which gives up yaw control [2].

Although the results presented in the paper are very promising, the introduced approach requires double derivatives of the gyroscope readings (angular velocity) which can be too noisy to be reliable. Even though gyroscopes have relatively low noise—at least compared to accelerometers—and the measurements are filtered by a low-pass filter, external disturbances and internal model uncertainties can destabilize the LMS convergence. Furthermore, according to [38], LMS filters lack proven convergence, therefore, their reliability is questionable compared to Lyapunov-based observers.

4.3 Multiple Model Adaptive Estimation

In order to mitigate abrupt changes in the adaptation transients, due to faults like, for example, a considerable LoE in an actuator of a multicopter, several models (obtained offline) can be adapted real time to reconfigure the online model used by the control system [31]. Using a bank of observers for FDD means that there are several pre-identified models

available for the multicopter (or any kind of dynamical system), each model describing the system's dynamics related to a different fault, as shown in Figure 4.3. Then, observers, such as Kalman filters, are simultaneously applied on each model. The output of these parallel observers are then evaluated, leading to the model best fitting the actual (faulty) dynamics of the multicopter. Choosing the right model adaptively requires additional tools, such as a Posterior Probability Evaluator (PPE). Hence the name MMAE for this approach in the literature.

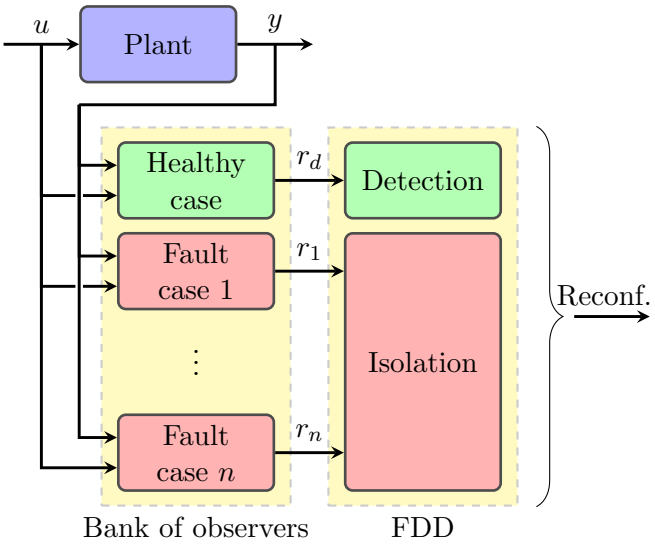

Figure 4.3: Bank of observers

[39] proposes an FDD method based on a bank of global observers and residual analysis. There is a separate observer allocated for the detection of the fault, i.e., a pre-identified model of the healthy system. This observer provides the detection residual r_d which first indicates if an anomaly has occurred, should r_d surpass a pre-defined or adaptive residual threshold $\bar{r}$. According to the definition given in the paper, if the fault occurrence time is denoted by t_f, the detection residual is given as seen in (4.7).

$$\begin{aligned} t < t_f \ &: \ \lim_{t\to\infty} \|r_d\| = 0 \\ t \geq t_f \ &: \ \|r_d\| \neq 0 \end{aligned} \tag{4.7}$$

Thus, if there is a delayed detection time t_d, then the fault is said to be detected if

$$t = t_d > t_f \ : \ \|r_d\| > \bar{r}$$

Then, there are multiple models available on-line which had been separately identified (offline) for damaged actuators. These models help to isolate the fault, i.e., they provide isolation residuals r_i which vanish if the model matches the real (faulty) condition of the damaged drone. By definition, the isolation residuals are given as seen in (4.8)

$$t \geq t_f \ : \ \lim_{t\to\infty} \|r_i\| = 0 \tag{4.8}$$

Since the paper used a linearized model of a quadrotor drone, $n + 1$ Luenberger observers (4.9) can be applied on all the healthy and faulty models to generate residuals,

which is the respective state reconstruction errors ($e = x - \hat{x}$) of the observers. As shown in Figure 4.3, each observer has the input u and output y of the plant and the error dynamics is asymptotically stable if $A - KC$ is Hurwitz.

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + K(y - \hat{y}) \\ \hat{y} &= C\hat{x} + Du \\ \dot{e} &= (A - KC)e \end{aligned} \tag{4.9}$$

If the models are reliable and only partial faults are modeled, this method helps to isolate faults accurately, as the residuals indicate immediately which fault case occurred. However, this also means that there need to be various models identified for the multicopter to match exactly the true model of the damaged model, which requires lot of time and effort to establish. Furthermore, full faults are hard to implement in a structure like this, as yaw control is given up in these models and it is not enough to just alter a few entries in the system matrices B and D, but their structure changes as well.

Another paper applies a similar method for fixed-wing UAV actuator fault detection [40]. The structure of the proposed FDD unit is similar to Figure 4.3. Multiple observers are applied on the pre-identified models of a faulty UAV and their output (state estimates and covariances) is processed by a PPE module. This module is responsible to match the pre-identified models to the actual (faulty) model of the UAV. Stability analysis of a MMAE module is provided in [41], proving that the PPE will converge to the right model. It means that the probability (updated recursively) of an identified model matching the real, faulty model of the UAV will converge to 1 if that model is indeed the best fitting among all models. At the same time, all other model probabilities will converge to 0. Thus this method also implements control reconfiguration by finding the right model for the drone.

In [42], the same structure is implemented for FDD purposes on a multirotor UAV. However, the observers are interacting with each other via a module placed before the filters. This module mixes and weighs all filter state estimates and covariances based on their PPE probabilities. These estimates and covariances serve as an input to the respective filters whose output is mixed again to provide a final state estimate for the control system. Furthermore, fault isolation is performed similarly as before, i.e., the model probabilities should converge to 1 and 0.

4.3.1 Double Model Adaptive Estimation

As mentioned previously, there are some major disadvantages of FDD using MMAE. First of all, each fault case requires a new model to be identified, which also means that yet another observer has to run online parallel to other processes, increasing online computational demand. Therefore, [43] proposes a Double Model Adaptive Estimation (DMAE) structure having a single observer for the healthy and the faulty model each. According to the paper, the double-model approach reduces computational complexity and false alarm rate, and increases the accuracy of FDD. False alarm rate is reduced due to the fact that having lots of models to cover the most possible fault cases leads to overlaps between model residuals which degrades performance and accuracy. Thus, the double model approach has a single state-space representation which includes all fault states, contrary to MMAE, where each fault is represented by a respective model.

The DMAE approach is based on the work [44], which introduces Selective Reinitialization of the bank of observers, based on the model probabilities. The essence of the algorithm is to find the maximal and minimal model probabilities and if the fault-free model probability is below a certain threshold then the fault-free state estimate and covariance matrix is replaced by the respective values of the most matching observer. Furthermore, in order to improve performance a Unscented Kalman Filter (UKF) is used for state estimation of the nonlinear aircraft model.

4.4 Chapter summary

As shown in this chapter, several adaptive control approaches can be tailored into a AFTC system for aerial vehicles. Seeing the working principles of a whole AFTC system is important to apprehend how FDD methodologies, aiming for LoE estimation, can be incorporated into these systems.

First of all, DAC methods were discussed showing their potential for adaptive LoE parameter estimation along with the entangled, complex design of the adaptive control law which ensures asymptotic reference tracking. However, as stated in [31, 6], DAC is not suitable to stabilize system with abrupt and severe uncertainties, meaning that a complete loss of an actuator leads to a scenario unrecoverable by the control system.

Thereafter, IAC was introduced via a few examples applied on multirotor UAV. It was shown that, contrary to DAC, IAC is more flexible to design, as the parameter estimator module is separated from the control system completely. Thus, according to this separation principle, the parameter estimation module can be designed as an individual subsystem whose output is then used to update the adaptive control (or even AFTC) system parameters. The introduced methods were estimating the LoE factors via online parameter estimation and LMS adaptive filtering. Due to the separated design of the control system and the LoE estimation module, the flexibility of the design was shown, and even though both methods are sensitive to disturbances, the method utilizing LMS filtering had been successfully applied in real flight conditions.

Finally, a third method of adaptive control was discussed, called MMAE. MMAE uses several (offline-obtained) models of the system, each representing a different fault case. Then, a switching FDD module is designed which chooses the model matching the faulty plant dynamics the most. The switching is based on outputs of the observers running simultaneously on a model, providing state estimates which are compared to the measured states of the actual plant. The issue with the MMAE approach is that obtaining models takes considerable investment of time and effort. Furthermore, the more complex a system is, the more faults can occur, meaning that more models must be obtained and more observers should run simultaneously while the UAV is actually flying. In order to mitigate these issues, DMAE approach was introduced which reduces the number of required models (and online observers) to two: a healthy and a faulty model. DMAE is said to increase the performance of the FDD framework, as having a lot models for different fault cases may mask the actual fault or trigger a false alarm, while these kinds of conflicts are resolved with a single faulty model.

All in all, the introduced adaptive control methodologies are serving as a frame to indicate the significance and integration of a FDD module within a flight control system. The next chapter will investigate certain kinds of observers which could be utilized as

part of a IAC framework for AFTC.

Chapter 5

Observer design methods

As shown in the previous chapter, there are various adaptive methods available for online Loss of Effectiveness (LoE) estimation. Most of these approaches depend on detailed models (either faulty or healthy) to be identified off/online, requiring significant effort to obtain. In order to overcome this issue, observers can be applied on the nonlinear dynamics of multicopters, depending on less model parameters mostly emanating from the multicopter geometry.

In this chapter, various model-based observers will be introduced, all suitable for Fault Detection and Diagnosis (FDD). Convergence speed, accuracy and flexibility will be the most important aspects serving for their comparison, as in the end, the chosen observer needs to be, above all, fast in LoE estimation.

Most papers discussed here utilize nonlinear observers to estimate LoE as external faults. The chapter starts with the most popular observer, named after Thau, used for multicopter FDD. Then, these observers are extended to account for unmodeled aerodynamic effects and other uncertainties. These robust observers are designed via sliding mode methodologies (Utkin and Walcott-Żak observers) aiming to inject a feedforward discontinuous signal for disturbance rejection. Some Linear Parameter-Varying (LPV) observers are also briefly reviewed. Additionally, nonlinear adaptive observers, a more flexible extension of the previous observers will be discussed. Finally, some Kalman-filter-based approaches are investigated, modeling the LoE as a step-like bias function.

5.1 Nonlinear Thau observer

A Thau observer is a model based, nonlinear observer, first developed by Thau [45] for a kind of nonlinear system of the form

$$\begin{aligned} \dot{x} &= Ax + Bu + f(x,u) \\ y &= Cx \end{aligned} \tag{5.1}$$

with x denoting the states, u the input and y the output of the system.

In order to design an asymptotically stable, nonlinear Thau observer for the system, according to [46], the following two criteria must hold.

- The pair (C, A) has to be observable

- The nonlinear term $f(x,u)$ has to be Lipschitz continuous, i.e. $\|f(x_1,u_1)-f(x_2,u_2)\| \leq \delta\|(x_1,u_1)-(x_2,u_2)\|$

A function is said to be Lipschitz continuous if it is differentiable and its rate of change is bounded by a Lipschitz constant $\delta < \infty$. This bound can be altered by design parameters to increase convergence speed [47].

Considering a system of the form (5.1) with the two assumptions regarding the existence of a Thau observer assumed to hold, an observer can be designed for the system as

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + f(\hat{x},u) - K_t(\hat{y}-y) \\ \hat{y} &= C\hat{x} \end{aligned} \tag{5.2}$$

where hat ($\hat{\bullet}$) symbol denotes estimated signals and K_t is the observer gain computed via

$$K_t = P_t^{-1}C^T \tag{5.3}$$

with $P_t = P_t^T$ being a positive definite solution of the Lyapunov equation

$$A^T P_t + P_t A - C^T C + \theta P_t = 0 \tag{5.4}$$

Equations (5.3) and (5.4) ensure that the observer (5.2) is exponentially stable, as proven in [47]. In this paper it is also stated that tuning parameter θ is chosen such that P_t is a positive definite solution and it determines the convergence dynamics of the observer.

Writing the observer's estimation error as $e = \hat{x}-x$ and the nonlinear term as $f(\hat{x},u) = \hat{f}$, the error dynamics can be written in the form

$$\dot{e} = (A - K_t C)e + \hat{f} - f \tag{5.5}$$

Having $Q = C^T C + \theta P_t \succ 0$ and expanding the observer gain as (5.3), the Lyapunov function (5.4) can be rewritten for the observer dynamics as

$$\begin{aligned} (A-K_tC)^T P_t + P_t(A-K_tC) + Q &= 0 \\ (A-P_t^{-1}C^TC)^T P_t + P_t(A-P_t^{-1}C^TC) + Q &= 0 \\ A^T P_t - C^T C + P_t A - C^T C + C^T C + \theta P_t &= 0 \end{aligned} \tag{5.6}$$

which is clearly equivalent with (5.4). Then, using (5.6), [47] shows that the stability of the error dynamics is ensured if the Lipschitz bound of the nonlinear function is

$$\delta < \frac{\lambda_{min}(Q)}{2\lambda_{max}(P)} \tag{5.7}$$

which is a key constraint for the synthesis of P and Q. In (5.7), $\lambda_{max}(\bullet)$ and $\lambda_{min}(\bullet)$ denotes the maximum and minimum eigenvalue of the argument respectively.

The Thau observer is ubiquitously applied in multicopter literature, as will be shown in the following implementation examples. It's popularity stems from the fact that multicopters have nonlinear dynamics that can be straightforwardly written in the form (5.1) with the nonlinear term being Lipschitz with respect to the states and inputs.

5.1.1 Thau observer for fault detection

One early paper on Thau observers for multicopters introduces a Thau observer applied to detect sensor faults in the system [48]. It shows that Thau observers are suitable to serve as a residual generator to perform FDD on multicopters. As a follow-up, [49] extends the findings of the previous paper for a generic unmanned vehicle. The method is validated on a quadcopter with sensor faults of the gyroscopes. The paper claims that too fast convergence of the observer, tuned via θ in (5.4), is undesirable, as it may hide information from the fault diagnosis unit. However, it is only the case if adaptive thresholds are applied to achieve robust fault detection.

Indeed, in [49, 50], the detection threshold is adapted based on statistical properties (mean, variance) of the residuals which are sensitive to the size of the window used for obtaining these properties. Furthermore, the observer's rate of convergence, if too high, can hide important changes, whereas a too slow convergence can lead to the multicopter becoming unrecoverable.

Residual threshold adaptation adds a certain robustness to the FDD system, as false alarms due to external disturbances are avoided, as shown in Chapter 3. Residual generation methods are derived in great detail in [51] and [50] utilizes these methodologies successfully for sensor fault detection. However, for a highly nonlinear system such as a multicopter, the residuals are obtained via rather complex expressions. Therefore, [49] takes the Thau observer error, i.e., position and attitude estimation errors. If the i^{th} residual is denoted by r_i then the upper ($\overline{\rho}$) and lower ($\underline{\rho}$) detection thresholds are adapted via the mean, variance and some constants as

$$\begin{aligned}\overline{\rho} &= \frac{\overline{k}_1}{T}\int_t^{t+T} r_i(\tau)d\tau + \frac{\overline{k}_2}{T}\int_t^{t+T} r_i(\tau) - \frac{1}{T}\int_\tau^{\tau+T} r_i(\nu)d\nu d\tau + \overline{k}_3 \\ \underline{\rho} &= \frac{\underline{k}_1}{T}\int_t^{t+T} r_i(\tau)d\tau + \frac{\underline{k}_2}{T}\int_t^{t+T} r_i(\tau) - \frac{1}{T}\int_\tau^{\tau+T} r_i(\nu)d\nu d\tau + \underline{k}_3\end{aligned} \tag{5.8}$$

where the window size T and adaptive gains $\overline{k}_j$ and $\underline{k}_j$ $(j = 1, 2, 3)$ are used for fine-tuning the behaviour of adaptive thresholds. Choosing T to be smaller, the thresholds will be more sensitive to small changes in the residuals and thus false alarms, therefore a trade-off between sensitivity and resolution has to be made when choosing the right window size.

5.1.2 Thau observer for fault detection and diagnosis

Papers introduced so far implemented Thau observers for only the detection of faults. However, [52] went a step further and implemented a complete FDD system with two Thau observers. The first observer generates residuals for fault detection and the second observer adaptively estimates the fault for diagnosis. Thus, based on the complete FDD module, an Active Fault-Tolerant Control (AFTC) system could be implemented on a multicopter.

$$\begin{aligned}(A - KC)^T P + P(A - KC) + \gamma^2 PP + \mathcal{I} &= -Q \\ E^T P &= GC\end{aligned} \tag{5.9}$$

The existence of $P, Q \succ 0$ being solutions of (5.9) leads to an adaptive algorithm for fault detection shown in (5.11).

In the following, the observer in [53] is examined. A nonlinear Thau observer is applied on the dynamics of a quadcopter, described in (5.1). Since the nonlinear term in the quadcopter dynamics (2.1) contains the gyroscopic moments, the nonlinearity is a cross product (a bilinear map), which is only locally Lipschitz continuous. This means, according to the paper, that in order to design a globally stable observer, the angular velocity Ω has to be measured when evaluating the nonlinear term. This is not an issue if a sensor-based observer, which converges faster than the Thau observer, yields an accurate, bias-free measurement of Ω.

The Thau observer (5.10) in [53] serves for detection and adaptive fault estimation as well. Furthermore, it provides auxiliary states for a second cascaded linearized (exogeneous) Kalman filter, which removes noise and estimates the fault size more accurately. The Thau observer implemented is augmented with a fault term $E\hat{\eta}$ such that

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + f(\hat{x}, u) + E\hat{\eta} + K(y - \hat{y}) \\ \hat{y} &= C\hat{x} \end{aligned} \tag{5.10}$$

with an adaptive update law for the fault estimate $\hat{\eta}$

$$\dot{\hat{\eta}} = WG^T(y - \hat{y}) - \sigma W\hat{\eta} \tag{5.11}$$

In (5.11), W is a symmetric, positive definite weight matrix, σ is a positive constant such that $\sigma - \lambda_{max}(W^{-1})$. $\lambda_{max}(\bullet)$ denotes the greatest eigenvalue of its argument matrix. Additionally, $G = CPE$ from (5.9), and thus the error dynamics are proven to be globally, asymptotically stable [53]. From (5.11) it is seen that the fault estimation error dynamics is given by

$$\dot{\eta} - \dot{\hat{\eta}} = \dot{e}_\eta = \dot{\eta} - WG^T(y - \hat{y}) + \sigma W\hat{\eta} = \dot{\eta} + \sigma W\eta - \sigma We_\eta - WG^T(y - \hat{y}) \tag{5.12}$$

Thus, it can be conveyed from (5.12) that the fault estimation error depends on the fault severity and some tuning parameters W and σ.

Finally, the actual (more accurate) fault estimation is performed by a linearized (exogeneous) Kalman filter using the state estimates from the Thau observers. It removes additional noise from the signals and provides the final state estimates.

It should be noted, however, that this model enables the estimation of the fault as an additive signal, i.e., an unknown thrust/torque input. Therefore, the estimated control effectiveness is not obtained this way, unless matrix E can be written such that the estimated fault $\hat{\eta}$ is actually the loss of effectiveness. Nevertheless, it is also possible to estimate fault torques by $\hat{\eta}$ and then compute the effectiveness based on the output that the model suggests and the estimated faulty output.

In order to mitigate the effects of disturbances and unmodeled dynamics robust observer synthesis is introduced in the next section. However, as an interesting solution, [54] implements a Thau observer for robust actuator fault estimation of a quadcopter with a three-step optimization method. First, a parameter estimation algorithm is used for online estimation of unmodeled terms, based on observer estimation error. Furthermore, since faults on the yaw channel have considerable less effect on the residuals, the yaw torque input is amplified in order to obtain the estimation error for yaw in the same magnitude as pitch and roll. Therefore, the observer estimates can be made more accurate. Finally, both sensor and actuator data are filtered by a Infinite Impulse Response (IIR) filter.

Remark
It should be pointed out, that [54] refers to [48] on how the Thau observer gain in (5.4) *is chosen. The latter defines P_t as the solution of*

$$A^T P_t + P_t A - C^T C + \theta C^T P_t = 0 \tag{5.13}$$

however, this definition does not match the original equation in [47, 30] and has dimension mismatch thereof. Since both [54, 48] define the observer gain as (5.3) *one may conduct the following analysis for matrices $P_t \in \mathbb{R}^{n\times n}$ and $C \in \mathbb{R}^{m\times n}$*

$$\begin{aligned} K = P_t^{-1} C^T \in \mathbb{R}^{n\times m} \rightarrow KC \in \mathbb{R}^{n\times n} \\ \textit{but } C^T P_t \rightarrow \mathbb{R}^{n\times m}\mathbb{R}^{n\times n} \\ \textit{has dimension mismatch.} \end{aligned}$$

Therefore, (5.13) *is feasible only if C is square, i.e. $m = n$, which holds in both papers with $C = \mathcal{I}_{n\times n}$, but not stated explicitly.*

5.2 Robust observers

Observers for uncertain nonlinear systems were discussed in the 80's by Walcott and Żak in [55] with detailed sliding mode extensions introduced in [56]. This type of sliding mode (Utkin) observer got its name after one of the authors and it introduces a discontinuous feedforward term to the system dynamics attenuate external disturbances. Therefore, the robustness of the nonlinear observer is greatly enhanced.

5.2.1 Discontinuous observers

Before investigating actual implementations of nonlinear sliding mode observers for multicopter FDD, some general sliding mode observer structures will be introduced. Considering a system of the form (5.1) with (A, C) observable and f being an unknown function describing bounded system uncertainty, an Utkin observer can be designed for the system [57]. It is assumed that the number of states are at least equal to the number of observed outputs, i.e. $A \in \mathbb{R}^{n\times n}$ and $C \in \mathbb{R}^{p\times n}$ with $p \leq n$. Since the aim is to drive both the state estimation error and the output estimation error asymptotically to zero, Utkin proposed to include the output in the state vector via

$$\begin{bmatrix} x_1 \\ y \end{bmatrix} = Tx \tag{5.14}$$

It is assumed that C can be partitioned such that $C = [C_1\ C_2]$, then the transformation matrix is given as

$$T = \begin{bmatrix} I_{n-p} & 0 \\ C_1 & C_2 \end{bmatrix} \tag{5.15}$$

Performing the state-transformation on the system matrices using T from (5.15), the Utkin observer is written as shown in (5.16).

$$\begin{aligned} \dot{\hat{x}}_1 &= A_{11}\hat{x}_1 + A_{12}\hat{y} + B_1 u + \Gamma\nu \\ \dot{\hat{y}} &= A_{21}\hat{x}_1 + A_{22}\hat{y} + B_2 u - \nu \end{aligned} \tag{5.16}$$

In (5.16), the following transformations were performed

$$TAT^{-1} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$
$$TB = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$$

The Utkin observer (5.16) introduces a discontinuous feedback signal ν depending on the output estimation error and defined as

$$\nu = H\text{sgn}(\hat{y} - y) \tag{5.17}$$

where $H \succ 0$ is a diagonal matrix having the same positive constants in its main diagonal.

Since (A, C) is observable, the pair (A_{11}, A_{21}) in (5.16) is observable as well; the proof—based on the Hautus-test—is found in [58]. Therefore, feedback gain matrix $\Gamma \in \mathbb{R}^{(n-p)\times p}$ can be designed such that all poles of $(A_{11}+\Gamma A_{21})$ lie in the open left half-plane.

In the same paper, Walcott-Żak observer design is introduced as well [57]. The system under consideration has the same form as (5.1), with matched uncertainty, i.e., the uncertainty enters through the input channel such that $f(x, u) = B\xi$ where ξ is a norm bounded uncertainty function. Then, if $\exists K$ such that $A - KC$ is Hurwitz, a Lyapunov pair (P, Q) has to be synthesized based on (5.6) such that, a structural constraint of the form

$$C^T F^T = PB$$

is satisfied. Finding a suitable matrix F is important to ensure sliding motion as the discontinuous feedforward signal is defined as

$$\nu = \begin{cases} -\rho \frac{P^{-1}C^T F^T FCe}{\|FCe\|} & \text{if } FCe \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.18}$$

Sufficient and necessary conditions for the existence of such matrices are given in [59].

5.2.2 Sliding mode Thau observers

A nonlinear sliding mode Thau observer is introduced in [60] to account for external disturbances acting on the system dynamics. The system is described as

$$\begin{aligned} \dot{x} &= Ax + Bu + f(x, u) + H_d d \\ y &= Cx \end{aligned} \tag{5.19}$$

with an additional (unknown) disturbance term $H_d d$. It is assumed that d is bounded by an unknown constant such that $\|d\| \leq \Delta_d$. Thus, a nonlinear (sliding mode) observer of the form (5.22) is designed.

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + f(\hat{x}, u) + \Gamma_d \nu + K(y - \hat{y}) \\ \hat{y} &= C\hat{x} \end{aligned} \tag{5.20}$$

Due to the nonzero uncertainty term in (5.19), the proposed observer (5.20) cannot guarantee asymptotic stability of the estimation error, however, it is able to keep the estimate within a close neighbourhood of the actual states.

Note the virtual feedforward term $\Gamma_d\nu$ in the observer dynamics (5.20). If $C\Gamma_d$ is nonsingular and the disturbance matching condition (5.21) holds

$$\exists\ D\ :\ H_d = \Gamma_d D \tag{5.21}$$

then the stability (at least in the sense of Lyapunov) of the error dynamics is guaranteed [60]. This relaxed stability might not be sufficient for accurate fault estimation but it could still yield useful indication of fault occurrence.

Although only tested for a 30% actuator fault, [11] applies a robust sliding mode Thau observer for a quadcopter. It has a very similar approach as the previous paper, but it augments the observer by a fault estimation term term ($E\hat{\eta}$) such that the observer has the form of

$$\begin{aligned}\dot{\hat{x}} &= A\hat{x} + Bu + f(\hat{x}, u) + E\hat{\eta} + H_d d + K(y - \hat{y})\\ \hat{y} &= C\hat{x}\end{aligned} \tag{5.22}$$

This observer is robust towards external noise due to the introduction of a sliding mode feedforward noise-canceling term. Furthermore, it is relatively fast and is able to accurately estimate fault severity. Synthesis of observer gains matrices is performed via construction of Linear Matrix Inequalities (LMIs) which lead to a convex optimization problem that can be solved offline via a dedicated solver, such as YALMIP [61].

5.2.3 LPV observer design

In order to overcome and attenuate the effects of external disturbances and model uncertainties decreasing the performance of observers, LPV-based synthesis methods can be utilized. LPV observers are especially useful for fixed-wing aircraft as a linear model is straightforward to obtain in trimmed flight conditions. However, since mass, turbulence, and aerodynamics parameters are constantly changing, LPV-based controller/observer synthesis can be used to expand the applicability of the linear models on a broad flight envelope. LPV-based sliding mode observer synthesis is frequently used in the literature for FDD purposes, mostly for aircraft. Even though LPV-based observers for FDD purposes might be cumbersome to implement on multicopters, there are interesting and relevant pieces of research on the topic which are worth investigation.

In [62], robust LPV-based sliding mode observer design is introduced for both actuator and sensor faults on rigid-wing aircraft. The general LPV system for the linearized model is given in (5.23)

$$\begin{aligned}\dot{x} &= A(\rho)x + B(\rho)u + E(\rho)\eta + H\xi\\ y &= Cx + d\end{aligned} \tag{5.23}$$

It is seen in (5.23) that the system matrices A, B, and E are all functions of the scheduling parameter ρ which represents the varying parameters in the uncertain model. ξ denotes the input disturbances while d denotes the output disturbance, assumed to be low-frequency signal. The sliding mode observer is then proposed in the form of (5.24).

$$\begin{aligned}\dot{\hat{x}} &= A(\rho)\hat{x} + B(\rho)u - K(\rho)(\hat{y} - y) + \Gamma\nu\\ \hat{y} &= C\hat{x}\end{aligned} \tag{5.24}$$

The feedforward injection signal ν has the same purpose as seen before, that is, to induce the sliding motion and estimate occurred faults. This estimate is called equivalent output injection which is the average value of ν while it is preserving sliding.

The essence of the design algorithm is to first define the structure of the discontinuous injection signal ν with tunable parameters providing design freedom. Then, synthesis of K and Γ are translated to an Linear Matrix Inequality (LMI) optimization problem aiming to minimize the effect of uncertainty on the fault estimate $\hat{\eta} = W\nu$ with a weight matrix W. The resulting observer, however, is rather complicated to tune and the method is synthesized using linearized models.

Authors of [4] also implement a robust, LPV-based observer to detect faults to the aileron of a rigid-wing aircraft. Its advantage is that it is able to handle model disturbances of the actuator and external effects as well, which can also be a disadvantage, as the accuracy of detection might be sacrificed for stability. The paper introduces multiple methods for FDD. The first method is a parity space method, where estimates of the aircraft state based on the actuator state and system dynamics are compared to measured signals, e.g., angular velocities. This way, a residual is generated between the online model and the actual faulty model which is suitable for fault detection, but not necessarily for diagnosis. The second method has an LPV-based robust observer for actuator (elevon) state estimation based on measured signals (from the Inertial Measurement Unit (IMU)). This estimate is then compared to the actuator states obtained from an online actuator model, i.e., a transfer function from a commanded actuator deflection to the actual actuator state.

Finally, an LPV observer had also been implemented on a quadcopter in [63] for FDD. The scheduling parameters (ρ) were the angular velocities and the cosine of the roll and pitch angle. These parameters are assumed to be perturbed at near hover conditions, furthermore, the pitch rate parameter was assumed to be prone to 50% uncertainty of its actual value due to IMU errors. Regardless of the erroneous scheduling parameters, the observer was able to detect actuator faults of the nonlinear system, even though the observer is based on the linearized model of the system. However, only 20 Revolutions Per Minute (RPM) additional rotor speed was introduced as actuator fault, which is relatively low, compared to the several thousand RPM rotor speeds in hover and certainly far from a full actuator fault.

5.2.4 Robust observer synthesis

LMI formulation for observers dedicated for aircraft actuator fault estimation has been introduced in [64]. The derived LMI formula serves as a basis for fast actuator fault estimation, as the proportional term (fault estimate) for the fault update law (5.11) has been developed in this paper. Previously, only an integrator term containing the estimation error $(y - \hat{y})$ was used in the update law but in this paper it was proven that the error dynamics performance is increased by the addition of a proportional term.

Since [64] used a linear system representation, and thus a linear observer, [65] derives robust synthesis methods leading to LMI-based optimization for nonlinear systems with dynamics in the form of (5.1). In this paper it is proven that the $\mathcal{H}_\infty$ norm of the transfer function from external disturbances to the residuals serving for FDD can be minimized via solving LMIs. Denoting the external disturbance vector by d, the residual vector by r and the Lyapunov function of the observer estimation error as V it is shown that the following inequality holds for the system

$$\dot{V} + r^T r - \gamma^2 d^T d = \dot{V} + \|r\|_2^2 - \gamma^2 \|d\|_2^2 < 0 \tag{5.25}$$

where γ denotes the $\mathcal{H}_\infty$ norm of the transfer function (T_d^r) from d and r. This result proves that the system describing the error dynamics is dissipative. Indeed, [66] defines dissipativity for a system in terms of a storage and a supply function. The storage function, in this case is the energy like Lyapunov function defined in terms of the observer estimation error e as

$$V = e^T P e$$

The system is said to be dissipative if the following "differential dissipation inequality" holds [66]:

$$\frac{d}{dt}V \leq s(d,r) \tag{5.26}$$

Thus, having the supply function defined as $s(d,r) = \gamma^2\|d\|^2 - \|r\|^2$, one obtains (5.25) from (5.26). Then, according to the bounded real lemma, the 2-norm of the performance variable (residuals) is bounded by γ^2 times the 2-norm of the disturbance variable, therefore

$$\frac{\|r\|_2}{\|d\|_2} \leq \gamma^2 \;\Rightarrow\; \|T_d^r\|_\infty \leq \gamma \quad \forall\, \|d\|_2 < \infty$$

meaning that the energy of the residual signals—due to disturbance inputs—is minimized if γ is minimized using LMI-synthesized observer gains.

Furthermore, [65] shows that another important relationship can be effectively included in LMI optimization, namely that the faults should appear in the residuals for FDD. Thus, the sensitivity of the transfer function (T_η^r) from faults (η) to residuals (r) must meet the following criterion

$$\inf \underline{\sigma}\left(T_\eta^r\right) = \frac{\|r\|}{\|\eta\|} > \beta^2 \tag{5.27}$$

Thus, (5.27) ensures that minimal singular value ($\underline{\sigma}$) is greater than a certain positive bound and the faults are detectable in the residuals.

5.3 Nonlinear adaptive observers

One of the early papers [67] on adaptive observers extends the Thau observer for a class of nonlinear systems of the form

$$\begin{aligned} \dot{x} &= Ax + f(x,u) + bg(x,u)\theta \\ y &= Cx \end{aligned} \tag{5.28}$$

where $\theta \in \mathbb{R}^p$ denotes a parameter of the system, $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, $b \in \mathbb{R}^{n\times k}$, $f : \mathbb{R}^n \to \mathbb{R}^n$, $g : \mathbb{R}^n \to \mathbb{R}^{k\times p}$. Then it is shown that the following assumptions must hold in order to design an asymptotically stable adaptive observer:

1. f and g must be Lipschitz continuous with respect to the states x with Lipschitz constants δ_1 and δ_2 respectively.
2. The unknown parameter vector is bounded such that $\|\theta\| \leq \delta_3$
3. An observer gain matrix K can be chosen such that $\delta_1 + \delta_2\delta_3\|b\| < \frac{\lambda_{min}(Q)}{2\lambda_{max}(P)}$ where Q is the positive definite solution of (5.6).

4. The pair (A, C) is observable (Implied by the previous point)

5. $\exists P \succ 0$ such that $b^T P = C_1$ with each row of C_1 being an element in $\text{im}(C^T)$

The first four requirements are emanating from the Thau observer structure, however, the last point is an interesting addition which is required to construct a suitable adaptive law of the form

$$\dot{\hat{\theta}} = \frac{1}{\psi} f^T(\hat{x}, u)(y - C\hat{x}) \tag{5.29}$$

with a positive constant ψ. (5.29) is the result of the parameter error dynamics that makes the system stable in the sense of Lyapunov.Denoting the parameter estimation error by $e_\theta = \theta - \hat{\theta}$ one obtains that Lyapunov stability is ensured by, among other conditions, (5.30) describing the parameter estimation error dynamics.

$$\dot{e}_\theta = -\frac{f^T(\hat{x}, u)}{\psi} b^T P e = -\frac{f^T(\hat{x}, u)}{\psi} C_1 e \tag{5.30}$$

It is now seen in (5.30) why each row of C_1 has to be in the row space of C: the parameters have to be observable from the measured states and if there are additional entries in C_1 pertaining to unmeasured states, then some unknown signals would enter the observer estimate for the parameters that cannot be accounted for. Since it is not straightforward to synthesize P and C_1 simultaneously, [68] goes through the synthesis process meticulously using LMIs and the Schur complement (see Appendix A).

Two types of observers, suitable for fault estimation, are listed in one of the most recent papers on quadcopter actuator FDD [10]. First a linear Proportional-Integral Observer is utilized, which has a straightforward synthesis process, namely solving an LMI problem. The paper only injected partial faults to the actuators, i.e., there was no actuator with 100% LoE. This means, that the system, linearized in hover, could still deal with fault estimation and recovery. However, when a full fault occurs, the multirotor might reach a configuration where the linearized dynamics do not hold, and the observer becomes unstable.

To deal with these extreme situations, a nonlinear adaptive observer is introduced in [10] as well. Assuming that the nonlinear dynamics is written as

$$\begin{aligned} \dot{x} &= f(x, u) + g(x, u)\eta \\ y &= h(x, u) \end{aligned} \tag{5.31}$$

then, according to [69], the system can be rewritten in a 'so called' nonlinear adaptive observer form,

$$\begin{aligned} \dot{y} &= \alpha(y, z, u, t) + \beta(y, z, u, t)\eta \\ \dot{z} &= \delta(y, z, u, t) \end{aligned}$$

with z denoting the unmeasurable states. Using this form, a nonlinear adaptive observer of the following structure is designed for the system (5.32).

$$\begin{aligned} \dot{\hat{y}} &= \alpha(y, \hat{z}, u, t) + \beta(y, \hat{z}, u, t)\hat{\eta} - k_y(\hat{y} - y) \\ \dot{\hat{z}} &= \delta(y, \hat{z}, u, t) \\ \dot{\hat{\eta}} &= -k_\eta \beta^T(y, \hat{z}, u, t)(\hat{y} - y) \end{aligned} \tag{5.32}$$

Functions α and β are Lipschitz continuous with respect to the unmeasurable states z with Lipschitz constants γ_α and γ_β respectively. k_y and k_η are positive design parameters. This observer is proven to drive the unknown fault parameter estimate, e.g., the LoE of the actuators, to the actual parameter, if β is persistently exciting. The definition of persistent excitation is given in [70] as

Definition 5.3.1 (Persistent excitation)
A time-varying signal $u : \mathbb{R}^+ \to \mathbb{R}^n$ *persistently excites if* $\exists T, \overline{k}, \underline{k} > 0$ *that* $\forall t \geq 0$*:*

$$\overline{k}\mathcal{I}_n \geq \int_t^{t+T} u(\tau)u^T(\tau)d\tau \geq \underline{k}\mathcal{I}_n \tag{5.33}$$

In other words, the energy introduced to a system through all channels of signal u in (5.33) is always positive and bounded for periods of T, therefore, no system behaviour the observer wants to capture remains masked.

Then, the observer is designed based on the Lyapunov function candidate

$$V = \frac{\epsilon}{2}e_y^T e_y + V(t, e_z) + \frac{\epsilon}{2k_\eta}e_\eta^T e_\eta$$

where $\epsilon > 0$, $e_y = \hat{y} - y$, $e_\eta = \hat{\eta} - \eta$, and $e_z = \hat{z} = z$ denote estimation errors. Thereafter, it is found that

$$\dot{V} \leq -\epsilon k_y \|e_y\|^2 + \epsilon\|e_y\|(\gamma_\alpha + \gamma_\beta\eta)\sqrt{\kappa(e_z)} - \kappa(e_z) \tag{5.34}$$

where $\kappa(e_z) \geq 0$, for example, $\kappa(e_z) = \kappa\|e_z\|^2$ with $\kappa > 0$. By introducing a new variable

$$e = \begin{bmatrix} \|e_y\| \\ \sqrt{\kappa(e_z)} \end{bmatrix}$$

(5.34) can be rewritten as

$$\dot{V} \leq -e^T \underbrace{\begin{bmatrix} \epsilon k_y & -\frac{\epsilon}{2}(\gamma_\alpha + \gamma_\beta\eta) \\ -\frac{\epsilon}{2}(\gamma_\alpha + \gamma_\beta\eta) & 1 \end{bmatrix}}_{M} e \tag{5.35}$$

with $M = M^T \succ 0$ for a sufficiently small ϵ and a bounded unknown parameter η. Due to the non-strict inequality in (5.35), Barbalat's lemma is used to prove the asymptotic convergence of the estimation errors. The detailed proof for asymptotic state and parameter estimation is found in [69]. It is assumed that the unknown parameter is slowly varying, i.e. $\dot{\eta} \approx 0$ which does not hold for LoE modeled as a piecewise constant step function whose derivative is zero except at the time of fault occurrence. However, this does not cause an implementation issue as shown in [10].

For further reference, some recent extensions of nonlinear adaptive filters can be found in [71, 72, 73], introducing observers for more general systems, for example, ones that have the estimated fault parameter embedded in the nonlinear dynamics, as opposed to (5.10). Furthermore, since this approach had been shown to outperform LPV and linear observers (in terms of convergence speed) on a nonlinear quadcopter model in simulation [10], the method is definitely worth to be implemented, at least for preliminary testing.

5.4 Kalman filters

In the final section of this chapter, some Kalman-filter-based methods for FDD will be introduced. A Kalman filter is an optimal observer widely used for state estimation in a stochastic environment. As shown in Figure 5.1, an aerial vehicle is a stochastic system, as system noise w is present in the dynamics of the vehicle due to unmodeled disturbances, furthermore, the sensor bias λ and sensor noise ϵ is added to the actual system output y. In order to reconstruct the system's true states x, the state vector is augmented by sensor biases, and some *a-priori* known noise characteristics are introduced to the system by setting up covariance matrices Q and R of the process and sensor noise signals respectively. Then, the Kalman filter is applied to estimate the system states (and sensor biases) along with their covariance. Thus, an optimal observer provides unbiased and filtered signals as feedback for the control system. Through the following pieces of literature, it will be shown how LoE signals can be regarded as an abrupt bias appearing in the system after a fault.

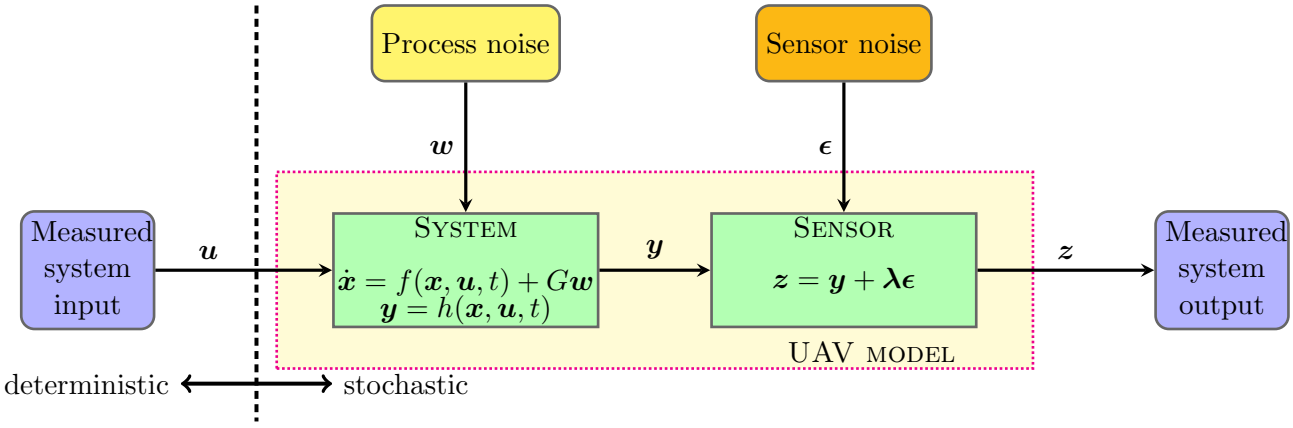

Figure 5.1: Stochastic process

First of all, [74] introduces a Two-Stage Kalman Filter (TSKF) for actuator FDD on quadcopters. The aim of the observer is to explicitly estimate the LoE of the actuators as unknown bias terms. Even though the model of the quadcopter is linearised in hover, restricting the applicability of the method to near-hover configurations, it is worth investigating the structure of the proposed TSKF, since it may be straightforward to extend it to a nonlinear model of the quadcopter.

Actuator LoE is modeled similarly to [16, 10] such that the actual input (u_a) exerted by a single actuator depends on the commanded input (u) and the fault term (η) according to (5.36).

$$u_a = (1 - \eta)u \tag{5.36}$$

Thus, $\eta = 0$ means a completely healthy, $\eta = 1$ a completely lost actuator. (5.36) is included in the discretized linear system dynamics as

$$\begin{aligned} x_{k+1} &= Ax_k + Bx_k - B \cdot \mathrm{diag}(u_k)\eta_k + w_k^x \\ \eta_{k+1} &= \eta_k + w_k^\eta \\ y_{k+1} &= Cx_{k+1} + v_{k+1} \end{aligned} \tag{5.37}$$

In (5.37), w_k^x denotes the system noise, whereas v_{k+1} stands for the measurement noise. Since the LoE is not a constant bias, but a unit step-like function, a large additive noise term w_k^η is introduced to mask low frequency changes in the LoE estimate [75]. This noise term, introduces a trade-off between the estimation convergence rate and the accuracy of the steady-state estimate of the LoE. Introducing individual forgetting factors for each actuator LoE estimate, however, increases the performance of the TSKF [75].

The first stage of the TSKF is to perform state estimation without the bias term on the system dynamics in (5.37). The first stage, therefore, provides the predicted state $\tilde{x}_{k+1|k}$ and the optimal state estimate $\tilde{x}_{k+1|k+1}$ without accounting for the LoE. LoE estimation is the task of the second Kalman filter which estimates the fault terms based on the residuals between the bias-free Kalman filter's state prediction $\tilde{x}_{k+1|k}$ and the sensor measurements y_{k+1}.

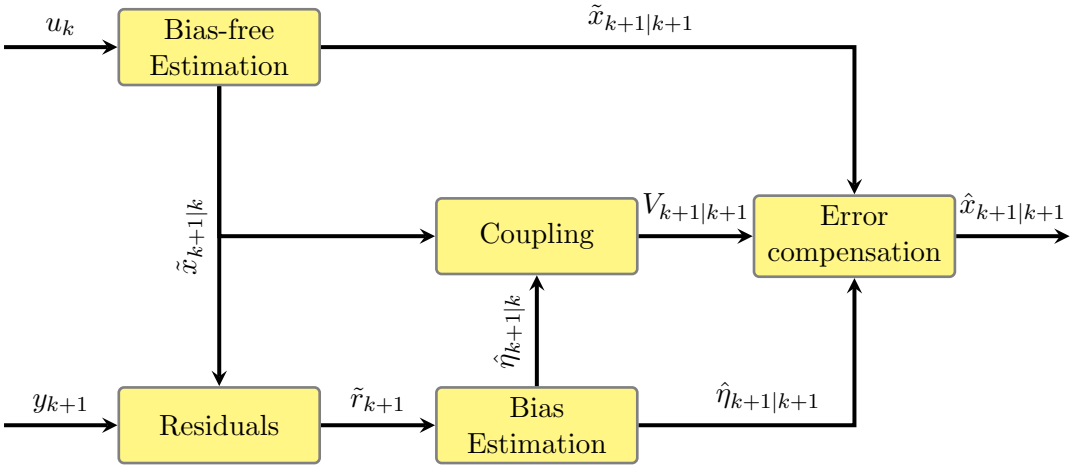

Figure 5.2: Two-stage Kalman Filter structure[74]

The structure of the TSKF is shown in Figure 5.2[74], with a linear, bias-free Kalman filter performing the initial state estimation followed by an optimal fault estimator. Since there are some statistical coupling between the states and the LoE parameters, the output of the two Kalman filters is needed to be merged via coupling equations. Finally an error compensation unit is applied on the state and LoE estimates with their statistical correcting terms. Thus, a corrected state estimate ($\hat{x}_{k+1|k+1}$) of the multicopter is acquired.

The TSKF method seems flexible enough to replace the bias-free state estimator by either an Extended Kalman Filter (EKF) or Unscented Kalman Filter (UKF) to account for the nonlinear dynamics of a multicopter, thus expanding the usefulness of this approach. Although the coupling equations need to be revised as well, the method is definitely useful for FDD purposes, especially since there is already a working state estimator on most multirotor Unmanned Aerial Vehicles (UAVs).

5.5 Chapter summary

This chapter discussed various observers used for FDD in the literature. Most applications dealt with nonlinear observers to enhance their usability even in non-hover situations. A

great advantage of these observers is the proven parameter estimation convergence which is crucial for the reliable reconfiguration of the control system.

Firstly, nonlinear Thau observers were introduced which are able to detect additive faults such as external wrench components (thrust and torque) acting on the rigid body due to LoE of the actuators. Examples were shown for actuator fault detection and fault diagnosis performed via a Thau observer on multicopters.

In order to make the detection more reliable and robust, adaptive thresholds can be implemented to provide active robustness to the FDD module. Another method is to synthesize robust observers which introduce passive robustness to the system. Several robust observers were reviewed, from discontinuous sliding mode observers to LPV observers along with robust observer gain synthesis methodologies.

Thereafter, it was shown how adaptive, nonlinear observers provide a powerful tool to estimate LoE parameters online. Their advantage is that the unknown parameters can be sewn into the nonlinear dynamics of the multicopter and still be estimated. However, certain excitation conditions must hold to achieve asymptotic convergence and observability.

Finally, a TSKF approach was investigated which has shown promising results for LoE estimation on a quadcopter. Even though the method was only used in hover conditions and the Kalman filter was linear, it provides an interesting lead to consider the extension of either a UKF or EKF used for state estimation and being already present on the platform.

Part II

Scientific paper

Multicopter Actuator Effectiveness Estimation for Fault Detection and Diagnosis

Mark T. Melczer * and Coen C. de Visser†
TU Delft Department of Aerospace Engineering, 2629HS Kluyverweg 1, Delft, The Netherlands

Pepijn van den Bos‡
Fusion Engineering, 2629HS Kluyverweg 1, Delft, The Netherlands

In order to meet the ever-increasing safety standards set by authorities for multicopter Unmanned Aerial Vehicles (UAVs), researchers immerse themselves in the analysis of fast Fault Detection and Diagnosis (FDD) methods augmenting Active Fault-Tolerant Control (AFTC) systems. This paper proposes a fast and reliable FDD system which is anticipated to be incorporated into an AFTC system saving the multirotor UAV from crashing after a fault is detected. The proposed framework is aimed to be flexible, meaning that its implementation on various multicopter platforms is straightforward and requires little effort, thus making it available for widespread use. Ensuring that multirotor UAVs are able to detect failures and update their control system accordingly will propel the development of AFTC systems for drones, leading to faster certification of such systems by authorities, enabling large scale and more flexible utilization of autonomous flying.

Acronyms

AFTC Active Fault-Tolerant Control

BLDC Brushless DC

CG Center of Gravity

ESC Electronic Speed Control

FDD Fault Detection and Diagnosis

HIL Hardware-in-the-loop

IMU Inertial Measurement Unit

INDI Incremental Nonlinear Dynamic Inversion

*M.Sc. student, TU Delft - Control & Simulation group, m.t.melczer@student.tudelft.nl
†Associate Professor, TU Delft - Control & Simulation group, c.c.devisser@tudelft.nl
‡Lead Control Systems Engineer, Fusion Engineering, pepijn@fusion.engineering

LMI Linear Matrix Inequality

LMIs Linear Matrix Inequalities

LoE Loss of Effectiveness

MMAE Multiple Model Adaptive Estimation

NDI Nonlinear Dynamic Inversion

OLS Ordinary Least Squares

PSD Power Spectral Density

PWM Pulse Width Modulation

UAV Unmanned Aerial Vehicle

Nomenclature

e_3	Unit vector of Z direction
g	Gravitational acceleration constant $\left[\frac{m}{s^2}\right]$
η	Actuator effectiveness parameter
J	Moment of inertia $\left[\frac{kg}{m^2}\right]$
m	Mass $[kg]$
r	Position vector $[m]$
R	Rotation matrix
T	Generated thrust $[N]$
v	Velocity vector $\left[\frac{m}{s}\right]$
$\mathcal{W}$	Wrench
κ_e	Back-EMF coefficient $\left[\frac{Vs}{m}\right]$
κ_{T}	Thrust coefficient $\left[\frac{Ns^2}{rad^2}\right]$
κ_τ	Torque coefficient $\left[\frac{Nms^2}{rad^2}\right]$
κ_q	Motor torque coefficient $\left[\frac{Nm}{A}\right]$
τ	Torque vector $[Nm]$
ω	Rotor speed $\left[\frac{rad}{s}\right]$
Ω	Angular velocity vector $\left[\frac{rad}{s}\right]$

I. Introduction

According to recent economic research reports, the drone market is predicted to grow significantly by 2020 [1, 2]. This projects an increasing number of UAVs present in the airspace which, apart from remote, hardly accessible locations, includes inhabited areas as well. Addressing the safety issues raised by the operation of multicopter drones, research is being conducted around the world to equip these aerial robots with AFTC systems able to steer even a damaged multicopter to safety.

One of the most researched failure types on multicopters are actuator failures, resulting in Loss of Effectiveness (LoE) in the damaged propulsion unit. Recent results show that apart from partial LoE of a multicopter's actuator [3] there exist control laws which ensure stable flight of multicopters with one [4], or even multiple propellers completely removed [5], meaning a complete LoE at those actuators. However, these control systems are switched, i.e., when a complete actuator failure occurs, the control system has to be restructured to accommodate the reduced model. Therefore, AFTC systems usually rely on FDD [6] or state reconstruction modules [7] which provide information on the effectiveness of the actuators.

There are also examples for designing AFTC systems with embedded FDD via adaptive backstepping, promising faster detection at the cost of a more complex system design [8]. Thus, a trade-off has to be made whether the AFTC system is designed as one complex system with unified stability analysis, or having a more flexible, modular design where the AFTC and the FDD modules are separately designed but their interactions are not straightforward. In this paper the latter design methodology is in focus, since the FDD module introduced in the paper is meant to augment an existing AFTC system.

An FDD module usually contains logic to detect the fault (compare a signal to a detection threshold), isolate a fault (identify the location of the fault in the system), identify the fault (unravel the nature/type of the fault), and evaluate the fault (estimate the fault size and severity) [9].

Multicopter FDD modules can be divided into two categories, according to the literature. The first category deals with low-level actuator signals such as rotor speeds and power [10], current signature [11], accelerometer vibrations [12–14] or even temperature [15]. Since most of these signals are related to the electrical dynamics of the motors, detecting faults using these signals could be much faster than using high-level signals from the multicopter dynamics. However, this also means that these signals are prone to more noise which may require computationally expensive Power Spectral Density (PSD) analysis to be performed online to detect faults [11, 13]. Furthermore, most of these approaches lack proof for convergence or stability analysis of the system in general.

The second category of FDD modules encountered on multicopters are focused on the high-level dynamics of the UAV and conveys information on the effectiveness of its actuators based on signals such as angular and translational velocities. Generally, these high-level FDD observers work with residuals, i.e., error between an

online observer model and the measured states of the actual system. In [16], a Multiple Model Adaptive Estimation (MMAE)-based approach is introduced where models of a multicopter are identified for various fault cases such as motor failures. The core idea of this approach is to run observers on each faulty model separately and, based on the residuals obtained for each model, a high-level logic decides which model matches the real system the most. Thus it is immediately seen which fault has occurred in the system if the fault models are well defined. However, an shortcoming of this approach is that it is hard to obtain these well defined models, especially for highly nonlinear systems where fault effects can overlap and mask each other [17]. Therefore, this approach seems to be more popular in fixed-wing UAV applications where linear models are more straightforward to obtain for cruising [18].

A more frequent observer type for multicopter FDD are nonlinear (adaptive) observers. Recently, [19] and [6] showed that a nonlinear Thau observer is suitable to detect and accommodate faults via estimating actuator effectiveness. These observers are proven to be asymptotically stable via Lyapunov analysis, contrary to other approaches. Hence, the effectiveness estimation is assumed to be suitable to detect partial and complete LoE as well, as the estimation will simply converge to a LoE estimate—usually between 0 and 1. However, since these observers are model-based, an observer model has to be constructed which makes the observer's performance highly dependent on model accuracy. There are however examples aiming to mitigate the effect of model uncertainty on Thau observers; such an observer is designed with sliding mode considerations in [20], aiming to counteract uncertainties entering the system. Sliding mode observers have the advantage over sliding mode control systems that even if the undesired effect of chattering occurs, it does so in the virtual observer model not the actuators [21], therefore they are quite popular for FDD design [22, 23]. The main drawback of using model-based observers for FDD is that the observer model has to be reidentified for different multicopter platforms, making it less flexible.

This paper focuses on actuator effectiveness estimation using a nonlinear Thau observer. Instead of utilizing the complete high-level dynamics of the multicopter, the observer is driven by fast, low-level current measurements from the motors. Thus, the detection time is aimed to be kept under 0.2 seconds.

Since the proposed observer is model based, the paper starts with constructing the observer model in Section II. It is shown how the individual thrust model of an actuator is included in the high-level attitude dynamics. Then, the proposed adaptive Thau observer synthesis is presented in Section III. Stability analysis of the observer has been included in Appendix A. The observer's FDD performance is demonstrated in Section IV and Section V based on simulation and Hardware-in-the-loop (HIL) estimation results. Finally, the paper ends with a conclusion in Section VI. As mentioned, some key notions, such as the observer's stability analysis and model parameter identification procedures, are included Appendix A.

II. System model description

A. Multicopter model

In this section, a model of a generic multicopter and its actuators is derived. Then, using these models, the online observer model is constructed. The observer,based on measurements from the actual system, estimates certain states of the actuators and the multicopter, thus the observer model has to be able to capture the dynamics of the system. At the end of the section, some important remarks are made on the nonlinear model used for FDD observer design.

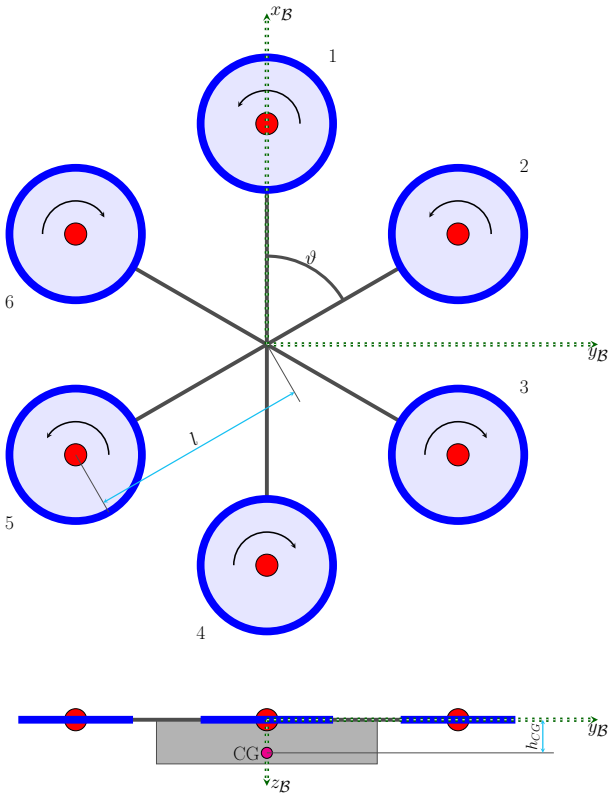

Fig. 1 Hexacopter model for simulation purposes. The specific layout of the actuator rotation directions ensures that controllability of the multicopter is preserved even when actuator 1 is completely damaged.

In order to design a model-based FDD module for a generic multicopter, the mathematical description of the mechanical system has to be constructed. As an example, Figure 1 shows a multicopter with six rotors. Two frames are used throughout the report to provide reference to the configuration (position and rotation) of the hexacopter. The first frame is the inertial frame $\mathcal{A}$ which is a global reference frame with its origin pinned in inertial space. Its axes are either in East-North-Up (ENU) or North-East-Down (NED) direction depending on the chosen convention. The second frame is body frame $\mathcal{B}$ whose origin is located in the Center-of-Gravity (CG) of the multicopter with axes fixed to its body, see Figure 1.

Equations of motion of a generic multicopter are shown in (1).

$$\begin{aligned} m\dot{v} &= mge_3 + Re_3\mathrm{T} \\ \dot{r} &= v \\ J_b\dot{\Omega} + \Omega \times J\Omega + d_\Omega\Omega &= \tau \\ \dot{R} &= R\hat{\Omega} \end{aligned} \tag{1}$$

The first two equations in (1) describe the translational dynamics of the vehicle as $v \in \mathbb{R}^3$ denotes the inertial velocity vector and $r \in \mathbb{R}^3$ is the position coordinate vector. Vehicle mass is denoted by m and g is the value of the gravitational acceleration coefficient. $e_3 = [0, 0, 1]^T$ is the unit axis in inertial Up/Down direction, depending on the choice of coordinate frame (ENU or NED). $\mathrm{T} \in \mathbb{R}$ is the total thrust generated in the body frame, always pointing in e_3 direction due to the system being underactuated. Rotational rate dynamics include the inertia of the rigid body J_b and its angular velocity Ω. The term d_Ω is a diagonal damping term which models counter torques due to drag effects [5]. $\tau \in \mathbb{R}^3$ denotes external torques acting on the system. In the last equation—which describes the multicopter attitude dynamics—R denotes a rotation matrix from body to inertial frame of reference. Furthermore, the $\hat{\bullet} : \mathbb{R}^n \to \mathfrak{so}(n)$ operator transforms a vector of size n to a skew-symmetric matrix of size $n \times n$ [24]. From (1), only the angular velocity dynamics is relevant for the FDD observer, however, the rest is also introduced as they are used in the simulations shown in Section IV.

B. Actuator model

Following the multicopter dynamics, an electric propulsion system model is modeled, ubiquitously found on multicopters. As shown in Figure 2 the actuator consists of an electric (Brushless DC (BLDC)) motor which rotates the propeller, based on electrical inputs from the Electronic Speed Control (ESC).

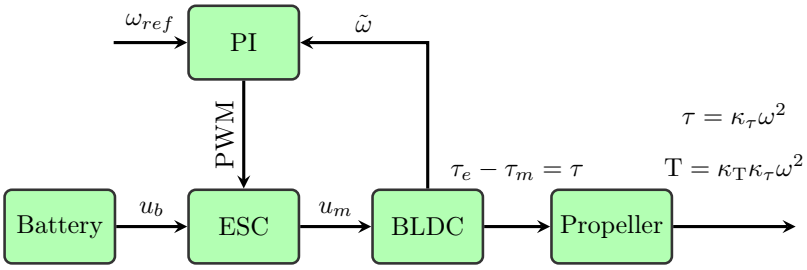

Fig. 2 Actuator model.

The input voltage (u_m) to the motor depends on the battery (source) voltage u_b and the Pulse Width Modulation (PWM) of the ESC unit. If the source voltage is AC, then the effective voltage is needed to be determined as shown in [25]. However, the battery voltage is DC, therefore the relationship between PWM

and average voltage at the motor is described as

$$u_m = u_b \cdot PWM \tag{2}$$

where u_b and PWM are measured.

Knowing the input voltage to the BLDC motor, the electrical description of the system is written as

$$u_m = Ri + L\frac{di}{dt} + \kappa_e \omega \tag{3}$$

where R denotes electrical resistance, i is the motor current, L is the motor inductance, κ_e is the counter-electromotive force of the motor, and ω denotes the speed of the motor. It is assumed that inductance is negligible, and the torque generated by the electromagnetic forces is written as the function of the armature current

$$\tau_e = \kappa_q i \tag{4}$$

where motor coefficient κ_q is assumed to be constant in the operational range of the motor and its value is provided by the manufacturer.

As for the mechanical description of the actuator model, assuming that the motor speed equals the propeller's speed, i.e., there is no gearing between these components, the actuator speed dynamics can be written as

$$J_p\dot{\omega} + \kappa_\tau \omega^2 + \kappa_f \omega = \kappa_q i \tag{5}$$

It is seen in (5) that the electrical torque generated τ_e by the motor is used to change its rotational speed, counteract blade drag and other internal damping factors ($\tau_m = \kappa_f \omega$) in the motor, shown in Figure 2 as well. It should be noted that J_p denotes the inertia of the propeller blade and the rotating part of the motor.

C. Control effectiveness matrix

After the angular velocity and actuator dynamics are constructed, the two can be merged via a control effectiveness matrix describing the thrust and torque generated by the actuators which drive the multicopter. First, the individual thrust model of an actuator is described. According to literature, for example [25], thrust and torque generated by an actuator in hover can be simply expressed as shown in (6)

$$\begin{aligned} \tau_p &= \kappa_\tau \omega^2 \\ \mathrm{T} &= \kappa_\mathrm{T} \omega^2 \end{aligned} \tag{6}$$

where the propeller-generated torque τ_p emanates from the aerodynamic drag on the blades and it acts in the body $z_{\mathcal{B}}$ direction. In (6), κ_τ and κ_{T} denote the actuator torque and thrust coefficients respectively.

In this paper, however, (6) is extended in a similar way as in [7] by adding an effectiveness parameter η to the actuation model such that

$$\tau_p = \kappa_\tau \eta \omega^2$$

$$\mathrm{T} = \kappa_{\mathrm{T}} \eta \omega^2$$

This effectiveness parameter is assumed to yield information on a fault occurrence, or even aerodynamic effects. For example, in the ideal case during hover, η should be 1. However, when the propeller blade is damaged its effectiveness parameter should decrease, with $\eta = 0$ indicating that the propeller is completely gone. It should be noted, that η may have a value greater than 1, implying that there are aerodynamic effects that increase the effectiveness of the propeller [26], or if the initial estimate of κ_τ is less than its actual value.

Thereafter the following definition is used to further specify the model at hand:

Definition II.1 (Wrench [27])
A wrench $\mathcal{W} \in \mathbb{R}^6$ is defined via appending torque and force vectors, expressed in the inertial frame $\mathcal{A}$, into a six-dimensional vector:

$$\mathcal{W} = [\mathfrak{m}_{\mathcal{A}}, \mathfrak{f}_{\mathcal{A}}]^T$$

Since—in this case—the actuators are assumed to exert thrust only in the body $z_{\mathcal{B}}$ direction, a more convenient, body-frame-fixed underactuation wrench $\mathcal{W}_u$ is defined as shown in (7), where the geometry of the multicopter was based on Figure 1.

$$\mathcal{W}_u = \begin{bmatrix} \tau \\ \mathrm{T} \end{bmatrix} = \underbrace{\begin{bmatrix} -\kappa_{\mathrm{T}} l \sin(\vartheta_1) & \cdots & -\kappa_{\mathrm{T}} l \sin(\vartheta_6) \\ \kappa_{\mathrm{T}} l \cos(\vartheta_1) & \cdots & \kappa_{\mathrm{T}} l \cos(\vartheta_6) \\ \kappa_\tau & \kappa_\tau \quad -\kappa_\tau \quad -\kappa_\tau & \cdots \\ \kappa_{\mathrm{T}} & \kappa_{\mathrm{T}} \quad \kappa_{\mathrm{T}} \quad \kappa_{\mathrm{T}} & \cdots \end{bmatrix}}_{G} \mathrm{diag}(\omega_1^2 \dots \omega_6^2) \begin{bmatrix} \eta_1 \\ \vdots \\ \eta_6 \end{bmatrix} \tag{7}$$

In (7) l denotes the rotor displacement from the Center of Gravity (CG) in the body $x_{\mathcal{B}} - y_{\mathcal{B}}$ plane (in this case same for all rotors) and ϑ the angle of the rotor's arm from body $x_{\mathcal{B}}$ axis. Matrix G in (7) is the control effectiveness matrix of the multicopter with an important role in control allocation as shown in Section IV.

The specific layout of rotation directions of the propellers is important because throughout testing, when a full fault occurs in actuator 1, the remaining system ensures that controllability of the system is maintained,

meaning that there is no need to give up yaw control (contrary to other actuator alignments or quadcopters). Thus, a single Incremental Nonlinear Dynamic Inversion (INDI) or Nonlinear Dynamic Inversion (NDI)-based controller is still able to steer the multicopter without the need to design a switched control law, such as primary-axis control [4], as the rank of the reduced control effectiveness matrix does not decrease. Therefore, the focus could be fixed solely on testing the capabilities of the FDD observer.

D. Observer model construction

Based on the models described above, the high level attitude dynamics of the multicopter can be merged with n number of actuator models to obtain a suitable online model for the model based FDD model.

$$\begin{aligned} &J_b\dot{\Omega} + \sum_k^n J_p\dot{\omega}_k + d_\Omega\Omega + \Omega \times (J_b\Omega + \sum_k^n J_p\omega_k) = G_\tau \mathrm{diag}\left(\omega^2\right)\eta \\ &J_p\dot{\omega} + \kappa_\tau\eta\omega^2 + \kappa_f\omega = \kappa_q i \end{aligned} \tag{8}$$

Note, that in (8) some additional terms have been included due to the gyroscopic effects of the rotating blades. Even though the effect of $\sum_k^n J_p\dot{\omega}_k$ is negligible [5], the term $\sum_k^n J_p\omega_k$ can have significant impact on the system [28], especially when propellers are damaged or even lost. G_τ denotes the sub-matrix of G in (7) created from the first three rows.

Finally, the nonlinear description of the whole system can be shown to be written in the form (9)

$$\begin{aligned} \dot{x} &= Ax + \Phi(x,u) + D\phi(x,u)\eta + Bu \\ y &= Cx \end{aligned} \tag{9}$$

where x denotes the state vector, y is the output vector, u is the input vector, and η is the parameter vector to be estimated. All these vectors are functions of time t, however, in order to simplify notation, this dependency is not written explicitly. The form of (9) provides the basis of an adaptive observer designed for estimating the parameter η [29]. Thus, the FDD observer will adaptively estimate LoE values of the actuators in order to detect faults and update the control system.

Supposing that the multicopter has n actuators, neglecting the effect of $\sum_k^n J_p\dot{\omega}_k$ in (8) and defining the state as $x = [\omega, \Omega]^T \in \mathbb{R}^{n+3}$, input as motor currents ($u = i$), and inertia matrix J as a block diagonal matrix

$$J = \mathrm{diag}(\mathrm{diag}(J_{p_1} \dots J_{p_n}), J_b) \in \mathbb{R}^{(n+3)\times(n+3)}$$

it is obtained that system matrices of (9) can be written as

$$
\begin{aligned}
A &= J^{-1} \begin{bmatrix} \operatorname{diag}(-\kappa_f) & 0_{n\times 3} \\ 0_{3\times n} & -d_\Omega \end{bmatrix} \in \mathbb{R}^{(n+3)\times(n+3)} \\
B &= J^{-1} \begin{bmatrix} \operatorname{diag}(\kappa_q) \\ 0_{3\times n} \end{bmatrix} \in \mathbb{R}^{(n+3)\times n} \\
C &= \mathcal{I} \in \mathbb{R}^{(n+3)\times(n+3)} \\
D &= J^{-1} \begin{bmatrix} \operatorname{diag}(-\kappa_\tau) \\ G_\tau \end{bmatrix} \in \mathbb{R}^{(n+3)\times n} \\
\phi(x) &= \operatorname{diag}(\omega^2) \in \mathbb{R}^{n\times n} \\
\Phi(x) &= \begin{bmatrix} 0_{n\times 1} \\ J^{-1}\left(\Omega \times \left(J_b\Omega + \sum_k^n J_p\omega_k\right)\right) \end{bmatrix} \in \mathbb{R}^{n+3}
\end{aligned}
\tag{10}
$$

E. Observability and Lipschitz continuity of the model

After the model has been constructed, some remarks are made on its observability and Lipschitz continuity which are requirements for observer synthesis—as discussed in Section III.

Remark (Model observability)

Since observability of the model is an important criterion when designing an observer, from (10) *it can be conveyed that the pair* (A, C) *is observable (according to the Kalman observability test, the respective observability matrix has full rank). Note that even if the state vector is reduced to* $x = \omega$*, meaning that decentralized observers might be designed for each individual actuator.*

As a result, a trade-off in the model choice emerges. If the high-level attitude dynamics are included in the model, i.e. Ω is part of the state, the slower dynamics of the multicopter angular velocity may dampen the faster dynamics of the propellers. This way it would make the observer more robust towards low level uncertainties and false alarms. On the other hand, including Ω in the state might increase fault detection time and requires accurate measurements of Ω which assumes accurate state estimation based on the Inertial Measurement Unit (IMU) signals.

Remark (Lipschitz continuity)

It should be noted that inherently none of the nonlinearities $\phi(x)$*,* $\Phi(x)$ *are globally Lipschitz continuous over*

the set of real numbers $\mathbb{R}$, since $\Phi(x)$ contains cross products and $\phi(x)$ is a quadratic polynomial function. The nonlinearities can be shown, however, to be locally Lipschitz continuous functions. In other words, there exists a bounded neighborhood $\mathcal{H}_{x_0}$ around all $x_0 \in \mathbb{R}$ such that $\phi(x)$ and $\Phi(x)$ are globally Lipschitz continuous for all $x \in \mathcal{H}_{x_0}$ [30].

It is shown in [31] that the angular velocity dynamics in (10) is Lipschitz continuous in a ball around the origin. Due to damping terms present in the system dynamics in (8), angular velocities and rotor speeds of the physical system can be shown to be constrained to a bounded state-space, yielding Lipschitz continuity for $\Phi(x)$ with Lipschitz bound $\gamma_\Phi < \infty$.

As for $\phi(x) = x^2$, using the property $|ab| = \sqrt{(ab)^2} = \sqrt{a^2 b^2} = \sqrt{a^2}\sqrt{b^2} = |a||b|$, it can be shown—for the one-dimensional case—that

$$\begin{aligned} |\phi(x_2) - \phi(x_1)| = |x_2^2 - x_1^2| = |(x_2 - x_1)(x_2 + x_1)| = |x_2 - x_1||x_2 + x_1| \\ \leq \gamma_\phi |x_2 - x_1| \qquad \forall\, x_1, x_2 \in \mathcal{X} \subset \mathbb{R} \end{aligned} \tag{11}$$

meaning that $\phi(x)$ is Lipschitz continuous in $\mathcal{X} \subset \mathbb{R}$ with Lipschitz bound $\gamma_\phi < \infty$. In the two-dimensional case, i.e., when $x_1, x_2 \in \mathcal{X} \subset \mathbb{R}^2$, it can be written that

$$\begin{aligned} x_1 = [a_1, a_2]^T \;\rightarrow\; \phi(x_1) = \phi(a_1, a_2) = [a_1^2, a_2^2]^T = x_1^2 \\ x_2 = [b_1, b_2]^T \;\rightarrow\; \phi(x_2) = \phi(b_1, b_2) = [b_1^2, b_2^2]^T = x_2^2 \end{aligned}$$

Thus, based on (11) and the triangle inequality property, it is seen that

$$\begin{aligned} \|\phi(x_2) - \phi(x_1)\| &= \|\phi(b_1, b_2) - \phi(a_1, a_2)\| = \|\phi(b_1, b_2) - \phi(a_1, a_2) + \phi(a_1, b_2) - \phi(a_1, b_2)\| \\ &= \|\phi(b_1, b_2) - \phi(a_1, b_2) + \phi(a_1, b_2) - \phi(a_1, a_2)\| \\ &\leq \|\phi(b_1, b_2) - \phi(a_1, b_2)\| + \|\phi(a_1, b_2) - \phi(a_1, a_2)\| \\ &\leq \gamma_\phi \|(b_1, b_2) - (a_1, b_2)\| + \gamma_\phi \|(a_1, b_2) - (a_1, a_2)\| \\ &= \gamma_\phi \|(b_1, b_2) - (a_1, b_2) + (a_1, b_2) - (a_1, a_2)\| = \gamma_\phi \|(b_1, b_2) - (a_1, a_2)\| = \gamma_\phi \|x_2 - x_1\| \end{aligned} \tag{12}$$

In (12), the Lipschitz bounds for $b_1 - a_1$ and $b_2 - a_2$ are equal due to the fact that all coordinates of x_1 and x_2 denote the speed of an actuator with the same dynamics.

Finally, based on equations (11) and (12), it can be shown by induction that the nonlinear term $\phi(x)$ is Lipschitz continuous in $\mathcal{X} \subset \mathbb{R}^n$, where n denotes the number of actuators of the multicopter. Thus, assuming that the rotor speed of each actuator is confined to the bounded state-space—due to the actuator damping term present in (8)—it is asserted that $\phi(x)$ is Lipschitz continuous.

III. Adaptive observer for LoE estimation

In this section, the synthesis of a model-based adaptive observer is introduced. Considering some initial assumptions in Assumption III.1 the observer is designed for a certain class of nonlinear systems described in (9). Proof for the observer's parameter estimation convergence (an important incentive for observers used for FDD) is given, along with the discussion of the assumptions made, see Appendix A. The observer synthesis is performed via solving a Linear Matrix Inequality (LMI)—a convex optimization problem—as shown at the end of this section. In the following, n will denote the number of states, m the number of measured states, $\succ 0$ and $\prec 0$ will stand for positive and negative definite matrices respectively, whereas $\succeq$ and $\preceq$ indicate semidefinite matrices.

A. Adaptive observer description

Assumption III.1 (Existence of an adaptive observer)

In order to design a feasible adaptive observer estimating parameter η, the following assumptions must hold for the system at hand [29]:

1) The nonlinearities of the system dynamics are Lipschitz continuous with respective bounds γ_Φ and γ_ϕ:

$$\begin{aligned} \|\Phi(x_1,u_1)-\Phi(x_2,u_2)\| &\leq \gamma_\Phi \|(x_1,u_1)-(x_2,u_2)\| \\ \|\phi(x_1,u_1)-\phi(x_2,u_2)\| &\leq \gamma_\phi \|(x_1,u_1)-(x_2,u_2)\| \end{aligned} \tag{13}$$

2) The norm of the estimated parameter η is bounded, i.e., $\|\eta\|_2 \leq \gamma_\eta$ and it is slowly varying, i.e. $\dot{\eta} \approx 0$ (such as sensor bias)

3) The linear part of the system—the pair (A, C)—in (9) is observable

4) There exists $P = P^T \succ 0 \in \mathbb{R}^{n\times n}$ such that all rows of $D^T P$ are in the row-space of C ($\boldsymbol{Im}\{C^T\}$)

5) There exists an observer gain matrix $K \in \mathbb{R}^{n\times m}$ such that

$$\underbrace{\gamma_\Phi + \gamma_\eta \gamma_\phi \|D\|}_{\alpha} < \frac{\lambda_{min}(Q)}{2\lambda_{max}(P)} \tag{14}$$

where

$$-Q = (A-KC)^T P + P(A-KC) \prec 0 \text{ with } Q = Q^T \in \mathbb{R}^{n\times n}$$

and $\lambda()$ denotes the eigenvalues of its argument matrix. Furthermore $\|D\| = \sqrt{(\lambda(D^T D))}$ denotes the matrix norm of the real valued matrix D. This prerequisite is referred to as the Thau-condition [32].

If all elements of Assumption III.1 hold, the adaptive observer is feasible to design for the system (9) in

the form

$$\begin{aligned}\dot{\hat{x}} &= A\hat{x} + \Phi(\hat{x}, u) + D\phi(\hat{x}, u)\hat{\eta} + Bu + K(y - \hat{y}) \\ \hat{y} &= C\hat{x} \\ \dot{\hat{\eta}} &= \frac{1}{\Gamma}\phi^T(x, u)D^T PC^+(y - \hat{y})\end{aligned} \tag{15}$$

where $\hat{x}$ is the estimated observer state, $\hat{y}$ is the output estimate, $\hat{\eta}$ denotes the estimated effectiveness of the actuators, and C^+ is the pseudoinverse of C. In (15) the parameter estimation update law is designed to ensure that the dynamics of the state estimation error $e_x = x - \hat{x}$ and the parameter estimation error $e_\eta = \eta - \hat{\eta}$ are asymptotically stable [33].

It was shown in Section II that the nonlinearities in the model dynamics are Lipschitz continuous on a bounded state space and that the pair (A, C) in (9) is observable. Furthermore, when a propeller is detached or damaged, its effectiveness changes as a step function, therefore, it is safe to assume $\dot{\eta} = 0$ because the change is instantaneous and the estimation problem remains the same—only with a different initial condition. The remaining assumptions are discussed in Appendix A.A, along with the stability analysis of the proposed observer.

B. Adaptive observer synthesis

Stability analysis of the observer shows that matrix

$$M = (A - KC)^T P + P(A - KC) + (\gamma_\Phi + \gamma_\phi\gamma_\eta\|D\|)(PP + \mathcal{I}) = -Q + \alpha(PP + \mathcal{I})$$

renders the system Lyapunov stable if it is negative definite. The interested reader is referred to Appendix A.A for the detailed discussion of the observer stability analysis. It should be noted, that Lipschitz bounds might change whenever a new multicopter/actuator model is considered, but the observer synthesis methodology remains the same.

In order to synthesize observer gain matrices K and P to generate a negative definite M matrix, a system of Linear Matrix Inequalities (LMIs) has to be constructed in the following way

$$\begin{aligned} &P \succ 0 \\ &\begin{bmatrix} (A - KC)^T P + P(A - KC) + \alpha\mathcal{I} & \sqrt{\alpha}P \\ \sqrt{\alpha}P & -\mathcal{I} \end{bmatrix} \prec 0 \end{aligned} \tag{16}$$

In (16) the second inequality is constructed based on the Schur complement of M which can be shown to be negative definite if M is negative definite, and vice versa [33, 34]. Solving (16), however, is not yet feasible,

as the expression is in fact nonlinear in K and P; therefore, by introducing $L = PK$, it is obtained that

$$\begin{bmatrix} A^T P - C^T L^T + PA - LC + \alpha\mathcal{I} & \sqrt{\alpha}P \\ \sqrt{\alpha}P & -\mathcal{I} \end{bmatrix} \prec 0 \qquad P \succ 0 \tag{17}$$

is indeed an LMI in the unknown L and P. Thus a convex optimization procedure yields L and P which allows to compute the observer gain matrix $K = P^{-1}L$ provided that P is invertible. Finally, if Assumption III.1 holds with the obtained gain matrices the observer is successfully synthesized.

Tuning of the observer is done via adding tolerance matrices $\epsilon\mathcal{I} \succ 0$ to the left hand side of the inequalities in (17) [33]. This way, the state estimation error dynamics become faster and more aggressive, eventually rendering the system unstable. Furthermore, gain Γ in (15) affects primarily the parameter estimation dynamics; response of the system is demonstrated with different values of Γ in Appendix A.D.

IV. Simulation results

In this section, simulation results are introduced, discussing the performance of the FDD observers and the shortcomings of effectiveness estimation.

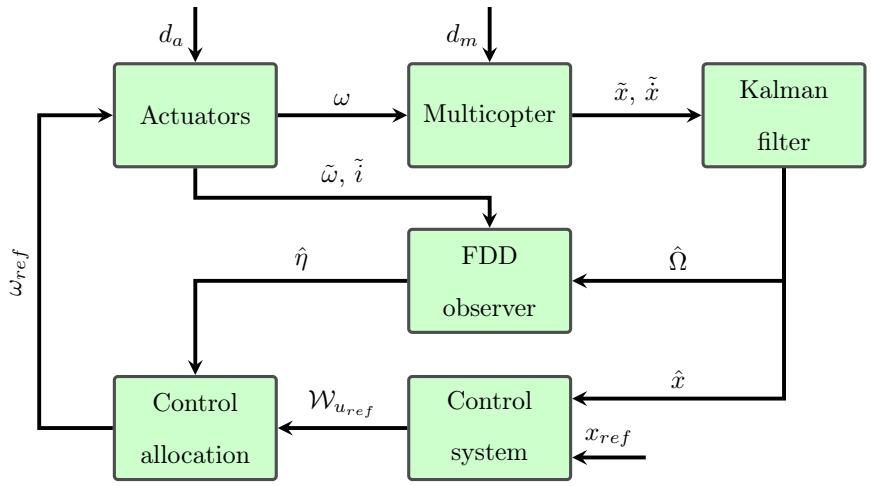

Fig. 3 Layout of the multicopter's subsystems.

The system architecture is shown in Figure 3, with all subsystems and their interactions. The real world multicopter receives the motor speed of individual actuators as input, and outputs its own states (with noise d_m) which are measured by an IMU ($\tilde{x}$, $\tilde{\dot{x}}$). In order to reconstruct the bias and noise free states, the noisy measurements enter a Kalman filter. The reconstructed states $\hat{x}$ are then used to provide feedback for the

high-level multicopter controller (in this case an INDI controller with nonlinear PD control [24]) and the FDD observer. The high-level controller provides a reference underactuation wrench $\mathcal{W}_{u_{ref}}$ for the control allocator which, based on the actuator effectiveness estimate $\hat{\eta}$, yields the reference motor speeds ω_{ref} for the individual motor speed controllers, using the pseudoinverse of G in (7). Finally, to close the loop, the reference rotor speeds enter a P(I) motor speed controller, which provides reference PWM for the ESC and the motors, based on motor speed feedback $\tilde{\omega}$. Speed $\tilde{\omega}$ and current $\tilde{i}$ of each actuator is measured (with noise d_a) which are then sent to the FDD observer. In simulation, it is assumed that $d_a = d_m = 0$ and the Kalman filter is omitted, thus the actual states of the multicopter and the actuators are used.

The multicopter is initialized with its rotors stopped and its attitude (roll, pitch, yaw angle) arbitrary between ± 1 $[rad]$, as seen in Figure 5. The multicopter is then instructed to recover from this initial pose and maintain hover. Then, after 4 seconds, full fault is introduced to one actuator (denoted by number 1 in Figure 1), and partial fault to the actuator next to it (number 2). Since the layout of the multicopter is as shown in Figure 1, the system maintains controllability. Thereafter, the multicopter is instructed to fly to a target 5 meters from its original position, which is successfully performed despite the damaged actuators, as seen in Figure 4. Figure 6 is intended to show that control allocation (simply the pseudoinverse updated with the effectiveness estimates), even with a lost propeller, is performing adequately with this specific hexacopter layout, as all actuation signals converge to the commanded ones.

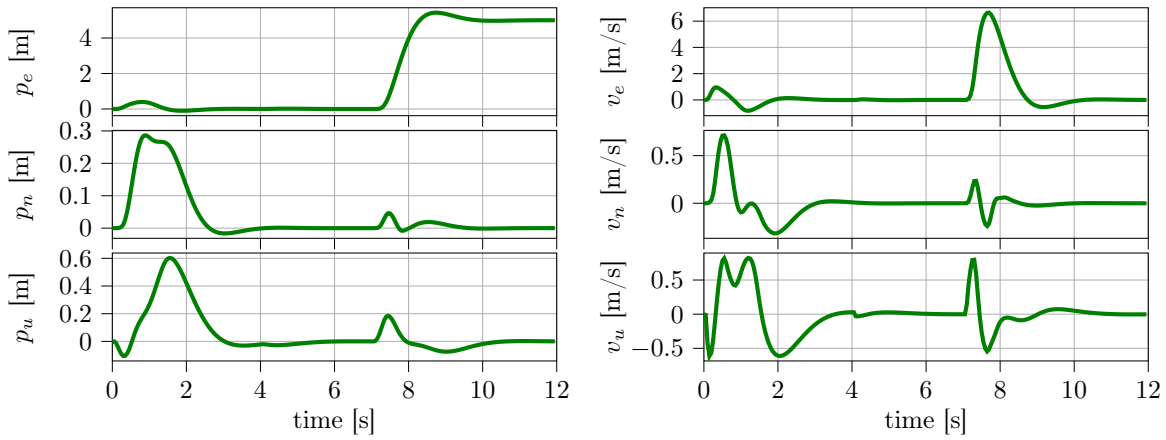

Fig. 4 Inertial position (p_e, p_n, p_u) and velocity (v_e, v_n, v_u) of the multicopter after starting from an arbitrary pose within ± 1 $[rad]$. After the control system successfully recovers the multicopter to hover faults are introduced in two actuators (at 4 seconds). Thereafter, the control system maintains hover even with the damaged actuators. Finally, the multicopter is instructed to fly to a target 5 meters away in the EAST direction which it performs successfully, despite the damaged actuators.

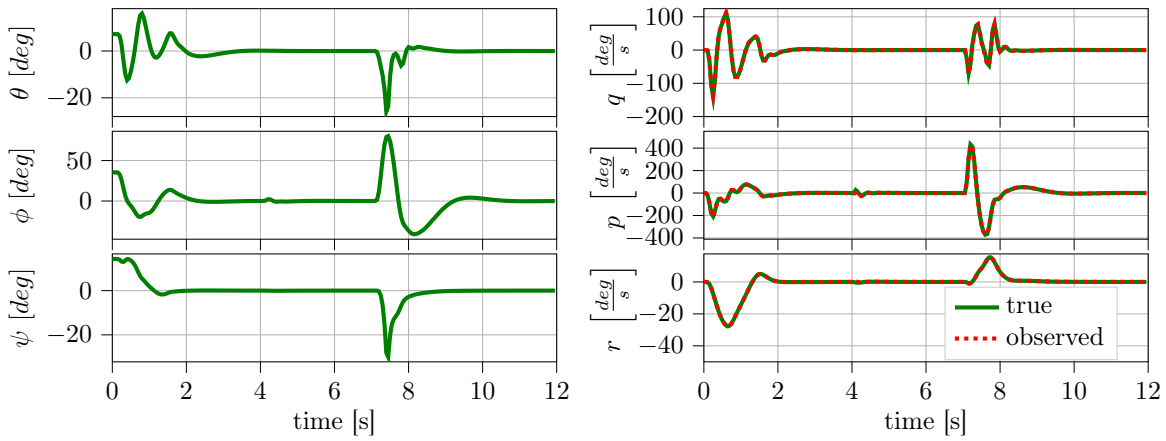

Fig. 5 **Pitch (θ), roll (ϕ), yaw (ψ) and respective angular velocities (q, p, r) of the multicopter. At 4 seconds, when fault is introduced, there is a barely distinguishable error in attitude due to the fast control system. In real flight conditions this error is anticipated to be more significant. It is also seen that due to the damage to the actuators, there is a large error in pitch and yaw when the multicopter flies to a new target, even though the it would only require rolling motion in the ideal case. As for the angular velocities, it is shown that the FDD observer tracks the true rates accurately.**

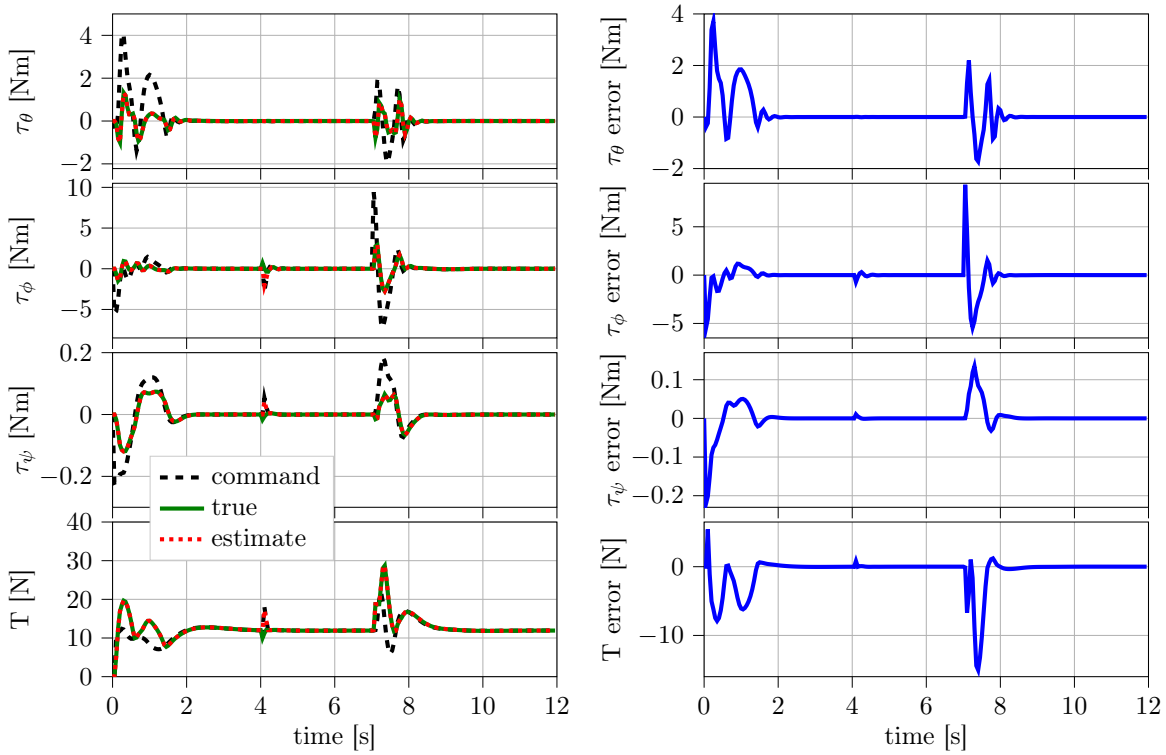

Fig. 6 **Reference, estimated and true underactuation wrench of the multicopter. This figure is intended to show that control allocation is properly maintained even after a full and a partial fault occurred in two actuators. It is seen that the error between the exerted (true) and commanded thrust (T) and torques (τ_θ, τ_ϕ, τ_ψ) converge to zero.**

$$w_{ref} = \sqrt{\text{diag}(\hat{\eta})^{-1} G^{+} \mathcal{W}_{u_{ref}}} \tag{18}$$

Control allocation is performed via (18), where $\hat{\eta} = 0$ imposes singularity when a full fault occurs. In order to account for this, a threshold is set in the control allocation module which deletes the respective column in G if a propeller gets completely damaged. Then, pseudoinverse G^{+} is recomputed from the reduced control effectiveness matrix. Although INDI is inherently able to deal with uncertainties in the model [28], the model changes completely, hence the need for recomputing the pseudoinverse.

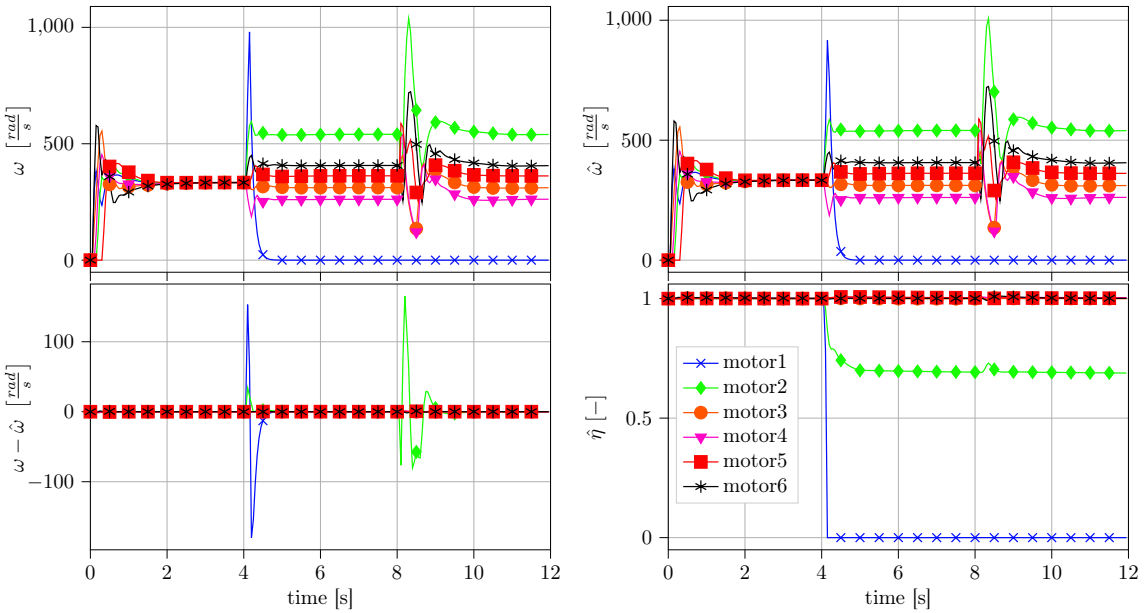

Fig. 7 Rotor speeds and effectiveness estimate on the hexacopter. Initially, the rotor speeds are allocated to bring the multicopter to hover. Thereafter, at 4 seconds, large estimation error ($\omega - \hat{\omega}$) occurs between the true and the observer rotor speed of the completely damaged actuator. The effectiveness estimate of the respective actuator goes to 0 almost immediately, triggering a switch-off signal which leads to 0 rotor speed. The partially damaged propeller does not exhibit such abrupt change in effectiveness estimation as the rotor speed error is less. However the effectiveness estimate still converges to a decreased value. Furthermore, note the increase in the rotor speed of the actuator, required to output the same thrust.

Finally, the most important signals for FDD are shown in Figure 7. It is seen that the completely damaged actuator spools up for a split second as it loses a significant amount of inertia due to the detached blade. This, along with the increasing reference speed, induces a large error in the observer's estimated rotor speed and the true rotor speed, which steers the effectiveness estimate towards zero in less than 0.2 seconds. As soon as the effectiveness estimate reaches a threshold specified at 0, the respective motor is instructed to switch off and its speed gradually reaches 0. At the same time, the observer detects the effectiveness decrease of the other damaged blade as well, converging to a value around 0.7, proving that the observer is indeed

Lyapunov stable and is able to handle partial and complete faults as well. Note the increased rotor speed of the partially damaged blade required to generate the same thrust as a healthy propeller.

However, there is an important shortcoming of the observer, which is not seen in the above figures. Namely, when a blade gets damaged, its inertia decreases as well, but the observer in itself does not account for these inherently. When the inertia of the real system decreases, its dynamics become faster while the observer model remains slow. Therefore, if the observer model matrices in (10) are not updated, there are large oscillations in the effectiveness estimate due to the motor speed estimation error, see Figure 8. These oscillations however only occur when the system leaves hover, i.e., when the drone is instructed to fly to a new target at 8 seconds, and the rotor speeds change. In Figure 7, the oscillations are not present because the inertia in the observer model is updated as soon as the effectiveness of the broken propeller converges. Nevertheless, it should be noted that regardless of this update, the effectiveness still converged to 0.7 even after the oscillations occur, see Figure 8.

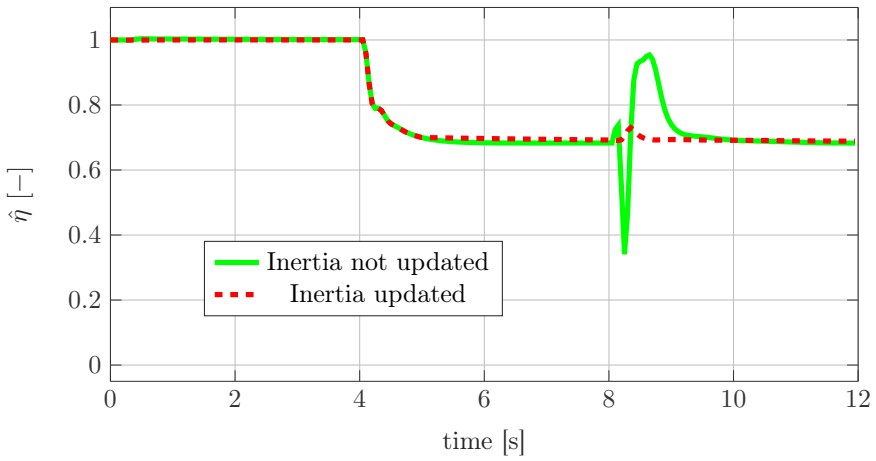

Fig. 8 Comparison of effectiveness estimation convergence in case of a partially damaged propeller. Since the true inertia of the broken blade decreases, the observer model exhibits slower dynamics if its inertia is not updated accordingly. This leads to transient errors when the multicopter leaves hover. Nevertheless, the effectiveness estimation still converges, however, not accounting for this phenomenon might lead to false alarms in real flight conditions.

V. Hardware-in-the-loop simulation

In this next section, the performance of the FDD observer is demonstrated via HIL simulation results. As mentioned in Section II, the system model is observable even if the high-level attitude dynamics is omitted from the model. Thus, a testbench setup was built consisting of a single actuator, as shown in Figure 10. The underlying architecture of the HIL simulation is shown in Figure 9. A computer is connected to the test setup via a serial port. The computer runs a script that sends a reference rotor speed ω_{ref} to a virtual PI speed controller. This speed controller sends the appropriate PWM to the ESC on the testbench setup.

The ESC, based on this PWM signal, channels power (u_s) from the power source to the motor whose speed ($\tilde{\omega}$) is measured via an optical sensor. The motor current ($\tilde{i}$) is also measured and this signal drives the FDD observer. Finally, the observer yields an estimated effectiveness $\hat{\eta}$, based on motor speed and current measurements.

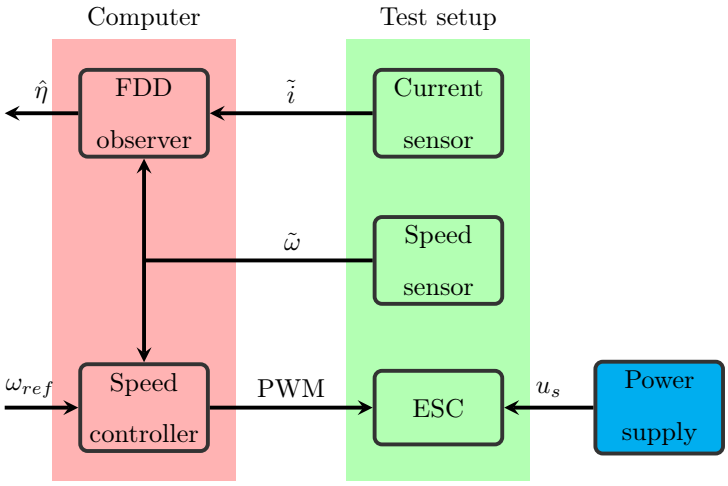

Fig. 9 Layout of the HIL simulation setup. The observer and motor speed controller run virtually on the computer which sends PWM inputs to the ESC. The FDD observer yields an effectiveness estimate ($\hat{\eta}$) based on speed ($\tilde{\omega}$) and current ($\tilde{i}$) measurements.

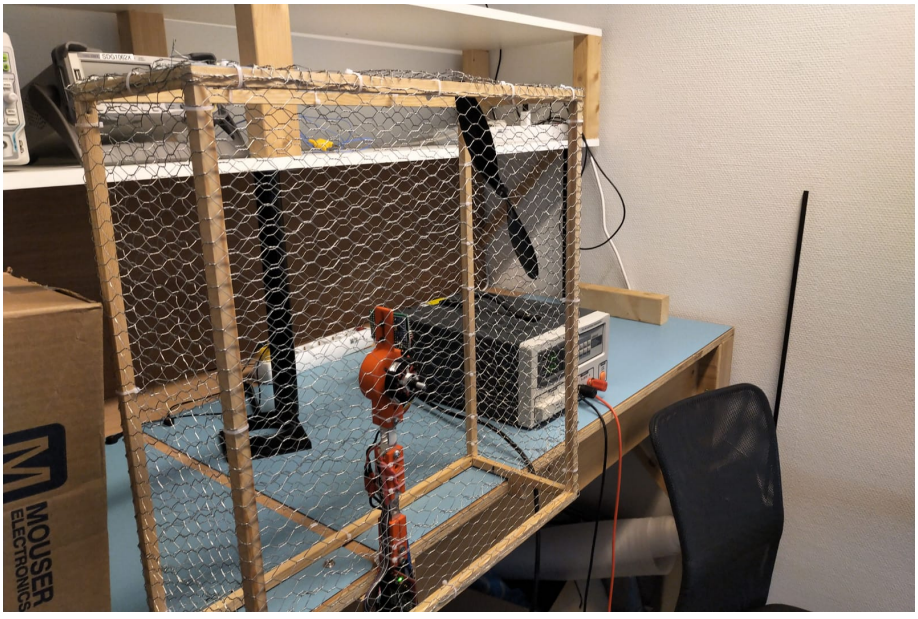

Fig. 10 Testbench setup. Note the detached propeller tangled in the safety net.

Since the actuator model is linear only locally (5), the motor speed controller has a feedforward mapping of w_{ref} to PWM. This mapping was identified as shown in Figure 11; a "staircase" PWM input was commanded

and the rotor speeds at each steady state were registered. Then, a quadratic curve was fitted on the obtained data using the Ordinary Least Squares (OLS) method. A proportional gain was chosen to increase the bandwidth of the motor speed controller, and an integrator term was added to account for minor uncertainties.

Furthermore, it was found that the relationship described in (6) does not hold for the actual system, as the value of κ_τ seemed to change considerably with speed. Therefore, κ_τ is scheduled with respect to the rotor speeds using a function identified offline. The detailed identification process is described in Appendix A.C.

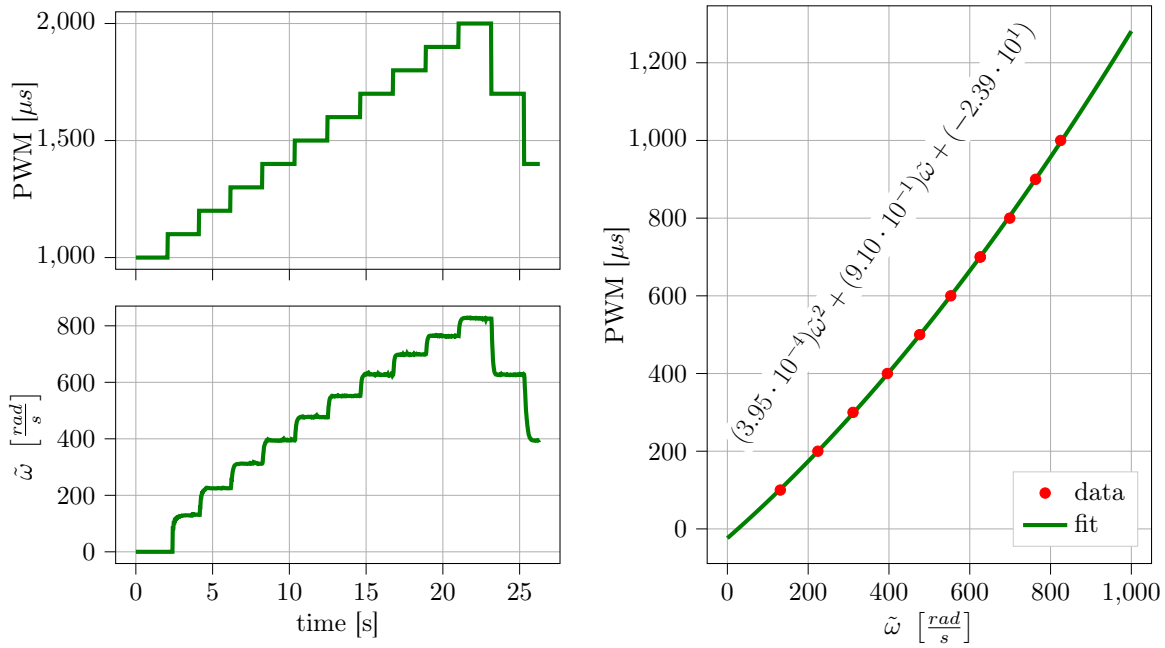

Fig. 11 Feedforward mapping identification procedure. Since it is assumed that the measured motor speed ($\tilde{\omega}$) matches the true motor speed (ω), a feedforward map is established between $\tilde{\omega}$ and input PWM. Thus, in order to make ω track ω_{ref}, the reference speed should be input into the identified expression to get the corresponding PWM input.

After the necessary model parameters were obtained as explained above, the HIL simulation was performed such that the rotor speed was gradually increased to almost full speed ($\approx 800\left[\frac{rad}{s}\right]$). Then, the motor was slowed to around hover speed ($\approx 500\left[\frac{rad}{s}\right]$) and approximately after 49 seconds a sudden deceleration command was introduced to the system. This command was required for detaching the propeller from the motor. It is seen in Figure 12 that as soon as the propeller is detached, the observer's estimate for effectiveness decreases drastically. Furthermore the effectiveness estimate converges to ~ 0, and the detection time, i.e., when the estimate crosses a threshold value (for example 0.1) is below 0.2 seconds, as shown in Figure 13. Thus, the high level AFTC control system has time to switch to a reduced control law which is able to maintain stable flight of the multicopter before the multicopter reaches an unrecoverable pose.

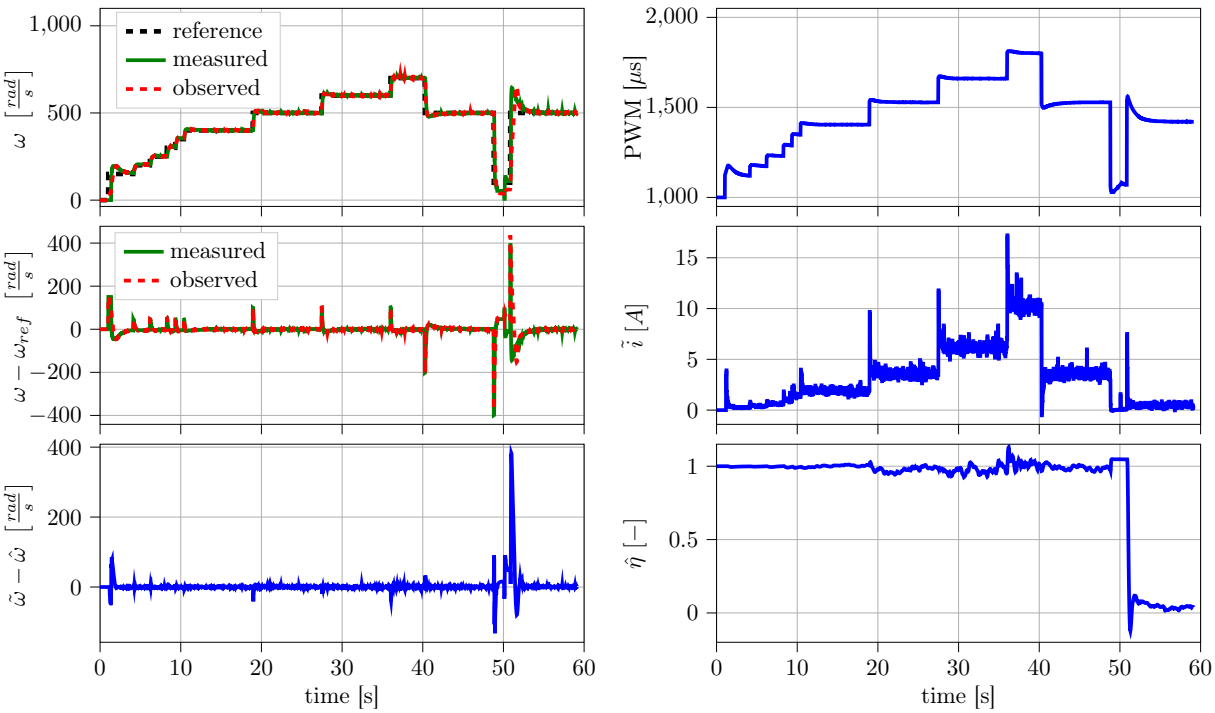

Fig. 12 Fault detection in HIL simulation. At ≈51 seconds, the propeller is detached from the actuator due to a sudden deceleration command. The adaptive FDD observer captures the loss of the propeller as a complete loss of effectiveness.

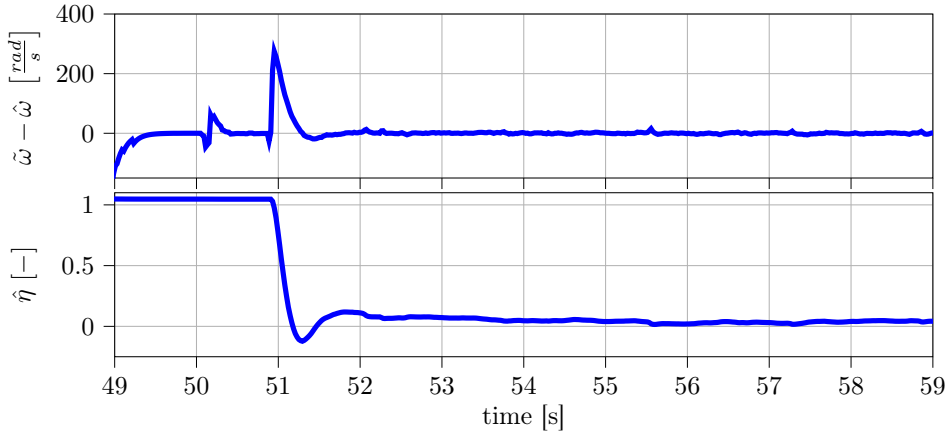

Fig. 13 Effectiveness estimation when the propeller gets detached from the actuator. Setting a detection threshold to 0.1, the detached blade is detected within 0.2 seconds. The error between measured ($\tilde{\omega}$) and estimated ($\hat{\omega}$) rotor speeds define the dynamics of the effectiveness estimation.

In this section it was shown that the fault detection time of the observer is sufficiently fast, although with a higher sampling rate, it is assumed that detection time could be decreased. On the HIL setup, the sampling rate was 50 Hz, however, in the upcoming tests, a 500 Hz current sensor will be utilized on an

actual multicopter, where AFTC control system will be present as well. Further fine-tuning of the observer may also decrease detection time. The effect of certain parameters on the effectiveness estimation dynamics are discussed in Appendix A.D.

VI. Conclusions

This paper introduced the design of a model-based FDD observer for a generic multicopter UAV equipped with electric propulsion units. The observer is applied on a nonlinear model consisting of high-level angular velocity dynamics of the multicopter and low-level dynamics of individual actuators. This modular approach enables the application of the observer on individual actuators. The observer adaptively estimates LoE parameters of the actuators which are good indicators of faults. The proposed system, apart from other similar applications in literature, is able to estimate not just partial, but complete LoE of the actuators as well. Estimated LoE of each actuator is used to update a high-level AFTC system which implements a switched, reduced control law to steer the damaged drone to safety. The observer is synthesized via LMIs and has profound stability analysis proving asymptotic stability of the state and parameter estimation.

The observer's performance was shown via simulations and HIL tests. Both reassured convergence of the observer along with its ability to detect LoE changes of partially or even completely damaged blades. The detection time on the testbench setup was found to be slightly below 0.2 seconds which was an initial design goal for the observer. Thus, the AFTC system can be updated before the multicopter reaches an unrecoverable pose.

A shortcoming of the presented observer is that it requires accurate model parameters to be identified making it less flexible as these parameters have to be re-estimated for every multicopter/actuator configuration. Nevertheless, the parameter estimation process, as shown in Appendix A is straightforward and not time consuming to perform. Another drawback of the proposed system is that a current sensor is required to drive the observer which might be infeasible, or at least cumbersome, to implement on an existing commercial drone.

Future work includes testing the observer on a real system. Provided that an AFTC system is available on the multicopter, the aim is to detect propeller failures quickly even in non-hover conditions and update the control system.

All in all, the main contribution of the paper is that by merging high-level and low-level dynamics of a multicopter system, there is a quick and relatively easy way to include an adaptive LoE estimator on electrically propelled UAVs for FDD purposes.

References

[1] Sachs, G., "Drones reporting for work," , 2017.

[2] Culus, J., Schellekens, Y., and Smeets, Y., "A drone's eye view," , 2018.

[3] Ortiz-Torres, G., Castillo, P., and Reyes-Reyes, J., "An Actuator Fault Tolerant Control for VTOL vehicles using Fault Estimation Observers: Practical validation *," *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, Dallas, TX, USA, 2018, pp. 1054–1062. doi:10.1109/ICUAS.2018.8453288, URL `https://ieeexplore.ieee.org/document/8453288/`.

[4] Sun, S., Sijbers, L., Wang, X., and de Visser, C., "High-Speed Flight of Quadrotor Despite Loss of Single Rotor," *IEEE Robotics and Automation Letters*, Vol. 3, No. 4, 2018, pp. 3201–3207. doi:10.1109/LRA.2018.2851028, URL `https://ieeexplore.ieee.org/document/8398406/`.

[5] Mueller, M. W., and D'Andrea, R., "Stability and control of a quadrocopter despite the complete loss of one, two, or three propellers," *2014 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2014, pp. 45–52.

[6] Avram, R. C., Zhang, X., and Muse, J., "Quadrotor actuator fault diagnosis and accommodation using nonlinear adaptive estimators," *IEEE Transactions on Control Systems Technology*, Vol. 25, No. 6, 2017, pp. 2219–2226.

[7] Lu, P., and van Kampen, E.-J., "Active fault-tolerant control for quadrotors subjected to a complete rotor failure," *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Hamburg, Germany, 2015, pp. 4698–4703. doi:10.1109/IROS.2015.7354046, URL `http://ieeexplore.ieee.org/document/7354046/`.

[8] Avram, R. C., Zhang, X., and Muse, J., "Nonlinear adaptive fault-tolerant quadrotor altitude and attitude tracking with multiple actuator faults," *IEEE Transactions on Control Systems Technology*, Vol. 26, No. 2, 2018, pp. 701–707.

[9] Isermann, R., *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*, Springer, Berlin ; New York, 2006. OCLC: ocm61703226.

[10] Wolfram, D., Vogel, F., and Stauder, D., "Condition monitoring for flight performance estimation of small multirotor unmanned aerial vehicles," *2018 IEEE Aerospace Conference*, IEEE, Big Sky, MT, 2018, pp. 1–17. doi:10.1109/AERO.2018.8396471, URL `https://ieeexplore.ieee.org/document/8396471/`.

[11] Zhongming Ye, Bin Wu, and Sadeghian, A., "Current signature analysis of induction motor mechanical faults by wavelet packet decomposition," *IEEE Transactions on Industrial Electronics*, Vol. 50, No. 6, 2003, pp. 1217–1228. doi:10.1109/TIE.2003.819682, URL `http://ieeexplore.ieee.org/document/1254628/`.

[12] Pourpanah, F., Zhang, B., Ma, R., and Hao, Q., "Anomaly Detection and Condition Monitoring of UAV Motors and Propellers," *2018 IEEE SENSORS*, IEEE, New Delhi, India, 2018, pp. 1–4. doi:10.1109/ICSENS.2018.8589572, URL `https://ieeexplore.ieee.org/document/8589572/`.

[13] Jiang, Y., Zhiyao, Z., Haoxiang, L., and Quan, Q., "Fault detection and identification for quadrotor based on airframe vibration signals: A data-driven method," *2015 34th Chinese Control Conference (CCC)*, IEEE, Hangzhou, China, 2015, pp. 6356–6361. doi:10.1109/ChiCC.2015.7260639, URL `http://ieeexplore.ieee.org/document/7260639/`.

[14] Kandaswamy, G., and Balamuralidhar, P., "Health monitoring and failure detection of electronic and structural components in small unmanned aerial vehicles," *World Academy of Science, Engineering and Technology, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, Vol. 11, No. 5, 2017, pp. 1074–1082.

[15] Lu, H., Li, Y., Mu, S., Wang, D., Kim, H., and Serikawa, S., "Motor anomaly detection for unmanned aerial vehicles using reinforcement learning," *IEEE internet of things journal*, Vol. 5, No. 4, 2018, pp. 2315–2322.

[16] Vey, D., and Lunze, J., "Experimental evaluation of an active fault-tolerant control scheme for multirotor UAVs," *2016 3rd Conference on Control and Fault-Tolerant Systems (SysTol)*, IEEE, Barcelona, Spain, 2016, pp. 125–132. doi:10.1109/SYSTOL.2016.7739739, URL `http://ieeexplore.ieee.org/document/7739739/`.

[17] Lu, P., Van Eykeren, L., van Kampen, E.-J., de Visser, C., and Chu, Q., "Double-model adaptive fault detection and diagnosis applied to real flight data," *Control Engineering Practice*, Vol. 36, 2015, pp. 39–57.

[18] Bauer, P., Venkataraman, R., Vanek, B., Seiler, P. J., and Bokor, J., "Fault Detection and Basic In-Flight Reconfiguration of a Small UAV Equipped with Elevons," *IFAC-PapersOnLine*, Vol. 51, No. 24, 2018, pp. 600–607. doi:10.1016/j.ifacol.2018.09.637, URL `https://linkinghub.elsevier.com/retrieve/pii/S2405896318323498`.

[19] Hasan, A., and Johansen, T. A., "Model-Based Actuator Fault Diagnosis in Multirotor UAVs," *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, Dallas, TX, USA, 2018, pp. 1017–1024. doi:10.1109/ICUAS.2018.8453420, URL `https://ieeexplore.ieee.org/document/8453420/`.

[20] Nguyen, N. P., and Hong, S. K., "Sliding Mode Thau Observer for Actuator Fault Diagnosis of Quadcopter UAVs," *Applied Sciences*, Vol. 8, No. 10, 2018, p. 1893. doi:10.3390/app8101893, URL `http://www.mdpi.com/2076-3417/8/10/1893`.

[21] Edwards, C., and Spurgeon, S. K., "On the development of discontinuous observers," *International Journal of control*, Vol. 59, No. 5, 1994, pp. 1211–1229.

[22] Alwi, H., Edwards, C., and Tan, C. P., *Fault detection and fault-tolerant control using sliding modes*, Springer Science & Business Media, 2011.

[23] Chandra, K. P. B., Alwi, H., and Edwards, C., "Fault Reconstruction for a Quadrotor Using an LPV Sliding Mode Observer," *IFAC-PapersOnLine*, Vol. 48, No. 21, 2015, pp. 374–379.

[24] Murray, R. M., *A mathematical introduction to robotic manipulation*, CRC press, 1994.

[25] Bangura, M., and Mahony, R., "Nonlinear Dynamic Modeling for High Performance Control of a Quadrotor," *Australian Robotics and Automation Association*, 2012.

[26] Sun, S., de Visser, C. C., and Chu, Q., "Quadrotor Gray-Box Model Identification from High-Speed Flight Data," *Journal of Aircraft*, 2018, pp. 1–17.

[27] Lynch, K. M., and Park, F. C., *Modern Robotics*, Cambridge University Press, 2017.

[28] Smeur, E. J., Chu, Q., and de Croon, G. C., "Adaptive incremental nonlinear dynamic inversion for attitude control of micro air vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 12, 2015, pp. 450–461.

[29] Rajamani, R., and Hedrick, J. K., "Adaptive observers for active automotive suspensions: theory and experiment," *IEEE Transactions on control systems technology*, Vol. 3, No. 1, 1995, pp. 86–93.

[30] Schreier, G., Ragot, J., Patton, R., and Frank, P., "Observer Design for a Class of Non-Linear Systems," *IFAC Proceedings Volumes*, Vol. 30, No. 18, 1997, pp. 483–488. doi:10.1016/S1474-6670(17)42448-7, URL `https://linkinghub.elsevier.com/retrieve/pii/S1474667017424487`.

[31] Hasan, A., and Johansen, T. A., "Model-based actuator fault diagnosis in multirotor UAVs," *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, 2018, pp. 1017–1024.

[32] Koshkouei, A. J., and Zinober, A. S., "Sliding mode state observation for non-linear systems," *International Journal of Control*, Vol. 77, No. 2, 2004, pp. 118–127.

[33] Cho, Y. M., and Rajamani, R., "A systematic approach to adaptive observer synthesis for nonlinear systems," *IEEE transactions on Automatic Control*, Vol. 42, No. 4, 1997, pp. 534–537.

[34] Gallier, J., "The Schur complement and symmetric positive semidefinite (and definite) matrices," *Penn Engineering*, 2010.

[35] Verhulst, F., *Nonlinear differential equations and dynamical systems*, Springer Science & Business Media, 2006.

[36] Thau, F., "Observing the state of non-linear dynamic systems," *International journal of control*, Vol. 17, No. 3, 1973, pp. 471–479.

[37] Barkana, I., "Defending the beauty of the Invariance Principle," *International Journal of Control*, Vol. 87, No. 1, 2014, pp. 186–206. doi:10.1080/00207179.2013.826385, URL `https://doi.org/10.1080/00207179.2013.826385`.

[38] Besançon, G., "Remarks on nonlinear adaptive observer design," *Systems & control letters*, Vol. 41, No. 4, 2000, pp. 271–280.

A. Appendix

In this section, derivation of the parameter update law in (15) is presented. This update law was chosen to render the observer estimation error dynamics Lyapunov stable.

A. Derivation of the observer Lyapunov equation

The state estimation error of the observer is defined as $e_x = x - \hat{x}$, and the parameter estimation error is given as $e_\eta = \eta - \hat{\eta}$. It is seen that the output error in (15) can be rewritten as

$$y - \hat{y} = C(x - \hat{x})$$

If the system was linear, the observer design would simply mean the task of choosing a suitable gain matrix K making $A - KC$ Hurwitz, i.e., having roots with strictly negative real parts, ensuring the convergence of the state estimation error dynamics

$$\dot{e}_x = (A - KC)e_x$$

and thus reconstructing the states of the system via the observer as

$$\lim_{t\to\infty} e_x = 0 \quad \rightarrow \quad \lim_{t\to\infty} \hat{x} = x$$

The necessary condition for the existence of such matrix is the observability of the pair (A, C) which can be asserted using Kalman's observability test stating that if the observability matrix W of the system is of full rank ($\boldsymbol{Ker}\{W\} = 0$), the system is observable.

If observability is ensured, the error dynamics of the state and parameter estimation can be expressed based on (9) and (15):

$$\begin{aligned}
\dot{e}_x &= \dot{x} - \dot{\hat{x}} = \\
&= Ax + \Phi(x,u) + D\phi(x,u)\eta + Bu - (A\hat{x} + \Phi(\hat{x},u) + D\phi(\hat{x},u)\hat{\eta} + Bu + K(y - \hat{y})) \\
&= A(x - \hat{x}) - KC(x - \hat{x}) + \Phi(x,u) - \Phi(\hat{x},u) + D\left(\phi(x,u)\eta - \phi(\hat{x},u)\hat{\eta}\right) \\
&= (A - KC)e_x + \Phi(x) - \Phi(\hat{x}) + D\left(\phi(x)\eta - \phi(\hat{x})\hat{\eta}\right) \\
&= f(e_x, e_\eta)
\end{aligned} \tag{19}$$

where, to keep the notations simple, input u is omitted from the argument list of the nonlinear terms Φ and ϕ.

Finding a parameter update law of the form

$$\dot{\hat{\eta}} = -K_\eta e_\eta \qquad K_\eta \succ 0$$

would ensure the convergence of the parameter estimation, however, there are no means available to access η directly. Therefore, an update law, function of solely the system states, has to be found.

First, a candidate (real valued, continuously differentiable) Lyapunov function is chosen in the form

$$V(e_x, e_\eta) = e_x^T P e_x + \Gamma e_\eta^T e_\eta \tag{20}$$

with $P \in \mathbb{R}^{n\times n} \succ 0$ and $\Gamma \in \mathbb{R} > 0$. It is seen that

$$V(e_x, e_\eta) > 0 \;\forall\; e_x, e_\eta \neq 0$$
$$V(e_x, e_\eta) = 0 \Leftrightarrow e_x, e_\eta = 0$$

meaning that $V(e_x, e_\eta)$ is positive definite in the neighborhood of the origin. According to Lyapunov's theorem for stability of dynamical systems, if the time derivative of the Lyapunov candidate is negative definite $\forall\; e_x, e_\eta \neq 0$, then the nonlinear error dynamics $\dot{e}_x = f(e_x, e_\eta)$ in (19) is asymptotically stable [35].

Using (19), the time derivative of the Lyapunov function candidate can be written as

$$\begin{aligned}
\dot{V}(e_x, e_\eta) &= \dot{e}_x^T P e_x + e_x^T P \dot{e}_x + 2\Gamma e_\eta^T \dot{e}_\eta \\
&= \left(e_x^T (A - KC)^T + (\Phi(x) - \Phi(\hat{x}))^T + (\phi(x)\eta - \phi(\hat{x})\hat{\eta})^T D^T \right) P e_x \\
&\quad + e_x P \left((A - KC) e_x + (\Phi(x) - \Phi(\hat{x})) + D\left(\phi(x)\eta - \phi(\hat{x})\hat{\eta}\right)\right) \\
&\quad + 2\Gamma e_\eta^T \dot{e}_\eta \\
&= e_x^T \left((A - KC)^T P + P(A - KC) \right) e_x \\
&\quad + (\Phi(x) - \Phi(\hat{x}))^T P e_x + e_x^T P \left(\Phi(x) - \Phi(\hat{x})\right) \\
&\quad + (\phi(x)\eta - \phi(\hat{x})\hat{\eta})^T D^T P e_x + e_x^T P D \left(\phi(x)\eta - \phi(\hat{x})\hat{\eta}\right) \\
&\quad + 2\Gamma e_\eta^T \dot{e}_\eta
\end{aligned}$$

In order to further simplify the above equation, $\phi(\hat{x})\eta$ is simultaneously added to and subtracted from the

term $\phi(x)\eta - \phi(\hat{x})\hat{\eta}$, resulting in

$$\begin{aligned}
&\phi(x)\eta - \phi(\hat{x})\hat{\eta} + \phi(\hat{x})\eta - \phi(\hat{x})\eta = \\
=&\phi(x)\eta - \phi(\hat{x})\eta + \phi(\hat{x})\eta - \phi(\hat{x})\hat{\eta} \\
=&\left(\phi(x) - \phi(\hat{x})\right)\eta + \phi(\hat{x})\underbrace{(\eta - \hat{\eta})}_{e_\eta}
\end{aligned}$$

Utilizing the above simplification and Assumption III.1 on the boundedness of the estimated parameter, the following is obtained for the time derivative of the Lyapunov candidate

$$\begin{aligned}
\dot{V}(e_x, e_\eta) \leq\ & e_x^T \left((A - KC)^T P + P(A - KC)\right) e_x \\
&+ \left(\Phi(x) - \Phi(\hat{x})\right)^T P e_x + e_x^T P \left(\Phi(x) - \Phi(\hat{x})\right) \\
&+ \gamma_\eta \left(\phi(x) - \phi(\hat{x})\right)^T D^T P e_x + e_x^T P D \left(\phi(x) - \phi(\hat{x})\right) \gamma_\eta \\
&+ e_\eta^T \phi(\hat{x})^T D^T P e_x + e_x^T P D \phi(\hat{x}) e_\eta \\
&+ 2\Gamma e_\eta^T \dot{e}_\eta
\end{aligned}$$

Furthermore, due to the nonlinear terms $\Phi(x, u)$ and $\phi(x, u)$ being Lipschitz continuous, an upper bound of $\dot{V}(e_x, e_\eta)$ is found in the form of

$$\begin{aligned}
\dot{V}(e_x, e_\eta) \leq\ & e_x^T \underbrace{\left((A - KC)^T P + P(A - KC)\right)}_{-Q} e_x \\
&+ \underbrace{2\gamma_\Phi e_x^T P e_x + 2\gamma_\phi \gamma_\eta \|D\| e_x^T P e_x}_{(\gamma_\Phi + \gamma_\phi \gamma_\eta \|D\|) 2 e_x^T P e_x} \\
&+ 2 e_\eta^T \phi(\hat{x})^T D^T P e_x \\
&+ 2\Gamma e_\eta^T \dot{e}_\eta
\end{aligned} \tag{21}$$

where the matrix norm of D denotes its maximum singular value.

Lemma A.1 (Quadratic inequality)

Suppose that $v \in \mathbb{R}^k$. Then, by definition of the Euclidean norm, $\|v\|_2 = \sqrt{v^T v}$ is always positive for a non-zero v vector, and zero if $v = 0$, meaning that $v^T v \geq 0\ \forall\ v$. Therefore

$$a^T a + b^T b \geq 2 a^T b\ \forall\ a, b \in \mathbb{R}^k$$

Proof. Consider an arbitrary set of vectors $a, b, v \in \mathbb{R}^k$ such that $v = a - b$. Note that

$$(a-b)^T(a-b) \geq 0 \;\forall\; a, b$$

which means that

$$a^T a - a^T b - b^T a + b^T b \geq 0$$
$$a^T a + b^T b \geq 2a^T b$$

■

Based on Lemma A.1, the following can be stated of expression $2e_x^T P e_x$:

$$\begin{aligned} 2e_x^T P e_x \leq 2\|e_x\|\|Pe_x\| &\leq \|e_x\|^2 + \|Pe_x\|^2 \\ &= e_x^T e_x + e_x^T P P e_x \\ &= e_x^T \left(\mathcal{I} + PP\right) e_x \end{aligned}$$

Additionally, choosing an update law for the parameter estimation error dynamics in the form

$$\dot{e}_\eta = -\frac{1}{\Gamma}\phi(\hat{x})^T D^T P e_x \tag{22}$$

will ensure that the time derivative of the Lyapunov function candidate satisfies the inequality presented in (23).

$$\begin{aligned} \dot{V}(e_x, e_\eta) \leq & -e_x^T Q e_x \\ & + (\gamma_\Phi + \gamma_\phi \gamma_\eta \|D\|) e_x^T (PP + \mathcal{I}) e_x \\ & + 2e_\eta^T \phi(\hat{x})^T D^T P e_x \\ & + 2\Gamma e_\eta^T \dot{e}_\eta \\ = & \, e_x^T \underbrace{\left(-Q + (\gamma_\Phi + \gamma_\phi \gamma_\eta \|D\|)(PP + \mathcal{I})\right)}_{M} e_x \end{aligned} \tag{23}$$

B. Stability analysis of the observer dynamics

Due to the assumption that the estimated parameter is slowly varying—as listed in Assumption III.1—the update law in (22) yields the parameter estimation update law of the observer as shown in (15) by writing

$$\dot{e}_\eta = \dot{\eta} - \dot{\hat{\eta}} = -\dot{\hat{\eta}} = -\frac{1}{\Gamma}\phi(\hat{x})^T D^T P e_x$$

Furthermore if $(D^T P)_i \in \boldsymbol{Im}\{C^T\} = \boldsymbol{Ker}\{C\}^\perp \; \forall \; i$ with $(P^T D)_i$ denoting the i^{th} row of $P^T D$ (as assumed in Assumption III.1) then $(D^T P)_i \underbrace{(C^T C)^{-1} C^T}_{C^+} C(x - \hat{x})$ is nonzero for any nonzero state estimation error. This assumption is important to hold because only the signal $y - \hat{y} = C(x - \hat{x})$ is available for feedback, therefore (22) in itself cannot be used due to the unknown x. Thus, the final parameter estimation update law is given in the form

$$\dot{\hat{\eta}} = \frac{1}{\Gamma} \phi(\hat{x})^T D^T P C^+ (y - \hat{y})$$

As for the time derivative of the candidate Lyapunov function in (23), matrix M is negative definite if the Thau-condition [36] listed in Assumption III.1 holds. The Thau-condition can be derived from (21) where both Q and P are real valued matrices and the update law in (22) has been applied, thus

$$\begin{aligned}
\dot{V}(e_x) &\leq -e_x^T Q e_x \\
&\quad + (\gamma_\Phi + \gamma_\phi \gamma_\eta \|D\|) 2 e_x^T P e_x \\
&\leq -\|e_x^T\| \|Q e_x\| + 2\alpha \|e_x^T\| \|P e_x\| \\
&\leq -\lambda_{min}(Q) \|e_x\|^2 + 2\alpha \lambda_{max}(P) \|e_x\|^2
\end{aligned}$$

which only holds if

$$\alpha < \frac{\lambda_{min}(Q)}{2\lambda_{max}(P)}$$

Therefore, to prove the convergence of the proposed observer, the following list can be established, based on the above assumptions and derivations:

1) $V\left(e_x(t)\right)$ is lower bounded, since $V\left(e_x(t)\right) \geq 0 \; \forall e_x(t)$ by definition.
2) $\dot{V}\left(e_x(t)\right)$ is negative semidefinite, as $\exists \mu > 0 \in \mathbb{R} \; : \; \dot{V}\left(e_x(t)\right) \leq -e_x^T(t) M e_x(t) = -\mu e_x^T(t) e_x(t)$.
3) $\ddot{V}\left(e_x(t)\right) \in \mathcal{L}^\infty$, i.e. $\dot{V}\left(e_x(t)\right)$ is uniformly continuous.

Then, by Barbalat's lemma, $\lim_{t\to\infty} \dot{V}\left(e_x(t)\right) = 0$ [37], meaning that $\lim_{t\to\infty} -\mu e_x^T(t) e_x(t) = 0$ which implies that $\lim_{t\to\infty} e_x(t) = 0$, and the state estimation error will converge to zero [33].

Finally, the parameter estimation error converges to zero asymptotically if the following persistence of excitation condition holds [33, 38]:

$$\exists \; T, \overline{k}, \underline{k} > 0 \; : \; \overline{k}\mathcal{I} \geq \int_{t_0}^{t_0+T} D\phi\left(x(t), u(t)\right) \phi^T\left(x(t), u(t)\right) D^T \mathrm{d}t \geq \underline{k}\mathcal{I} \tag{24}$$

Since $\phi(x) > 0 \; \forall \; x \neq 0$ and $\|\phi(x)\|_2 < \infty \; \forall \; x \in \mathcal{X}$, where $\mathcal{X}$ is a bounded state space, the system is persistently excited unless $x \equiv 0$.

C. Parameter identification for observer model construction

In this section, the parameter identification process of a single actuator is introduced. The methodology for establishing the thrust model and obtaining the actuator inertia is presented.

The testbench setup, seen in Figure 10 has current sensors to read the current flowing through the motor and optical sensors that read the motor speed. Furthermore, there are two load cells aligned in such way, that one of them provides thrust force measurements, the other torque measurements around the propeller z axis. Unfortunately, the torque sensor precision was insufficient to a degree that it was unusable, however, the thrust sensor performed impeccably, as seen in Figure 14. Thus, the thrust coefficient was obtained based on (6) for each steady state data section plotted in Figure 15. Thereafter the final value for κ_{T} was yielded by a weighted average of the identified thrust coefficients, with greater weights assigned to near-hover rotor speeds. However, the torque coefficient had to be determined using a different method.

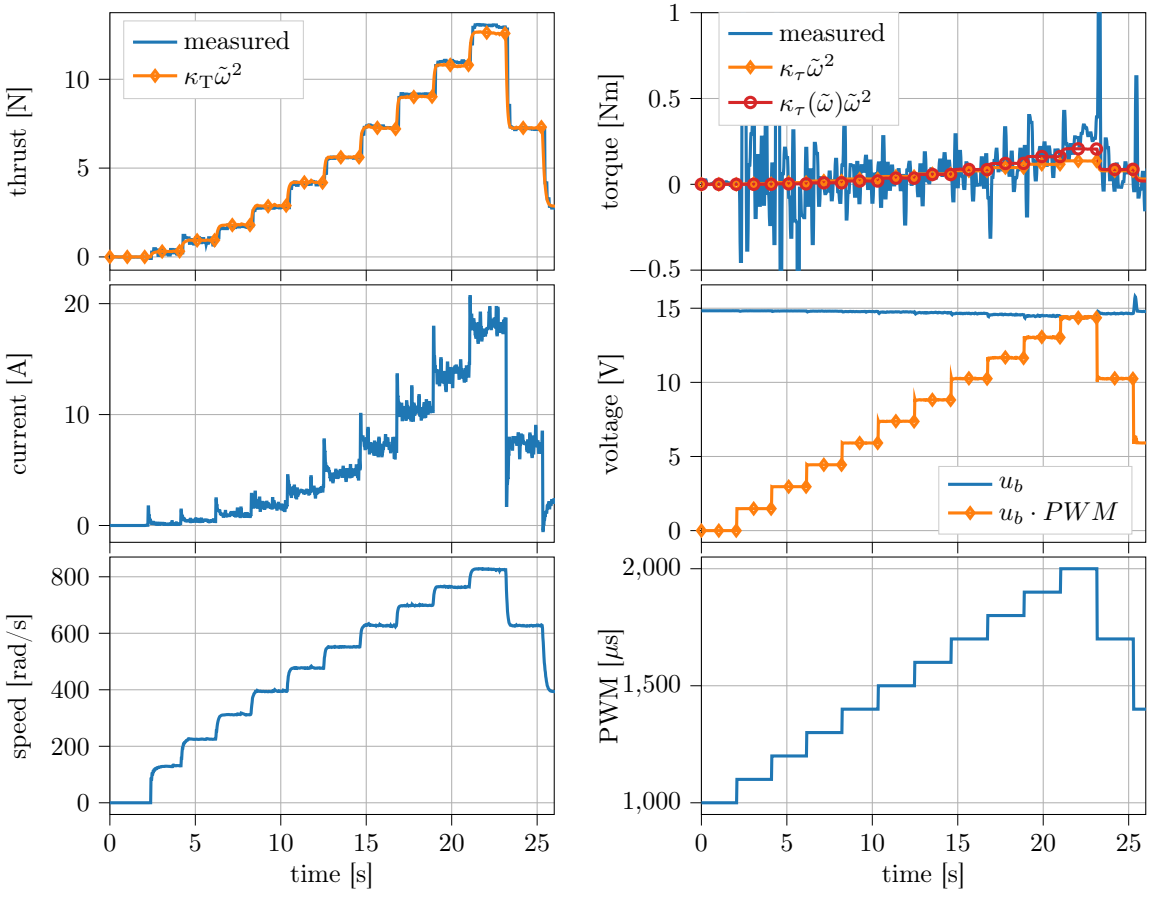

Fig. 14 All data gathered from the testbench setup.

First of all, a staircase-like PWM input was given to the system as shown in Figure 14, with the propeller detached. Then, the speed and current measurements—similarly as shown in Figure 15—was plotted for all

PWM levels. From (5), the steady state equation without a propeller is found to be

$$\kappa_f \omega_{ss} = \kappa_q i$$

which, since κ_q can be computed from the motor velocity constant given by the manufacturer, yields the value of κ_f via an OLS parameter estimation. Then, the same input was given to the system with the propeller attached with Figure 15 showing the corresponding speed and current measurements. Since κ_f was already identified, another OLS estimation procedure was performed on the measurement data based on

$$\kappa_\tau \omega_{ss}^2 = \kappa_q i - \kappa_f \omega_{ss}$$

for each "step" shown in Figure 15. As a result, it was found, that the value of κ_τ varies significantly with the rotor speed as seen in Figure 16 and Table 1; therefore, the value of κ_τ is actively scheduled in the observer model based on the linear approximation of the fitted function ($\kappa_\tau(\omega)$) describing the relationship between κ_τ and ω, see Figure 16.

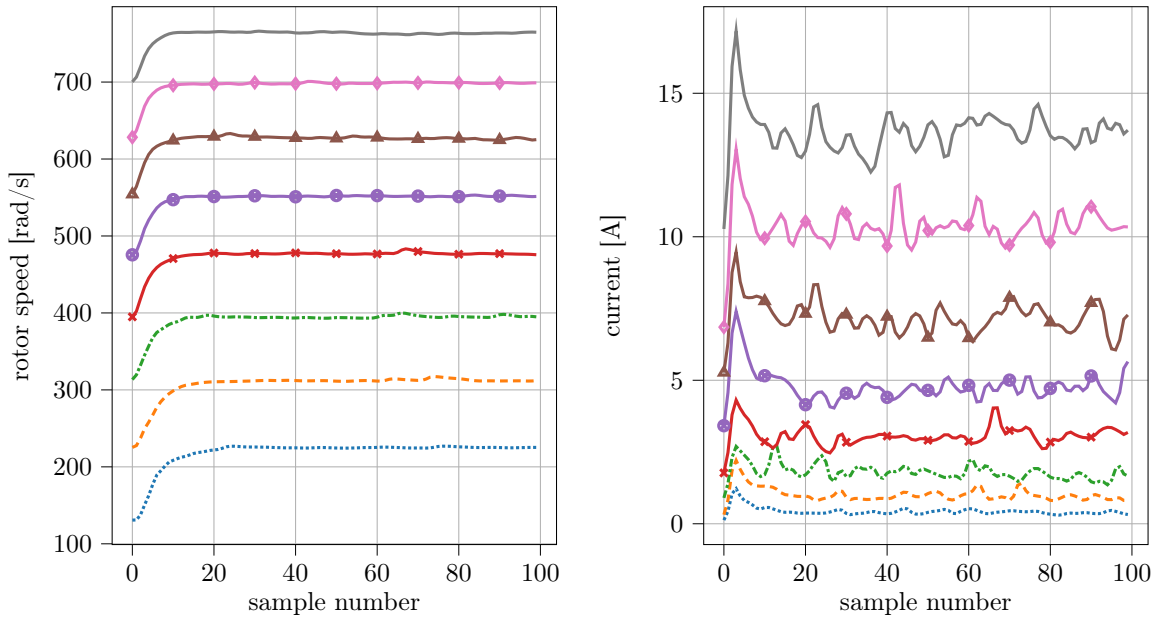

Fig. 15 System response at various PWM levels. The data slices were taken from system responses to a "staircase-like" pwm input sequence.

Finally, the inertia of the blade and the motor's rotating part was obtained via computing $\dot{\omega}$ via numerically differentiating the rotor speed samples in the transient phases of each step response in Figure 15. Then, since the values of κ_τ and κ_f were already determined, the inertia of the actuator was also found via an OLS

procedure applied on the complete actuator dynamics in (5).

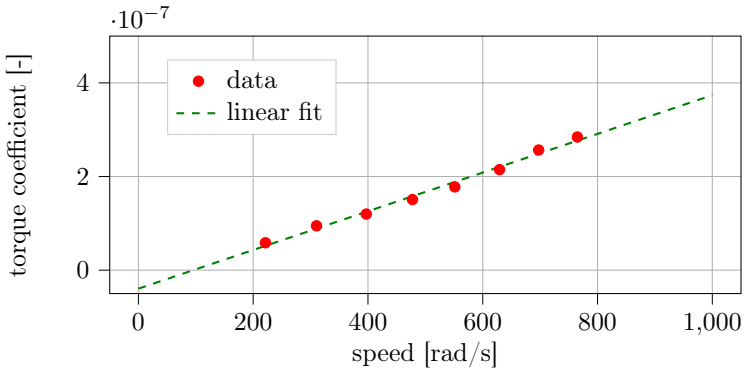

Fig. 16 κ_τ changes considerably with the rotor speed. Therefore a linear fit was applied to obtain a scheduling function.

Table 1 Parameters identified on the HIL setup for each steady state (κ_T, κ_τ, κ_f) and transient data slice (J)

Parameters			Data sections							
PWM		$[\mu s]$	1300	1400	1500	1600	1700	1800	1900	2000
ω_{ss}		$\left[\frac{rad}{s}\right]$	221.32	310.28	397.02	477.59	511.35	629.26	697.43	764.84
κ_T	$(\cdot 10^{-5})$	$\left[\frac{N\cdot s^2}{rad^2}\right]$	1.7878	1.7348	1.7815	1.8193	1.8390	1.8615	1.8783	1.8853
κ_τ	$(\cdot 10^{-7})$	$\left[\frac{Nm\cdot s^2}{rad^2}\right]$	0.5850	0.9467	1.1980	1.5083	1.7781	2.1473	2.5663	2.8437
κ_f	$(\cdot 10^{-6})$	$\left[\frac{Nm\cdot s}{rad}\right]$	9.5810	8.5110	11.039	9.3082	8.9735	9.8699	8.9732	8.1637
J	$(\cdot 10^{-5})$	$[kg\cdot m^2]$	1.5770	2.7314	2.0508	2.0887	5.2432	3.5305	3.0275	5.9222

Table 1 shows the respective parameters identified for each steady state and transient section of the measured variables shown in Figure 15. Motor friction coefficient κ_f is plotted for the detached-propeller case. It is seen that the thrust coefficient is accurately modeled by (6), however, the torque coefficient significantly varies with rotor speed. Therefore, the observer model used the weighted average of κ_T, κ_f, and J as a constant, whereas κ_τ was actively scheduled as a function of ω.

As shown in Table 2, the thrust coefficient (κ_T) and the motor friction coefficient (κ_f) are estimated with a relatively low variance. However, the variance of the estimated inertia for each transient in Figure 15 indicates that the dynamics of the actuator can become considerably different than the online model. For this reason, in Appendix A.D, it is shown that the detection speed barely changes even with 50% error between the true and model inertia.

Table 2 Statistical parameters of the identified constants. In the first row, u denotes the unit of the respective statistical parameter, meaning that variance has unit u^2. Low variance of the thrust coefficient (κ_{T}) indicates an accurate estimate. Greater variances of the torque coefficient (κ_τ) and motor friction coefficient (κ_f) indicate that the mean represents the system's behaviour less accurately compared to the thrust coefficient. Finally, it is seen that estimated inertia parameters are widely spread around the mean which indicates that there would be significant errors between the true and estimated inertia at various motor speeds. The weighted average of the coefficients is supposed to alleviate modeling errors around hover conditions, i.e., greater weights are assigned to constants identified near hover speeds.

Parameters			Mean (u)	Variance (u^2)	Weighted Average (u)
κ_{T}	$(\cdot 10^{-5})$	$\left[\frac{N\cdot s^2}{rad^2}\right]$	1.8234	0.0024	1.8465
κ_τ	$(\cdot 10^{-7})$	$\left[\frac{Nm\cdot s^2}{rad^2}\right]$	1.6967	0.5455	1.9960
κ_f	$(\cdot 10^{-6})$	$\left[\frac{Nm\cdot s}{rad}\right]$	9.3024	0.6944	9.4782
J	$(\cdot 10^{-5})$	$[kg\cdot m^2]$	3.2714	2.1366	3.9876

D. Sensitivity analysis of the observer dynamics

In this final part of the appendix, the effect of identified parameters is discussed. The essence of the proposed adaptive filter is to estimate the value of κ_τ based on measurements such as ω and i. Should the initial (or scheduled) parameter of κ_τ be off by a certain amount, the observer is supposed to estimate this parameter error in the actuator effectiveness.

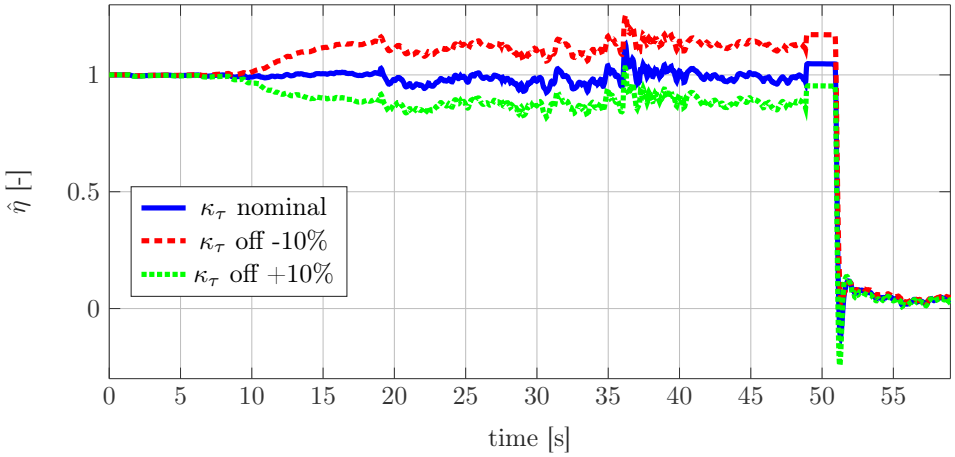

Fig. 17 The effect of the online value of κ_τ on effectiveness estimation. Note, that if a $\pm 10\%$ error is present, the effectiveness estimate converges to ~ 0.9 or ~ 1.1 respectively. Thus, if κ_τ is lower than the true value, this error appears as an increased effectiveness of the blades and vice versa. However, when the full fault occurs, the effectiveness estimate goes to zero regardless of the error in κ_τ

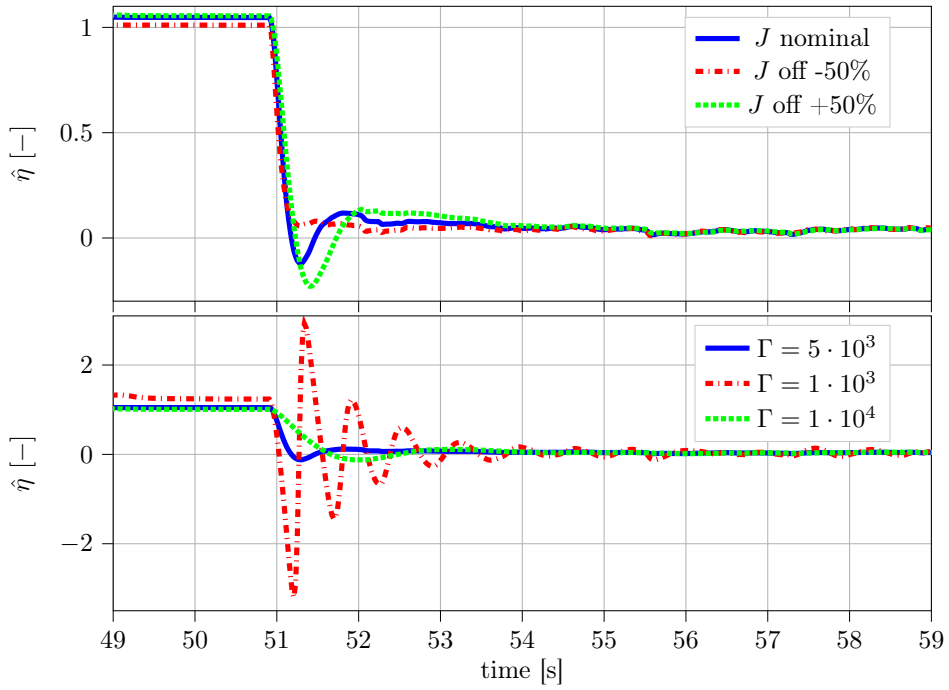

Fig. 18 Fault detection speed affected by various parameters. It is seen that if the online inertia estimate is off from the true value the effectiveness estimation speed is affected. If the estimated inertia is much less than the actual value, the detection speed may increase slightly, however, every transient in rotor speed will introduce a large error in estimated and true rotor speeds, due to the different step responses. The more significant value affecting convergence speed is the tuning gain seen in (15). A larger Γ results in less overshoot with a trade-off in detection speed. It is shown that smaller Γ values decrease the detection time at the cost of poorly damped oscillations.

Based on Figure 17, it is assumed that, in real flight conditions, aerodynamic effects causing an effectiveness change in propellers are captured if their frequency is within the bandwidth of the observer. Higher frequency effects are assumed to not change propeller effectiveness considerably. Furthermore, minor errors in κ_τ are also assumed to be mitigated by an inherently robust INDI flight controller. Additionally, errors in parameter κ_f appear as additive torques which are also captured by the effectiveness estimate in a similar way as errors in κ_τ, see Figure 17.

As shown in Figure 18, the detection speed is affected by the online estimate of actuator inertia J. If the online model has a lower inertia value then the true system, the dynamics of the actuator model, see (5), of the observer has faster response time resulting in error between the estimated and measured motor speed at every transient. Thus, every time there is a step in the reference rotor speeds, the effectiveness estimation will be off by a large margin. This effect is shown in Figure 8 where the inertia of the real system is actually less than the online parameter after it became damaged. This may lead to false alarms, therefore, it is important to have an accurate estimate of the inertia. However, results in Figure 18 show that there is

considerable room for the inertia parameter estimation error.

As for the tuning parameter of the adaptive update law of the effectiveness in (15), the rise time of the effectiveness estimate is greatly affected by its value. Figure 18 shows that a greater Γ value results in slower response with less overshoot, whereas a small Γ reduces rise time at the cost of poorly damped oscillations. Therefore, a value had to be found that kept the rise (detection) time around 0.2 seconds with acceptable damping. This value might vary considerably with the actuator model at hand.

Part III

Discussion and Appendices

Chapter 6

Conclusions

In this final chapter, the contributions of this project are discussed with respect to the research questions posed in Chapter 1. First of all, various methods for multicopter Fault Detection and Diagnosis (FDD) were researched in a literature survey. The findings of this survey listed methods suitable for multicopter FDD in a way that it facilitated laying out the path of the project. As shown through examples from FDD literature, signal analysis is a popular approach for detecting anomalies in the monitored system. Current research also applies Neural Networks (NNs) trained via the extraction of vibration or current signal features. Several adaptive control methodologies were introduced as well with the emphasis on Direct Adaptive Control (DAC), Indirect Adaptive Control (IAC), and Multiple Model Adaptive Estimation (MMAE). It was shown that IAC is generally more flexible to design and implement than DAC when FDD modules are included in the models as well. MMAE provides a powerful switching adaptive law which is suitable for Loss of Effectiveness (LoE) estimation, however, several models have to be identified offline for each fault to be accounted. This also means that that more observers/filters have to be simultaneously running online on each model which can be computationally expensive. Finally, several observer design methods were discussed which were mostly nonlinear and model based. From the nonlinear Thau observer, through discontinuous sliding mode observers, to adaptive observers, a great variety of modules are used for FDD purposes in the literature. These observers are powerful tools for LoE estimation, with Lyapunov-based estimation convergence. Furthermore they are generally straightforward to implement within a IAC framework, making them even more desirable to use within a Active Fault-Tolerant Control (AFTC) system.

Based on the preliminary literature survey, and the following requirements towards the FDD framework

1. fault detection time should be below 0.2 seconds
2. fault isolation must be ensured among actuators
3. stability and reliability of the FDD system has to be asserted

a nonlinear adaptive Thau observer was proposed for LoE estimation of the actuators. Since fault isolation is an important requirement, i.e., it has to be straightforward to show

which actuator is faulty, low-level (decentralized) signals were investigated to be utilized for FDD purposes. It was found that current and motor speed measurements play a valuable role in resolving observability issues reported in multiple pieces of literature. It was argued that using high-level translational and angular dynamics of the multicopter does not necessarily ensure the observability of effectiveness parameters of the actuators, especially if the multicopter has 6 or more actuators. The proposed observer model, augmenting the angular velocity dynamics with decentralized actuator models, was shown to be observable for any number of actuators. It was also shown that the observer model can be scaled down to a single actuator model, meaning that isolation of faulty actuators are ensured by decentralized signals.

Furthermore, simulation and Hardware-in-the-loop (HIL) results indicate that decentralized signals (due to the fast dynamics of the actuators) enable fault detection within 0.2 seconds. This is an important achievement, since the control system of the multicopter can be updated to act before the multicopter reaches an unrecoverable pose due to the imbalance in propulsion. Additionally, a key property of the proposed observer is that it is able to estimate partial and complete LoE as well with proven asymptotic stability. Profound stability analysis was given in the document to prove the convergence of the FDD observer.

The last research question addressed suitable methods used to interpret the observer LoE estimates for the AFTC system governing the multicopter. Since the design of the observer was modular, the control law itself is not affected directly by the observer. Instead, the observer augments a switched control system which is designed to steer the multicopter to safety even if one or multiple actuators suffer LoE. If the control system is based on Incremental Nonlinear Dynamic Inversion (INDI) or Nonlinear Dynamic Inversion (NDI), the control allocator is updated with the estimated effectiveness of each actuator. Thus, the allocator will assign greater reference speed for an actuator if a partial LoE is detected. However, in case of complete LoE the observer instructs the allocator to update the allocation law and the controller to switch to a reduced controller.

It was shown that adaptive Thau observers estimating actuator effectiveness parameters are a powerful foundation for a FDD framework on multirotor Unmanned Aerial Vehicles (UAVs). Such an observer enables the multicopter to monitor the state of its actuators and update its control system or operator to intervene and address LoE issues in its propulsion system. Hence, integrity and safety of multirotor UAVs can be greatly enhanced.

Last but not least, some shortcomings of the observer are pointed out. It was shown that even though the observer needs several model parameters to be accurately identified beforehand, these parameters are straightforward to obtain. However, the observer assumes that current and rotor speed sensors are available on-board which might limit its applicability on commercially available multicopters. Nevertheless, as an immediate continuation of this project, actual flights are planned to be performed with multicopters equipped with AFTC systems. Based on the introduced results, the FDD observer is anticipated to successfully augment the AFTC system and provide fast and accurate information on actuator availability.

Appendix A

Schur complement

In this appendix, some useful insights are discussed which can give a better understanding of some phases in this thesis project. Here, the Schur complement of a symmetric matrix will be examined, as it has fundamental ramifications in the field of Linear Matrix Inequality (LMI) optimization for robust observer synthesis.

Suppose $\exists\ M = M^T \in \mathbb{R}^{m\times m}$ partitioned as follows

$$M = \begin{bmatrix} P & S \\ S^T & Q \end{bmatrix} \tag{A.1}$$

Assumption A.1
Matrix Q is assumed to be invertible.

Due to Assumption A.1, Q has to be square, i.e. $Q \in \mathbb{R}^{q\times q}$ with $q \leq m$. Then S must be $S \in \mathbb{R}^{p\times q}$ with $p = m - q$, therefore, $P \in \mathbb{R}^{p\times p}$.

Assume that the inverse of M is sought, such that the solution of (A.2) is found [76].

$$\begin{bmatrix} P & S \\ S^T & Q \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix} \tag{A.2}$$

Since Assumption A.1 holds, (A.3) can be written to find y in (A.2).

$$y = Q^{-1}(b - S^T x) \tag{A.3}$$

Substituting (A.3) into

$$Px + Sy = a$$

it is obtained that

$$\begin{aligned} Px + SQ^{-1}(b - S^T x) &= a \\ (P - SQ^{-1}S^T)x &= a - SQ^{-1}b \end{aligned} \tag{A.4}$$

Definition A.1 (Schur complement [77])
Matrix $P - SQ^{-1}S^T$ is defined as the Schur complement of block Q in matrix M.

Assumption A.2
It is assumed that the Schur complement of Q, i.e. $P - SQ^{-1}S^T$, is invertible.

If Assumption A.2 holds, then based on (A.4) it is found that

$$x = (P - SQ^{-1}S^T)^{-1}(a - SQ^{-1}b)$$

which, along with (A.3) can be rewritten into matrix form as

$$\begin{bmatrix} x \\ y \end{bmatrix} = \underbrace{\begin{bmatrix} (P - SQ^{-1}S^T)^{-1} & -(P - SQ^{-1}S^T)^{-1}SQ^{-1} \\ -Q^{-1}S^T(P - SQ^{-1}S^T)^{-1} & Q^{-1} + Q^{-1}S^T(P - SQ^{-1}S^T)^{-1}SQ^{-1} \end{bmatrix}}_{M^{-1}} \begin{bmatrix} a \\ b \end{bmatrix} \tag{A.5}$$

Then, with $\mathcal{I}$ denoting an identity matrix of arbitrary size, it is found that the inverse of M in (A.5) can be factorized as

$$M^{-1} = \underbrace{\begin{bmatrix} \mathcal{I} & 0 \\ -Q^1 S^T & \mathcal{I} \end{bmatrix}}_{(N^T)^{-1}} \underbrace{\begin{bmatrix} (P - SQ^{-1}S^T)^{-1} & 0 \\ 0 & Q^{-1} \end{bmatrix}}_{\Lambda^{-1}} \underbrace{\begin{bmatrix} \mathcal{I} & -SQ^{-1} \\ 0 & \mathcal{I} \end{bmatrix}}_{N^{-1}} \tag{A.6}$$

Definition A.2 (Matrix inverse)
Suppose $A \in \mathbb{R}^{n \times n}$ is an invertible matrix of the form

$$A = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{1n} \\ a_{21} & \ddots & & \\ \vdots & & \cdots & \\ a_{n1} & & & a_{nn} \end{bmatrix}$$

The inverse matrix of A is defined as

$$A^{-1} = \frac{adj(A)}{det(A)}$$

where $det(A)$ denotes the determinant of matrix A. $adj(A)$ denotes the adjungate matrix of A which is computed as

$$adj(A) = \begin{bmatrix} \alpha_{11} & \alpha_{21} & \cdots & \alpha_{1n} \\ \alpha_{21} & \ddots & & \\ \vdots & & \cdots & \\ \alpha_{n1} & & & \alpha_{nn} \end{bmatrix}^T$$

where α_{ij} denotes the cofactor of matrix A at element ij.

Lemma A.1
Based on (A.6)*, M can be factorized as*

$$M = N\Lambda N^T$$

where

$$N = \begin{bmatrix} \mathcal{I} & SQ^{-1} \\ 0 & \mathcal{I} \end{bmatrix}$$

and

$$\Lambda = \begin{bmatrix} P - SQ^{-1}S^T & 0 \\ 0 & Q \end{bmatrix}$$

Proof. Due to the associative property of the matrix product, it can be written that

$$\begin{aligned} MM^{-1} &= (N\Lambda N^T)M^{-1} = \mathcal{I} \\ &= N\Lambda \underbrace{\left(N^T(N^T)^{-1}\right)}_{\mathcal{I}} \Lambda^{-1}N^{-1} = \\ &= N \underbrace{(\Lambda\Lambda^{-1})}_{\mathcal{I}} N^{-1} = \\ &= NN^{-1} = \mathcal{I} \end{aligned}$$

Now it has to be checked whether the inverse of N and Λ is indeed defined correctly in (A.5). First consider the problem of finding the inverse of N. Since $NN^{-1} = \mathcal{I}$ must hold, using N^{-1} from (A.6), it can be asserted that

$$\begin{bmatrix} \mathcal{I} & SQ^{-1} \\ 0 & \mathcal{I} \end{bmatrix} \begin{bmatrix} \mathcal{I} & -SQ^{-1} \\ 0 & \mathcal{I} \end{bmatrix} = \begin{bmatrix} \mathcal{I} & SQ^{-1} - SQ^{-1} \\ 0 & \mathcal{I} \end{bmatrix} = \mathcal{I}$$

Then, since Λ is a block diagonal matrix with blocks A and B, using Definition A.2, it can be written that the inverse of Λ is given as

$$\Lambda^{-1} = \begin{bmatrix} A^{-1} & 0 \\ 0 & B^{-1} \end{bmatrix}$$

which indeed yields

$$\Lambda\Lambda^{-1} = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix} \begin{bmatrix} A^{-1} & 0 \\ 0 & B^{-1} \end{bmatrix} = \begin{bmatrix} AA^{-1} & 0 \\ 0 & BB^{-1} \end{bmatrix} = \begin{bmatrix} \mathcal{I} & 0 \\ 0 & \mathcal{I} \end{bmatrix}$$

■

Corollary A.0.1
It was found that the symmetric matrix M can be factorized via the Schur complement of block Q, such that

$$M = N\Lambda N^T$$

meaning that the M is invertible if and only if Q and the Schur complement of Q is invertible.

Definition A.3 (Matrix definiteness)
Consider a symmetric matrix $R \in \mathbb{R}^{r\times r}$. Then, $\forall\ x \in \mathbb{R}^r,\ x \neq 0$, R is said to be

- *Positive definite ($R \succ 0$) if $x^T Rx > 0$*
- *Positive semidefinite ($R \succeq 0$) if $x^T Rx \geq 0$*
- *Negative definite ($R \prec 0$) if $x^T Rx < 0$*
- *Negative semidefinite ($R \preceq 0$) if $x^T Rx \leq 0$*

Lemma A.2
If matrix P is positive definite then all its eigenvalues must be positive.

Proof. Consider Definition A.3 and take v as a non-trivial eigenvector of $P \succ 0$. Then, having $\lambda \in \mathbb{R}$ defined as the corresponding eigenvalue of P, it is seen that

$$\lambda v = Pv$$

Suppose, that $\lambda = 0$ implying $v \in \boldsymbol{Ker}\{P\}$. Therefore, it is seen that $v^T P v = 0$, making P at most positive semidefinite. Now take $\lambda < 0$. Then, using the definition of the 2-norm of a nonzero vector, it can be asserted that $v^T v = \|v\| > 0$, thus

$$v^T P v = v^T(\lambda v) = \lambda v^T v < 0$$

which means, according to Definition A.3, that P is negative definite. ■

After these preliminary findings, an important lemma, based on the Schur complement, can be stated which can be utilized when setting up and solving LMIs.

Lemma A.3
Consider $M = M^T$ of the form (A.1)*. Then the following statements hold*

1. $M \succ 0 \;\Leftrightarrow\; Q \succ 0$ *and* $P - SQ^{-1}S^T \succ 0$
2. *if* $Q \succ 0$ *then* $M \succeq 0 \;\Leftrightarrow\; P - SQ^{-1}S^T \succeq 0$

Proof. • Prove 1): $M \succ 0 \;\Rightarrow\; Q \succ 0$ and $P - SQ^{-1}S^T \succ 0$

Since M is symmetric, $\exists\ \Lambda$ diagonal and $\exists\ N$ invertible such that the decomposition of M is written as

$$M = N\Lambda N^T$$

Since N is invertible it is full rank, therefore, $\boldsymbol{Im}\{N\} \in \mathbb{R}^m$, meaning that $\forall\ y \in \mathbb{R}^m,\ y \neq 0\ \exists x \in \mathbb{R}^m,\ x \neq 0\ :\ N^T x = y$. Hence, using Definition A.3 and the fact $M \succ 0$, it can be shown that $\forall\ x, y \in \mathbb{R}^m,\ x, y \neq 0$

$$\begin{aligned} x^T M x &> 0 \\ x^T N \Lambda N^T x &> 0 \\ y^T \Lambda y &> 0 \\ \Lambda &\succ 0 \end{aligned}$$

Taking N and Λ as shown in Lemma A.1 it is found that

$$\Lambda = \begin{bmatrix} P - SQ^{-1}S^T & 0 \\ 0 & Q \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix} \succ 0$$

therefore, $\forall x \in \mathbb{R}^m,\ x = [x_1^T, x_2^T]^T \neq 0$ with x_1, x_2 having appropriate sizes

$$x_1^T A x_1 + x_2^T B x_2 > 0$$

thus, if $x_2 = 0$ then x_1 cannot be zero, hence $x_1^T A x_1 > 0$ must hold for all nonzero vector x_1, implying that $A \succ 0$. The same must hold for all x with $x_1 = 0$ and $x_2 \neq 0$ which leads to $B \succ 0$.

• Prove 1) $M \succ 0 \;\Leftarrow\; Q \succ 0$ and $P - SQ^{-1}S^T \succ 0$

Since $Q \succ 0$ and $P - SQ^{-1}S^T \succ 0$, the diagonal matrix

$$D = \begin{bmatrix} P - SQ^{-1}S^T & 0 \\ 0 & Q \end{bmatrix}$$

must be positive definite, as it has the eigenvalues of Q and $P - SQ^{-1}S^T$ which are all positive. Furthermore according to [78], $\forall$ R of full rank, it holds that if $D \succ 0$ then $RDR^T \succ 0$. Thus, if $R = N$ as defined in Lemma A.1, then it is seen that $M = NDN^T \succ 0$.

- Prove 2) if $Q \succ 0$ then $M \succeq 0 \;\Rightarrow\; P - SQ^{-1}S^T \succeq 0$

Since M is symmetric and positive semidefinite, it is diagonalizable as shown in Lemma A.1. Due to N being invertible, Σ must be at positive semidefinite and since all eigenvalues of Q are strictly positive, $P - SQ^{-1}S^T$ must be positive definite in order for $M \succeq 0$ to hold.

- Prove 2) if $Q \succ 0$ then $M \succeq 0 \;\Leftarrow\; P - SQ^{-1}S^T \succeq 0$

Since $Q \succ 0$ and $P - SQ^{-1}S^T \succeq 0$ then the eigenvalues of

$$D = \begin{bmatrix} P - SQ^{-1}S^T & 0 \\ 0 & Q \end{bmatrix}$$

are nonnegative, meaning that $D \succeq 0$ has only nonnegative eigenvalues. Note, that D is symmetric, due to $P - SQ^{-1}S^T$ and Q being both symmetric matrices. Therefore, the square root of matrix D can be taken, and for any matrix R, the following holds

$$RDR^T = RD^{\frac{1}{1}}D^{\frac{1}{2}}R^T = RD^{\frac{1}{2}}\left(RD^{\frac{1}{2}}\right)^T = UU^T \succeq 0$$

because for any nonzero z, $z^TUU^Tz \neq 0$. Since R can be any matrix, having $R = N$ shows that $M \succeq 0$. ■

Appendix B

Multibody dynamics of a multicopter

This appendix discusses the multibody dynamics of a multicopter with some attached propellers. Since torques emanating from the rotating propeller disc may have non-negligible effects on the complete dynamics of the system the interactions between the rotating bodies are examined. As it will be shown, most of these effects cancel each other out if the propellers are aligned symmetrically, however, if a propeller is lost, then the equilibrium of these effects gets disrupted. It is hypothesized that the disrupted balance in propeller effects may carry beneficial information for Fault Detection and Diagnosis (FDD) observers.

This chapter starts with the discussion of relative angular momenta of the rigid body (Appendix B.1) and a single propeller attached to it (Appendix B.2). Following, forces and moments acting on the rigid body (Appendix B.3) and on a propeller (Appendix B.4) are examined.Finally, the found effects are lumped into a single dynamic model in Appendix B.5.

In this chapter the subscripts and superscripts will have the following meaning:

$$\bullet_{xy}^{z}$$

• in frame y with respect to frame x expressed in frame z.

B.1 Relative angular momentum of the rigid body

Firstly, the relative angular momentum $h_{\mathcal{BB}}^{\mathcal{A}}$ of the multicopter body - excluding the propellers - will be derived in the Center of Gravity (CG) of the body, i.e., the origin of the body-fixed reference frame $\mathcal{B}$. The mass of the body is denoted by m_b and its volume is V_b. The rotation matrix from $\mathcal{B}$ to $\mathcal{A}$ is denoted by $R_{\mathcal{AB}}$.

$$h_{\mathcal{AB}}^{\mathcal{A}} = \int_{V_b} \rho_{\mathcal{B}}^{\mathcal{A}} \times \dot{\rho}_{\mathcal{B}}^{\mathcal{A}} \mathrm{d}m$$

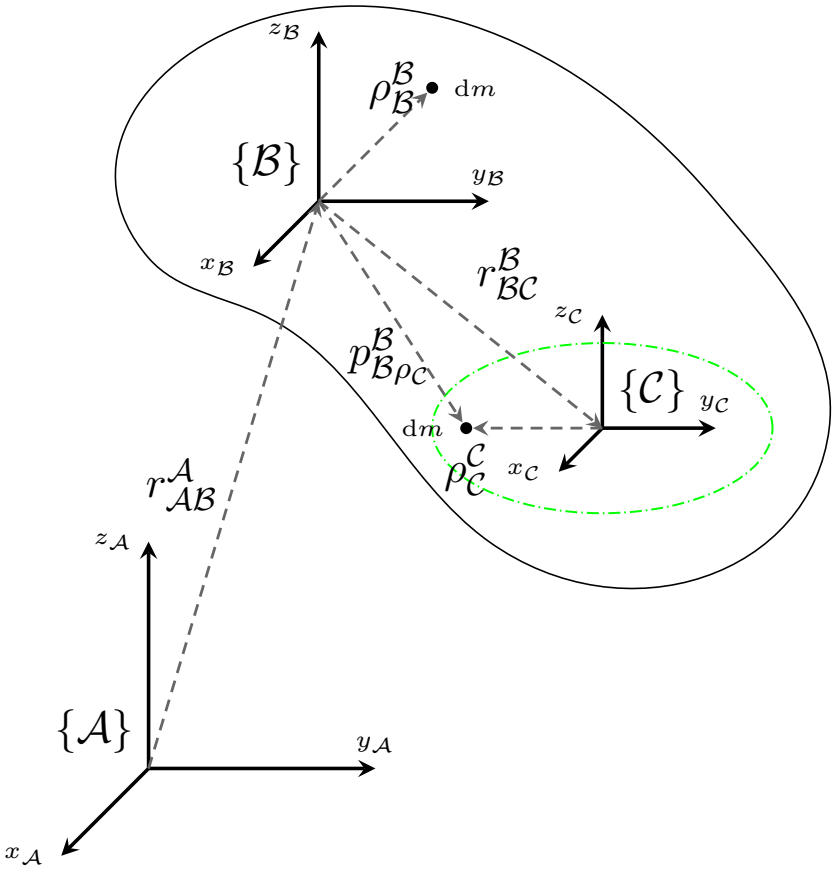

Figure B.1: Multibody diagram of a multicopter body and an attached propeller

According to Figure B.1, one has the following relationships

$$\begin{aligned}\rho^{\mathcal{A}}_{\mathcal{B}} &= R_{\mathcal{AB}}\rho^{\mathcal{B}}_{\mathcal{B}}\\ \dot{\rho}^{\mathcal{A}}_{b} &= R_{\mathcal{AB}}\left[\Omega\right]\rho^{\mathcal{B}}_{\mathcal{B}} + R_{\mathcal{AB}}\dot{\rho}^{\mathcal{B}}_{\mathcal{B}}\end{aligned}$$

where Ω is the angular velocity of $\mathcal{B}$ with respect to $\mathcal{A}$.

Definition B.1
The operator $[\bullet] : \mathbb{R}^n \to \mathfrak{so}(n)$ *maps an n-dimensional vector to its respective skew-symmetric matrix in* $\mathfrak{so}(n)$.

Assumption B.1
Since the multicopter body is assumed to be rigid, the position of an elementary mass of the body with respect to the body frame is constant, meaning that $\dot{\rho}^{\mathcal{B}}_{\mathcal{B}} = 0$. *Thus, the following is obtained*

With Assumption B.1 in mind, the following can be written

$$h^{\mathcal{A}}_{\mathcal{AB}} = \int_{V_b} R_{\mathcal{AB}} \rho^{\mathcal{B}}_{\mathcal{B}} \times R_{\mathcal{AB}} \left[\Omega\right] \rho^{\mathcal{B}}_{\mathcal{B}} \mathrm{d}m$$

Since the volume integral does not depend on the relative attitude of the rigid body in the inertial frame, the rotation matrix can be pulled out of the integral and the following expression can be derived

$$h^{\mathcal{A}}_{\mathcal{AB}} = R_{\mathcal{AB}} \int_{V_b} \rho^{\mathcal{B}}_{\mathcal{B}} \times \left[\Omega\right] \rho^{\mathcal{B}}_{\mathcal{B}} \mathrm{d}m$$

Due to the linearity of the integral, the above equation can be expanded as

$$\begin{aligned}
&h^{\mathcal{A}}_{\mathcal{AB}} \\
&\quad = R_{\mathcal{AB}} \int_{V_b} \rho^{\mathcal{B}}_{\mathcal{B}} \times \Omega \times \rho^{\mathcal{B}}_{\mathcal{B}} \mathrm{d}m = \\
&\quad = R_{\mathcal{AB}} - \int_{V_b} \left[\rho^{\mathcal{B}}_{\mathcal{B}}\right]^2 \mathrm{d}m \Omega = \\
&\quad = R_{\mathcal{AB}} \underbrace{\begin{bmatrix} \int_{V_b} \rho_2^2 + \rho_3^2 \mathrm{d}m & -\int_{V_b} \rho_1 \rho_2 \mathrm{d}m & -\int_{V_b} \rho_1 \rho_3 \mathrm{d}m \\ -\int_{V_b} \rho_2 \rho_1 \mathrm{d}m & \int_{V_b} \rho_1^2 + \rho_3^2 \mathrm{d}m & -\int_{V_b} \rho_2 \rho_3 \mathrm{d}m \\ -\int_{V_b} \rho_3 \rho_1 \mathrm{d}m & -\int_{V_b} \rho_3 \rho_2 \mathrm{d}m & \int_{V_b} \rho_1^2 + \rho_2^2 \mathrm{d}m \end{bmatrix}}_{J_b^{\mathcal{B}}} \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix} = \\
&\quad = R_{\mathcal{AB}} J_b^{\mathcal{B}} \Omega
\end{aligned} \tag{B.1}$$

with $\rho^{\mathcal{B}}_{\mathcal{B}} = [\rho_1, \rho_2, \rho_3]^T$, $\Omega = [\Omega_1, \Omega_2, \Omega_3]^T$ and $J_b^{\mathcal{B}}$ is the inertia tensor of the rigid body with respect to the body-fixed frame of reference. Differentiating the multicopter's relative angular momentum with respect to time it is found that the sum of inertial torques $\sum_i \tau_i^{\mathcal{A}}$ equals $\dot{h}^{\mathcal{A}}_{\mathcal{AB}}$, that is

$$\dot{h}^{\mathcal{A}}_{\mathcal{AB}} = R_{\mathcal{AB}} \left[\Omega\right] J_b^{\mathcal{B}} + R_{\mathcal{AB}} J_b^{\mathcal{B}} \dot{\Omega} = \sum_i \tau_i^{\mathcal{A}}$$

where the rotation matrix dynamics $\dot{R} = R\left[\Omega\right]$ was used [79].

Multiplying by $R^T_{\mathcal{AB}}$ yields the following, well known equation for a single body's attitude kinematics

$$J_b^{\mathcal{B}} \dot{\Omega} + \left[\Omega\right] J_b^{\mathcal{B}} \Omega = \sum_i \tau_i^{\mathcal{B}} \tag{B.2}$$

since the rotation matrix is orthogonal, i.e. $R^T R = \mathcal{I}$.

B.2 Relative angular momentum of the propeller

As for a single propeller positioned $r^{\mathcal{B}}_{\mathcal{BC}}$ from the rigid body CG, its relative angular momentum in the inertial frame, with respect to the origin of the body-fixed frame of reference, can be written as

$$h^{\mathcal{A}}_{\mathcal{AC}} = \int_{V_p} r^{\mathcal{A}}_{bm} \times \dot{r}^{\mathcal{A}}_{bm} \mathrm{d}m \tag{B.3}$$

where V_p denotes the volume of the propeller and, according to Figure B.1, $r_{bm}^{\mathcal{A}}$ is defined as

$$\begin{aligned} r_{bm}^{\mathcal{A}} &= R_{\mathcal{AB}} r_{bm}^{\mathcal{B}} = R_{\mathcal{AB}} \left(r_{\mathcal{BC}}^{\mathcal{B}} + R_{\mathcal{BC}} \rho_{\mathcal{C}}^{\mathcal{C}} \right) \\ \dot{r}_{bm}^{\mathcal{A}} &= R_{\mathcal{AB}} \left[\Omega \right] r_{bm}^{\mathcal{B}} + R_{\mathcal{AB}} \dot{r}_{bm}^{\mathcal{B}} \\ &= R_{\mathcal{AB}} \left[\Omega \right] \left(r_{\mathcal{BC}}^{\mathcal{B}} + R_{\mathcal{BC}} \rho_{\mathcal{C}}^{\mathcal{C}} \right) + R_{\mathcal{AB}} \left(\dot{r}_{\mathcal{BC}}^{\mathcal{B}} + R_{\mathcal{BC}} \left[\omega \right] \rho_{\mathcal{C}}^{\mathcal{C}} + R_{\mathcal{BC}} \dot{\rho}_{p}^{\mathcal{C}} \right) \end{aligned}$$

$R_{\mathcal{BC}}$ is the rotation matrix from the propeller frame $\mathcal{C}$ which is attached to the propeller's mount point on the rigid body. ω is the angular speed of $\mathcal{C}$ with respect to $\mathcal{B}$, i.e., the rotational speed of the propeller.

Assumption B.1
$\dot{r}_{\mathcal{BC}}^{\mathcal{B}} = 0$, as the propeller is assumed to be at a constant location in the body frame, due to the multicopter's body being rigid. Furthermore, $\dot{\rho}_{p}^{\mathcal{C}} = 0$ as well, since the all elementary masses within the propeller, modeled as a disk, are at a constant position with respect to the CG of the propeller, as it is also a rigid body.

Remark
Due to the propeller being modeled as a disk, the inertia tensor of an infinitely thin disk is used to describe the dynamics of the propeller. According to [79], the said inertia tensor has the form

$$J_p^{\mathcal{C}} = \begin{bmatrix} \frac{1}{4} m_p r_p^2 & 0 & 0 \\ 0 & \frac{1}{4} m_p r_p^2 & 0 \\ 0 & 0 & \frac{1}{2} m_p r_p^2 \end{bmatrix}$$

Assumption B.2
It is assumed that the propeller cannot tilt with respect to the body frame.

Corollary B.0.1
Due to Assumption B.2 the rotational speed of the propeller frame, with respect to the body frame, will have the form

$$\omega = [0, 0, \omega_z]^T$$

and the rotation matrix $R_{\mathcal{BC}}$ will describe a single rotation around $z_{\mathcal{C}}$ by an arbitrary angle ϑ whose cosine will be denoted by c_ϑ and its sine by s_ϑ. Therefore, $R_{\mathcal{BC}}$ will always have the form

$$R_{\mathcal{BC}} = \begin{bmatrix} c_\vartheta & s_\vartheta & 0 \\ -s_\vartheta & c_\vartheta & 0 \\ 0 & 0 & 1 \end{bmatrix} \in SO(3)$$

Hence, the following holds

$$R_{\mathcal{BC}} \left[\omega \right] = \left[\omega \right] R_{\mathcal{BC}}$$

Proof. It is proven in [79] that the following relationship holds for any pair of $R \in SO(3)$ and $x \in \mathbb{R}^3$:

$$R^T \left[x \right] R = \left[Rx \right] \tag{B.4}$$

Due to the special structure of $R_{\mathcal{BC}} \in SO(3)$ and $\omega \in \mathbb{R}^3$ shown in Corollary B.0.1 it holds that $R_{\mathcal{BC}} \omega = \omega$, therefore, (B.4) can be written as

$$R_{\mathcal{BC}}^T \left[\omega \right] R_{\mathcal{BC}} = \left[R_{\mathcal{BC}} \omega \right] = \left[\omega \right]$$

where the orthogonal property of R was utilized. Multiplying both sides by $R_{\mathcal{BC}}$, due to the identity property of $SO(3)$ matrices, the proof is completed. ■

Since the attitude of the propeller with respect to the inertial frame does not affect its volume integral and according to Assumption B.1, (B.3) can be written as

$$\begin{aligned} h^{\mathcal{A}}_{\mathcal{AC}} &= R_{\mathcal{AB}} \int_{V_p} \left(r^{\mathcal{B}}_{\mathcal{BC}} + R_{\mathcal{BC}}\rho^{\mathcal{B}}_{\mathcal{C}}\right) \times \left([\Omega]^r_{\mathcal{BC}}\,\mathcal{B} + [\Omega]\,R_{\mathcal{BC}}\rho^{\mathcal{C}}_{\mathcal{C}} + R_{\mathcal{BC}}\,[\omega]\,\rho^{\mathcal{C}}_{\mathcal{C}}\right) \mathrm{d}m = \\ &= R_{\mathcal{AB}}\Bigg(\int_{V_p} r^{\mathcal{B}}_{\mathcal{BC}} \times [\Omega]^r_{\mathcal{BC}}\,\mathcal{B}\mathrm{d}m + \int_{V_p} r^{\mathcal{B}}_{\mathcal{BC}} \times [\Omega]\,R_{\mathcal{BC}}\rho^{\mathcal{C}}_{\mathcal{C}}\mathrm{d}m + \int_{V_p} r^{\mathcal{B}}_{\mathcal{BC}} \times [\Omega]\,R_{\mathcal{BC}}\,[\omega]\,\rho^{\mathcal{C}}_{\mathcal{C}}\mathrm{d}m + \\ &+ \int_{V_p} R_{\mathcal{BC}}\rho^{\mathcal{C}}_{\mathcal{C}} \times [\Omega]\,r^{\mathcal{B}}_{\mathcal{BC}}\mathrm{d}m + \int_{V_p} R_{\mathcal{BC}}\rho^{\mathcal{C}}_{\mathcal{C}} \times [\Omega]\,R_{\mathcal{BC}}\rho^{\mathcal{C}}_{\mathcal{C}}\mathrm{d}m + \int_{V_p} R_{\mathcal{BC}}\rho^{\mathcal{C}}_{\mathcal{C}} \times R_{\mathcal{BC}}\,[\omega]\,\rho^{\mathcal{C}}_{\mathcal{C}}\mathrm{d}m \Bigg) \end{aligned} \tag{B.5}$$

Integrals containing both $r^{\mathcal{B}}_{\mathcal{BC}}$ and $\rho^{\mathcal{C}}_{\mathcal{C}}$ can be manipulated until the integral is reduced to

$$\int_{V_p} \rho^{\mathcal{C}}_{\mathcal{C}}\mathrm{d}m$$

which is zero by definition of the CG.

Remark
Let $\rho^{\mathcal{B}}_{\mathcal{C}} = R_{\mathcal{BC}}\rho^{\mathcal{C}}_{\mathcal{C}}$ which, using the structure of $R_{\mathcal{BC}}$ from Corollary B.0.1 and defining $\rho^{\mathcal{C}}_{\mathcal{C}} = [\rho_1, \rho_2, \rho_3]^T$, can be expressed as

$$\rho^{\mathcal{B}}_{\mathcal{C}} = \begin{bmatrix} c_\vartheta\rho_1 + s_\vartheta\rho_2 \\ -s_\vartheta\rho_1 + c_\vartheta\rho_2 \\ \rho_3 \end{bmatrix}$$

Using the structure of the inertia tensor shown in (B.1) *and the inertia of a disk shown in Appendix B.2, it can be shown that $R_{\mathcal{BC}}$, due to its special structure, does not change the inertia of the propeller disk, regardless of the reference point being in the origin of $\mathcal{B}$ or $\mathcal{C}$.*

Firstly, note the fact that

$$\int_{V_p} \rho_1^2 dm = \int_{V_p} \rho_2^2 dm$$

holds. Then, note that

$$\int_{V_p} \rho_1\rho_2 dm = 0$$

also holds. Thus it can be shown that

$$\begin{aligned} \int_{V_p} \rho_1^2 dm &= \int_{V_p} \left(c_\vartheta\rho_1 + s_\vartheta\rho_2\right)^2 dm \\ &= \int_{V_p} c_\vartheta^2\rho_1^2 + 2c_\vartheta s_\vartheta\rho_1\rho_2 + s_\vartheta^2\rho_2^2 dm \\ &= c_\vartheta^2 \int_{V_p} \rho_1^2 dm + 2c_\vartheta s_\vartheta \underbrace{\int_{V_p} \rho_1\rho_2 dm}_{0} + s_\vartheta^2 \int_{V_p} \rho_1^2 dm \\ &= \underbrace{\left(c_\vartheta^2 + s_\vartheta^2\right)}_{1} \int_{V_p} \rho_1^2 dm \end{aligned}$$

Thus, using Corollary B.0.1 and Appendix B.2, (B.5) can be rewritten as

$$\begin{aligned} h_{\mathcal{AC}}^{\mathcal{A}} &= R_{\mathcal{AB}} \left(r_{\mathcal{BC}}^{\mathcal{B}} \times [\Omega]_{\mathcal{BC}}^{r} \mathcal{B} \int_{V_p} \mathrm{d}m + \int_{V_p} \rho_{\mathcal{C}}^{\mathcal{C}} \times [\Omega] \rho_{\mathcal{C}}^{\mathcal{C}} \mathrm{d}m + \int_{V_p} \rho_{\mathcal{C}}^{\mathcal{C}} \times [\omega] \rho_{\mathcal{C}}^{\mathcal{C}} \mathrm{d}m \right) \\ &= R_{\mathcal{AB}} \left(J_p^S \Omega + J_p^{\mathcal{C}} \Omega + J_p^{\mathcal{C}} \omega \right) \end{aligned}$$

where

$$J_p^S = m_p \left(\left(r_{\mathcal{BC}}^{\mathcal{B}} \right)^T r_{\mathcal{BC}}^{\mathcal{B}} \mathcal{I} - {}_{\mathcal{BC}}^{r} \mathcal{B} \left(r_{\mathcal{BC}}^{\mathcal{B}} \right)^T \right)$$

with $\mathcal{I} \in \mathbb{R}^3$ being an identity matrix. This equation describes Steiner's theorem which introduces an apparent inertia J_p^S of a point mass with respect to a rotation axis not coincident with the point mass [79]. In this case, the propeller can be described as a mass with m_p rotating with the body frame, $r_{\mathcal{BC}}^{\mathcal{B}}$ away from its origin. $J_p^{\mathcal{C}}$ is the propeller disk inertia tensor with respect to the propeller frame.

Differentiating the relative angular momentum of the propeller, one obtains

$$\dot{h}_{\mathcal{AC}}^{\mathcal{A}} = R_{\mathcal{AB}} [\Omega] \left(J_p^S \Omega + J_p^{\mathcal{C}} \Omega + J_p^{\mathcal{C}} \omega \right) + R_{\mathcal{AB}} \left(J_p^S \dot{\Omega} + J_p^{\mathcal{C}} \dot{\Omega} + J_p^{\mathcal{C}} \dot{\omega} \right)$$

where there are three terms, namely

- $J_p^S \dot{\Omega} + [\Omega] J_p^S \Omega$ is the torque required to rotate the body frame (rotating by Ω) containing a rigid body with mass m_p at distance $r_{\mathcal{BC}}^{\mathcal{B}}$ from its origin.
- $J_p^{\mathcal{C}} \dot{\Omega} + [\Omega] J_p^{\mathcal{C}} \Omega$ is the torque required to rotate the body frame (rotating by Ω) as if the CG of the propeller was in its origin.
- $J_p^{\mathcal{C}} \dot{\Omega} + [\Omega] J_p^{\mathcal{C}} \omega$ is the torque required to rotate the propeller frame (rotating by Ω due to being attached to the body frame) around its z axis.

These results mean that the torque applied at the actuation points, i.e., the origin of the propeller-fixed frame, is distributed to rotate the propeller around $z_{\mathcal{A}}$ axis and the body around $z_{\mathcal{B}}$, however, with an increased inertia due to the mass and inertia of the propeller. Thus, the first two terms in the above list can be lumped together with the derived expression in (B.2) to obtain the inertia of the whole body with the propellers attached. Therefore it is found that

$$\begin{aligned} \dot{h}_{\mathcal{AB}}^{\mathcal{B}} + \dot{h}_{\mathcal{AC}}^{\mathcal{B}} &= \sum_i \tau_i^{\mathcal{B}} \\ \left(J_b^{\mathcal{B}} + J_p^S + J_p^{\mathcal{C}} \right) \dot{\Omega} + [\Omega] \left(J_b^{\mathcal{B}} + J_p^S + J_p^{\mathcal{C}} \right) \Omega + & \\ + J_p^{\mathcal{C}} \dot{\omega} + [\Omega] J_p^{\mathcal{C}} \omega &= \sum_i \tau_i^{\mathcal{B}} \end{aligned}$$

stating that all torques applied with respect to the body frame need to rotate two rigid bodies accordingly. It is straightforward to extend the given relationship to multiple propellers, as each propeller adds inertia to the multicopter body similarly.

B.3 Forces on the rigid body

In order to derive the sum of forces acting on the multicopter, consider body frame $\mathcal{B}$ as a frame not attached, but aligned with another 'body frame' $\mathcal{B}'$ which is attached to the rigid body. Thus, at every instant, the rigid body—with $\mathcal{B}'$ attached—wants to move away from $\mathcal{B}$. Hence, the position of an elementary mass within the rigid body in the body frame can be written as

$$r_\rho^\mathcal{B} = r_{bb'}^\mathcal{B} + R_{\mathcal{B}\mathcal{B}'}\rho_\mathcal{B}^{\mathcal{B}'} \tag{B.6}$$

where $r_{bb'}^\mathcal{B}$ is the relative position of the origin of $\mathcal{B}'$ from $\mathcal{B}$, which is 0 at every instant, since it is assumed that they are aligned and coincident. Therefore, the rotation matrix from $\mathcal{B}'$ to $\mathcal{B}$ ($R_{\mathcal{B}\mathcal{B}'}$) is the identity matrix. $\rho_\mathcal{B}^{\mathcal{B}'}$ is the vector pointing from the origin of $\mathcal{B}'$ to the elementary mass.

Although $r_{bb'}^\mathcal{B}$ is always 0, and $R_{\mathcal{B}\mathcal{B}'}$ is always the identity matrix, the linear and angular velocity of $\mathcal{B}'$ with respect to $\mathcal{B}$ is not necessarily zero. Differentiating (B.6) with respect to time it is obtained that

$$\begin{aligned} \dot{r}_\rho^\mathcal{B} &= \dot{r}_{\mathcal{B}\mathcal{B}'}^\mathcal{B} + R_{\mathcal{B}\mathcal{B}'}\left[\Omega\right]\rho_\mathcal{B}^{\mathcal{B}'} + R_{\mathcal{B}\mathcal{B}'}^{\mathcal{B}'}\dot{\rho}_\mathcal{B}^{\mathcal{B}'} = \\ &= v_b^\mathcal{B} + \left[\Omega\right] r_\rho^\mathcal{B} \end{aligned} \tag{B.7}$$

since the velocity of $\mathcal{B}'$ with respect to $\mathcal{B}$ is the body-fixed velocity ($v_b^\mathcal{B}$) of the multicopter, $R_{\mathcal{B}\mathcal{B}'} = \mathcal{I}$, and $\dot{\rho}_\mathcal{B}^{\mathcal{B}'} = 0$ as the body is assumed to be rigid.

Differentiating (B.7) with respect to time, the acceleration of $\mathcal{B}'$ in frame $\mathcal{B}$ is obtained as

$$\begin{aligned} \ddot{r}_\rho^\mathcal{B} &= \dot{v}_b^\mathcal{B} + \left[\Omega\right] r_\rho^\mathcal{B} = \\ &= \dot{v}_b^\mathcal{B} + \left[\dot{\Omega}\right] \underbrace{r_{bb'}^\mathcal{B}}_{0} + \left[\dot{\Omega}\right] R_{\mathcal{B}\mathcal{B}'}\rho_\mathcal{B}^{\mathcal{B}'} + \left[\Omega\right] dotr_{\mathcal{B}\mathcal{B}'}^\mathcal{B} + \left[\Omega\right] R_{\mathcal{B}\mathcal{B}'}\left[\Omega\right]\rho_\mathcal{B}^{\mathcal{B}'} = \\ &= \dot{v}_b^\mathcal{B} + \left[\Omega\right] v_b^\mathcal{B} + \left(\left[\dot{\Omega}\right] + \left[\Omega\right]^2\right)\rho_\mathcal{B}^{\mathcal{B}'} \end{aligned} \tag{B.8}$$

Multiplying (B.8) by an elementary mass $\mathrm{d}m$ and integrating it over the volume of the rigid body, it is found that terms with $\rho_\mathcal{B}^{\mathcal{B}'}\mathrm{d}m$ vanish due to $\mathcal{B}'$ being fixed in the CG of the multicopter body. Thus, it is found that all external forces $f_{ext}^\mathcal{B}$ acting on the body in body frame $\mathcal{B}$ will accelerate the body as

$$f_b^\mathcal{B} = m_b\left(\dot{v}_b^\mathcal{B} + \left[\Omega\right] v_b^\mathcal{B}\right)$$

where $v_b^\mathcal{B}$ is the velocity of the multicopter expressed in the body frame, whereas Ω is the angular velocity of the body frame. Therefore, the body-twist can be defined as

$$\mathcal{T}^\mathcal{B} = \begin{bmatrix} \Omega \\ v_b^\mathcal{B} \end{bmatrix}$$

B.4 Forces on a propeller

Using the same principles as in Appendix B.3, take an elementary mass particle of a propeller in propeller frame $\mathcal{C}$. The origin of $\mathcal{C}$ is located at $r_{\mathcal{B}'\mathcal{C}}^{\mathcal{B}'} = r_{\mathcal{B}\mathcal{C}}^\mathcal{B}$ from the origin

of $\mathcal{B}'$ and $\mathcal{B}$ respectively. The position of this mass particle within the propeller frame is denoted by $\rho_{\mathcal{C}}^{\mathcal{C}}$. Then, the elementary mass in the propeller is located at

$$r_{\rho}^{\mathcal{B}} = r_{bb'}^{\mathcal{B}} + R_{\mathcal{B}\mathcal{B}'} r_{\mathcal{B}'\mathcal{C}}^{\mathcal{B}'} + R_{\mathcal{B}\mathcal{B}'} R_{\mathcal{B}'\mathcal{C}} \rho_{\mathcal{C}}^{\mathcal{C}} \tag{B.9}$$

in the body frame $\mathcal{B}$. Differentiating (B.9) with respect to time it is found that

$$\dot{r}_{\rho}^{\mathcal{B}} = \underbrace{\dot{r}_{\mathcal{B}\mathcal{B}'}^{\mathcal{B}}}_{v_b^{\mathcal{B}}} + R_{\mathcal{B}\mathcal{B}'}\left[\Omega\right] r_{\mathcal{B}'\mathcal{C}}^{\mathcal{B}'} + R_{\mathcal{B}\mathcal{B}'} \underbrace{\dot{r}_{\mathcal{B}'\mathcal{C}}^{\mathcal{B}'}}_{0} + \underbrace{\left(R_{\mathcal{B}\mathcal{B}'}\left[\Omega\right] R_{\mathcal{B}'\mathcal{C}} + R_{\mathcal{B}\mathcal{B}'} R_{\mathcal{B}'\mathcal{C}}\left[\omega\right]\right)}_{(\ldots)} \rho_{\mathcal{C}}^{\mathcal{C}} + R_{\mathcal{B}\mathcal{B}'} R_{\mathcal{B}'\mathcal{C}} \underbrace{\dot{\rho}_{\mathcal{C}}^{\mathcal{C}}}_{0} \tag{B.10}$$

Differentiating (B.10) the acceleration of an elementary mass of a propeller, with respect to the body frame, is found to be

$$\ddot{r}_{\rho}^{\mathcal{B}} = \dot{v}_b^{\mathcal{B}} + \left[\dot{\Omega}\right] r_{\mathcal{B}\mathcal{C}}^{\mathcal{B}} + \left[\Omega\right] \dot{r}_{\mathcal{B}'\mathcal{C}}^{\mathcal{B}} + (\cdots)\rho_{\mathcal{C}}^{\mathcal{C}} + (\cdots) \underbrace{\dot{\rho}_{\mathcal{C}}^{\mathcal{C}}}_{0} \tag{B.11}$$

Since the velocity of the origin of $\mathcal{C}$ with respect to the body frame is seen as the sum of the linear and angular speeds around the origin of $\mathcal{B}$, the following can be written

$$\dot{r}_{\mathcal{B}'\mathcal{C}}^{\mathcal{B}} = v_b^{\mathcal{B}} + \left[\Omega\right] r_{\mathcal{B}'\mathcal{C}}^{\mathcal{B}}$$

Thereafter, multiplying (B.11) by the elementary mass and integrating over the volume of the propeller, one obtains that due to the origin of $\mathcal{C}$ is fixed in the CG of the propeller the terms with $\rho_{\mathcal{C}}^{\mathcal{C}} \mathrm{d}m$ vanish and the final equation describing the linear acceleration of a propeller in the body frame is given as

$$\mathfrak{f}_b^{\mathcal{B}} = m_p \left(\dot{v}_b^{\mathcal{B}} + \left[\Omega\right] v_b^{\mathcal{B}} + \underbrace{\left[\dot{\Omega}\right] r_{\mathcal{B}\mathcal{C}}^{\mathcal{B}}}_{\text{angular acc.}} + \underbrace{\left[\Omega\right]^2 r_{\mathcal{B}\mathcal{C}}^{\mathcal{B}}}_{\text{centripetal acc.}} \right) \tag{B.12}$$

B.5 External wrench

In [1], it is stated, that as the rigid body is rotating around its $z_{\mathcal{B}}$ axis, a torque is generated in the form $\tau_r^{\mathcal{B}} = d_r[0, 0, \Omega_z]^T$ where d_r is a drag coefficient and Ω_z is the yaw rate of the body. Thus the external wrench acting on the rigid body is consists of this counter torque on the yaw channel and the drag force modeled as

$$\mathfrak{f}_d^{\mathcal{B}} = C_d \left(v_b^{\mathcal{B}}\right)^2$$

where $C_d \in \mathbb{R}^{3\times3}$ is the drag coefficient matrix. Thus, the body wrench is

$$\mathcal{W}_b^{\mathcal{B}} = \begin{bmatrix} \tau_r^{\mathcal{B}} \\ \mathfrak{f}_d^{\mathcal{B}} \end{bmatrix}$$

The external wrench applied on the propeller in the propeller frame consists of the torque generated due to drag ($\tau_a = d_a\omega^2$) and blade flapping ($\tau_{fl} = d_f\left(v_b^{\mathcal{B}}, \omega\right)$) and the thrust ($\mathfrak{f}_t = \kappa_t\tau_m$), such that

$$\mathcal{W}_p^{\mathcal{C}} = \begin{bmatrix} \tau_a^{\mathcal{C}} + \tau_f^{\mathcal{C}} \\ f_t^{\mathcal{C}} \end{bmatrix}$$

B.6 Twist-wrench formulation

The aim is to obtain a model of the form

$$M\dot{\mathcal{T}} + C(\mathcal{T}) + G(q) = Q(q, \mathcal{T}, \mathcal{W}(u))$$

where M is the mass matrix containing masses and inertia tensors, $\mathcal{T}$ is called a twist which is a vector containing linear velocities and rotational velocities, q is the configuration (attitude and position) of the rigid body, $C(\mathcal{T})$ is the Coriolis matrix, $G(q)$ is a matrix containing potential terms, and $Q(q, \mathcal{T}, u)$ is the matrix with input u and wrench $\mathcal{W}$. The input to the system is the torque ($\tau_m = \kappa_q i$) generated by the Brushless DC (BLDC) motors connecting the multicopter body and the propeller.

$$\begin{bmatrix} \left(J_b^{\mathcal{B}} + \sum_i \left(J_p^S + J_p^{\mathcal{C}}\right)\right)\dot{\Omega} + [\Omega]\left(J_b^{\mathcal{B}} + \sum_i \left(J_p^S + J_p^{\mathcal{C}}\right)\right)\Omega + \sum_i J_p^{\mathcal{C}}\dot{\omega} + \sum_i [\Omega] J_p^{\mathcal{C}}\omega \\ \left(\sum_i m_p + m_b\right)\left(\dot{v}_b^{\mathcal{B}} + [\Omega] v_b^{\mathcal{B}}\right) + \sum_i m_p \left(\left[\dot{\Omega}\right] r_{\mathcal{BC}}^{\mathcal{B}} + [\Omega]^2 r_{\mathcal{BC}}^{\mathcal{B}}\right) \end{bmatrix} = \\ (G(q))^{\mathcal{B}} + \mathcal{W}_b^{\mathcal{B}} + \mathrm{Ad}_{T_{\mathcal{CB}}}^T \mathcal{W}_p^{\mathcal{C}} + \mathcal{W}_\Delta^{\mathcal{B}} \tag{B.13}$$

where $\mathrm{Ad}_{T_{\mathcal{CB}}}^T$ is the transpose of the adjoint transformation that maps a wrench acting in the propeller frame $\mathcal{C}$ to the body frame $\mathcal{B}$.

Definition B.1 (Adjoint transformation [79])
Suppose there is a frame y with its origin being r_{xy} from another frame x. The attitude of y with respect to x is described by a rotation matrix R_{xy}. A wrench $\mathcal{W}^y$ acting in the y frame is transformed to x via the adjoint map

$$Ad_{T_{yx}}^T = \begin{bmatrix} R_{yx} & 0 \\ [r_{yx}]\, R_{yx} & R_{yx} \end{bmatrix}^T \in SE(3)$$

such that

$$\mathcal{W}_x = Ad_{T_{yx}}^T \mathcal{W}_y$$

In (B.13) the term $\mathcal{W}_\Delta$ denotes uncertain external effects and $G(q)$ denotes the effect of gravity on the multicopter.

Bibliography

[1] Mark W Mueller and Raffaello D'Andrea. Stability and control of a quadrocopter despite the complete loss of one, two, or three propellers. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 45–52. IEEE, 2014.

[2] Sihao Sun, Leon Sijbers, Xuerui Wang, and Coen de Visser. High-Speed Flight of Quadrotor Despite Loss of Single Rotor. *IEEE Robotics and Automation Letters*, 3(4):3201–3207, October 2018.

[3] Rolf Isermann. *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*. Springer, Berlin ; New York, 2006. OCLC: ocm61703226.

[4] Raghu Venkataraman, Péter Bauer, Peter Seiler, and Bálint Vanek. Comparison of fault detection and isolation methods for a small unmanned aircraft. *Control Engineering Practice*, 84:365–376, March 2019.

[5] Ewoud JJ Smeur, Qiping Chu, and Guido CHE de Croon. Adaptive incremental nonlinear dynamic inversion for attitude control of micro air vehicles. *Journal of Guidance, Control, and Dynamics*, 38(12):450–461, 2015.

[6] Remus C Avram, Xiaodong Zhang, and Jonathan Muse. Nonlinear adaptive fault-tolerant quadrotor altitude and attitude tracking with multiple actuator faults. *IEEE Transactions on Control Systems Technology*, 26(2):701–707, 2018.

[7] Farhad Pourpanah, Bin Zhang, Rui Ma, and Qi Hao. Anomaly Detection and Condition Monitoring of UAV Motors and Propellers. In *2018 IEEE SENSORS*, pages 1–4, New Delhi, India, October 2018. IEEE.

[8] Huimin Lu, Yujie Li, Shenglin Mu, Dong Wang, Hyoungseop Kim, and Seiichi Serikawa. Motor anomaly detection for unmanned aerial vehicles using reinforcement learning. *IEEE internet of things journal*, 5(4):2315–2322, 2018.

[9] Remus C Avram, Xiaodong Zhang, and Jonathan Muse. Quadrotor actuator fault diagnosis and accommodation using nonlinear adaptive estimators. *IEEE Transactions on Control Systems Technology*, 25(6):2219–2226, 2017.

[10] G. Ortiz-Torres, P. Castillo, and J. Reyes-Reyes. An Actuator Fault Tolerant Control for VTOL vehicles using Fault Estimation Observers: Practical validation *. In *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1054–1062, Dallas, TX, USA, June 2018. IEEE.

[11] Ngoc Phi Nguyen and Sung Kyung Hong. Sliding Mode Thau Observer for Actuator Fault Diagnosis of Quadcopter UAVs. *Applied Sciences*, 8(10):1893, October 2018.

[12] Robert Mahony, Tarek Hamel, and Jean-Michel Pflimlin. Nonlinear complementary filters on the special orthogonal group. *IEEE Transactions on automatic control*, 53(5):1203–1217, 2008.

[13] Richard M Murray. *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[14] Robert Mahony, Vijay Kumar, and Peter Corke. Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor. *IEEE Robotics & Automation Magazine*, 19(3):20–32, September 2012.

[15] Gordon J Leishman. *Principles of helicopter aerodynamics with CD extra*. Cambridge university press, 2006.

[16] Peng Lu and Erik-Jan van Kampen. Active fault-tolerant control for quadrotors subjected to a complete rotor failure. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4698–4703, Hamburg, Germany, September 2015. IEEE.

[17] Sihao Sun, Coen C de Visser, and Qiping Chu. Quadrotor gray-box model identification from high-speed flight data. *Journal of Aircraft*, pages 1–17, 2018.

[18] Moses Bangura and Robert Mahony. Nonlinear Dynamic Modeling for High Performance Control of a Quadrotor. *Australian Robotics and Automation Association*, 2012.

[19] Tommaso Bresciani. Modelling, identification and control of a quadrotor helicopter. *Master Thesis, Lund University, Sweden*, 2008.

[20] Amit Aherwar and Saifullah Khalid. VIBRATION ANALYSIS TECHNIQUES FOR GEARBOX DIAGNOSTIC: A REVIEW. page 10, 2012.

[21] Adam Bondyra, Przemysław Gasior, Stanisław Gardecki, and Andrzej Kasiński. Fault diagnosis and condition monitoring of uav rotor using signal processing. In *2017 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, pages 233–238. IEEE, 2017.

[22] Yan Jiang, Zhao Zhiyao, Liu Haoxiang, and Quan Quan. Fault detection and identification for quadrotor based on airframe vibration signals: A data-driven method. In *2015 34th Chinese Control Conference (CCC)*, pages 6356–6361, Hangzhou, China, July 2015. IEEE.

[23] Peter Caselitz and Jochen Giebhardt. Rotor Condition Monitoring for Improved Operational Safety of Offshore Wind Energy Converters. *Journal of Solar Energy Engineering*, 127(2):253, 2005.

[24] Zhongming Ye, Bin Wu, and A. Sadeghian. Current signature analysis of induction motor mechanical faults by wavelet packet decomposition. *IEEE Transactions on Industrial Electronics*, 50(6):1217–1228, December 2003.

[25] Satish Rajagopalan, José A. Restrepo, José M. Aller, Thomas G. Habetler, and Ronald G. Harley. Nonstationary Motor Fault Detection Using Recent Quadratic Time–Frequency Representations. *IEEE Transactions on Industry Applications*, 44(3):735–744, 2008.

[26] Satish Rajagopalan, Jos M. Aller, Jos A. Restrepo, Thomas G. Habetler, and Ronald G. Harley. Detection of Rotor Faults in Brushless DC Motors Operating Under Nonstationary Conditions. *IEEE Transactions on Industry Applications*, 42(6):1464–1477, November 2006.

[27] Daniel Wolfram, Florian Vogel, and Dominik Stauder. Condition monitoring for flight performance estimation of small multirotor unmanned aerial vehicles. In *2018 IEEE Aerospace Conference*, pages 1–17, Big Sky, MT, March 2018. IEEE.

[28] Michal Podhradsky, Jarret Bone, Calvin Coopmans, and Austin Jensen. Battery model-based thrust controller for a small, low cost multirotor Unmanned Aerial Vehicles. In *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 105–113, Atlanta, GA, USA, May 2013. IEEE.

[29] Paul Pounds, Robert Mahony, and Peter Corke. System Identification and Control of an Aerobot Drive System. In *2007 Information, Decision and Control*, pages 154–159, Adelaide, Australia, February 2007. IEEE.

[30] Jie Chen and Ron J Patton. *Robust model-based fault diagnosis for dynamic systems*, volume 3. Springer Science & Business Media, 1990.

[31] Ioan Doré Landau, Rogelio Lozano, Mohammed M'Saad, and Alireza Karimi. *Adaptive control*, volume 51. Springer New York, 1998.

[32] Brian Whitehead and Stefan Bieniawski. Model reference adaptive control of a quadrotor uav. In *AIAA Guidance, Navigation, and Control Conference*, page 8148, 2010.

[33] Liguo Sun, Coen C de Visser, Q Ping Chu, and Wouter Falkena. A hybrid sensor based backstepping control approach with its application to fault-tolerant flight control. In *AIAA Guidance, Navigation, and Control (GNC) Conference*, page 4777, 2013.

[34] Rowena L Eberhardt and David G Ward. Indirect adaptive flight control system interactions. *International Journal of Robust and Nonlinear Control*, 9(14):1013–1031, 1999.

[35] Peng Lu, Erik-Jan van Kampen, and Qiping P. Chu. Nonlinear Quadrotor Control with Online Model Identification. In *Advances in Aerospace Guidance, Navigation and Control*, pages 81–98. Springer International Publishing, 2015.

[36] Mina Ranjbaran and Khashayar Khorasani. Fault recovery of an under-actuated quadrotor aerial vehicle. In *49th IEEE Conference on Decision and Control (CDC)*, pages 4385–4392. IEEE, 2010.

[37] Mina Ranjbaran and Khashayar Khorasani. Generalized fault recovery of an under-actuated quadrotor aerial vehicle. In *2012 American Control Conference (ACC)*, pages 2515–2520. IEEE, 2012.

[38] Simon Haykin and Bernard Widrow. *Least-mean-square adaptive filters*, volume 31. John Wiley & Sons, 2003.

[39] Daniel Vey and Jan Lunze. Experimental evaluation of an active fault-tolerant control scheme for multirotor UAVs. In *2016 3rd Conference on Control and Fault-Tolerant Systems (SysTol)*, pages 125–132, Barcelona, Spain, September 2016. IEEE.

[40] Peter Bauer, Raghu Venkataraman, Balint Vanek, Peter J. Seiler, and Jozsef Bokor. Fault Detection and Basic In-Flight Reconfiguration of a Small UAV Equipped with Elevons. *IFAC-PapersOnLine*, 51(24):600–607, 2018.

[41] Vahid Hassani, Joao Pedro Hespanha, Michael Athans, and Antonio M Pascoal. Stability analysis of robust multiple model adaptive control. *IFAC Proceedings Volumes*, 44(1):350–355, 2011.

[42] Mohammad Hadi Amoozgar, Abbas Chamseddine, and Youmin M Zhang. Experimental test of an interacting multiple model filtering algorithm for actuator fault detection and diagnosis of an unmanned quadrotor helicopter. In *International Conference on Intelligent Robotics and Applications*, pages 473–482. Springer, 2012.

[43] Peng Lu, Laurens Van Eykeren, Erik-Jan van Kampen, Coen de Visser, and Qiping Chu. Double-model adaptive fault detection and diagnosis applied to real flight data. *Control Engineering Practice*, 36:39–57, 2015.

[44] Peng Lu and Erik-Jan Van Kampen. Selective reinitialisation multiple model adaptive estimation for fault detection and diagnosis. In *AIAA Guidance, Navigation, and Control Conference*, page 0965, 2014.

[45] FE Thau. Observing the state of non-linear dynamic systems. *International journal of control*, 17(3):471–479, 1973.

[46] SHAUYING R Kou, David L Elliott, and Tzyh Jong Tarn. Exponential observers for nonlinear dynamic systems. *Information and control*, 29(3):204–216, 1975.

[47] G. Schreier, J. Ragot, R.J. Patton, and P.M. Frank. Observer Design for a Class of Non-Linear Systems. *IFAC Proceedings Volumes*, 30(18):483–488, August 1997.

[48] A Freddi, S Longhi, and A Monteriù. A Model-Based Fault Diagnosis System for a Mini-Quadrotor. page 7, 2009.

[49] Alessandro Freddi, Sauro Longhi, and Andrea Monteriù. A Diagnostic Thau Observer for a Class of Unmanned Vehicles. *Journal of Intelligent & Robotic Systems*, 67(1):61–73, July 2012.

[50] A. Monteriù, P. Asthana, K. P. Valavanis, and S. Longhi. Real-Time Model-Based Fault Detection and Isolation for UGVs. *Journal of Intelligent and Robotic Systems*, 56(4):425–439, November 2009.

[51] Mogens Blanke, Michel Kinnaert, Jan Lunze, Marcel Staroswiecki, and J Schröder. *Diagnosis and fault-tolerant control*, volume 2. Springer, 2006.

[52] Zhaohui Cen, Hassan Noura, and Younes Al Younes. Systematic fault tolerant control based on adaptive thau observer estimation for quadrotor uavs. *International Journal of Applied Mathematics and Computer Science*, 25(1):159–174, 2015.

[53] Agus Hasan and Tor Arne Johansen. Model-based actuator fault diagnosis in multirotor UAVs. In *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1017–1024. IEEE, 2018.

[54] Zhaohui Cen, Hassan Noura, Tri Bagus Susilo, and Younes Al Younes. Robust Fault Diagnosis for Quadrotor UAVs Using Adaptive Thau Observer. *Journal of Intelligent & Robotic Systems*, 73(1-4):573–588, January 2014.

[55] B Walcott and S Żak. State observation of nonlinear uncertain dynamical systems. *IEEE Transactions on automatic control*, 32(2):166–170, 1987.

[56] AG Bondarev, SA Bondarev, NE Kostyleva, and Vadim Ivanovich Utkin. Sliding modes in systems with asymptotic state observers. *Avtomatika i Telemekhanika*, (6):5–11, 1985.

[57] Christopher Edwards and Sarah K Spurgeon. On the development of discontinuous observers. *International Journal of control*, 59(5):1211–1229, 1994.

[58] Christopher Edwards and Sarah Spurgeon. *Sliding mode control: theory and applications*. Crc Press, 1998.

[59] Ji Xiang, Hongye Su, and Jian Chu. On the design of walcott-zak sliding mode observer. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 2451–2456. IEEE, 2005.

[60] Ali J Koshkouei and Alan SI Zinober. Sliding mode state observation for non-linear systems. *International Journal of Control*, 77(2):118–127, 2004.

[61] Johan Löfberg. Yalmip: A toolbox for modeling and optimization in matlab. In *Proceedings of the CACSD Conference*, volume 3. Taipei, Taiwan, 2004.

[62] Halim Alwi and Christopher Edwards. Robust fault reconstruction for linear parameter varying systems using sliding mode observers. *International Journal of Robust and Nonlinear Control*, 24(14):1947–1968, 2014.

[63] Kumar Pakki Bharani Chandra, Halim Alwi, and Christopher Edwards. Fault reconstruction for a quadrotor using an lpv sliding mode observer. *IFAC-PapersOnLine*, 48(21):374–379, 2015.

[64] Ke Zhang, Bin Jiang, and Vincent Cocquempot. Adaptive observer-based fast fault estimation. *International Journal of Control, Automation, and Systems*, 6(3):320–326, 2008.

[65] Bingyong Yan, Huifeng Wang, and Huazhong Wang. Robust fault detection for a class of uncertain nonlinear systems based on multiobjective optimization. *Mathematical Problems in Engineering*, 2015, 2015.

[66] Carsten Scherer and Siep Weiland. *Linear matrix inequalities in control*, volume 3. 2000.

[67] Rajesh Rajamani and J Karl Hedrick. Adaptive observers for active automotive suspensions: theory and experiment. *IEEE Transactions on control systems technology*, 3(1):86–93, 1995.

[68] Young Man Cho and Rajesh Rajamani. A systematic approach to adaptive observer synthesis for nonlinear systems. *IEEE transactions on Automatic Control*, 42(4):534–537, 1997.

[69] Gildas Besançon. Remarks on nonlinear adaptive observer design. *Systems & control letters*, 41(4):271–280, 2000.

[70] Gildas Besançon. *Nonlinear observers and applications*, volume 363. Springer, 2007.

[71] Mondher Farza, Mohammed M'Saad, Tarak Maatoug, and Mohamed Kamoun. Adaptive observers for nonlinearly parameterized class of nonlinear systems. *Automatica*, 45(10):2292–2299, 2009.

[72] Ivan Y Tyukin, Erik Steur, Henk Nijmeijer, and Cees Van Leeuwen. Adaptive observers and parameter estimation for a class of systems nonlinear in the parameters. *Automatica*, 49(8):2409–2423, 2013.

[73] Gildas Besançon and A Ţiclea. On adaptive observers for systems with state and parameter nonlinearities. *IFAC-PapersOnLine*, 50(1):15416–15421, 2017.

[74] M. Hadi Amoozgar, Abbas Chamseddine, and Youmin Zhang. Experimental Test of a Two-Stage Kalman Filter for Actuator Fault Detection and Diagnosis of an Unmanned Quadrotor Helicopter. *Journal of Intelligent & Robotic Systems*, 70(1-4):107–117, April 2013.

[75] N Eva Wu, Youmin Zhang, and Kemin Zhou. Detection, estimation, and accommodation of loss of control effectiveness. *International Journal of adaptive control and signal processing*, 14(7):775–795, 2000.

[76] Jean Gallier. The schur complement and symmetric positive semidefinite (and definite) matrices. *Penn Engineering*, 2010.

[77] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[78] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

[79] Kevin M Lynch and Frank C Park. *Modern Robotics*. Cambridge University Press, 2017.

Glossary

Acronyms

AFTC Active Fault-Tolerant Control

BLDC Brushless DC

CG Center of Gravity

CSA Current Signature Analysis

DAC Direct Adaptive Control

DFT Discrete Fourier Transform

DMAE Double Model Adaptive Estimation

EKF Extended Kalman Filter

ESC Electronic Speed Control

FDD Fault Detection and Diagnosis

FDI Fault Detection and Identification

FFT Fast Fourier Transform

GPS Global Positioning System

HIL Hardware-in-the-loop

IAC Indirect Adaptive Control

IIR Infinite Impulse Response

IMU Inertial Measurement Unit

INDI Incremental Nonlinear Dynamic Inversion

LMI Linear Matrix Inequality

LMIs Linear Matrix Inequalities

LMS Least Mean Squares

LoE Loss of Effectiveness

LPV Linear Parameter-Varying

MMAE Multiple Model Adaptive Estimation

MRAC Model Reference Adaptive Control

NDI Nonlinear Dynamic Inversion

NN Neural Network

OLS Ordinary Least Squares

PID Proportional-Integral-Differential

PPE Posterior Probability Evaluator

PSD Power Spectral Density

PWM Pulse Width Modulation

RPM Revolutions Per Minute

SoC State of Charge

STFT Short-Time Fourier Transform

TFR Time-Frequency Representation

TSKF Two-Stage Kalman Filter

UAV Unmanned Aerial Vehicle

UKF Unscented Kalman Filter

WPD Wavelet Packet Decomposition

Nomenclature

e_1	Unit vector of X direction
e_2	Unit vector of Y direction
e_3	Unit vector of Z direction
$\mathfrak{f}$	Force vector $[N]$
g	Gravitational acceleration constant $\left[\frac{m}{s^2}\right]$
η	Actuator effectiveness parameter
l	Actuator distance from body frame origin
$\mathfrak{m}$	Moment vector $[Nm]$

q	Configuration vector
r	Position vector $[m]$
R	Rotation matrix
T	Homogeneous transformation matrix
T	Generated thrust $[N]$
v	Velocity vector $\left[\frac{m}{s}\right]$
$\mathcal{A}$	Inertial frame of reference
$\mathcal{B}$	Body-fixed frame of reference
$\mathcal{C}$	Rotor-fixed frame of reference
$\mathcal{I}$	Identity matrix
$\mathcal{T}$	Twist
$\mathcal{W}$	Wrench
$\mathfrak{q}$	Attitude quaternion
κ_e	Back-EMF coefficient $\left[\frac{Vs}{m}\right]$
κ_T	Thrust coefficient $\left[\frac{Ns^2}{rad^2}\right]$
κ_τ	Torque coefficient $\left[\frac{Nms^2}{rad^2}\right]$
κ_f	Motor friction coefficient $\left[\frac{Nm \cdot s}{rad}\right]$
κ_q	Motor torque coefficient $\left[\frac{Nm}{A}\right]$
τ	Torque vector $[Nm]$
ω	Rotor speed $\left[\frac{rad}{s}\right]$
Ω	Angular velocity vector $\left[\frac{rad}{s}\right]$
$\mathbb{C}$	Set of complex numbers
$\mathbb{H}$	Set of quaternions
$\mathbb{N}$	Set of natural numbers
$\mathbb{Q}$	Set of rational numbers
$\mathbb{R}$	Set of real numbers
$\mathbb{Z}$	Set of integer numbers
$SE(n)$	n dimensional special euclidean group
$\mathfrak{so}(n)$	n dimensional set of skew-symmetric matrices
$SO(n)$	n dimensional special orthogonal group