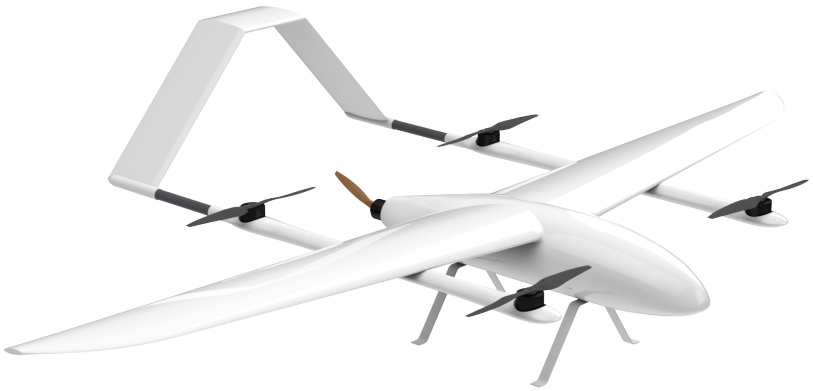
Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

Bernhard Paus Græsdal

Full Nonlinear System Identification for a Vertical-Takeoff-and-Landing Unmanned Aerial Vehicle

Master's thesis in Cybernetics and Robotics
Supervisor: Kostas Alexis
November 2021



NTNU
Norwegian University of
Science and Technology

Bernhard Paus Græsdal

Full Nonlinear System Identification for a Vertical-Takeoff-and-Landing Unmanned Aerial Vehicle

Master's thesis in Cybernetics and Robotics
Supervisor: Kostas Alexis
November 2021

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

Abstract

This thesis presents a novel dynamic model for a widely used Vertical-TakeOff-and-Landing (VTOL) Unmanned Aerial Vehicle (UAV) named the *FoxTech Babyshark 260 VTOL*. Through its hybrid design, the aircraft is capable of taking off and landing vertically, by combining the principles of fixed-wing and quadcopter aircrafts. Having a good model for the system is crucial for many applications, such as model-based control, planning and fault-detection. Through proposing a publicly available dynamic model of the Babyshark VTOL aircraft, the author hopes to facilitate future research in the field of VTOL UAVs, and to enable the use of model-based methods on the Babyshark for both industrial and academic applications.

The complete UAV model is derived from first principles of physics and parameters are estimated with optimization methods through system identification. 3D modeling is used to find the inertial properties of the aircraft, while on-the-ground testing is used to identify separate actuator models. Flight-Test Data (FTD) is obtained from real flight experiments, and kinematic relationships, piecewise splines, and analytical derivation are used for Flight Path Reconstruction (FPR). The aerodynamic model is then derived from FTD, where a stepwise-regression procedure is used together with the Equation-Error method to identify a suitable aerodynamic model structure for both longitudinal and lateral-directional motion. Next, the Output-Error method is implemented with the Gauss-Newton step optimization algorithm to arrive at the final parameter estimates that minimize the squared simulation error. Finally, a single parameter is added to the model to capture rudder-pitch coupling caused by the aircraft's V-tail configuration.

The result of this thesis is a model which captures the observed behavior of the aircraft well. Confidence in the model is built through residual analysis on validation data, investigation of the static curves of the model, invidividual parameter analysis, and eigenmode analysis of the linearized aircraft model. Finally, future directions are proposed, both for further model development and applications of the model. A ready-to-use implementation of the final model can be found at [1].

Sammendrag

I denne masteroppgaven presenteres en ny dynamisk modell for den populære VTOL-droneplattformen *FoxTech Babyshark 260 VTOL*. På grunn av sitt design er fartøyet i stand til å ta av og lande vertikalt, gjennom å kombinere prinsipper fra moderne kvadrotordroner og tradisjonelle fly. For mange bruksområder er det essensielt å ha en god modell av systemet, for eksempel for modellbasert kontroll, planlegging og feildetektering. Ved å tilgjengeliggjøre en dynamisk modell av VTOL-dronen håper forfatteren å fasilitere for fremtidig forskning på VTOL-droner, og å muliggjøre bruk av modellbaserte metoder på fartøyet for både industrielle og akademiske applikasjoner.

Den fullstendige dronemodellen er utledet fra grunnprinsipper i fysikk, og modellparametrene er estimert gjennom optimeringsmetoder fra systemidentifikasjon. 3D-modellering er brukt for å beregne treghetsegenskapene til fartøyet, og separate aktuatormodeller er identifisert gjennom enkle labtester. Flytestdata er innhentet fra ekte eksperimenter, og kinematiske forhold, stykkvise polynomiske funksjoner og analytisk derivering er brukt for å rekonstruere flyets fullstendige tilstand gjennom eksperimentene. Deretter er den aerodynamiske modellen utledet fra flytestdataen, der en stegvis regresjonsprosedyre kombinert med Equation-Error-metoden er brukt for å identifisere en passende modellstruktur for både langsgående og sideveis bevegelse. Videre så er Output-Error-metoden implementert med Gauss-Newton-algoritmen brukt for å finne modellparametrene som minimerer den kvadratiske simuleringsfeilen til modellen. Til slutt er modellen utvidet med en siste parameter for å fange kobling mellom flyets sideror og pitchvinkel, forårsaket av flyets V-hale.

Resultatet av denne oppgaven er en model som beskriver den observerte oppførselen til flyet på en god måte. Tilliten til modellen bygges gjennom residualanalyse på valideringsdata, etterforskning av modellens statiske kurver, individuell parameteranalyse og egenmodusanalyse av den lineariserte modellen for fartøyet. Til slutt så er lovende retninger for fremtidig arbeid foreslått, både for videre utvikling og applikasjoner av modellen. En offentlig tilgjenglig, ferdig implementasjon av modellen finnes på [1].

Preface

This thesis is the culmination of my M.Sc. degree in Cybernetics and Robotics at the Norwegian University of Science and Technology (NTNU). The work is carried out in collaboration with the company Aviant and focuses on the company's current aircraft platform.

First of all, I would like to thank my supervisor, Prof. Kostas Alexis, for his guidance with this thesis. Kostas's expertise has been invaluable throughout my work, and I am grateful for everything that I have learned from him. I would also like to thank Thomas Stastny for his help and advice along the way. In addition, I would like to thank the rest of the Aviant team for their support with work on the company's platform.

Finally, I would like to thank my friends and family for their continuous support during my work.

Bernhard Paus Græsdal

Trondheim, November 2021

Contents

List of Figures

List of Tables

Acronyms

AoA Angle of Attack.

CFD Computational Fluid Dynamics.

DP Dynamic Programming.

EKF Extended Kalman Filter.

EOMs Equations of Motion.

FC Flight Controller.

FPR Flight Path Reconstruction.

FTD Flight-Test Data.

GNSS Global Navigation Satellite System.

GOF Goodness-of-Fit.

IMU Inertial Measurement Unit.

IMUS Incremental Model Update Schemes.

LQG Linear Quadratic Gaussian.

LQR Linear Quadratic Regulator.

MAC Mean Aerodynamic Chord.

MAE Mean Absolute Error.

MPC Model Predictive Control.

NMAE Normalized Mean Absolute Error.

NMPC Nonlinear Model Predictive Control.

nrmse Normalized Root-Mean-Squared Error.

OLS Ordinary Least-Squares.

RMSE Root-Mean-Squared Error.

RPS pevolutions per second.

RSS Residual Sum of Squares.

SSA SideSlip Angle.

TIC Theil's Inequality Coefficient.

TSS Total Sum of Squares.

TVLQR Time-Varying Linear Quadratic Regulator.

UAV Unmanned Aerial Vehicle.

UAVs Unmanned Aerial Vehicles.

VLM Vortex Lattice Method.

VTOL Vertical-TakeOff-and-Landing.

Chapter 1

Introduction

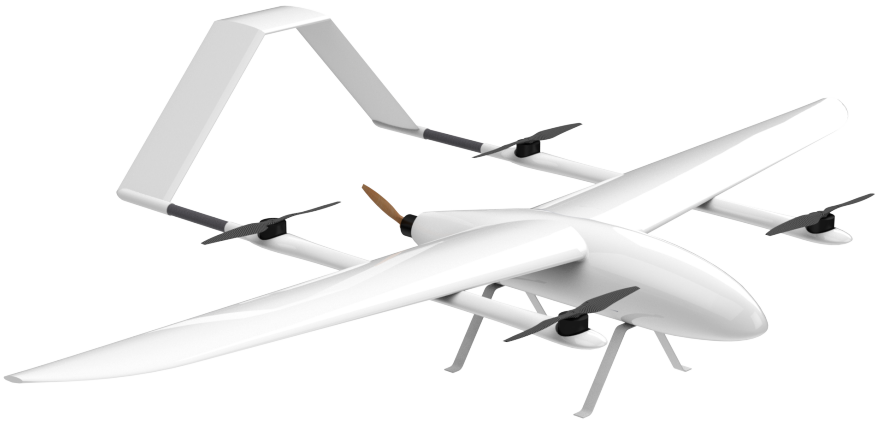


Figure 1-1: The focus of this thesis: The FoxTech Babyshark 260 VTOL.

The field of small Unmanned Aerial Vehicles (UAVs) has seen significant progress over recent years, both academically and commercially. This is caused by various reasons, ranging from advances in research to the availability and price of hardware. Technical advancements in the field of multirotor and fixed-wing vehicles, combined with reduced prices for components, significant improvements in computational power

on lightweight microcontrollers, and weight reductions in sensors and other modules, are all reasons that have made it possible for both businesses and scientists to acquire and develop modern, small UAVs.

Two of the most common types of Unmanned Aerial Vehicle (UAV)s are fixed-wing vehicles and multirotor vehicles. The respective UAV types have their advantages and disadvantages, and it is customary to employ them for different tasks. Multirotor vehicles are typically used for short-flights requiring high precision, while fixed-wing UAVs are employed where long-range capability is a key factor. This can be seen in the industry, where multirotor UAVs such as quadcopters are used for inspection and precision-farming [2, 3, 4], and fixed-wing UAVs are used for routine surveillance or search-and-rescue of larger areas [5, 6, 7, 8, 9]

In particular, multirotor UAVs have the advantage that they can take off from nearly anywhere. In addition to this, multirotor UAVs are highly maneuvrable. However, this all comes at the cost of spending much energy to stay in the air, as the vehicle constantly has to generate an upwards force that balances out the gravitational force pulling the aircraft down. On the other hand, fixed-wing UAVs have the advantage of being energy efficient, as the aerodynamic properties of a wing are exploited to generate lift. Therefore, the amount of energy required to keep the vehicle airborne is significantly reduced compared to that of multirotor vehicles. Typically, this advantage comes at the cost of requiring large runways for takeoff and landing.

In recent years, a new type of unmanned aerial vehicle called Vertical-TakeOff-and-Landing (VTOL) has grown more and more popular [7, 8, 9, 10, 11]. Although this type of design has seen use for manned aircraft, such as the Osprey V-22 aircraft as seen in fig. 1-2, its application to small, unmanned VTOL aircraft are far from mature. Different VTOL designs exist, but the main purpose of these vehicles is the same across designs: VTOL vehicles maintain the energy advantages of fixed-wing vehicles while still keeping essential advantages of multirotor vehicles; most notably, VTOL vehicles are capable of takeoff and landing anywhere.

Typical VTOL designs can be described as tailsitters, tilrotors or hybrid designs. Tailsitter VTOL vehicles are aircraft where the thrust generating propellers are sta-

tionary, either on the front or back of the aircraft, and are generating a thrust in the opposite direction of the nose. These aircraft take off from a standing configuration and tilt into the cruise configuration, where the propellers generate thrust in the direction of flight. Tiltrotors have propellers capable of tilting and can keep the aircraft's body stationary while tilting the propellers to move from hover flight to cruise flight. These designs are both lightweight and reduce the number of moving parts. However, they come at the cost of being vulnerable to disturbances and faults during the transition phase, both from hover flight to cruise flight and back.

Figure 1-2: The Bell Boeing V-22 Osprey, one of the most well-known examples of manned VTOL aircraft. *Photographer: Airwolfhound from Hertfordshire, UK. Picture reused under the Creative Commons BY-SA 2.0 license via Wikimedia Commons.*

This work deals with the VTOL UAV named the *FoxTech Babyshark 260 VTOL* [12], an aerial platform that is used across the world for research and commercial applications. The aircraft employs a hybrid VTOL design, and the aircraft can be seen in fig. 1-3. This design is essentially a traditional fixed-wing aircraft, where multirotor motors and propellers are added to the design in a stationary fashion. The multirotor propellers are used only during the hover phase of flight, before the aircraft transitions into conventional fixed-wing cruise flight, where the traditional fixed-wing pusher motor is being used to generate forward thrust. This type of design has a significant advantage of additional safety and robustness, as the vehicle will, in most cases, be able to use the multirotor propellers to land safely. The aircraft is also less

Figure 1-3: Pictures of the FoxTech Babyshark 260 VTOL in flight. *Photographer: Otto Barsten Johnson. Pictures published with permission from the photographer.*

prone to disturbances and errors during the most critical flight stage, the transition between cruise and hover flight. However, this design comes with the drawback of additional drag during fixed-wing cruise flight when the multirotor propellers are not in use.

1.1 Thesis Contributions and Research Objectives

The goal of this thesis is to develop an accurate dynamic model of the FoxTech Babyshark 260 VTOL. Although the aircraft is widely used for both research and commercial applications, to the author's knowledge, no dynamic model of the aircraft

exists today. Therefore, all known applications of the Babyshark VTOL employ PID controllers that are experimentally tuned. Due to the lack of an adequate dynamic model, it has not been possible to employ rigorous model-based control, planning, and estimation methods on the aircraft. By proposing a novel model of the aircraft, this thesis aims to facilitate the use of advanced, model-based methods on the aircraft in the future, both for industrial and academic applications.

The ultimate result of this thesis is a novel, full-state, nonlinear model of the aircraft. The final nonlinear model is presented in its entirety in section 6.2. As the aircraft is essentially a combination between a multirotor and a fixed-wing aircraft, the model of the aircraft combines modeling from two traditionally separated fields. For multirotors, simple models are typically employed, where identification primarily focuses on identifying the inertia properties and the propeller properties, modeling the UAV as a standard rigid body [13, 14, 15, 16, 17, 18]. On the other hand, for fixed-wing aircraft, modeling is usually more complex, and requires complex aerodynamic modeling [19, 20, 21]. Therefore, fixed-wing system identification makes up the central part of the work in this thesis.

In the process of obtaining a dynamic aircraft model, a combination of many methods is used. As will be seen, the field of system identification is mature, especially for manned, full-scale fixed-wing aircraft. However, for small UAVs, the field is not as mature, and not all principles can be applied directly. Therefore, to obtain a satisfactory model of the aircraft, this work employs principles from traditional system identification methods for full-scale aircraft, combined with new variations on methods for data-processing, parameter estimation, and model development based on recent relevant research.

Finally, to facilitate future work, the obtained aircraft model is made publicly available. As is shown in the final section of this thesis, the model performs well upon validation. Currently, much of UAV research focuses on either multirotors or fixed-wing aircraft. As the Babyshark is a widespread VTOL platform both for research and commercial applications, the author hopes that the freely available model of the aircraft will contribute to research on small, unmanned VTOL aircraft in the future.

1.2 Motivation and Applications of UAV System Identification

The goal of UAV system identification is to obtain a dynamic model of the aircraft, which is desirable for many reasons. Typical reasons for performing system identifications of UAVs are summarized in the survey by Hoffer et al. in [22], where the authors propose that everything that requires a-priori knowledge of the aircraft is dependent on having a model. Examples include control system design, control system verification, fault detection, online and offline parameter estimation, and adaptive control to compensate for time-varying changes in the system or to faults. In addition, having an accurate model lets the engineer employ strong methods for planning and state estimation.

Many of the most rigorous modern control and planning methods are model-based methods. That is, these methods require knowledge of the expected behavior of the system, which usually means having a dynamic model of the system. Examples of model-based control methods employing dynamic system models are many. In particular, within optimal control, methods such as Model Predictive Control (MPC), Linear Quadratic Regulator (LQR), Linear Quadratic Gaussian (LQG), H_∞ control, Dynamic Programming (DP), or variations of these, all require a dynamic model, either linear or nonlinear, of the system that is to be controlled, with examples for fixed-wing aircraft in [23, 24, 25]. Other examples are found within standard nonlinear control methods: most Lyapunov-based methods require a model of the system, for example, Sliding Mode Control, Feedback Linearization, or Backstepping Control [26]. Even for Adaptive Control, when the goal is to estimate and react to unknown parameters and disturbances online, the methods usually require an a-priori dynamic model [27].

All of the control methods mentioned above stabilize the system around a nominal point or trajectory, and many methods for trajectory generation are also model-based. However, some methods are model-free, such as the workflow employed in the famous open-source autopilot PX4 [28, 29], which is often used to control the Babyshark

aircraft. In PX4, trajectories are generated as straight lines between user-selected waypoints, and then a guidance logic named L1 guidance [30] is used to decide how the aircraft should track the trajectory. This trajectory optimization scheme works for many simple applications. However, because no information about the aircraft behavior is employed when generating the trajectory, it is usually not dynamically feasible, which causes significant tracking errors.

In contrast, model-based trajectory methods such as trajectory optimization may be employed to generate complex, dynamically feasible trajectories. There exists a number of these methods, and one such method is Direct Collocation [31]. Here, a nonlinear optimization scheme is employed to generate dynamically feasible trajectories represented by piecewise polynomials. One of the strengths of such methods is that additional constraints may be imposed on the trajectories, and collision avoidance or other task-specific behavior may be included in the trajectory planning. When the trajectories are dynamically feasible, tracking errors are significantly reduced compared to model-free trajectory generation methods.

In summary, there are several strong reasons to obtain a dynamic model of an aircraft. Applications including control, planning, and fault detection are significant reasons for developing such a model. In addition, many other valuable applications exist, such as more accurate simulation, estimation methods, or adaptive control methods.

1.3 Thesis Outline

In chapter 2, an overview of relevant literature for this thesis is presented. Next, the general mathematical model that is employed is presented in chapter 3, including rigid-body modeling, aerodynamic modeling, and actuator modeling. The identification of the model is divided into two parts: a preliminary airframe analysis and aerodynamic modeling based on Flight-Test Data (FTD). The preliminary analysis is presented in chapter 4, where instrumentation, static properties of the aircraft, and actuator models are found. Next, the theoretical principles employed for aerodynamic

modeling are outlined in chapter 5. Then, the results of the aerodynamic modeling are presented together with the entire model, model validation, and discussion of the final model in chapter 6. Finally, conclusions, future work, and promising directions for the model are presented in chapter 7.

1.4 Implementation

A complete implementation of the final model, including example simulations and 3D visualization, can be found at [1]. The source code for the identification procedure can be found at [32]. All the methods are implemented as outlined in chapter 5. In the interest of open-source, all of the system identification source code is implemented without third-party libraries, contrary to other relevant works, where closed-source libraries are often used for fixed-wing system identification. Most of the implementation in this work is done in Matlab, although parts of the optimization routine outlined in chapter 5 are implemented in C++ to speed up the final optimization routine.

Chapter 2

Literature Review

In this chapter, an overview of relevant literature for system identification of UAVs is given. This thesis deals with the system identification of a VTOL UAV, and it is, therefore, relevant to cover methods for multirotor modeling, fixed-wing modeling, VTOL modelling, as well as applications of these models. As will be seen, the literature on system identification for UAVs is rich, especially for full-scale traditional fixed-wing aircraft. The objective of this chapter is to provide the reader with a sense of common methods and to locate this thesis in the literary landscape of system identification for UAVs.

First, the reader is pointed to both a recent and a traditional survey on fixed-wing system identification. Next, literature on traditional full-scale aircraft identification is covered, as these are still highly relevant references for modern system identification. Further, identification of modern fixed-wing UAVs is covered, split into the time domain and the frequency domain. These methods typically deal with the identification of the aircraft around a reference flight condition. Following this, some work on expanding the aircraft modeling to the nonlinear flight envelope at close-to-stall conditions is presented. Next, multiple recent works covering system identification of a VTOL aircraft are presented. Further, as the estimation of the aerodynamic quantities AoA, SSA and airspeed are highly relevant for fixed-wing system identification, a short coverage of a few relevant works on this is provided. Finally, typical multirotor modeling is briefly introduced, and example works are given.

2.1 UAV System Identification Surveys

Before diving into specific relevant work, the reader may be interested in recent surveys and categorizations of literature on system identification for low-cost, small UAVs. One such survey is given by Hoffer et al. in [22]. The survey is from 2014, and contains an overview over literature for helicopter UAVs, fixed-wing UAVs, multirotor UAVs, flapping-wing UAVs, and lighter-than-air UAVs. An introduction to the main elements of UAV system identification is given, as well as an overview of typical applications. The literature for fixed-wing system identification is organized after model use, system identification method, data source, and application, and summarizes which works employ which methods.

Another survey that may be of interest is [33] that was presented in 1996 and covers many of the mature system identification methods traditionally used for full-scale aircraft. However, this material is meant for full-scale aircraft. In addition, most of the material is covered in the traditional textbooks [19, 20, 21], which are introduced in the next subsection.

2.2 Traditional Full-Scale Fixed-Wing Aircraft Identification

For traditional full-scale aircraft, several well-regarded textbooks for system identification exist. One of these is the textbook written by Klein et al. [19], where a comprehensive overview and theoretical foundation for modeling in both the time-domain and the frequency domain is given. Another excellent source for time-domain system identification is [20]. In this thesis, time-domain methods are used, and [19, 20] are therefore used extensively as resources. In addition, [21] gives an overview over frequency-based approaches. Although classical textbooks such as [19, 20, 21] are very relevant for UAV system, identification, the downside of these works is that the material is intended for full-scale aircraft. In fact, many of the recent works in the field of system identification for small, fixed-wing aircraft mentioned throughout

this thesis focus on applying and adapting traditional methods from [19, 20, 21] for smaller aircraft.

In particular, [19, 20] cover methods for data gathering, including input signal design, experiment design, maneuver design, and instrumentation setup. Both sources provide a fundamental introduction to general mathematical modeling for aircraft and data reconstruction and compatibility analysis techniques. Comprehensive breakdowns of identification methods such as the Filter-Error, the Equation-Error, or the Output-Error methods, in addition to regression methods for both parameter estimation and model structure determination, make up much of the content in the books. In addition, methods for model validation, such as statistical accuracy of parameters, or residual analysis, are described and come in handy when one seeks to validate the model. Other topics, such as estimation theory, realtime parameter estimation, and unstable aircraft identification, are also covered in the books.

A relevant paper on traditional system identification for fixed-wing aircraft is [34], where the methods now known in [19, 20] as the Equation-Error method and the Output-Error method are reviewed and compared. While [34] was published in 1994, it provides foundations for the methods used in much of the related literature presented here. At the time, the Equation-Error method was named the Two-Step method [35] and is compared to the One-Step Maximum Likelihood method, which is now called the Output-Error method. In the work, both methods are employed and analyzed for aircraft system identification. The One-Step Maximum Likelihood method proposes estimating both the aircraft state trajectory and the aerodynamic parameters simultaneously, and as such, assumes an appropriate model structure a-priori. On the other hand, the Two-Step method proposes estimating the state trajectory in the first step while estimating the aerodynamic parameters in the second step. The advantage of the latter is that multiple model structures can be rapidly explored. The results prove that, in many situations, the Two-Step method possesses the same estimation properties as the One-Step Maximum Likelihood method. However, the Two-Step method makes the aerodynamic parameter estimation step linear in the parameters, making it possible to investigate different model structures quickly.

2.3 Time Domain Identification of Modern Fixed-Wing UAVs

Moving on to more recent work, in [36] published in 2015, Arifianto et al. develops a small, low-cost fixed-wing UAV from commercially available off-the-shelf components. In addition, a mathematical model of the aircraft is developed. Several steps are taken in the modeling of the aircraft: The simple pendulum method [37] is used for determining the moments of inertia of the vehicle. A frequency response test of the servo motors is used to derive a second-order model of the control surfaces. A sophisticated propulsion model is developed using a lookup table generated from CFD modeling of the propeller in the JavaProp applet and validated through static tests. Finally, the Output-Error method, as formulated in [19, 20], is used to estimate both the longitudinal and lateral-directional aerodynamic parameters from flight-test data in the time domain, based on an aerodynamical model postulated a-priori. According to the authors, this is the first time this method has been employed for small fixed-wing UAVs. In this work, in-house built air data probes are used to get accurate information on the entire aerodynamic state of the system, such as Angle of Attack (AoA), SideSlip Angle (SSA) and airspeed, which are used in the identification procedure. Having sensors that directly measure these aerodynamic angles are uncommon for small, fixed-wing UAVs. Finally, the obtained model is validated through real-life flight-data.

In the work by Grymin et al. from 2016 [38], the Two-Step method from [34, 35] is used to identify a suitable aerodynamic model for the small UAV developed in [36]. As mentioned earlier, the Two-Step method provides a framework for rapid investigation of various model structures, which the authors take advantage of by using stepwise multiple regression to determine relevant explanatory variables for the aerodynamic coefficients, as is proposed in [19]. In the first step, the authors use an Extended Kalman Filter (EKF) for the Flight Path Reconstruction (FPR), which is the reconstruction of the aircraft state trajectory from measured data. For the second step, Ordinary Least-Squares (OLS) is used to estimate the aerodynamic

parameters. This allows multiple model structures to be rapidly explored, and the statistical metric known as the partial F-value is used to determine which explanatory variables should be included in the regression. In the end, the aerodynamic forces and moments are described as functions of multiple linear and second-degree spline functions of the state variables. After the aerodynamic model is obtained, it is used to obtain dynamically feasible trajectories through the solution of an optimal control problem, and discrete-time feedback controllers are used to stabilize the trajectories. Finally, the results are verified in simulation and flight tests.

Even more recently, in the work by Simmons et al. in [39, 40] from 2019, system identification of a small, low-cost UAV is performed using a time-domain formulation of the Output-Error method. The Equation-Error method [19, 20] (similar to the Two-Step method from [34, 35]) is first used to determine the aerodynamic model structure, and the resulting parameters are then used as initial guessed for the Output-Error method, with the model structure developed with the Equation-Error method. A large portion of the procedure follows the practices outlined in [19], where the Output-Error method is initialized with results from the Equation-Error method. Novel additions to these methods include supplementing Flight-Test Data (FTD) with information obtained using the Vortex Lattice Method (VLM) and synthesizing information obtained from different flight maneuvers into the successful identification of a lateral-directional model. Separate models for the longitudinal and the lateral-directional maneuvers are developed, using the body-axis formulation for aerodynamic coefficients rather than the stability axes. The models show good predictive behavior when validated on flight test data, and the certainty of the parameters is shown to decrease when supplementing FTD with data from VLM.

A different perspective on system identification is given by Hale et al. in [41], where the authors focus on uncertainty analysis of identified aircraft models and their parameters. Employing the Output-Error method from [19], the authors first identify a model for a small, fixed-wing UAV with a V-tail configuration. After estimating the parameters of the model, the Cramér-Rao lower bound is used to represent the minimum variance of the estimated value, as proposed in [19]. A novel addition to

the procedure outlined in [19] is a method for determining model-form uncertainty represented by prediction intervals by using a separate validation data set to calculate the residuals between the aerodynamic coefficients in the identified model and the data. The method is tested both with synthetic data and actual flight-test data and shows that the estimated model-form uncertainty increases when the assumed model structure moves away from the true model.

An entirely different approach to system identification is taken by Kamel et al. in [23]. The authors seek to model the dynamics of a fixed-wing UAV for the purpose of using Nonlinear Model Predictive Control (NMPC) for lateral-directional trajectory tracking. A high-level closed-loop model is developed for the lateral-directional dynamics of the aircraft based on flight data, assuming that a low-level autopilot consisting of state estimation and control of attitude, airspeed, and altitude is already running on the aircraft. With successful open-source flight controllers such as the Pixhawk [42] running the PX4 software [28, 29], this is a common scenario. This way of modeling aircraft dynamics focuses only on the high-level behavior of the system and significantly reduces the order of the model, which in turn simplifies the system identification procedure. However, this comes at the cost of reducing the physical interpretability of the model parameters. A high-level NMPC is successfully implemented based on the obtained closed-loop model, and the controller is tested in real flights in cascade with the low-level autopilot controller.

The work in [23] is further extended by Stastny et al. in [24], where the lateral-directional model is extended with an open-loop model for the longitudinal dynamics. The longitudinal model assumes that the dynamics can be described in a quasi-steady manner, where airspeed, relative flight path angle, and heading are influenced by the commanded throttle and AoA. The model is validated on free-form flight data and shows good prediction capabilities. Finally, a high-level NMPC is implemented for simultaneous airspeed stabilization, 3D path following, and handling of soft angle of attack constraints.

A different approach is taken to time-domain modeling in [43], where a nonlinear adaptive estimation scheme is used for online learning of an aerodynamic model

for the fixed-wing aircraft. The purpose of using an adaptive model is to accurately capture unsteady aerodynamics in the presence of adverse and exconditions and disturbances. In the work, the entire nonlinear rigid body EOMs for an aircraft is used, and the aerodynamic forces and moments are modeled as a single-layer adaptive neural network, where the adaptive parameters are updated in realtime based on a moving window of past data. By minimizing the squared prediction error for the acceleration, the parameters are iteratively updated through a Newton-type search method.

Further, in [25, 43], the adaptive nonlinear model is used as the basis for a robust Nonlinear Model Predictive Control (NMPC) controller. The robust NMPC employs frequency-dependent weighting matrices to reject low-frequency process noise and high-frequency sensor noise. Further, a quadratic cost function is minimized, subject to the nonlinear and time-changing system dynamics. The controller is successfully tested in simulation, showing robustness to both lift reduction and drag increase in the model.

In [6], Lee et al. propose a global method for system identification of High Altitude Long Endurance vehicles, which reduces the need for extensive flight testing over multiple flight envelopes. The technique is an extension of traditional Incremental Model Update Schemes (IMUS), where virtual experiments based on Computational Fluid Dynamics (CFD) are used to avoid extensive gathering of flight data. In the work, VLM is used to obtain a baseline linear aerodynamics model, which is combined with nonlinear rigid-body dynamics. Next, estimated forces and moments from real FTD are compared to those predicted by the model from VLM to calculate model correction terms that are linear in the airspeed. Finally, the correction terms are used to develop a model valid for a broader flight envelope than initially captured by the data.

2.4 Frequency Domain Identification of Modern Fixed-Wing UAVs

Moving on to modeling and identification in the frequency domain, in [44, 45] published in 2011 and 2013, Dorobantu et al. proposed a system identification procedure for modeling of small, low-cost, fixed-wing UAVs using flight data from a single Inertial Measurement Unit (IMU). The method estimates frequency responses from obtained flight data and then identifies aircraft dynamics in the frequency domain by fitting linear parametric models to the estimated frequency responses. The aerodynamic derivatives obtained from the linear models are integrated into the nonlinear equations of motion for an aircraft, and both models are validated in the time domain. The model accuracy is evaluated using sensitivity and residual analysis and shows good model performance.

Later, [46] published by Venkataraman et al. in 2019, a model of a small, rudderless, fixed-wing UAV is developed using frequency-domain methods. First-principles are used to obtain an initial model, where the initial parameters are obtained as stability and control derivatives from the vortex lattice method using the AVL software [47]. Black-box models in the form of transfer functions are used to find the initial parameter values. Due to the lateral-directional dynamics being highly coupled, a scheme for updating only some of the most uncertain parameters with a linear state-space model is proposed, based on the principles of flight dynamics outlined in [48]. The model is validated with flight data.

2.5 Near Stall Identification of Fixed-Wing UAVs

In [49], Hoburg et al. focus on aircraft identification for post-stall aerodynamics for UAV perching. Here, a longitudinal aircraft model is developed, focusing on nonlinear aerodynamic effects that arise due to unsteady flow at high AoAs. In the work, a set of physically inspired basis functions is used to model the behavior of the aircraft close to the stall regime. Flight data is collected using a motion capture system. The

obtained model accurately predicts the measured aircraft trajectories.

In [50], work on fixed-wing UAV perching is expanded. This time, a flat-plate model for the lift and drag of the aircraft is employed. However, to better match the obtained data, the flat-plate model is augmented with radial basis functions as first proposed in [51]. Using the trajectory optimization technique called Direct Collocation, optimal perching trajectories are computed for the aircraft. Next, the model is further used to implement a control scheme known as LQR-Trees, where multiple TVLQR controllers are computed to stabilize a variety of different initial conditions around the nominal trajectory. Sums-of-Squares optimization is used to calculate the region of attraction for the controllers. A library is built from the controllers and their regions of attraction are used to select the appropriate controller in realtime. The method is extensively tested on a real physical system, showing excellent results with a success rate of 95%.

2.6 System Identification of Modern VTOL UAVs

In [7], Verling et al. propose a model-based nonlinear control law for a tailsitter VTOL. The tailsitter VTOL is a flying wing that uses two elevons combined with the differential thrust from two propellers for control. The aircraft is modeled as a rigid body system, and wind-tunnel testing is used to develop an aerodynamic model of the aircraft. The authors individually model the two sides of the aircraft corresponding to each elevon and propeller for the aerodynamic modeling. This way of modeling, combined with an aerodynamic model in the body frame, makes it possible to derive all moments from only modeling the forces along the x and y-axis, as well as the y-axis moment. Aerodynamic models similar to this thesis are employed, but the model is extended to take propeller slipstream into account. The aerodynamic coefficients are calculated from wind-tunnel measurements. A geometric control law on SO(3) is implemented to reach the reference attitude and reference angular speed, which calculates generalized forces and moments as inputs. An inversion of the UAV model is then used to calculate the required system inputs to achieve the commanded forces

and moments. The controller is implemented in PX4 and tested together with the L1 navigation law from [30].

Verling et al. further extend the work on the tailsitter UAV from [7] in the work in [9], where an augmented model is used to compute optimal trajectories for the back transition from fixed-wing flight to hover flight for the aircraft. Through an offline, nonlinear trajectory optimization scheme, optimal trajectories for the back transition that minimize the altitude change is computed based on the aircraft model. A trajectory is found for the required pitch and throttle, which is used as feed-forward by the low-level controller from [7]. Feedback is added on the altitude tracking to improve robustness to disturbances and modeling errors. The feedback is a simple P-gain, where the input is added directly to the thrust input. The trajectories and tracking with the augmented controller are tested in real-world experiments with good results.

In [10], the authors further expand the work in [7, 9] with a model-based wind estimation framework. This time, a model that captures the relation between the aircraft attitude and the freestream velocity in the North and East directions during steady-state flight is developed from a flat-plate model. The model is fit from FTD through a weighted least squares method. From this model, the airspeed can be implied from attitude measurements, making it possible to estimate the wind in the North and East directions from the wind triangle. A Kalman filter is implemented based on the model to estimate the wind vector in realtime. The estimation algorithm is tested in real experiments.

Further, in the recent work by Olsson et al. in [8], the model of the VTOL tailsitter from [7, 9, 10] is expanded to a full, nonlinear aircraft model that is valid over the entire flight regime. First principles of physics are used to derive a rigid-body model of the aircraft. Aerodynamic coefficients are identified via nonlinear, time domain, grey-box parameter estimation in Matlab from FTD. Detailed modeling of the interaction between the propeller and wings, with individual modeling of propeller moments and thrust, is used to arrive at an accurate model.

2.7 Estimation of Aerodynamic Angles and Airspeed

In [52], Johansen et al. propose a method for estimating the AoA, SSA and airspeed of an aircraft without the use of an accurate aircraft model. Instead, kinematic relationships are used rather than specific aircraft models, and a sensor suite consisting of GNSS, IMU, and a single pitot tube is assumed. A Kalman filter driven by the GNSS velocities and the airspeed sensor measurements is designed to estimate the aerodynamic quantities. Under the conditions that the input signal is persistently exciting, meaning that the aircraft is changing its pitch and yaw, the estimates converge. Actual experiments seem to indicate parameter convergence; however, the results are deemed preliminary, as the results are not compared to actual measurements of the aerodynamic quantities.

The work is further extended by Wenz et al. in [53]. Here, the goal is to estimate the wind velocity vector to compute the AoA, SSA and airspeed of the aircraft. The wind velocity vector is estimated from an Extended Kalman filter, which employs both kinematic relationships, aerodynamic relationships relating the lift and drag to the AoA with a flat-plate model as proposed in [54], and Drydens gust model for the wind, also as proposed in [54]. Finally, the implementation is tested in simulation, where it shows promising results.

2.8 Multirotor System Identification

In general, research on quadcopters tends to employ the same modeling principles. The quadrotor is usually modeled as a rigid body, and the inputs of the system are defined as the total thrust and the moments around the body axes. The attitude may be represented by Euler angles, such as in [13, 15, 16, 18], quaternions as in [14, 55, 56], or rotation matrices such as in [17]. Usually, the propellers are assumed to generate an instantaneous thrust and moment, where the thrust and moment are modeled as quadratic in the propeller rotational speed. Sometimes, the propellers are

modeled as a first-order response, such as in [55].

An example of a quadrotor model and its usage is given in [17], where Lee et al. propose a geometric, nonlinear, model-based controller for a quadrotor. The controller is similar to that proposed for a VTOL tailsitter in [7]. In [17], the quadrotor is modeled as a rotating rigid-body, where the attitude is modeled with rotation matrices. The inputs to the model are the total force and the moments around the body axes. The propellers are assumed to generate an instantaneous thrust or moment, where the moment is proportional to the generated thrust. In the work, the position and heading of the quadrotor are controlled. The controller commands total thrust and moment, and the quadrotor model is used to calculate the corresponding inputs.

Chapter 3

Background and Aircraft Modeling

In this chapter, a complete mathematical model for the VTOL UAV is presented. The model is based on rigid-body Equations of Motion (EOMs), combined with traditional principles for aerodynamic modelling for fixed-wing aircraft as found in litterature such as [19, 20, 38, 39, 40, 44, 54, 57]. First, relevant coordinate frames and the state and input formulations are defined. Then, standard rigid-body EOMs are presented: the kinematic relationships known as the navigation equations and the rotational kinematic equations are presented, followed by the dynamics in the form of force equations and moment equations. A general model for aerodynamic forces and moments is presented, where a separation is made between longitudinal and lateral-directional motion. Next, an additional parameter is added to capture rudder-pitch coupling caused by the aircraft V-tail configuration. Finally, the actuators of the system are modeled, both the fixed-wing propeller, control surfaces, and the multirotor propellers.

3.1 Reference Frames and State Representation

In aircraft modeling, a common assumption is that the aircraft is a rigid body. Hence, aircraft are often modeled by the standard Newton-Euler equations that describe rigid bodies' combined translational and rotational dynamics. However, the aerodynamics forces and moments that affect an aircraft are not dependent on the inertial position

and velocities of the aircraft but rather depend on the movement of the aircraft relative to the surrounding airflow. For that reason, it is helpful to define a set of reference frames to aid in the modeling of the aircraft. As will be seen, the dynamics of an aircraft can be defined differently depending on which reference frame is chosen to model the state, and it is essential how the state of the aircraft is described. In this section, both relevant reference frames and the state representation are covered.

3.1.1 Reference Frames

As is common in aircraft modeling, four reference frames are defined and used in this work. These are the inertial NED frame $\mathcal{N}$, the body frame $\mathcal{B}$, the stability frame $\mathcal{S}$, and the wind frame $\mathcal{W}$. Here, the reference frames are presented briefly. For a comprehensive walk-through, the reader is referred to [54], which employs the same reference frames.

First, let the NED frame be denoted as $\mathcal{N}$. The NED frame $\mathcal{N}$ is defined as an inertial frame, with the x-axis pointing North, the y-axis pointing East, and the z-axis pointing Down. $\mathcal{N}$ is used to define inertial positions and velocities, and several of the available sensor measurements are measured in the NED frame, such as Global Navigation Satellite System (GNSS) data and magnetometer readings.

Second, define the body frame $\mathcal{B}$ as a reference frame centered with the origin at the center of gravity of the aircraft, with the x-axis pointing out of the aircraft's nose, the y-axis pointing out of the right wing, and the z-axis pointing down. The transformation from $\mathcal{N}$ to $\mathcal{B}$ is given by the Z-Y-X Euler sequence, that is, by the following sequence of right-handed rotations:

$$\boldsymbol{R}_{\mathcal{N}}^{\mathcal{B}} = \boldsymbol{R}_x(\phi)\boldsymbol{R}_y(\theta)\boldsymbol{R}_z(\psi) \tag{3.1}$$

Next, let $\mathcal{S}$ denote the stability frame. The stability frame is helpful in the modeling of lift and drag forces and is defined such that the lift force acts along the negative y-axis, and the drag force acts along the negative x-axis. $\mathcal{S}$ is obtained from $\mathcal{B}$ by a negative right-handed rotation of the Angle of Attack (AoA), represented by α,

around the y-axis:

$$\boldsymbol{R}_{\mathcal{S}}^{\mathcal{B}} = \boldsymbol{R}_y(-\alpha) \tag{3.2}$$

Finally, let the wind frame be denoted by $\mathcal{W}$. Although this frame is not directly used, it is useful as it defines the SideSlip Angle (SSA), which we represent by β. Let $\mathcal{W}$ be defined as a right-handed rotation of β around the z-axis in $\mathcal{S}$:

$$\boldsymbol{R}_{\mathcal{W}}^{\mathcal{S}} = \boldsymbol{R}_z(\beta) \tag{3.3}$$

3.1.2 State and Input Representation

First, let the position of the aircraft in the NED frame $\mathcal{N}$ be defined as $\boldsymbol{p} = [x,\, y,\, h]^\intercal$, where x and y denotes the position along the North and East axes, and h indicates the height of the aircraft, defined in the negative direction along the Down axis. Next, let the attitude of the aircraft be represented by the Euler angles $\boldsymbol{\Phi} = [\phi,\, \theta,\, \psi]^\intercal$, where ϕ, θ, and ψ denote the roll, pitch, and yaw of the aircraft, respectively. Next, let the angular rates of the aircraft be described by $\boldsymbol{\omega} = [p,\, q,\, r]^\intercal$, where p, q and r describe the rotational speed around the x, y, and z-axis in the body frame $\mathcal{B}$. Further, let $\boldsymbol{v} = [u,\, v,\, w]^\intercal$ describe the body velocities of the aircraft, where u, v and w describe the velocities along the x, y, and z body axes. Finally, from the reference frames defined in the preceding subsection, we now represent the following aerodynamic quantities:

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right) \tag{3.4a}$$

$$\beta = \sin^{-1}\left(\frac{v}{V}\right) \tag{3.4b}$$

$$V = \sqrt{u^2 + v^2 + w^2} \tag{3.4c}$$

where α denotes the Angle of Attack (AoA) of the aircraft, β denotes the SideSlip Angle (SSA) of the aircraft, and V denotes the total airspeed of the aircraft. Note that in this work, it is assumed that there is no wind present.

Not all of the variables above are needed to define the dynamics of an aircraft fully. For example, one may choose between two state representations: the first represents the state in terms of the body axes in $\mathcal{B}$, using the body velocities $\boldsymbol{v} = [u, v, w]^\intercal$. The second defines the state in terms of the wind axes defined in $\mathcal{W}$, and uses the aerodynamic angles α and β, in addition to the total airspeed of the aircraft V. In both cases, the angular rates $\boldsymbol{w}$, attitude $\boldsymbol{\Phi}$, and position $\boldsymbol{p}$ are included in the state. The two models are equivalent in terms of rigid body dynamics, although the dynamics are formulated in different state vectors. However, in terms of modeling aerodynamic forces and moments, the two representations typically lead to slightly different aerodynamic models, which is seen later in this chapter. The difference lies in whether the aerodynamic forces and moments are modeled in the body frame or the wind frame.

In this thesis, the dynamics are modeled in the body frame $\mathcal{B}$ rather than the stability frame $\mathcal{S}$. The main reason is that the aircraft in question does not have sensors to measure α and β directly. On larger aircraft, these aerodynamic angles are typically measured using wind vanes mounted on the aircraft. However, for small aircraft, this is usually not the case. For small aircraft, it is common to model both the dynamics and the aerodynamic forces in the body frame [38, 39, 40, 44, 45, 46]. The author has not seen any recent examples where the dynamics are formulated entirely in the wind frame as is it formulated in traditional textbooks such as [19, 20], the closest being the somewhat similar state formulations used in [23, 24]. However, many researchers use the body frame formulation of the state in combination with a wind frame formulation of the aerodynamic model, as is also done in this thesis. In this formulation used in for example [8, 36, 41, 54, 57], the aerodynamic angles are estimated, and forces and moments are calculated in terms of aerodynamic angles before being rotated back to the body frame where the system state and dynamics are defined. In these cases, the aerodynamic angles are typically approximated as in eq. (3.4c) by assuming that there is no wind.

In this work, not all aircraft states are included in the state formulation, and the state also consists of actuator states. In particular, only the attitude, angular

velocity, and translational velocities are of interest for the aircraft dynamics, as the inertial position does not impact the dynamic equations. This is seen in section 3.2.1. Hence, the aircraft state consists only of the attitude, angular body velocities, and translational body velocities. In addition to the aircraft state, the control surface deflections are included in the state. This is because the control surfaces have internal dynamics, which play a significant part in the aircraft dynamics, and therefore need to be included in the model. Let the actual control surface deflections be denoted by $\boldsymbol{\delta} = [\delta_a,\, \delta_e,\, \delta_r]^\intercal$, which corresponds to the deflection angles of the ailerons, elevators and rudders given in radians.

Finally, the inputs to the system consists of both fixed-wing inputs and multirotor inputs. The fixed-wing inputs are the control surface deflection setpoints δ_a^{sp}, δ_e^{sp} and δ_r^{sp}, in addition to the squared commanded pevolutions per second (RPS) of the fixed-wing propeller, $\delta_t \triangleq \eta_{\text{FW}}^2$. The multirotor inputs consist of the squared RPS of each of the $i \in [1, 4]$ multirotor propellers, $\delta_{\text{MR},i} \triangleq \eta_{\text{MR},i}^2$

The entire state and input vector are thus given by

$$\boldsymbol{x} = [\boldsymbol{\Phi}^\intercal\, \boldsymbol{\omega}^\intercal\, \boldsymbol{v}^\intercal\, \boldsymbol{\delta}^\intercal]^\intercal = [\phi,\, \theta,\, \psi,\, p,\, q,\, r,\, u,\, v,\, w,\, \delta_a,\, \delta_e,\, \delta_r]^\intercal \tag{3.5}$$

$$\boldsymbol{u} = [\boldsymbol{u}_{\text{FW}}^\intercal,\, \boldsymbol{u}_{\text{MR}}^\intercal]^\intercal = [\delta_a^{sp},\, \delta_e^{sp},\, \delta_r^{sp},\, \delta_t,\, \delta_{\text{MR},1},\, \delta_{\text{MR},2},\, \delta_{\text{MR},3},\, \delta_{\text{MR},4}]^\intercal \tag{3.6}$$

3.2 Rigid-Body Equations of Motion for Aircrafts

The EOMs for a rigid body consists of both kinematic and dynamic equations. The kinematic relationships are integrations dependent only on states and are not directly affected by inputs or disturbances to the system. In particular, there are two sets of kinematic equations for an aircraft, namely the navigation equations and the rotational kinematic equations. The dynamics of an aircraft consist of the force and moment equations, which are where inputs and disturbances enter the system.

3.2.1 Navigation Equations

The navigation equations describe the kinematic relationships between the inertial position $\boldsymbol{p}$, attitude $\boldsymbol{\Phi}$ and body velocities $\boldsymbol{v}$:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{h} \end{bmatrix} = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ \sin\theta & -\cos\theta\sin\phi & -\cos\theta\cos\phi \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{3.7a}$$

Note that the relationship is kinematic, and that the position can be seen as a pure output of the system. As the position does not play any role in the rest of the aircraft dynamics, the position and navigation equations are omitted from the model, and included here only for the sake of completeness.

3.2.2 Rotational Kinematic Equations

The rotational kinematic equations relate the rate of change of the Euler angles to the body-axis components of the angular velocity:

$$\dot{\phi} = p + \tan\theta(q\sin\phi + r\cos\phi) \tag{3.8a}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{3.8b}$$

$$\dot{\psi} = \frac{q\sin\phi + r\cos\phi}{\cos\theta} \tag{3.8c}$$

3.2.3 Force Equations

The aircraft is assumed to be a rigid body. By using Newtons Second Law of Motion for a rigid body undergoing translational motion, one obtains the expression for the

translational dynamics of the aircraft described in the body frame $\mathcal{B}$ as

$$\boldsymbol{F}_{\text{tot}} = \mathsf{m}\left(\dot{\boldsymbol{v}} + \boldsymbol{\omega} \times \boldsymbol{v}\right) \tag{3.9}$$

where $\boldsymbol{F}_{\text{tot}} = [f_x, f_y, f_z]^\intercal$ denotes the sum of forces acting in the body frame $\mathcal{B}$, and m denotes the mass of the aircraft (note the different font to differentiate it from the aerodynamic x-moment m). Expanding the cross product in eq. (3.10) and rearranging gives the individual translational dynamics:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} rv - qw \\ pw - ru \\ qu - pv \end{bmatrix} + \frac{1}{\mathsf{m}} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \tag{3.10}$$

There are multiple forces acting on the aircraft. The total sum of forces acting on the rigid body are modelled in the body frame as

$$\boldsymbol{F}_{\text{tot}} = \boldsymbol{F}_{\text{G}} + \boldsymbol{F}_{\text{A}} + \boldsymbol{F}_{\text{T}} + \boldsymbol{F}_{\text{MR}} \tag{3.11}$$

where $\boldsymbol{F}_G$ denotes the gravitational force acting on the body, $\boldsymbol{F}_{\text{A}} = [X,\, Y,\, Z]^\intercal$ denotes the total aerodynamic forces, $\boldsymbol{F}_{\text{T}} = [T,\, 0,\, 0]^\intercal$ denotes the total thrust force from the fixed-wing propeller acting along the body x-axis, and $\boldsymbol{F}_{\text{MR}} = [0,\, 0,\, -T_{\text{MR}}]^\intercal$ denotes the total force generated by the multirotor propellers in the direction of the z-body axis. Modeling of both the aerodynamic forces and the propellers forces are covered in their own sections.

The gravitational force is defined in the inertial frame $\mathcal{N}$, and is rotated to the body frame $\mathcal{B}$. Using Euler angles, the gravitational force is given in the body frame as

$$\boldsymbol{F}_G = \boldsymbol{R}_{\mathcal{N}}^{\mathcal{B}} \begin{bmatrix} 0 \\ o \\ \mathsf{m}g \end{bmatrix} = \mathsf{m}g \begin{bmatrix} -\sin\theta \\ \sin\phi\cos\theta \\ \cos\phi\cos\theta \end{bmatrix} \tag{3.12}$$

Using the result in eq. (3.12) and writing the component form of all the forces acting on the aircraft, the total force equations are given as

$$\dot{u} = rv - qw + \frac{1}{\mathsf{m}}(X + T - \mathsf{m}g\sin\theta) \tag{3.13a}$$

$$\dot{v} = pw - ru + \frac{1}{\mathsf{m}}(Y + \mathsf{m}g\sin\phi\cos\theta) \tag{3.13b}$$

$$\dot{w} = qu - pv + \frac{1}{\mathsf{m}}(Z - T_{\mathrm{MR}} + \mathsf{m}g\cos\phi\cos\theta) \tag{3.13c}$$

3.2.4 Moment Equations

Equivalently, by using Newtons Second Law of Motion for a rigid body undergoing rotational motion, the angular dynamics of the aircraft are described in the body frame $\mathcal{B}$ as

$$\boldsymbol{\tau}_{\mathrm{tot}} = \boldsymbol{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \boldsymbol{I}\boldsymbol{\omega} \tag{3.14}$$

where $\boldsymbol{\tau}_{\mathrm{tot}} = [\tau_x, \tau_y, \tau_z]^\intercal$ denotes the sum of moments acting in the body frame $\mathcal{B}$. $\boldsymbol{I}$ denotes the inertia matrix of the aircraft. The individual attitude dynamics in eq. (3.14) are formulated as:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \Gamma_1 pq - \Gamma_2 qr \\ \Gamma_5 pr - \Gamma_6(p^2 - r^2) \\ \Gamma_7 pq - \Gamma_1 qr \end{bmatrix} + \begin{bmatrix} \Gamma_3\tau_x + \Gamma_4\tau_z \\ \frac{1}{J_{yy}}\tau_y \\ \Gamma_4\tau_x + \Gamma_8\tau_z \end{bmatrix} \tag{3.15}$$

Where it is assumed that the aircraft is symmetric around the plane spanned by the body x-axis and y-axis, such that the moment of inertia of the aircraft is given as

$$\boldsymbol{I} = \begin{bmatrix} J_{xx} & 0 & -J_{xz} \\ 0 & J_{yy} & 0 \\ -J_{xz} & 0 & J_{zz} \end{bmatrix} \tag{3.16}$$

From this, the constants $\Gamma_i, i \in [1, 8]$ are defined as

$$\Gamma = J_{xx}J_{zz} - J_{xz}^2 \tag{3.17a}$$

$$\Gamma_1 = \frac{J_{xz}(J_{xx} - J_{yy} + J_{zz})}{\Gamma} \tag{3.17b}$$

$$\Gamma_2 = \frac{(J_{zz}(J_{zz} - J_{yy}) + J_{xz}^2)}{\Gamma} \tag{3.17c}$$

$$\Gamma_3 = \frac{J_{zz}}{\Gamma} \tag{3.17d}$$

$$\Gamma_4 = \frac{J_{xz}}{\Gamma} \tag{3.17e}$$

$$\Gamma_5 = \frac{(J_{zz} - J_{xx})}{J_{yy}} \tag{3.17f}$$

$$\Gamma_6 = \frac{J_{xz}}{J_{yy}} \tag{3.17g}$$

$$\Gamma_7 = \frac{J_{xx}(J_{xx} - J_{yy}) + J_{xz}^2}{\Gamma} \tag{3.17h}$$

$$\Gamma_8 = \frac{J_{xx}}{\Gamma} \tag{3.17i}$$

The sum of moments acting on the rigid body are modelled in the body frame as

$$\boldsymbol{\tau}_{\text{tot}} = \boldsymbol{\tau}_{\text{A}} + \boldsymbol{\tau}_{\text{MR}_Q} + \boldsymbol{\tau}_{\text{MR}_T} \tag{3.18}$$

where $\boldsymbol{\tau}_{\text{A}} = [l,\, m,\, n]^\intercal$ denotes the total aerodynamic moments, $\boldsymbol{\tau}_{\text{MR}_T} = [\tau_{\text{MR}_{T,x}},\, \tau_{\text{MR}_{T,y}},\, 0]^\intercal$ denotes the total moments due to the multirotor propeller thrust forces and their displacement from the center of mass, and $\boldsymbol{\tau}_{\text{MR}_Q} = [0,\, 0,\, Q_{\text{MR}}]^\intercal$ denotes the aerodynamic propeller moments form the multirotor propellers. The fixed-wing propeller moment is neglected, as this is measured from experiments to be small. Modeling of the aerodynamic moments and the multirotor propeller moments are covered in the next subsections.

During normal fixed-wing flight, the multirotor actuators are assumed to be

switched off. In that case, the moment equations are given simply as

$$\dot{p} = \Gamma_1 pq - \Gamma_2 qr + \Gamma_3 l + \Gamma_4 n \tag{3.19a}$$

$$\dot{q} = \Gamma_5 pr - \Gamma_6 (p^2 - r^2) + \frac{1}{J_{yy}} m \tag{3.19b}$$

$$\dot{r} = \Gamma_7 pq - \Gamma_1 qr + \Gamma_4 l + \Gamma_8 n \tag{3.19c}$$

3.3 Aerodynamic Modeling

The main task of aircraft system identification is typically modeling aerodynamic forces and moments acting on the aircraft in flight. As stated at the beginning of the thesis, there is rich literature on this topic for conventional aircraft, as it has been an active area of research for decades. For small fixed-wing UAVs, the literature is not as mature, although research has been emerging over the past years. In short, there are many established conventions for aerodynamic modeling, and researchers tend to use different techniques and combinations of techniques. Most of the modeling principles used for this thesis are from [19], which covers a variety of practices typically employed. Similar modeling techniques are found in textbooks as [20, 54]. This chapter outlines the specific modeling principles used for this work.

For full-scale conventional aircraft in quasi-steady flow at low Mach numbers, $M = \frac{\text{flow speed}}{\text{speed of sound}}$, the aerodynamic forces and moments are typically functions of α, β, V, angular rates $\boldsymbol{\omega}$, aircraft controls $\boldsymbol{\delta}$, $\dot{\alpha}$ and $\dot{\beta}$. In reality and for the full flight envelope, this relationship can be nonlinear and quite complex. Although it can be tempting to create complex models to fit the data better, it is important to identify a suitable model structure that matches the flight data well without overfitting the data through overly complex model structures. A simpler model that explains the data well is preferred over a complex model, as the addition of more parameters increases the parameter uncertainty [19, 20, 39].

For the aerodynamic modeling in this thesis, the following assumptions are adapted from [19]:

Assumption 1 *The aerodynamic model is developed around a reference flight condition, implying that model validity is localized to conditions close to the reference condition.*

Assumption 2 *The reference flight regime follows quasy-steady aerodynamics, meaning that the flowfield adjusts instantaneously to changes.*

Assumption 3 *The dependence of longitudinal forces and moments on lateral variables can be neglected, and vice versa.*

Assumption 4 *For subsonic flight, airspeed changes do not affect the aerodynamic coefficients.*

Using standard notation within the aerospace literature, let the aerodynamic forces in the body frame be denoted as $\boldsymbol{F}_{\mathrm{A}} = [X, Y, Z]^{\intercal}$, and the aerodynamic moments in the body frame be given as $\boldsymbol{\tau}_{\mathrm{A}} = [l, m, n]^{\intercal}$. This chapter covers the modeling of forces and moments as functions of the aircraft states and inputs and presents the chosen modeling techniques and assumptions. Methods for finding the actual parameter values and the best-suited model structure are presented in the next chapter, which is devoted to principles of system identification for fixed-wing aircraft.

From assumption 3, individual aerodynamic models for the longitudinal and the lateral-directional aircraft dynamics are developed. The longitudinal motion of an aircraft refers to translational motion along the x-axis and z-axis, in addition to rotational motion around the y-axis. The corresponding forces and moments are X, Z and m, with the related states $\boldsymbol{x}_{\mathrm{lon}} = [u, w, q, \theta]^{\intercal}$ and the inputs $\boldsymbol{u}_{\mathrm{lon}} = [\delta_e^{sp}, \delta_t]^{\intercal}$. The lateral-direction motion of an aircraft refers to translational motion along the body y-axis, in addition to rotational motion around both the x-axis and z-axis. The corresponding forces and moments are Y, l and n, with the related states $\boldsymbol{x}_{\mathrm{lat}} = [v, p, r, \phi, \psi]^{\intercal}$ and the inputs $\boldsymbol{u}_{\mathrm{lat}} = [\delta_a^{sp}, \delta_r^{sp}]^{\intercal}$. However, as will be seen, a single coupling parameter between longitudinal motion and lateral-directional motion is added in the end, to capture the rudder-pitch coupling found to be significant for large rudder deflections.

The details of the aerodynamic models are derived and presented in this section. From principles outlined in [19] combined with more recent works on smaller fixed-wing aircraft, the final result is a model where all the aerodynamic coefficients are modeled as a sum of terms that are nonlinear in the aerodynamic angles α, β, linear in the angular rates p, q, and r, and linear in control surfaces δ_a, δ_e, and δ_r. The linear terms for angular rates and control surfaces are also nonlinear in α and β. As a simple model is always preferable, the derivatives associated with the angular rates and control surfaces are only considered constant or linear functions of the aerodynamic angles. Functional dependencies of airspeed dependencies are not considered in this work, following assumption 4.

First, the traditional aerodynamic model typically used for large-scale aircraft is presented. Then, with the aim of expanding the model's validity to to a larger flight envelope, the model is augmented to employ the aerodynamic angles α, β rather than the body velocities u, v, w. Finally, the model is augmented with nonlinear terms that are functions of the aerodynamic angles. The longitudinal model is presented first, followed by the lateral-directional model. Finally, the augmentation to capture the rudder-pitch coupling is presented.

3.3.1 Traditional Perturbation Model

In [19, 20, 48], traditional aerodynamic models are developed around certain reference conditions. These conditions are referred to as trim, which for normal flight typically means a specific airspeed, AoA and the constant control surface deflections required to cancel out all moments about this condition. If the airplane is modeled in body velocities, airspeed and aerodynamic angles are replaced by body velocities u, v, and w. Such a model is typically linear and usually describes the motion close to the trim conditions well. For example, in [39, 40], such a model is employed successfully for a small fixed-wing airplane flying at a constant throttle setting. As the perturbation model is the foundation for the final aerodynamic model in this work, it is presented here for completeness. However, in the next sections, this model is expanded and changed before arriving at the final model.

Following the assumption of quasi-steady aerodynamics in assumption 2, the aerodynamic forces and moments in the body frame $\mathcal{B}$ can be expressed as linear Taylor series expansion about a reference condition, giving the following expressions for the longitudinal forces and moments:

$$X = X_0 + X_u \Delta u + X_w \Delta w + X_q q + X_{\delta_e} \Delta \delta_e \tag{3.20a}$$

$$Z = Z_0 + Z_u \Delta u + Z_w \Delta w + Z_q q + Z_{\delta_e} \Delta \delta_e \tag{3.20b}$$

$$m = m_0 + m_u \Delta u + m_w \Delta w + m_q q + m_{\delta_e} \Delta \delta_e \tag{3.20c}$$

where $\Delta u = u - u^*$, $\Delta w = w - w^*$ and $\Delta \delta_e = \delta_e - \delta_{e,0}$ denote the perturbations from the reference condition $\boldsymbol{x}^*$, $\boldsymbol{u}^*$. Note that for the angular rates, the reference condition is defined as $p^*, q^*, r^* = 0$.

Simmilarly, first order Taylor expansion about the reference condition gives the following lateral-directional aerodynamic model

$$Y = Y_0 + Y_v v + Y_p p + Y_r r + Y_{\delta_a} \Delta \delta_a + Y_{\delta_r} \Delta \delta_r \tag{3.21a}$$

$$l = l_0 + l_v v + l_p p + l_r r + l_{\delta_a} \Delta \delta_a + l_{\delta_r} \Delta \delta_r \tag{3.21b}$$

$$n = n_0 + n_v v + n_p p + n_r r + n_{\delta_a} \Delta \delta_a + n_{\delta_r} \Delta \delta_r \tag{3.21c}$$

where $\Delta \delta_a = \delta_a - \delta_{a,0}$, $\Delta \delta_r = \delta_r - \delta_{r,0}$,denote the perturbations from the reference condition $\boldsymbol{x}^*$, $\boldsymbol{u}^*$. For the angular rates, the reference condition is still $p^*, q^*, r^* = 0$, and for the body velocity in the y-direction $v^* = 0$.

It is often more convenient to work with nondimensional aerodynamic coefficients expressed in terms of nondimensional quantities, in order to eliminate the known dependence on airspeed V and air density ρ, and to normalize the partial derivatives. Let the nondimensional longitudinal aerodynamic coefficients be defined as

$$c_X = \frac{X}{\bar{q}S}, \quad c_Z = \frac{Z}{\bar{q}S}, \quad c_m = \frac{m}{\bar{q}S\bar{c}}, \tag{3.22}$$

and the lateral-directional coefficients as

$$c_Y = \frac{Y}{\bar{q}S}, \quad c_l = \frac{l}{\bar{q}Sb}, \quad c_n = \frac{n}{\bar{q}Sb}, \tag{3.23}$$

where S denotes the total surface area of the wings on the aircraft, $\bar{c}$ denotes the Mean Aerodynamic Chord (MAC) of the aircraft, b is the wingspan, and $\bar{q} = \frac{1}{2}\rho V^2$ is the dynamic pressure.

Further, define the following nondimensional quantities:

$$\hat{u} = \frac{u}{V_0}, \quad \hat{w} = \frac{w}{V_0}, \quad \hat{q} = \frac{\bar{c}q}{2V_0} \tag{3.24}$$

$$\hat{v} = \frac{v}{V_0}, \quad \hat{p} = \frac{bp}{2V_0}, \quad \hat{r} = \frac{br}{2V_0} \tag{3.25}$$

where V_0 denotes the reference airspeed.

It is now possible to express the aerodynamic coefficients as linear functions of the nondimensional quantities, which gives the traditional model of a full-scale aircraft operating around a reference condition and modelled in the body frame:

$$c_X = c_{X_0} + c_{X_u}\Delta\hat{u} + c_{X_w}\Delta\hat{w} + c_{X_q}\hat{q} + c_{X_{\delta_e}}\Delta\delta_e \tag{3.26a}$$

$$c_Z = c_{Z_0} + c_{Z_u}\Delta\hat{u} + c_{Z_w}\Delta\hat{w} + c_{Z_q}\hat{q} + c_{Z_{\delta_e}}\Delta\delta_e \tag{3.26b}$$

$$c_m = c_{m_0} + c_{m_u}\Delta\hat{u} + c_{m_w}\Delta\hat{w} + c_{m_q}\hat{q} + c_{m_{\delta_e}}\Delta\delta_e \tag{3.26c}$$

Similarly, for the lateral-directional motion:

$$c_Y = c_{Y_0} + c_{Y_v}v + Y_p p + c_{Y_r}r + c_{Y_{\delta_a}}\Delta\delta_a + c_{Y_{\delta_r}}\Delta\delta_r \tag{3.27a}$$

$$c_l = c_{l_0} + c_{l_v}v + c_{l_p}p + c_{l_r}r + c_{l_{\delta_a}}\Delta\delta_a + c_{l_{\delta_r}}\Delta\delta_r \tag{3.27b}$$

$$c_n = c_{n_0} + c_{n_v}v + c_{n_p}p + c_{n_r}r + c_{n_{\delta_a}}\Delta\delta_a + c_{n_{\delta_r}}\Delta\delta_r \tag{3.27c}$$

For symmetric aircraft, usually c_{Y0}, c_{l0}, and c_{n0} are zero. The fixed-wing propeller generally causes a moment about the x-axis, which is trimmed by having some nonzero

deflection in the ailerons, $\delta_a^* \neq 0$. For symmetric aircraft, $\delta_r^* = 0$.

3.3.2 Linear Aerodynamic Model in AoA and SSA

In this work, modifications to the traditional aerodynamic model presented above are made. To more accurately model the aerodynamics over a flight envelope consisting of different airspeeds, the aerodynamic model is developed around the absolute aerodynamic angles α, β, rather than around the local reference conditions u^*, v^*, and w^*. This is similar to many recent works on small fixed-wing UAVs where the body velocity is employed to model the aircraft state while using the aerodynamic angles to model the aerodynamic forces and moments, such as in [8, 11, 54].

Therefore, we instead model the aerodynamic forces as they appear in reality. The aerodynamic x and z-forces originate from the lift force L and the drag force D acting in the stability frame $\mathcal{S}$. The aerodynamic pitch moment is still modeled in the body frame $\mathcal{B}$. Further, we now model all the forces and moments in the body frame as functions of the aerodynamic angles α, β, the nondimensionalized pitch rate $\hat{q}$, and the deflection of the elevator from the reference condition, $\Delta\delta_e \triangleq \delta_e - \delta_e^*$. This results in the following longitudinal aerodynamic coefficients acting in the body frame $\mathcal{B}$:

$$c_X = -c_D \cos\alpha + c_L \sin\alpha \tag{3.28a}$$

$$c_Z = -c_D \sin\alpha - c_L \cos\alpha \tag{3.28b}$$

$$c_m = c_{m0} + c_{m\alpha}\alpha + c_{m\hat{q}}\hat{q} + c_{m\delta_e}\Delta\delta_e \tag{3.28c}$$

where the lift and drag coefficients are modelled as

$$c_L = c_{L0} + c_{L\alpha}\alpha + c_{L\hat{q}}\hat{q} + c_{L\delta_e}\Delta\delta_e \tag{3.29a}$$

$$c_D = c_{D0} + c_{D\alpha}\alpha + c_{D\hat{q}}\hat{q} + c_{D\delta_e}\Delta\delta_e \tag{3.29b}$$

Similarly, the lateral-directional aerodynamics are all modelled in the body frame $\mathcal{B}$ as functions of the lateral-directional aerodynamic angle β, lateral-directional nondimensionalized angular rates $\hat{p}$, $\hat{r}$, and the lateral-directional control surface deflections

from trim $\Delta\delta_a \triangleq \delta_a - \delta_a^*$, $\Delta\delta_r \triangleq \delta_r - \delta_r^*$:

$$c_Y = c_{Y0} + c_{Y\beta}\beta + c_{Y\hat{p}}\hat{p} + c_{Y\hat{r}}\hat{r} + c_{Y\delta_a}\Delta\delta_a + c_{Y\delta_r}\Delta\delta_r \tag{3.30a}$$

$$c_l = c_{l0} + c_{l\beta}\beta + c_{l\hat{p}}\hat{p} + c_{l\hat{r}}\hat{r} + c_{l\delta_a}\Delta\delta_a + c_{l\delta_r}\Delta\delta_r \tag{3.30b}$$

$$c_n = c_{n0} + c_{n\beta}\beta + c_{n\hat{p}}\hat{p} + c_{n\hat{r}}\hat{r} + c_{n\delta_a}\Delta\delta_a + c_{n\delta_r}\Delta\delta_r \tag{3.30c}$$

The aerodynamic coefficients defined in eqs. (3.28) and (3.30) are called nomdimensional stability and control derivatives. The static stability derivates are those associated with the aerodynamic angles (α, β) (or in the perturbation model, the air-relative velocities (u, v, w)). The dynamic stability derivatives are those associated with angular rates (p, q, r). The control derivatives are those associated with the controls $(\delta_a, \delta_e, \delta_r)$. In addition to this, there are the derivatives associated with unsteady aerodynamics $\dot{\alpha}$ (or $\dot{w}$). However, the latter is often hard to differentiate as an independent coefficient, as $\dot{\alpha}$ (or $\dot{w}$) and q often follows the same time-histories, and these parameters are therefore lumped together as c_{a_q} for $a = D$, L and m as proposed in [19].

3.3.3 Nonlinear Aerodynamic Model

The linear dependencies in section 3.3.2 typically provide a good representation for the aerodynamic forces and moments in most practical situations. However, it may be desirable to extend the model with nonlinear terms for large amplitudes or rapid deviations from the reference flight conditions [19]. Here, the aerodynamic model will be augmented with nonlinear terms.

As stated in the opening, the linearly involved terms may be nonlinear in α and sometimes in β. To keep the model structure simple, the derivatives associated with the angular rates and control surfaces are only considered constant or linear functions of the aerodynamic angles. For the aerodynamic static stability derivatives, squared terms are considered. Adding these nonlinear dependencies, we arrive at the final aerodynamic models that is considered in this work. For the longitudinal aerodynamic model, the cofficients are modelled as follows:

$$c_D = c_{D0} + c_D(\alpha, \alpha^2) + c_{D\hat{q}}(\alpha)\hat{q} + c_{D\delta_e}(\alpha)\Delta\delta_e \tag{3.31a}$$

$$c_L = c_{L0} + c_L(\alpha, \alpha^2) + c_{L\hat{q}}(\alpha)\hat{q} + c_{L\delta_e}(\alpha)\Delta\delta_e \tag{3.31b}$$

$$c_m = c_{m0} + c_m(\alpha, \alpha^2) + c_{m\hat{q}}(\alpha)\hat{q} + c_{m\delta_e}(\alpha)\Delta\delta_e \tag{3.31c}$$

Similarly, for the lateral-directional model:

$$c_Y = c_{Y0} + c_Y(\beta, \beta^2) + c_{Y\hat{p}}(\beta)\hat{p} + c_{Y\hat{r}}(\beta)\hat{r} + c_{Y\delta_a}(\beta)\Delta\delta_a + c_{Y\delta_r}(\beta)\Delta\delta_r \tag{3.32a}$$

$$c_l = c_{l0} + c_l(\beta, \beta^2) + c_{l\hat{p}}(\beta)\hat{p} + c_{l\hat{r}}(\beta)\hat{r} + c_{l\delta_a}(\beta)\Delta\delta_a + c_{l\delta_r}(\beta)\Delta\delta_r \tag{3.32b}$$

$$c_r = c_{r0} + c_r(\beta, \beta^2) + c_{r\hat{p}}(\beta)\hat{p} + c_{r\hat{r}}(\beta)\hat{r} + c_{r\delta_a}(\beta)\Delta\delta_a + c_{r\delta_r}(\beta)\Delta\delta_r \tag{3.32c}$$

where $c_k(x, x^2) = c_{kx}x + c_{kx^2}x^2$ and $c_k(x) = c_{kx}x$ denote quadratic and linear functions, with $k = D, Y, L, l, m, n$ and $x = \alpha, \beta$.

Equation (3.31) and eq. (3.32) represents the full aerodynamic models that is considered for the identification and modelling of the aerodynamic coefficients in this thesis. Although the entire nonlinear models are presented here, the specific model structure for each coefficient is developed based on experimental data. As will be seen, individual subsets of the model structure is chosen for each coefficient. The details of the procedure for determining this model structure are presented in chapter 5.

3.3.4 Rudder-Pitch Coupling

Because of the control surface configuration on the aircraft, there is a coupling between the rudder deflection δ_r and the pitch, which cannot be neglected. This is caused by the V-tail control surfaces illustrated in fig. 3-1, where a differential deflection causes a change in the pitch moment. An important fact to note is that the sign of the differential deflection does not matter for the generated pitch moment. In this work, FTD is used to find the function that best captures the rudder-pitch coupling. In chapter 6, it is found that a quadratic relationship between c_m and δ_r best describes

the data. Therefore, the pitch moment model is augmented as follows:

$$c_m = c_{m0} + c_m(\alpha, \alpha^2) + c_{m\hat{q}}(\alpha)\hat{q} + c_{m\delta_e}(\alpha)\Delta\delta_e + c_{m\delta_r^2}\delta_r^2 \tag{3.33}$$

As this parameter only affects the aircraft motion for large rudder deflections, $c_{m\delta_r^2}$ is estimated in a final, separate step, described in section 6.1.4

3.4 Linear Aircraft Models around Trim

The nonlinear equations presented in the preceding sections can be linearized around trim conditions. This is useful for both analysis and control design. In this work, the linear dynamics models is used to analyze the modes which describe the aircraft's fixed-wing behavior around trim. Here, the linear models are only summarized and explained briefly. For a complete derivation, the reader is referred to for example [54], [48] or [44]. The reader should note that this linearized model only includes the fixed-wing regime of flight, and that neither control surface dynamics or multirotor dynamics are included in this linear model.

First, the longitudinal dynamics may be described by the state $\boldsymbol{x}_{\text{lon}} = [u,\, w,\, q,\, \theta]^\intercal$, and the input $\boldsymbol{u}_{\text{lon}} = [\delta_e,\, \delta_t]^\intercal$. This gives the following linear system for the longitudinal motion of the aircraft around trim $\boldsymbol{x}_{\text{lon}} = \boldsymbol{x}^*_{\text{lon}}$:

$$\dot{\boldsymbol{x}}_{\text{lon}} = \boldsymbol{A}_{\text{lon}}\boldsymbol{x}_{\text{lon}} + \boldsymbol{B}_{\text{lon}}\boldsymbol{u}_{\text{lon}} \tag{3.34}$$

where

$$\boldsymbol{A}_{\text{lon}} = \begin{bmatrix} X_u & X_w & X_q & -g\cos\theta^* \\ Z_u & Z_w & Z_q & -g\sin\theta^* \\ M_u & M_w & M_q & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \boldsymbol{B}_{\text{lon}} = \begin{bmatrix} X_{\delta_e} & X_{\delta_t} \\ Z_{\delta_e} & 0 \\ M_{\delta_e} & 0 \\ 0 & 0 \end{bmatrix} \tag{3.35}$$

and trim conditions defined as

$$\boldsymbol{x}^*_{\text{lon}} = [u^*,\, w^*,\, 0,\, \theta^*]^\intercal \tag{3.36a}$$

$$\boldsymbol{u}^*_{\text{lon}} = [\delta^*_e,\, \delta^*_t]^\intercal \tag{3.36b}$$

Through an eigenmode decomposition of $\boldsymbol{A}_{\text{lon}}$ one obtaines two complex-conjugated pole-pairs. These are called the short-period mode and the phugoid mode. The short-period mode is associated with damped oscillations in pitch about the y-axis, and usually has a natural frequency between 1rad/s and 10rad/s. The airspeed usually remains constant during an excitation of the short-period mode. The phugoid mode is a very light damped low-frequency mode, with oscillations in the speed u, where α remains constant. The phugoid mode is coupled with all states, including the height of the system. Usually, the natural frequency of the phugoid mode ranges from 0.1rad/s to 1rad/s.

Similarly, the lateral-directional dynamics may be described by the state $\boldsymbol{x}_{\text{lat}} = [v,\, p,\, r,\, \phi]^\intercal$ and the input $\boldsymbol{u}_{\text{lat}} = [\delta_a, \delta_r]^\intercal$. This gives the following linear system for the lateral-directional motion of an aircraft around trim $\boldsymbol{x} = \boldsymbol{x}^*$:

$$\dot{\boldsymbol{x}}_{\text{lat}} = \boldsymbol{A}_{\text{lat}}\boldsymbol{x}_{\text{lat}} + \boldsymbol{B}_{\text{lat}}\boldsymbol{u}_{\text{lat}} \tag{3.37}$$

where

$$\boldsymbol{A}_{\text{lat}} = \begin{bmatrix} Y_v & Y_p & Y_r & g\cos\theta^* \\ L_v & L_p & L_r & 0 \\ N_v & N_p & N_r & 0 \\ 0 & 1 & \tan\theta^* & 0 \end{bmatrix}, \quad \boldsymbol{B}_{\text{lat}} = \begin{bmatrix} Y_{\delta_a} & Y_{\delta_r} \\ L_{\delta_a} & L_{\delta_r} \\ N_{\delta_a} & N_{\delta_r} \\ 0 & 0 \end{bmatrix} \tag{3.38}$$

and trim conditions defined as

$$\boldsymbol{x}^*_{\text{lat}} = [0,\, 0,\, 0,\, 0]^\intercal \tag{3.39a}$$

$$\boldsymbol{u}^*_{\text{lat}} = [\delta^*_a,\, \delta^*_r]^\intercal \tag{3.39b}$$

From an eigenmode decomposition of $\boldsymbol{A}_{\text{lat}}$ one obtains three modes: two real poles and one complex-conjugated pole-pair. The two real poles correspond to the roll mode and the spiral divergence mode, and the complex-conjugated pole corresponds to the Dutch roll mode. The roll mode is is nonoscillatory and is typically fast, and is decoupled from the spiral and Dutch roll modes. The spiral mode is typically very slow and marginally stable, slightly stable, or even slightly unstable. It is very slow, in the order of 100s. If the spiral mode is unstable, this causes the aircraft to slowly spiral into the ground. Having an unstable mode may sound strange at first, but this is actually the case for some aircraft [48]. However, in real applications, the spiral mode is compensated for by the control system or pilot inputs, and as long as it is slow, this mode is not an issue. Finally, the Dutch roll mode is lightly damped oscillations around the z-axis. It is visible as oscillations in roll and yaw, with 90° phase lag between the motions.

3.5 Actuator Modeling

There are multiple actuators on the aircraft, which are separately modeled. The actuators consist of the fixed-wing actuators and the multirotor actuators. The fixed-wing actuators are the control surface deflections δ_a, δ_e, and δ_r, and the squared RPS of the fixed-wing propeller δ_t. The multi-rotor actuators consist only of the four multirotor propellers, where their squared RPS are denoted by $\delta_{\text{MR},i}$.

3.5.1 Control Surfaces Modeling

First, the input signals δ_a, δ_e, and δ_r are defined in terms of their signs and their corresponding control surfaces on the aircraft. Then, the control surfaces are modeled with their own dynamics, separate from the rest of the aircraft.

The input signals δ_a, δ_e and δ_r follow the standard sign conventions of aviation: a positive aileron angle δ_a leads to a positive rolling moment around the x-axis, a positive elevator angle δ_e leads to a negative pitching moment around the y-axis, and a positive rudder angle δ_r leads to a negative yawing moment around the z-axis. An

illustration of this can be seen in fig. 3-1.

Figure 3-1: Sign definitions for control surfaces.

The control surfaces are modelled as if it is possible to control the aileron deflection angle δ_a, elevator deflection angle δ_e and rudder deflection angle δ_r separately. In reality, the aircraft only has two pairs of control surfaces, namely a pair of ailerons and a pair of ruddervators in an inverted V-tail configuration. Therefore, the elevator and rudder signals, δ_e and δ_r, are combined to compute the true deflection angles of the actual control surfaces. Deflecting the ruddervators together has the same effect as an elevator, and deflecting them differentially has the same effect as a rudder. Conversion from δ_e and δ_r is performed by the autopilot mixer. In this thesis, the relationship between the elevator, rudder, and ruddervators is defined as

$$\begin{bmatrix} \delta_e \\ \delta_r \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \delta_{rr} \\ \delta_{rl} \end{bmatrix} \tag{3.40}$$

where δ_{rr} and δ_{rl} denote the right and left ruddervator deflection angles measured in radians. The convention in eq. (3.40) is the same as the one proposed in [54]. Note

that the autopilot commands and logs signals in δ_a, δ_e, and δ_r, which are the signals that are used for the identification procedure.

Next, the control surfaces have dynamics of their own that need to be modeled. The control surfaces of the aircraft are controlled with servos, which have limitations in terms of rates. If these servos are fast enough compared to the dynamics of the aircraft, their dynamics can be neglected. However, through the experiments, it is found that the bandwidth of the servos is too close to the aircraft dynamics to be ignored. There are a total of four servos for the ailerons and ruddervators, and the servos are identical. Through experiments, it was found that the servos seem to follow a rate-limited first-order response, described by the following model:

$$\dot{\delta}_i = \text{sat}\left(\frac{-\frac{1}{T}\delta_i + \frac{1}{T}\delta_i^{sp}}{\dot{\delta}_{max}}\right)\dot{\delta}_{max}, \quad \text{sat}(x) = \begin{cases} x & \text{for } |x| \leq 1 \\ 1 & \text{for } x > 1 \end{cases} \tag{3.41}$$

The experiments and resulting model parameters are covered in section 4.3.2.

3.5.2 Fixed-Wing Propeller Modeling

In this work, the fixed-wing propeller thrust force is identified separately from the rest of the aerodynamics. This is in contrast to many other existing works for system identification of aircraft. The aircraft is often assumed to fly at a constant throttle setting, and the propeller thrust is included in the aerodynamic model. Examples of this can be found in [19, 38, 39, 40, 44, 45]. Separately modeling the fixed-wing propeller has some advantages, one being that it is easier to update the model if the fixed-wing propeller setup is changed.

The thrust generated by the fixed-wing propeller only acts along the body x-axis, and is denoted by T. It is modelled as a linear function of the quadratic RPS value η_{FW}, with $\delta_t \triangleq \eta_{\text{FW}}^2$:

$$T = \rho D_{\text{FW}}^4 c_{T_{\text{FW}}} \delta_t \tag{3.42a}$$

where D_{FW} denotes the diameter of the fixed-wing propeller, and $c_{T_{\text{FW}}}$ denotes the fixed-wing thrust coefficient.

The fixed-wing propeller spins along the positive body x-axis and generates a negative aerodynamic moment around the x-axis. This torque causes requires the aircraft to hold a constant aileron deflection at steady-state flight [58]. However, for the modeling in this thesis, this torque is assumed to be small and is hence neglected. Instead, this torque is captured in the aileron trim δ_a^* and c_{l0}.

It should be mentioned that the proposed propeller model does not capture the effect of a propeller that is moving through an airfield. For a propeller rotating at a constant RPS, increasing the airspeed decreases the effective AoA of the propeller blade. As a result, this reduces the propeller thrust. This phenomenon can be modeled with a more sophisticated model, such as using the advance ratio as is done in [8] or a numerically obtained lookup table as is done in [36]. However, this was not found necessary in this work, and the simple static model proposed in this section captures the aircraft motion well.

The entire aircraft is modeled as a rigid body with no moving parts. In reality, the propellers are rotating, which cause a gyroscopic moment that acts on the rigid body. However, this gyroscopic effect is considered negligible for the aircraft in question due to its large inertia properties.

3.5.3 Multirotor Modeling

As for regular quadcopters, there are four propellers mounted along the body z-axis of the aircraft. These generate moments around all body axes and a thrust along the body z-axis. The aircraft in this thesis follows the standard PX4 convention for propeller direction and order, as can be seen in fig. 3-2.

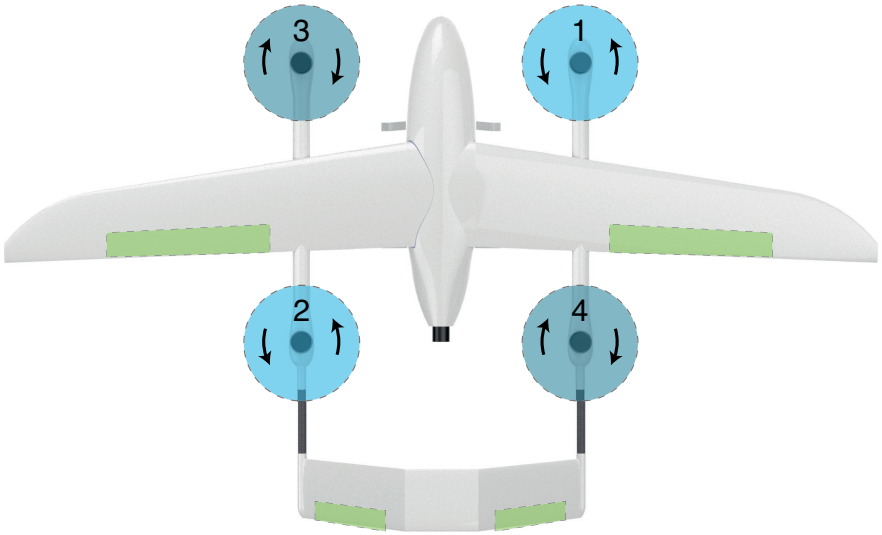


Figure 3-2: Ordering and direction of multirotor propellers.

Thrust

The thrusts generated by the multirotor propellers are modelled as a linear function of the multirotor inputs, which are defined as the squared RPS $\delta_{\mathrm{MR},i} \triangleq \eta_{\mathrm{MR},i}^2$:

$$T_{\mathrm{MR},i} = \rho D_{\mathrm{MR}}^4 c_{T_{\mathrm{MR}}} \delta_{\mathrm{MR},i} \tag{3.43a}$$

where D_{MR} is the diameter of the multirotor propellers, $\eta_{\mathrm{MR},i}$ denotes the rotational speed (RPM) of the i-th multirotor propeller, and $c_{T_{\mathrm{MR}}}$ is the thrust coefficient for the multirotor propellers.

The total force T_{MR} given by the propellers along the body z-axis is given as

$$T_{\mathrm{MR}} = \sum_{i=1}^{4} T_{\mathrm{MR},i} \tag{3.44}$$

Aerodynamic moments

Similarly, the aerodynamic moments generated by the multirotor propellers are modeled as

$$Q_{MR,i} = \rho D_{MR}^5 c_{Q_{\text{MR}}} \delta_{\text{MR},i} \tag{3.45a}$$

where $c_{Q_{\text{MR}}}$ denotes the moment coefficient of the multirotor propellers. There are four multirotor propellers, divided into two pairs of propellers that spin in the opposite direction. This is common for quadcopters, as it allows the aircraft to use the four propellers to control the moments around all the body axes.

The total moment around the body z-axis is given as

$$Q_{\text{MR}} = Q_{\text{MR},1} + Q_{\text{MR},2} - Q_{\text{MR},3} - Q_{\text{MR},4} \tag{3.46}$$

where the signs of the terms in $\boldsymbol{\tau}_Q$ are given by the rotational direction of the individual propellers, as is shown in fig. 3-2.

Moment due to thrust

As the thrust forces from the top propellers are not acting on the center of mass, they generate a moment around the x-axis and the y-axis. The resulting moment from the thrust forces generated by the propellers are given by

$$\boldsymbol{\tau}_{\text{MR}_T} = \sum_{i=1}^{4} \boldsymbol{r}_{p,i} \times \boldsymbol{T}_{\text{MR},i} = \sum_{i=1}^{4} \begin{bmatrix} r_{i,x} \\ r_{i,y} \\ r_{i,z} \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ -T_{\text{MR},i} \end{bmatrix} = \sum_{i=1}^{4} \begin{bmatrix} -r_{i,y} T_{\text{MR},i} \\ r_{i,x} T_{\text{MR},i} \\ 0 \end{bmatrix} \tag{3.47}$$

where $\boldsymbol{r}_{t,i}$ denotes the vector describing the location of the i-th multirotor propeller relative to the center of mass.

Chapter 4

UAV Airframe and Preliminary Analysis

In this chapter, a preliminary analysis of the VTOL UAV is presented. The UAV airframe is described in general in terms of design, actuation principle, instrumentation, and avionics. Static properties of the aircraft are investigated, such as inertial properties, propeller constants, and dynamics of the control surfaces. Finally, the trim properties of the aircraft obtained from preliminary manual experiments are summarized, and an initial aerodynamic model is obtained through the use of simplified surface modeling and a Computational Fluid Dynamics (CFD) method called Vortex Lattice Method (VLM).

4.1 Instrumentation and Avionics

There are several digital electronic components on the aircraft. For example, the Flight Controller (FC) running on the aircraft is a Pixhawk [42] 2.1, which is a microcontroller with multiple onboard sensors. The sensor suite on the Pixhawk 2.1 consists of three accelerometers, three gyroscopes, three magnetometers, and two barometers. In addition, an external GNSS system combined with a magnetometer is mounted on the outside of the airframe. In addition to this, there is a digital differential airspeed sensor aboard the vehicles, which measures both static and dynamic pressure from

a pitot tube mounted at the aircraft's nose, which allows estimation of the vehicle airspeed.

The FC onboard the aircraft is running PX4 [29], an open-source autopilot that is widely used. PX4 is running an elaborate estimation library in the form of an Extended Kalman Filter (EKF) running at a slightly delayed time horizon. As the EKF is running at a delayed time horizon, PX4 also employs a faster but less elaborate output predicter that is running in realtime, which is used for low-level control at high frequencies. The output predicter predicts states at realtime rates and uses the EKF to correct its states to prevent drifting. Both the states of the EKF and the output predictor outputs are logged at 200 Hz during the flight experiments performed in this work and combined to obtain the entire state of the aircraft.

4.2 Static Airframe Properties

Initially, neither the total mass of the aircraft, the inertia matrix, or the properties of the lifting surfaces are known. However, when seeking to accurately model the dynamics of an aircraft, these properties are essential to identify before aerodynamic system identification can be started. In this section, the mass, the inertia matrix, the properties of the lifting surfaces are found. A summary of the static aircraft properties can be found in table 4.1. Note how only four of the inertia terms are included, as the rest are assumed to be zero from symmetry arguments are described in chapter 3.

Parameter	Value
b	2.5m
$\bar{c}$	0.242m
S	0.6617m
m	12.140 kg
J_{xx}	0.7316
J_{yy}	1.0664
J_{zz}	1.6917
J_{xz}	0.1277

Table 4.1: Summary of all the static properties of the aircraft.

4.2.1 Mass

First, the mass of the aircraft is found. The aircraft is too large to be weighted as a single part, and the individual parts have to be weighed separately and summed up. The total aircraft mass with the specific avionics setup used for the work in this thesis is found to be $\mathsf{m} = 12.140$kg. To ease potential future work for researchers working with the same airframe but with a different avionics setup, the details of the masses can be found in table 4.2.

Part	Mass	Quantity
Battery 6500mah	0.950kg	2
Battery 11000mah	1.270kg	3
Wing (with servo)	0.770kg	2
Tail (with servos)	0.630kg	1
Fuselage (with avionics)	2.310kg	1
VTOL arm (with wiring)	0.280kg	2
Top motor (with propeller)	0.260kg	4
Pusher motor (with propeller)	0.350kg	2
Total	12.140kg	

Table 4.2: Masses of the individual parts of the aircraft.

4.2.2 Inertia Matrix

A critical static property of the aircraft is the inertia matrix $\boldsymbol{J}$, as introduced in chapter 3. In this work, an accurate 3D model is carefully created from physical measurements on the airframe, as it was desirable to have a 3D model of the aircraft for future work and visualization. Another common method is the compound pendulum method used in [39], and [57]. All parts assumed to contribute significantly to the aircraft inertia are included in the 3D model. The measured part masses are uniformly distributed on the part, and is accurately located on the model. The parts included in the model are the fuselage, wings, tail, VTOL arms, batteries, ESCs, and motors. Renderings of the 3D model can be seen in fig. 4-1. The 3D model is created in Solidworks. Note that the 3D renderings do not show the batteries, ESCs, and servos, as these are located within the airframe.

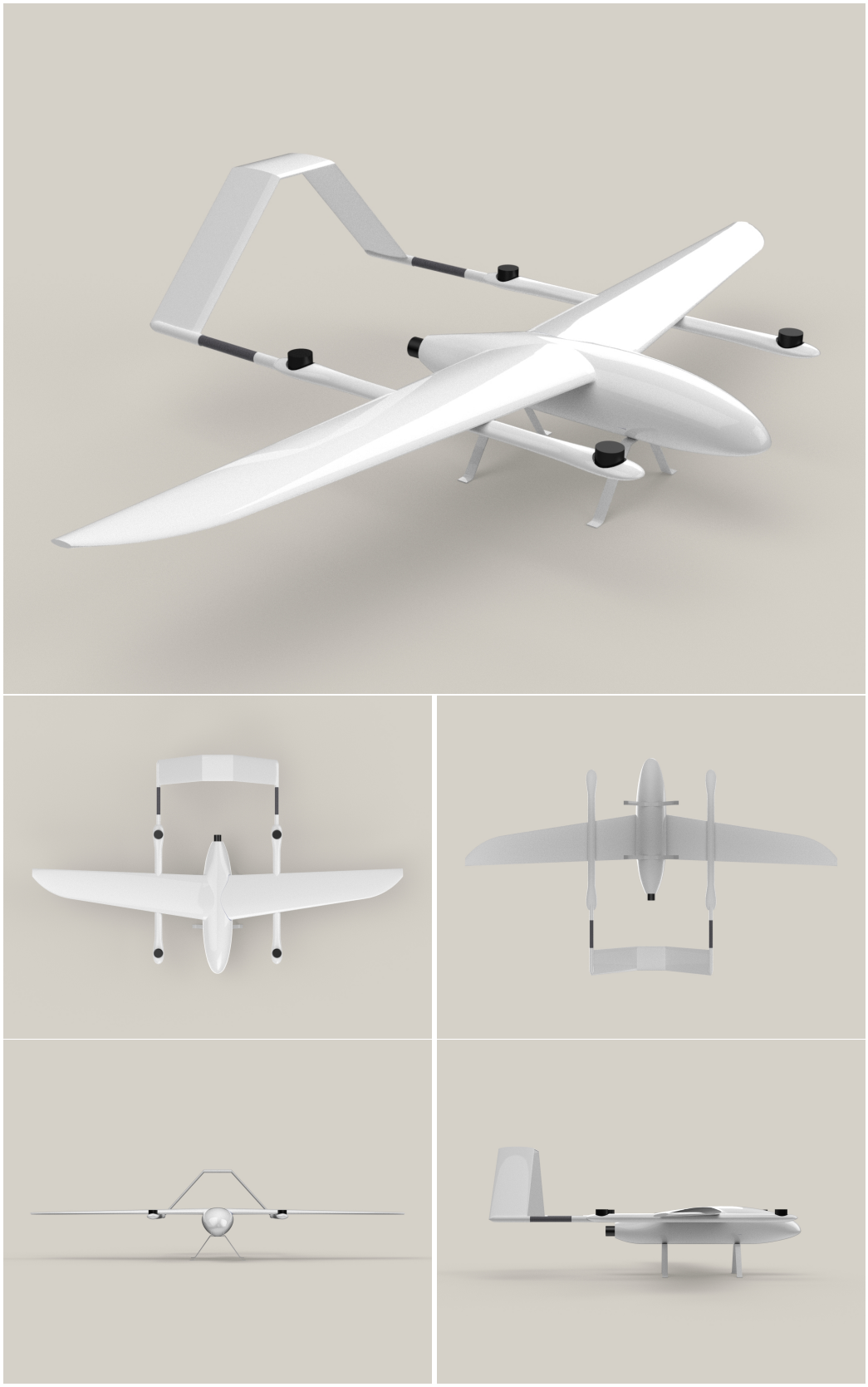


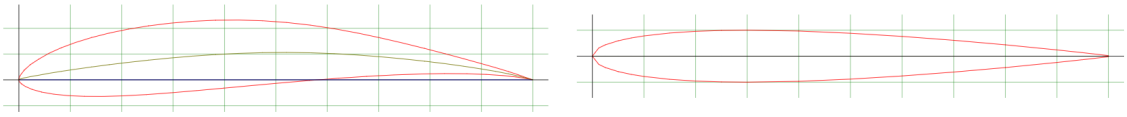
Figure 4-1: Detailed 3D model of the Babyshark.

The inertia matrix $\boldsymbol{J}$ is directly obtained from the 3D modeling software after the individual part masses are added. One advantage of this method is that the products of inertias on the off-diagonal of $\boldsymbol{J}$ can also be estimated, in contrast to using the compound pendulum method which only estimates the moments of inertia on the diagonal of $\boldsymbol{J}$. However, a downside is that recreating an accurate 3D model from measurements can be a time-demanding process. The final inertia matrix parameters obtained from the 3D model are summarized in table 4.1.

4.2.3 Lifting Surfaces

Next, the static properties of the wings are measured. There are a total of three lifting surfaces on the aircraft: two wings and one tail. The wingspan b is readily found from direct measurements on the aircraft to be 2.5m. The total lifting surface area of the wing $S = 0.6617\text{m}^2$ is found by projecting the wings and tail on the x-y-plane in the 3D model of the aircraft. The MAC is defined as the average chord length of a tapered, swept wing, and is found from a numerical approximation of the wing to be $\bar{c} = 0.242\text{m}$. A figure of the numerical approximation can be seen in section 4.2.3.

From dialogue with the aircraft manufacturer, the wing airfoil is known to be EPPLER 397, and the tail airfoil is known to be NACA0010. Plots of these airfoils can be seen in fig. 4-2.



(a) Wing airfoil: EPPLER 397.

(b) Tail airfoil: NACA 0010.

Figure 4-2: The airfoils used for the aircraft.

4.3 Actuator Identification

As shown in the previous chapter, the different actuators are modeled individually. In this section, the model parameters are found through preliminary tests and ex-

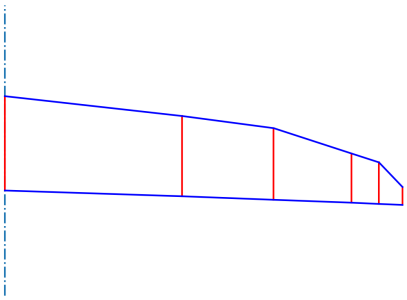


Figure 4-3: Wing approximation used to calculate the Mean Aerodynamic Chord (MAC) of the aircraft.

periments on the actuators. A summary of the actuator parameters can be seen in table 4.3.

Parameter	Value
T_{servo}	0.028s
$\dot{\delta}_{\max}$	$200^\circ\,\text{s}^{-1} = 3.491\ \text{rad}\,\text{s}^{-1}$
$\delta_{a,\max}$	25°
$\delta_{e,\max}$	25°
$\delta_{r,\max}$	22°
ρ	1.225kg/m^3
D_{FW}	0.3810m
D_{MR}	0.4064m
$c_{T_{\text{FW}}}$	0.0840
$c_{T_{\text{MR}}}$	0.0994
$c_{Q_{\text{MR}}}$	0.006338
$r_{1,x}$	0.353m
$r_{1,y}$	0.400m
$r_{2,x}$	-0.447m
$r_{2,y}$	-0.400m
$r_{3,x}$	0.353m
$r_{3,y}$	-0.400m
$r_{4,x}$	-0.447m
$r_{4,y}$	0.400m

Table 4.3: Summary of all the parameters describing the actuators of the aircraft.

4.3.1 Propellers and Motors

The models for propeller thrust and moment are presented in eqs. (3.42), (3.43) and (3.45). These models introduces three unknown parameters, namely $c_{T_{\text{FW}}}$, $c_{T_{\text{MR}}}$, and $c_{Q_{\text{MR}}}$. The rest of the parameters are known a-priori or can be measured directly. Due to the chosen model presented in chapter 3 where the propellers forces are modelled separately from the aerodynamics, it is simple to update the model with a new propeller configuration. One only has to do the propeller tests for the new configuration and update the corresponding coefficients in the model. To ease future work, the current propeller and motor setup used in this work is shown in table 4.4

	Motor	Propeller
Fixed-Wing	T-Motor AT4120 500KV	15x8 Wood
Multirotor	T-Motor MN505-S 320KV	16x5.5 Carbon Fiber

Table 4.4: Current setup of motors and propeller.

Figure 4-4: Experimental setup used for measurements of the propeller constants.

For the tests, a thrust stand of the type [59] is used, which is capable of measuring both the total generated thrust force and torque while measuring the RPS of the motor through an optical sensor. A picture of the setup can be seen in fig. 4-4. After performing the tests for the two different motor types, the unknown coefficients are found through Ordinary Least-Squares (OLS) as defined in section 5.4.2, by using the measured forces as outputs, and the squared RPS multiplied with ρD^4 as input, where D is the diameter of the used propeller.

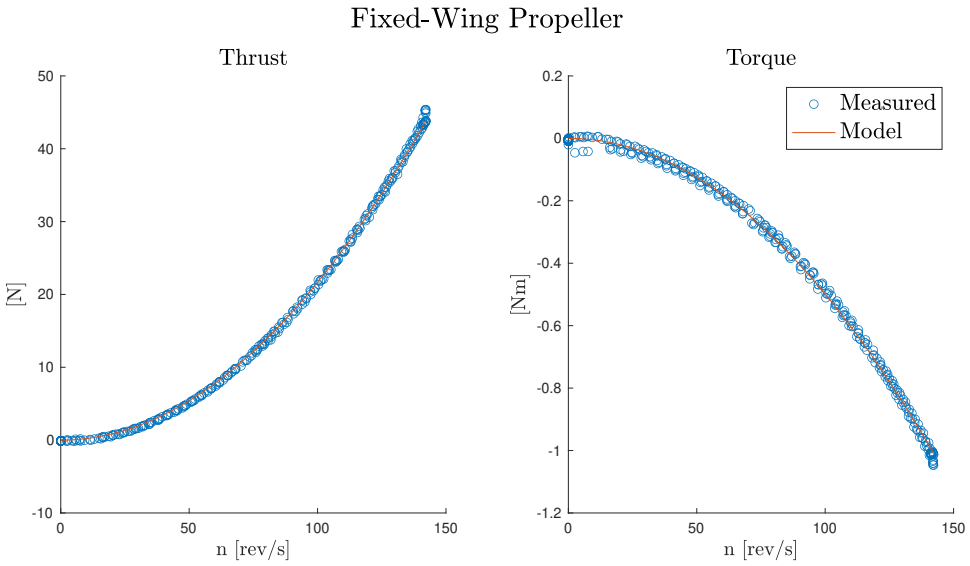


Figure 4-5: Fixed-wing propeller identification results.

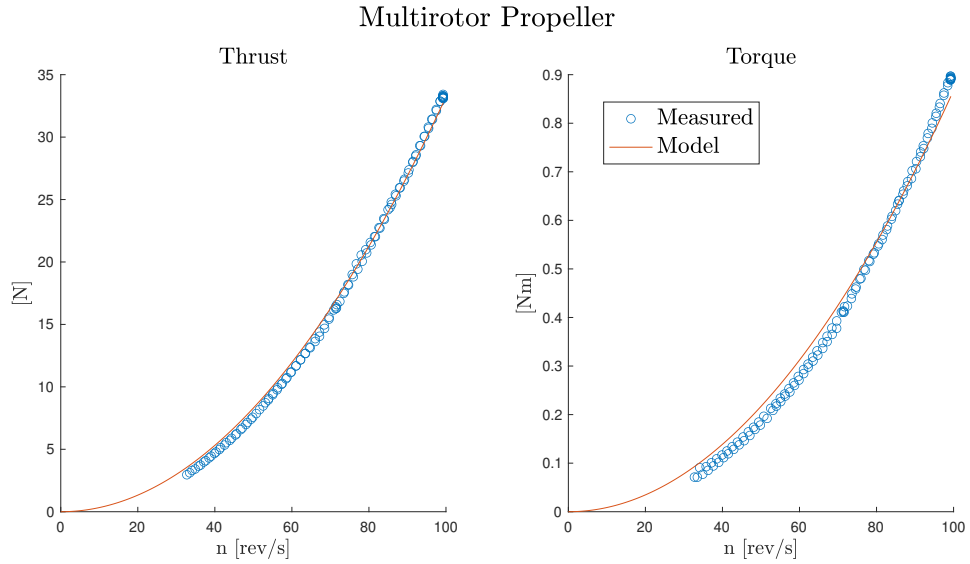


Figure 4-6: Multirotor propeller identification results.

A plot of the data and the obtained functions can be seen for the fixed-wing propeller in fig. 4-5, and for the multirotor propeller in fig. 4-6. As can be seen from the plot, the propeller model presented in chapter 3 accurately predicts both the thrust and the moments for both of the propellers when there is no incoming airflow.

4.3.2 Control Surface Dynamics

Next, the servo dynamics are identified. The servo experiments are performed with the servos mounted in the wing and a-tail, as they are when the aircraft is flying. This is to make sure that they are tested with a realistic load. The servo model used for the aircraft is the *KST FZ10 Servo*. A step input of different amplitudes is given to the servos, and the use of a high-speed camera measures the total deflection angles. The recorded responses for the motors are given in fig. 4-7.

At first, a simple first-order response was assumed for the servos. However, by looking at the data, it is found that a pure first-order response does not accurately reflect the behavior of the motors. Different amplitudes give different time constants, which is not the case for a simple first-order system. Therefore, a more detailed model is needed.

By looking at the data, the responses of the servos seem to be rate-limited. This can be seen as a linear part of the response, which is visible in fig. 4-7. It also corresponds with the specifications from the servo producer, where a max rate under no load is specified. Therefore, the model presented in eq. (3.41) is used. Note that the model in eq. (3.41) is nothing more than a simple first-order response with a rate limit expressed in mathematical terms.

As the motors are identical for the different control surfaces, it is expected that a single model will fit all of the motors. The parameter values are obtained through trial and error, and the simulated responses compared with the obtained data can be seen in fig. 4-7. It is clear that the model accurately described the behavior of the motors. The parameters are summarized in table 4.1.

4.4 Trim Conditions

It is necessary to trim all fixed-wing aircraft. First, the airspeed affects the lift of the aircraft, which gives rise to a trim AoA, α^*, where the wings generate enough lift to stay at a constant altitude. Second, at this α^*, the aircraft will generate a specific drag, which will require a constant trim throttle δ_t^* to fly at a constant airspeed.

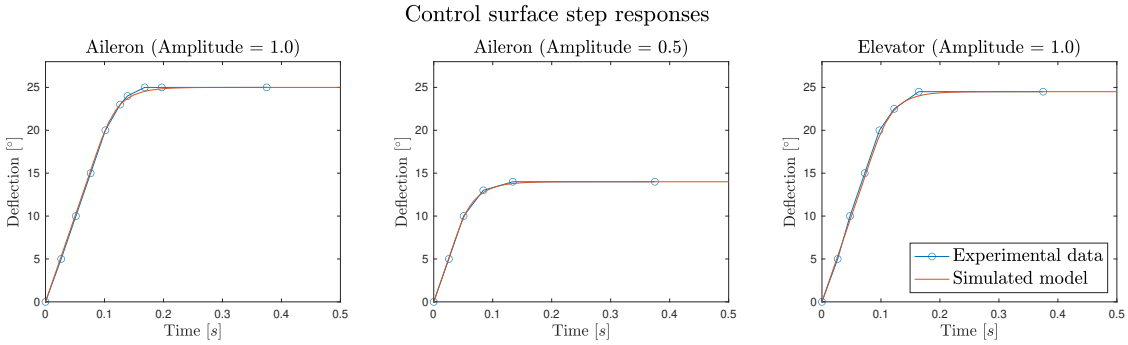


Figure 4-7: Step responses of the control surface servos.

Third, the lift at α^* will cause a pitching moment as the center of gravity is in front of the center of pressure, which will require an elevator trim δ_e^* that cancels out the pitching moment. Finally, the fixed-wing propeller will generate a moment at the constant throttle setting, which will require an aileron trim δ_a^* to keep the vehicle at a constant bank angle $\phi^* = 0$.

The trim settings are found through preliminary manual flights with an RC pilot. Aircraft trim settings are likely to vary slightly between instances of the same model due to minor physical differences in the aircraft. The specific trim settings used for the aircraft in this thesis are shown in table 4.5. It may be necessary for another aircraft instance of the same model to find new trim settings, depending on how different they are from the ones presented here.

Parameter	Value
α^*	$3°$
β^*	$0°$
V^*	$21\,\mathrm{m\,s^{-1}}$
δ_a^*	$3.0309°$
δ_e^*	$-5.6436°$
δ_r^*	$0°$
δ_t^*	${\eta_{\mathrm{FW}}^*}^2 = 125^2$
θ^*	$3°$
u^*	$20.971\mathrm{m/s}$
w^*	$1.099\mathrm{m/s}$

Table 4.5: Trim settings for the aircraft in this thesis.

From the trim airspeed V^* and the trim AoA α^* it is possible to calculate the trim

body velocities u^* and w^*. Some manipulation of the equations in eq. (3.4c) gives

$$u^* = \frac{V^*}{\sqrt{1 + \tan \alpha^{*2}}} \tag{4.1a}$$

$$w^* = \sqrt{(V^*)^2 - (u^*)^*} \tag{4.1b}$$

4.5 Aerodynamic Modeling with the Vortex Lattice Method

Although the aerodynamic modeling of aircraft is most accurately performed through the use of wind tunnel data or flight data [19, 21], it is possible to obtain theoretical values for the aerodynamic parameters using numerical methods. These parameters are expected to be inaccurate but may provide valuable insight into the aircraft's time constants and expected behavior. For example, in [39, 40] such modeling is used to design input signals to the aircraft that excite the desired modes of the system. Having a baseline model before starting system identification procedures obtained from similar aircraft, small-scale aircraft models, wind-tunnel tests, or numerical calculations is common. An example of this is given in [44], where the researchers use combined data from similar aircraft to obtain a baseline model. In addition, in [57], parameters obtained from wind tunnel tests are compared with parameters obtained through VLM, where stability coefficients from VLM are used directly in the final model.

In this work, the Extended Vortex Lattice Method (VLM) is employed, through the software AVL [47]. The coefficients obtained from numerical simulations are not expected to be exact because of weaknesses in purely computational approaches and uncertainty in the aircraft geometry. Still, they will be helpful as a baseline model for the complete nonlinear system identification and preliminary analysis. Through AVL, both linear state-space models around trim and nonlinear models using aerodynamic coefficients are developed. These models are tested on validation data in chapter 6, but as will be seen, they do not accurately capture the aerodynamics acting on the

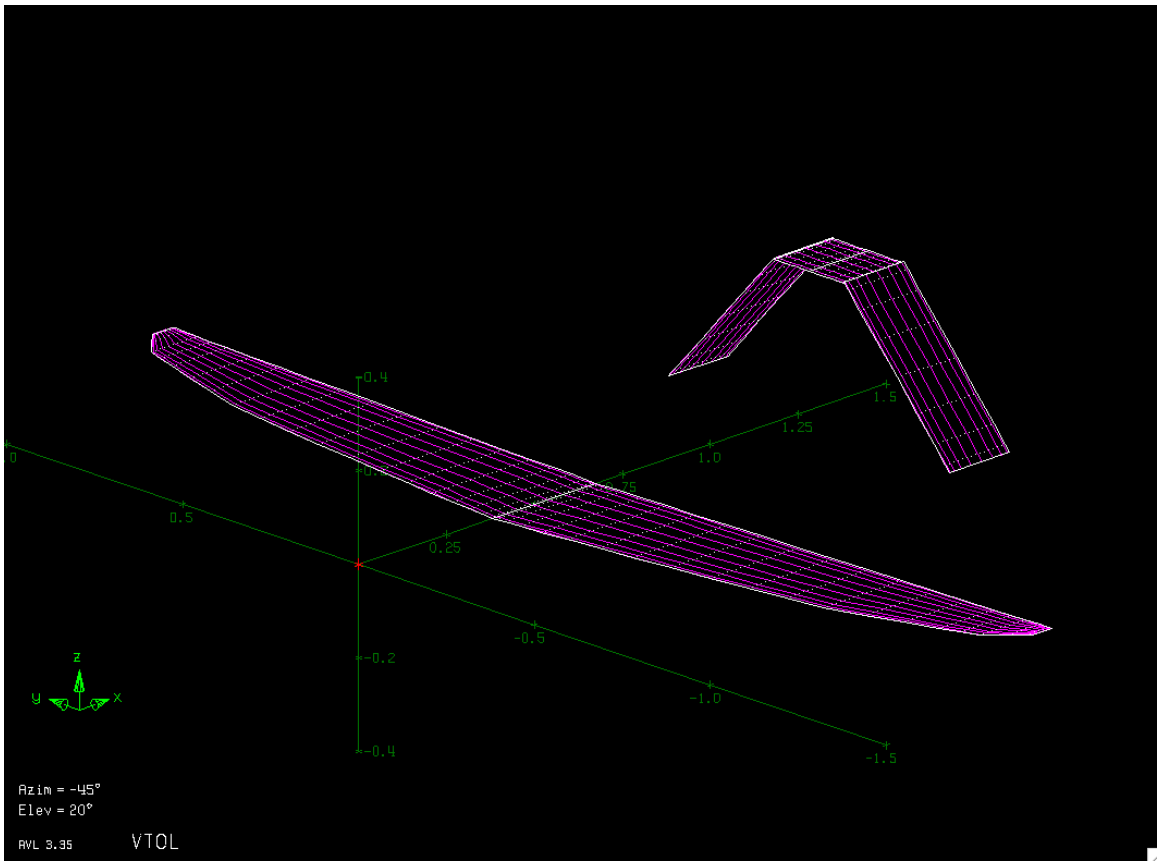


Figure 4-8: AVL model of the lifting surfaces on the UAV.

aircraft.

To employ VLM, one needs to specify the geometry of the lifting surfaces on the aircraft. That is, only the wings and the V-tail and the respective control surfaces are modeled. A visual representation of the model can be seen in fig. 4-8. As recommended in the documentation, the fuselage, and the VTOL modules are not modeled, as this may give inaccurate results. Some geometry of the aircraft is known precisely. The rest of the geometry has to be matched based on physical measurements of the aircraft. For example, it is hard to manually determine whether there is a twist in the wings, which will affect the pitch moment stability [58, 60]. In particular, the pitch moment coefficients are found to be very sensitive to small changes such as the twist of the wings or the center of gravity. VLM is also known to over-estimate the effect of the control surfaces [39].

Both longitudinal and lateral-directional state-space models are obtained through

VLM. In the following subsections, eigenmode analysis is performed to analyze the results. For further details on the implementation of VLM and the use of AVL, the reader is referred to the AVL documentation.

4.5.1 Eigenmode Analysis of State-Space Model from VLM

For the longitudinal model, eigenmode analysis of the state-space matrix obtained through VLM gives the modes shown in table 4.6, where ζ is the damping of the mode, f is the mode frequency, and T_c is the time-constant. As is typical for aircraft,

Mode	Eigenvalue	ζ	f[cycles/s]	T_c[s]
Short-Period	$-3.09 \pm 7.44i$	0.384	1.2812	0.324
Phugoid	$-0.01 \pm 0.62i$	0.0192	0.0988	83.7

Table 4.6: Eigenmode analysis of longitudinal state-space model obtained through VLM.

the short-period mode is damped with a short time constant of 0.324s. For the phugoid mode, the time constant is much larger and is 83.7s. The damping of the phugoid mode is very light, with $\zeta = 0.019$.

Similarly, eigenmode analysis of the lateral-directional state-space matrix obtained through VLM gives the results in table 4.7. The first thing to note from the eigenmode

Mode	Eigenvalue	ζ	f[cycles/s]	T_c[s]
Roll	-16	$--$	$--$	0.0624
Dutch roll	$-1.06 \pm 5.66i$	0.184	0.9167	0.946
Spiral	0.107	$--$	$--$	-9.35

Table 4.7: Eigenmode analysis of lateral-directional state-space model obtained through VLM.

analysis of the lateral-directional model is that the spiral mode of the aircraft is slow, in the order of 10s, and is slighly unstable. Further, as is quite common [48], the Dutch roll mode is lightly damped with a damping of $\zeta = 0.184$, with a relatively slow time constant of 0.174s. The roll mode is fast and critically damped, with a time constant of 0.0625s.

There are some interesting takeways from the preliminary analysis performed with VLM. First, the aircrafts time constants are of expected magnitudes given the aircraft size. Second, it seems that the experiments in this work will not be able to capture all five modes accurately. As will be seen, due to physical restrictions during the tests, the flight experiments that are performed are on the order of 5s. By the assumption that the general order of magnitude of the time-constants obtained through VLM are correct, it, therefore, seems that both the short-period mode, the roll mode, and the Dutch roll mode will get captured in the experiments. However, this also means that the data obtained through flight testing will not capture the slow modes of the aircraft. Therefore, it is expected that there will be uncertainty associated with identifying both the phugoid mode and the spiral mode.

Chapter 5

Principles of Time Domain System Identification

System Identification is a method for building mathematical models of dynamic systems, using measurements of input and output signals of the system. In general, the purpose of doing system identification is to obtain a model of the system in question, such that advanced methods within control and planning, as mentioned in section 1.2, can be employed.

In this chapter, an introduction to system identification for fixed-wing aircraft is given. There are many different methods for performing system identification, depending on the system in question, the goal, the available data, and other resources available. The VTOL aircraft in this thesis consists of a multirotor and a fixed-wing part, where the multirotor part was already identified in the preliminary analysis in chapter 4. As mentioned in the introduction in chapter 1, fixed-wing system identification is a large field that has received a lot of focus over the years, particularly for full-scale aircraft. Research on small fixed-wing UAVs is less mature. As will be seen, the principles for fixed-wing system identification are often quite involved, and the methods used in this thesis are presented in this chapter.

5.1 Introduction to System Identification for Fixed-Wing UAVs

The process of system identification for fixed-wing UAVs is complex and involves many steps, and several of the steps usually have to be reperformed as one gains information about the system. These steps involve preliminary analysis, model class choice, experiment design, data processing, model structure determination, parameter estimation, and model validation. In the aviation industry, it is common to spend much time and resources on developing more accurate and correct models. A relatively coarse model may be developed initially, and iterations with higher and higher fidelity may be developed over several years, as seen in, for example, the work by Wingtra in [7, 8, 9].

First, depending on the purpose and a-priori knowledge of the airframe, a model class must be selected for the identification procedure. Then, experiments are designed such that the chosen model is best identified from the available data. This involves the design of input signals to excite the desired system frequencies or target only some of the states or axes of the system while keeping the rest of the system stable. Next, one must prepare the instrumentation so that the data is logged with the appropriate fidelity and retrieved after the experiments. Post-processing must be performed on the data, where the data first needs to be organized into input-output datasets. Then, the data must be cleaned and filtered, and relevant signals must be calculated from the data. Then, a model from the selected class can be initialized and fit the input-output data, typically during an optimization routine. Finally, the validity of the obtained model has to be ascertained before the system identification routine is complete.

System identification is seldom an iterative process where the steps are only repeated once. Instead, the steps are repeated multiple times after new information about the system is uncovered. One may have to go back a few steps several times before the procedure is over. During the process, one might, for example, discover that the model class is not rich enough to describe the system or that the model class

is overly complex, causing over-fitting of the data. One might discover that additional data is needed to estimate specific parameters accurately or that the initial data was too noisy or otherwise corrupted and not usable for the last steps.

For aircraft, system identification is usually a question of parameter estimation, as the model is typically more or less known a-priori and does not have to be determined from the data [19]. For normal aircraft, the unknowns generally are the aerodynamic coefficients that describe the total aerodynamic forces and moments. The way these forces and moments enter the system is known from first principles of physics. This is illustrated in the proposed model from chapter 3, where most of the model is known or can be found from simple tests as in chapter 4, except for the aerodynamic forces and moments.

5.1.1 Problem Formulation

It is possible to write the system idenfitication problem as a general optimization problem, where the system is represented by an arbitrary state-space model. Let the system be instrumented with sensors and actuators, where one may apply input commands $\boldsymbol{u}[n]$ at some discrete rate, and measure the outputs $\boldsymbol{y}[n]$ at some discrete rate. Without diving into the details of the state-space model, we then have a system of the following form

$$\hat{\boldsymbol{x}}[n+1] = \boldsymbol{f}_{\boldsymbol{\theta}}(\hat{\boldsymbol{x}}[n], \boldsymbol{u}[n]) \tag{5.1a}$$

$$\hat{\boldsymbol{y}}[n] = \boldsymbol{g}_{\boldsymbol{\theta}}(\hat{\boldsymbol{x}}[n], \boldsymbol{u}[n]) \tag{5.1b}$$

where $\hat{\boldsymbol{x}}[n]$ is the predicted state of the system at timestep n, $\hat{\boldsymbol{y}}[n]$ is the predicted output of the system, and $\boldsymbol{\theta}$ describes a vector of all the model parameters. $\boldsymbol{f}$ is a model of the state dynamics, and $\boldsymbol{g}$ is some model of the system output. Note that this formulation allows for virtually any type of system model, where the model could be represented by anything ranging from explicit EOMs derived from first principles of physics, to nonlinear neural-network models.

Generally in system identification, one seeks to minimize the squared simulation

error:

$$\min_{\boldsymbol{\theta},\,\boldsymbol{x}[0]} \sum_{n=1}^{N-1} \|\hat{\boldsymbol{y}}[i] - \boldsymbol{y}[i]\|^2, \quad \text{subject to eq. (5.1)} \tag{5.2}$$

That is, one seeks to find the model parameters $\boldsymbol{\theta}$ and the initial state $\boldsymbol{x}[0]$, that minimize the squared error between the predicted output $\hat{\boldsymbol{y}}[i]$ and the measured output $\boldsymbol{y}[i]$. In the general formulation, both initial conditions $\boldsymbol{x}[0]$ and the parameters $\boldsymbol{\theta}$ are unknown, as is the case for many practical applications. As will be seen, only the parameters $\boldsymbol{\theta}$ are estimated in the optimization routine in this thesis, as $\boldsymbol{x}[0]$ is assumed to be known exactly through the data preprocessing steps.

It should be noted that the formulation in eq. (5.1) is simplified, as it does not take any noise into account. However, measurements will always be corrupted with measurement noise in the real world, and the dynamics will be affected by process noise. Thus, generally, one needs to adjust the system identification procedure depending on the significance of the noise and the way it enters the system. This is especially true for small, fixed-wing aircraft flying in the atmosphere, where vibrations, sensor noise, and wind disturbances always affect measurements.

5.1.2 Classes of System Identification

Given the general framework for system identification given in eqs. (5.1) and (5.2), one may employ a variety of model classes and different parametrizations for the system in question. Here, a brief overview to system identification for fixed-wing aircrafts is presented. It should be noted that system identification in general is a field much to broad to accurately cover in a few pages, and that the intention of this subsection is only to give the reader a sense of where this work fits in the world of fixed-wing system identification.

Generally, system identification methods can be differentiated into two groups: parametric and non-parametric [21, 22]. In the survey in [22], the authors provide the following, precise explanation: Non-parametric methods require no a-priori information about the system dynamics or model structure and are identified from

impulse responses or frequency responses. On the other hand, parametric models assume a structure for the dynamic model and identify the model parameters from training data. From the research on relevant literature in this thesis, most fixed-wing applications apply a parametric method.

There exist many different parametric models used for system identification. Following [22], parametric models are be divided into the following four categories:

- Time-varying and time-invariant systems
- Static and dynamic systems
- Linear and nonlinear systems
- Continuous and discrete systems

Aircraft identification typically works with identifying a time-invariant, dynamic system, which may be either linear or nonlinear, and modeled as either continuous or discrete. However, most works found for the literature review in this thesis employ continuous models.

In addition to the four categories given above, another distinction can be made on whether the model is a full black-box model or a grey-box model. A black-box model is a purely data-driven modeling tool, where one employs no a-priori knowledge about the underlying model. A famous example of this is neural networks, where the unknown function is approximated by nonlinear over-parametrization, and the parameters are found through nonlinear optimization. An example of this model type employed to aircraft can be found in [61]. Another example of black-box models are discrete, parametric models such as ARX, ARMAX, NARMAX, or Box-Jenkins, which employ variations of auto-regression and moving averages to describe the dynamics, such as used for fixed-wing aircraft in [62]. Yet another black-box representation is transfer functions, as employed for system identification of a small fixed-wing aircraft in [46].

In contrast to black-box models, grey-box models are based on a-priori knowledge of the system, typically first principles of physics. This thesis employs grey-box modeling, and an example of this was presented in chapter 3. In this thesis, the

nonlinear grey-box model is based on rigid-body motion, and the coefficients are individually identified as part of the nonlinear model. Similar approaches are taken by researchers in [8, 38, 39, 40, 57]. Another approach to grey-box modeling is to linearize the nonlinear aircraft EOMs, and identify the parameters of the linear model instead. This approach is taken by researchers in [44, 45, 46]. It is also possible to combine grey-box models with black-box models, such as in [49], where a flat-plate aerodynamic model is augmented by using local radial basis functions.

In this thesis, a parametric, grey-box model is employed. The model is time-invariant, dynamic, nonlinear, and continuous and is based on rigid-body motion. Many of the parameters are already estimated through preliminary analysis in chapter 4. In particular, the part of the system identification that remains is the aerodynamic modeling, where the aerodynamic forces and moments are modeled as functions of the relevant states and inputs. The exact model structure and the parameters are determined from the data, using the procedures outlined in this chapter.

5.2 Experiment Design

The design of the experiments for collecting data is one of the most critical aspects of system identification. In particular, the chosen input signals are vital, as the input signal to the system affects all other aspects of the identification process [22]. While designing the experiments, there are several key aspects to keep in mind. In this section, a brief overview of the experiment design used for the work in this thesis is given.

All of the flights are performed by a professional RC pilot that controls the aircraft. The pilot flies laps with the aircraft at the desired flight conditions and initiates preprogrammed identification maneuvers on the long side of the lap. During an entire flight, the pilot, therefore, executes several maneuvers for system identification. The aircraft is flown in the `Stabilized` mode, and the throttle is kept more or less constant during the entirety of a maneuver.

The maneuvers themselves are automated and mapped to a switch on the RC

controller, such that the pilot may initiate a maneuver when desired. The maneuvers are preprogrammed into PX4, and the maneuver parameters are programmed as parameters in the flight control software. This allows the parameters to be changed in flight, as the aircraft is communicated with through MAVlink messages transmitted over the RC connection. In this way, an engineer may change the parameters and prepare the maneuvers while a pilot is flying the aircraft. When the aircraft is flying straight and level, the pilot initiates a maneuver, letting the aircraft settle before another maneuver may be performed. Due to physical restrictions, the maneuvers are limited to a maximum duration of approximately 5s, before the pilot needs to command a turn.

Several different input signals are being used in relevant work on small, fixed-wing aircraft. The input signals for the maneuvers have to be carefully designed, and there are several aspects to this. Hoffer et al. highlights multiple aspects in [22]:

- Applicable frequency range for the system
- Persistent excitation of all the desired modes
- Duration of excitation
- Types of inputs
- Chosen set of inputs for model verification

One may, for example, use signals such as frequency sweeps, doublets, manual maneuvers, or a combination. Some researchers also use different signals for different axes of the aircraft. For example, in [44, 45], frequency signals are used for frequency-domain identification, and doublets are used for time-domain identification. In [39, 40], doublets are used for the elevator and rudder, while 1-2-1 signals are used for the aileron. In [46], chirp signals are superimposed over the manual pilot stick inputs. In [8], 2-1-1 signals are used for all axes of the aircraft.

In this work, 2-1-1 signals are used to excite the different axes. This signal structure is recommended by [23], as it excites a large frequency range while being more efficient than frequency sweeps [63]. A 2-1-1 signal is a multistep signal consisting of three steps, where the first step has a period that is twice as long as the next two

steps. For the work in this thesis, different amplitudes and periods are used for the signal, as recommended in [23]. A plot of a 2-1-1 signal can be seen in fig. 5-1.

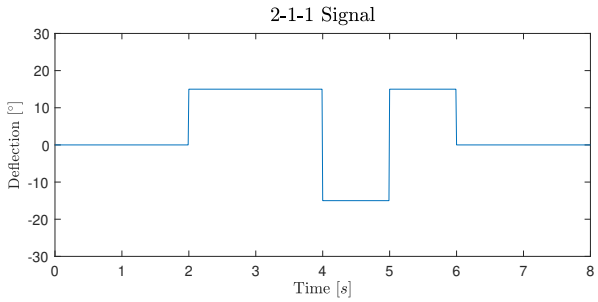


Figure 5-1: 2-1-1 signal.

The maneuvers are grouped into three different types: roll maneuvers, pitch maneuvers, and yaw maneuvers. 3D plots of recorded maneuvers with input signals of different signs and amplitudes can be seen in figs. 5-2 to 5-4. First, 2-1-1 signals are commanded to the aileron, which excites the roll mode. Between every maneuver, the input sign is switched, such that the aircraft performs the maneuver to both sides. Next, 2-1-1 signals are commanded to the elevators. This time, due to safety reasons while flying, the signal is designed such that the first step always pitches the aircraft upwards. This signal will excite the short-period mode. Finally, 2-1-1 signals are commanded to the aircraft's rudder, again while alternating the sign between maneuvers. This last signal is designed to excite the Dutch roll mode. For the aileron and pitch maneuvers, periods ranging from 0.3s to 0.5s are used, and for the rudder maneuvers periods ranging from 0.9s to 1.1s are used. The different maneuver types are designed to excite the modes of the system corresponding to each axis. However, because of physical limitations on the maneuver durations, it is expected that neither the phugoid mode nor the spiral roll mode will be captured in the data. These modes usually have time constants in the order of 10 to 100s, much longer than what the experiment setup allows. In addition, these modes are known to be hard to excite due to lack of excitation in the low-frequency domain [22, 44].

All maneuvers are designed for the onboard flight controller to stabilize all other aircraft axes while the maneuver is performed. That is, the control surfaces corresponding to the axis in question perform the 2-1-1 deflection, while PX4 controls

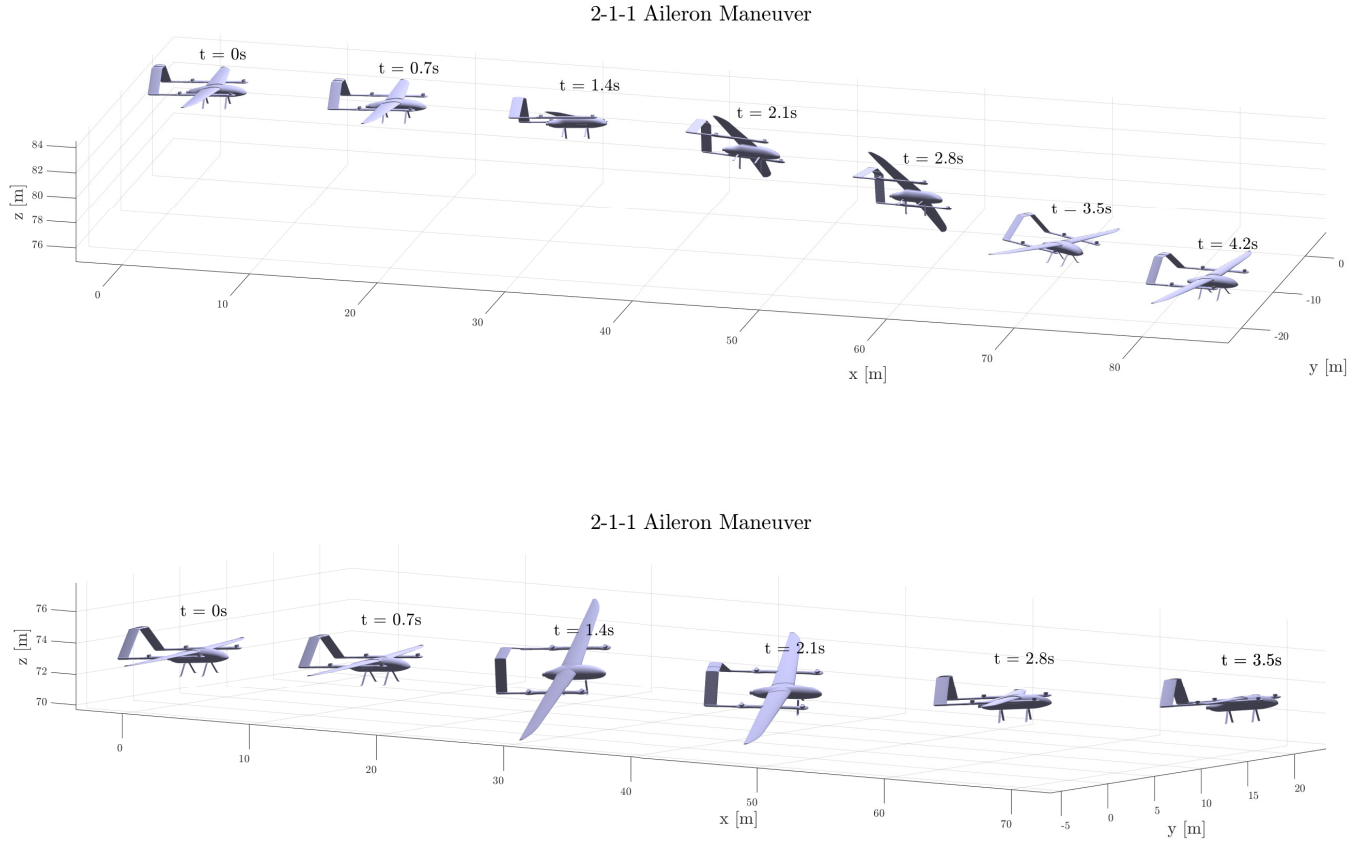


Figure 5-2: 2-1-1 aileron maneuvers (scale of the aircraft is 5:1).

the other control surfaces to stabilize the aircraft around those axes. Although this degrades the data, it is necessary to prevent the aircraft from crashing, as also explained in [19]. This is similar to works such as [8]. It should be pointed out that this is troublesome for the Dutch roll mode, which is known to excite both rolling and yawing motion, where the rolling motion in the Dutch roll mode will be canceled out by feedback control in these experiments.

For the work in this thesis, several attempts of data collection are made. Due to bad weather, logging dropouts, and disturbances in pilot inputs, between a third to half of all the data is discarded. In the end, all of the selected data is obtained when there is no detectable wind on the ground. The wind is known to increase higher up in the atmosphere, and as such, it is impossible to rule out any wind at the height

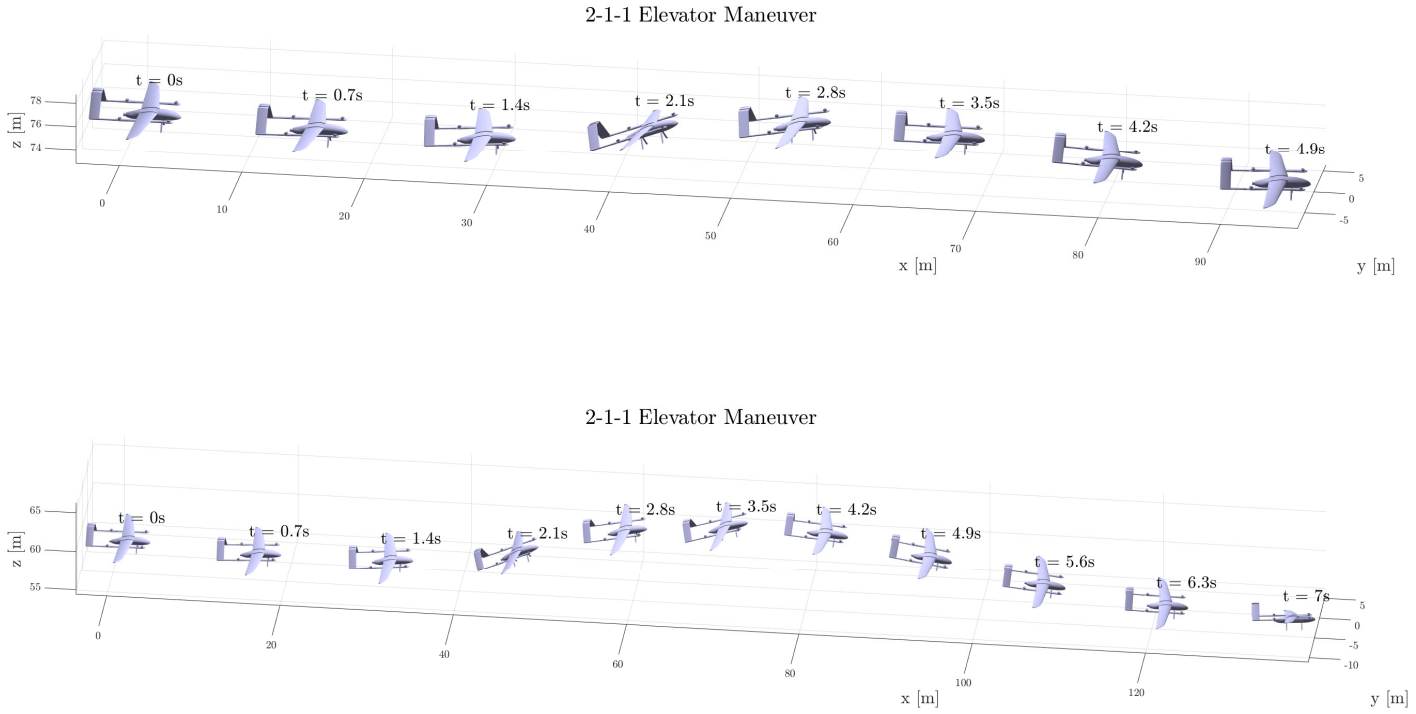


Figure 5-3: 2-1-1 elevator maneuvers (scale of the aircraft is 5:1).

the maneuvers are performed at. However, in this work, it is assumed that any wind disturbances are negligible, as there was no detectable wind during the experiments. A valuable lesson learned from the experiment planning in this work is that there is usually significantly less wind during the early morning. During the testing period, it was found that the wind levels started to increase considerably from 9 am.

5.3 Preprocessing and Initial Data Analysis

Before the system identification routine can begin, the available data needs to be preprocessed. This means both cleaning and sorting the raw data and obtaining the required signals from the data. First, the data itself needs to be processed, as it is corrupted with sensor noise, biases, and drifts. Second, the relevant signals have to be constructed from the processed data through the process known as Flight Path Reconstruction (FPR). Depending on the available sensors and avionics installed on

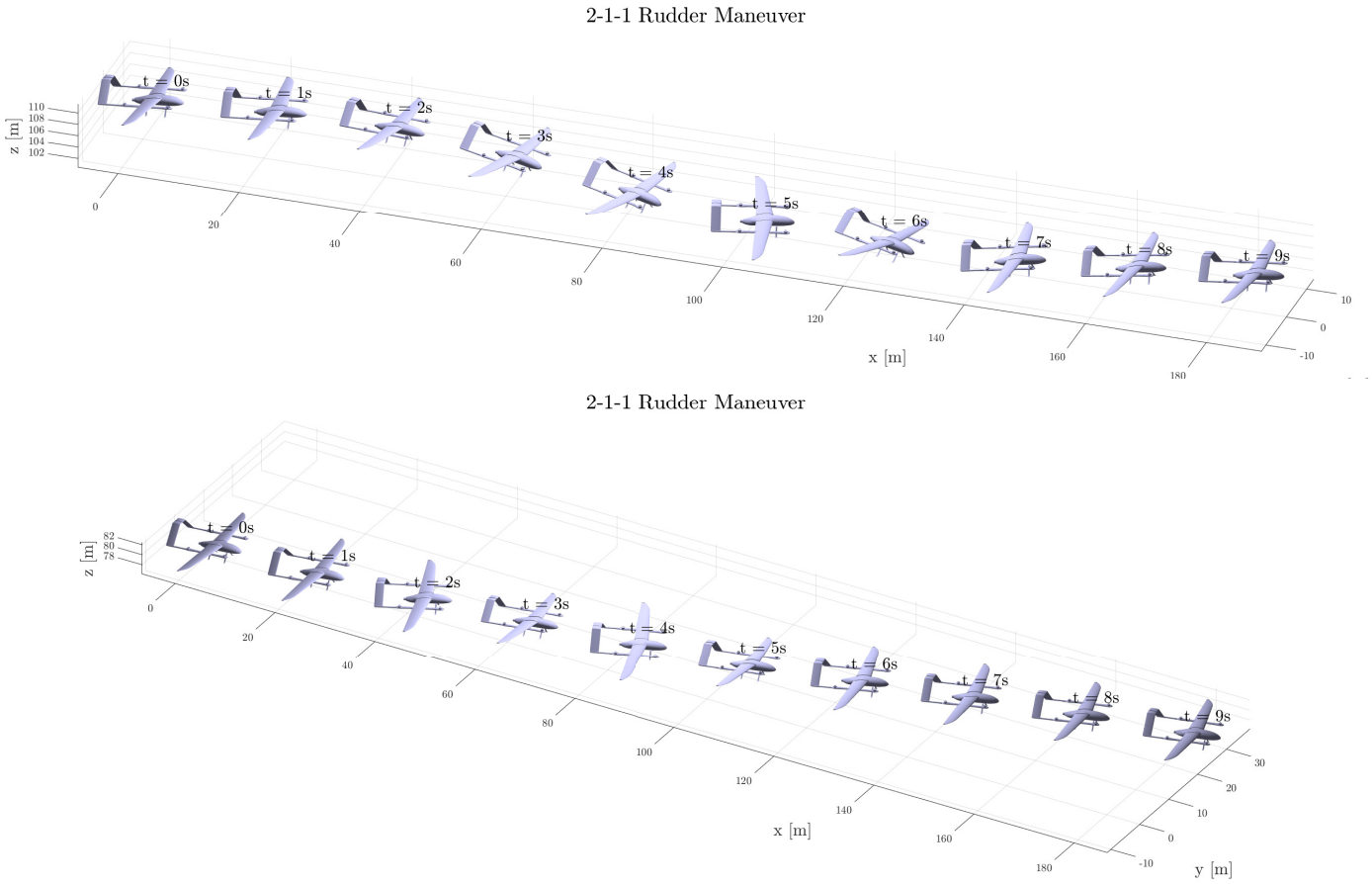


Figure 5-4: 2-1-1 rudder maneuvers (scale of the aircraft is 10:1).

the aircraft, different quantities are measured directly. Usually, many of the signals used for system identification need to be created as a preprocessing step. Finally, the data obtained through preprocessing and FPR need to be verified through a step known as checking for data compatibility or kinematic consistency.

5.3.1 Cleaning, Smoothing and Derivation of the Data

Raw sensor data is noisy and usually has to be processed before system identification can be performed [22]. Noise levels depend on sensor quality and the size of the aircraft, but there will always be noise in a sensor signal. Commonly in systems theory, one makes the distinction between pure sensor noise and process disturbances. The sensor noise is modeled as Gaussian noise and is always present in any sensor. Process disturbance accounts for other processes in the system that degrades the

data, such as actual disturbance in the form of wind gusts or unmodelled modes of the system, such as high-frequency vibrations from the motor or oscillations in the wings. A large portion of the external system disturbances for an aircraft, such as static wind and wing gusts, can be eliminated by flying only on days with little wind. Yet, some of the system disturbances are usually seen as noise in the data.

First, the data is cleaned and organized by maneuver types. That means that only the relevant parts of the experiment flights are retained and that any maneuvers with significant data degradation due to dropouts or large disturbances are rejected. The process of finding only the maneuver data can be entirely automated, as the maneuvers are initiated with a switch on the RC controller, and the RC signals are logged. The process of flagging data with dropouts can be automated too. However, the process of removing data with significant disturbances, or too little stable flight before or after a maneuver, is manual and quite tedious work.

After the data is cleaned and organized, relevant signals intended for FPR are selected from the data. The aircraft used in this work is running an Extended Kalman Filter (EKF) as part of the flight controller suite [64], which is estimating the attitude and inertial velocity of the aircraft, by fusing sensor data at a delayed time horizon. The PX4 flight suite is beginning to be quite common in the UAV literature, and in [8, 39, 40] the states from the EKF are used for identification rather than raw sensor data. This is also done in this work. Before experiments are performed, modifications are made to the PX4 software to ensure that the EKF data is logged at an appropriate sample rate. Finally, the logged attitude and inertial velocities are taken from the onboard EKF and used as the basis for the remaining steps. As the raw sensor data is already processed by an EKF, it is expected that most noise in the data is already removed.

In this work, a Savitsky-Golay filter is used to additionally reduce the noise level in the FTD. This is similar to the work in [39, 40]. The Savitsky-Golay filter is a digital filter that can be applied for smoothing the data without distorting the signal tendency. Other than this, there is little particular reason for choosing this filter type over any other, and the filter is found to work well with standard settings. As stated,

the noise level in the data is already relatively low due to the EKF. A filter order of 5 and a frame length of 11 is found to give satisfiable results for the kinematic consistency tests as described in section 5.3.3.

For FPR, both the smoothed attitude and the inertial velocities are used. Their derivatives are needed too, and the signals need to be fused to the desired time horizon. In this work, derivatives are computed using analytical differentiation of local polynomial regressions, as proposed in [19, 39, 40]. This is in contrast to numerical differentiation by using, for example, finite differences, which is known to be troublesome for signals with noise [19]. Piecewise splines are chosen for the polynomial regression, and if the spline intervals are small enough, it can be seen in fig. 5-5 how the data is accurately approximated. The derivatives are obtained by analytically differentiating the splines in the interval and evaluating the function at the desired time. The open-source Matlab package SLM [65] is found to work well with the polynomial spline regression and is used for the work in this thesis. In this work, a spline interval of $dt = 0.1$s is found to perform well.

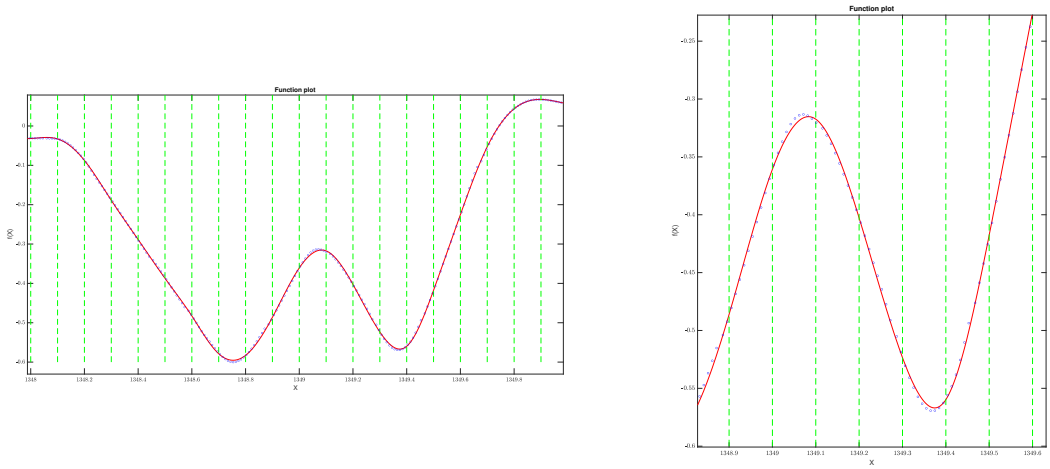


Figure 5-5: Data approximation using piecewise polynomial splines.

5.3.2 Flight-Path Reconstruction

The goal of Flight Path Reconstruction (FPR) is to reconstruct the aircraft state history based on measurements and kinematic relations. Specific details of FPR depend on which variables are chosen as states in the aircraft model and on which data is used as the basis for the FPR. Details and variations on FPR are given in [19, 20].

In this work, the inertial velocities $\boldsymbol{V}_{NED} = [V_N\, V_E\, V_D]^\intercal$ and the Euler angles $\boldsymbol{\Phi} = [\phi\, \theta\, \psi]^\intercal$ are available from the EKF, which are filtered and used as the basis for FPR. The full aircraft state used in modelling the aircraft dynamics consists of the body velocities $\boldsymbol{v}$, the angular rates $\boldsymbol{\omega}$ and the Euler angles. That means that both $\boldsymbol{v}$ and $\boldsymbol{\omega}$ has to be computed from the available measurements $\boldsymbol{V}_{NED}$ and $\boldsymbol{\Phi}$ from kinematic relationships. In addition to this, aerodynamic force coefficients c_X, c_Y, c_Z and aerodynamic moment coefficients c_l, c_m, c_n will be used for the regression analysis in section 5.4. Calculation of these coefficients requires both the angular accelerations $\dot{\boldsymbol{\omega}} = [\dot{p}\, \dot{q}\, \dot{r}]^\intercal$ and translational accelerations $\boldsymbol{a} = [a_x\, a_y\, a_z]^\intercal$, so these signals also have to be obtained through FPR.

The kinematic relationships described in this section follow standard rigid-body physics and modeling. For completeness and the convenience of the reader, all of the kinematic relationships used for FPR in this thesis work are given here, with brief explanations. For complete derivations and further details, the reader is referred to traditional references such as [19, 20] which do an excellent job of covering the material.

First, the body velocities are obtained by constructing a rotation matrix from the Euler angles $\boldsymbol{R}_B^N = \boldsymbol{R}_x(\phi)\boldsymbol{R}_y(\theta)\boldsymbol{R}_z(\psi)$ such that $\boldsymbol{V}_{NED} = \boldsymbol{R}_B^N\boldsymbol{v}$. This gives the following translational kinematic relationship for reconstructing the body velocities

from the inertial velocities:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin(\theta) \\ \cos\psi\sin\theta\sin\phi - \cos\phi\sin\psi & \cos\phi\cos\psi + \sin\theta\sin\phi\sin\psi & \cos\theta\sin\phi \\ \cos\psi\sin\theta\cos\phi + \sin\phi\sin\psi & \sin\theta\cos\phi\sin\psi - \sin\phi\cos\psi & \cos\theta\cos\phi \end{bmatrix} \begin{bmatrix} V_N \\ V_E \\ V_D \end{bmatrix} \tag{5.3a}$$

Next, the angular rates are obtained from an inversion of the rotational kinematic relationship in eq. (3.8), resulting in the following kinematic relationship:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{5.4a}$$

This relationship requires not only the Euler angles $\mathbf{\Phi}$ but also their derivatives $\dot{\mathbf{\Phi}}$. These derivatives are obtained by analytical derivation of piecewise polynomial regressions rather than numerical derivatives, as described in section 5.3.1.

The translational accelerations are needed to compute the aerodynamic force coefficients. Translational accelerations are obtained from the following kinematic relationship

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} \dot{u} + qw - rv + g\sin\theta \\ \dot{v} + ru - pw - g\cos\theta\sin\phi \\ \dot{w} + pv - qu - g\cos\theta\cos\phi \end{bmatrix} \tag{5.5a}$$

Also here, the derivatives of the body velocities are first obtained through analytical derivation of piecewise polynomial regressions approximating $\boldsymbol{v}$, before the translational accelerations can be computed. The angular accelerations $\dot{\boldsymbol{\omega}}$ are readily obtained from derivation of $\boldsymbol{\omega}$.

At this point in the FPR procedure, the resulting signals are $\boldsymbol{v}$, $\boldsymbol{\omega}$, $\boldsymbol{a}$, $\dot{\boldsymbol{\omega}}$ and $\mathbf{\Phi}$. The final step is to compute the aerodynamic coefficients, which depend on the

aerodynamic forces. The aerodynamic forces are given as

$$X = ma_x - T \tag{5.6a}$$

$$Y = ma_y \tag{5.6b}$$

$$Z = ma_z \tag{5.6c}$$

where T is the force generated by the fixed-wing propeller. The aerodynamic force coefficients follow readily as

$$c_X = \frac{X}{\bar{q}S} \tag{5.7a}$$

$$c_Y = \frac{X}{\bar{q}S} \tag{5.7b}$$

$$c_Z = \frac{X}{\bar{q}S} \tag{5.7c}$$

where S is the total planform area of the wings, and $\bar{q} = \frac{1}{2}\rho V_a^2$ as defined in chapter 3.

Further, the aerodynamic lift and drag coefficients are obtained as

$$\begin{bmatrix} c_D \\ c_L \end{bmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} -c_X \\ -c_Z \end{bmatrix} \tag{5.8}$$

The aerodynamic moments are given as

$$\begin{bmatrix} l \\ m \\ n \end{bmatrix} = \begin{bmatrix} J_{xx}\dot{p} - J_{xz}(\dot{r} + pq) + qr(J_{zz} - J_{yy}) \\ J_{yy}\dot{q} + pr(J_{xx} - J_{zz}) + J_{xz}(p^2 - r^2) \\ J_{zz}\dot{r} - J_{xz}(\dot{p} - qr) + pq(J_{yy} - J_{xx}) \end{bmatrix} \tag{5.9a}$$

The aerodynamic moment coefficients are computed as

$$c_l = \frac{L}{\bar{q}Sb} \tag{5.10a}$$

$$c_m = \frac{M}{\bar{q}S\bar{c}} \tag{5.10b}$$

$$c_n = \frac{N}{\bar{q}Sb} \tag{5.10c}$$

with b being the wingspan of the aircraft, and $\bar{c}$ being the Mean Aerodynamic Chord (MAC) as defined in chapter 3.

5.3.3 Data Compatibility, or Kinematic Consistency

Checking for data compatibility, or kinematic consistency, between the different data obtained through the preprocessing and FPR step, is essential before system identification can be performed. Data compatibility analysis is the process of determining whether the data obtained from different sensors onboard the aircraft are mutually consistent with the kinematic relationships that govern the aircraft motion. Data compatibility is well established in the literature for system identification [19, 20], and is also covered in most recent research works such as [38, 39, 40].

Usually, the instrumentation on a small aircraft includes sensors that measure accelerations, angular rates, inertial velocities, and inertial position, as well as airspeed. Data compatibility is determined by checking whether the obtained measurements from different sensors satisfy the kinematic relationships known from first principles about rigid-body motion. Sensor bias and sensor noise may lead to drifting in the states when the kinematic relationships are integrated, resulting in poor data compatibility, which will cause problems in the system identification step. Data compatibility analysis may be used to identify and eliminate such biases and give valuable insight into how the data should be preprocessed.

In work in this thesis, minor issues are expected with data compatibility, as the data is extracted from the onboard EKF and not directly from the sensors. However, checking the data compatibility is still an essential step in the data preprocessing step and is used to determine the parameters for smoothing and derivation of the data. It should be noted that only the states and variables that have a purely kinematic relationship can be checked for kinematic consistency. Input signals are not included in the data compatibility analysis, as the inputs directly affect the accelerations of the system and are hence part of the dynamics, not the kinematics, of the system.

Data compatibility analysis is performed by integrating the kinematic equations in eq. (3.8), as well as by inverting the kinematic relationship in eq. (5.5) where the

accelerations are now known:

$$\dot{u} = rv - qw - g\sin\theta + a_x \tag{5.11a}$$

$$\dot{v} = pw - ru + g\sin\phi\cos\theta + a_y \tag{5.11b}$$

$$\dot{w} = qu - pv + g\cos\phi\cos\theta + a_z \tag{5.11c}$$

The kinematics relationships are unstable, which means that noise or biases in the measurements will cause drifts over time. As stated in [19], the random sensor noise implies that the kinematic differential equations are stochastic. If the noise is small with zero mean, this can often be ignored, and the data will be compatible and ready for system identification. However, smoothing, filtering, or de-trending techniques need to be applied if the noise is significant. The body-axis angular rates $\boldsymbol{\omega}$, and the translational accelerations $\boldsymbol{a}$ are considered inputs in the kinematic relationships. Suppose the recorded measurements of the Euler angles $\boldsymbol{\Phi}$ and the body velocities $\boldsymbol{v}$ correspond with the resulting outputs from integrating the kinematic relationships outlined in this section. In that case, the flight data is said to be kinematically consistent. If the reconstructed outputs do not match the measured outputs, the data needs to be processed further.

Through kinematic consistency testing, the preprocessing parameters used in this work are found. As stated earlier, standard parameters for the Savitsky Golay filter are found to work well. Next, the intervals for the piecewise splines are selected to 0.1s, a number that is arrived at through the kinematic consistency checking. In fig. 5-6, a plot comparing different intervals for the spline approximation can be seen.

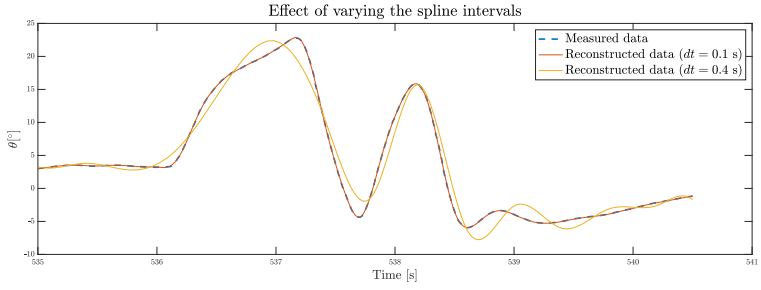


Figure 5-6: Kinematic consistency test for different spline intervals.

5.4 Equation-Error and Model Determination

A powerful estimation tool is the Equation-Error method. In this work, this method is used in combination with a stepwise regression procedure to arrive at a proper model structure that describes the data while keeping the model structure as simple as possible. The method is also used to obtain initial parameter estimates for the Output-Error method covered in the next section. Due to the implementation of the preprocessing step in the previous section, approximate signals for all aerodynamic coefficients are available, making it possible to employ the Equation-Error method as a linear regression procedure. This section outlines both the Equation-Error method and the stepwise regression procedure used for this work.

5.4.1 Equation-Error Parameter Estimation

The Equation-Error method formulates the problem of system identification as a linear regression problem. In the context of system identification, this refers to estimating the relationships between measured variables. These measured variables are divided into regressors and outputs, or equivalently, independent and dependent variables. The Equation-Error method is well established and outlined in, for example, [19, 20, 39, 40].

The general formulation of the Equation-Error method for a general dynamical model is found by minimizing the sum of the one-step prediction errors:

$$\min_{\boldsymbol{\theta}} \sum_{i=0}^{N-2} \|\boldsymbol{f}_\theta(\hat{\boldsymbol{x}}[i], \boldsymbol{u}[i]) - \hat{\boldsymbol{x}}[i+1]\|^2, \quad \text{subject to eq. (5.1)} \tag{5.12}$$

Note that in the previous cost function in eq. (5.1), the sum of squared errors between the simulated system outputs $\hat{\boldsymbol{y}}$ and the measured outputs $\boldsymbol{y}$ are minimized. In the cost function in eq. (5.12), only the sum of the squared errors in the next predicted state is minimized. This means that in eq. (5.12), the error at every timestep is independent. This is what is meant by one-step predictions, as the cost function only sums up the predictions one step ahead for each timestep.

The cost function in eq. (5.12) may look nonlinear. However, assuming one knows the states and inputs perfectly, the cost function can be re-formulated as a linear regression problem for many model structures. In the case of the aerodynamic model presented in chapter 3, this regression problem is linear in all the parameters. This means that Ordinary Least-Squares (OLS) can be employed to obtain the parameter estimates from the experimental data.

Take for example the full pitch moment coefficient model proposed in eq. (3.31):

$$c_m = c_{m0} + c_{m\alpha}\alpha + c_{m\alpha^2}\alpha^2 + c_{m\hat{q}}\hat{q} + c_{m\hat{q}\alpha}\hat{q}\alpha + c_{m\delta_e}\delta_e + c_{m\delta_e\alpha}\delta_e\alpha \tag{5.13}$$

$$= \begin{bmatrix} c_{m0} & c_{m\alpha} & c_{m\alpha^2} & c_{m\hat{q}} & c_{m\hat{q}\alpha} & c_{m\delta_e} & c_{m\delta_e\alpha} \end{bmatrix} \begin{bmatrix} 1 \\ \alpha \\ \alpha^2 \\ \hat{q} \\ \hat{q}\alpha \\ \delta_e \\ \delta_e\alpha \end{bmatrix} = \boldsymbol{\theta}^\intercal \boldsymbol{\phi} \tag{5.14}$$

where $\boldsymbol{\theta}$ is a vector of unknown parameters and $\boldsymbol{\phi}$ is the known regressor vector. All of the coefficients in both eqs. (3.31) and (3.32) follow the same model structure. Although the model for the coefficient contains nonlinear terms, the terms themselves are linearly involved. Therefore, OLS, as defined in the next subsection, may be employed directly.

5.4.2 Ordinary Least Squares Regression

Here, a brief introduction to OLS is given. For a more comprehensive treatment, the reader is referred to excellent sources such as [19].

Let measurements $z = y + v$ be obtained from a model of the form $y = \boldsymbol{\theta}^\intercal \boldsymbol{\phi}$, where v is a normally distrubed measurement error. Collect all the N output measurements in a vector $\boldsymbol{z} \in \mathbb{R}^{N\times 1}$, and the N data points of m-dimensional regressors in the regressor vector $\boldsymbol{\Phi} \in \mathbb{R}^{m\times N}$. Collect the m unknown parameters in the vector $\boldsymbol{\theta} \in \mathbb{R}^{m\times 1}$. Let

the parameters be estimated by $\hat{\boldsymbol{\theta}}$, such that the outputs can be estimated as $\hat{\boldsymbol{y}} = \boldsymbol{\Phi}^\intercal \hat{\boldsymbol{\theta}}$. Further, define the squared error as

$$\frac{1}{2}\|\boldsymbol{z} - \hat{\boldsymbol{y}}\|^2 = \frac{1}{2}\|\boldsymbol{z} - \boldsymbol{\Phi}^\intercal \hat{\boldsymbol{\theta}}\|^2 \tag{5.15}$$

The squared error is then minimized simply by setting the derivative with respect to $\boldsymbol{\theta}$ to zero, from which we obtain the least squares parameter estimate:

$$\hat{\boldsymbol{\theta}} = (\boldsymbol{\Phi}\boldsymbol{\Phi}^\intercal)^{-1}\boldsymbol{\Phi}\boldsymbol{z} \tag{5.16}$$

5.4.3 Stepwise Regression and Model Determination

There are several methods for determining the appropriate model structure from the obtained data. In this work, a technique named Stepwise Regression is used. The procedure is a method for deciding which regressors to include, evaluating if the regressor increased the model performance significantly, and determining when to stop including regressors.

There are different ways of implementing stepwise regression. For example, one may manually use specific parts of the procedure to evaluate regressors and model structure, as in [39, 40]. The method may also be entirely automated, designed to analytically pick the best regressors out of a generated regressor pool, as is done in this work. The stepwise regression method was first proposed in [66]. In [19], Klein et al. provide an excellent overview of the technique in general, with various means of validation and implementation, stopping rules, and statistical metrics. Other methods exist, such as multivariate-orthogonal-function modeling, as proposed in [67] and employed in [68]. In this subsection, the stepwise regression method as implemented explicitly for this work is outlined.

In this work, stepwise regression is entirely automated. In [39, 40], it is proposed that this procedure needs to be performed manually and that the engineer needs to employ intuition and knowledge of physics to develop a suitable model structure successfully. However, in this work, it is found that when using a-priori knowledge from

physics in handpicking the regressor candidates while fully automating everything else, an automated procedure can better determine a suitable model structure than a manual process. In fact, manually developing and comparing each possible model structure rapidly becomes a huge problem that may be intractable for a human to do successfully. Instead, we propose that the engineer use insight to formulate a meaningful stepwise regression problem and then leave the actual model development up to the algorithm.

Stepwise Regression Algorithm

The stepwise regression algorithm consists of two steps, which are repeated until convergence. The first step is the *Forward selection* step, and the second step is the *Backward elimination* step. The next regressor is chosen from the remaining regressors in the regressor pool at every step of the forward selection step. If the new potential regressor sufficiently increases the model performance, it is included in the model. As adding a new regressor may make previously added regressors superficial, the backward elimination step now has to be performed. In the backward regression step, regressors that no longer increase the model performance are removed. This step is repeated until no more regressors should be removed. Then, the procedure starts over again, with another step of the forward selection procedure. When a step is performed where no changes to the model are made, the stepwise regression procedure terminates.

The model starts with adding a bias term as the first regressor. Then, the procedure evaluates linear terms, and upon termination, repeats, now with the nonlinear terms in the regressor candidate pool. A general summary of the stepwise regression algorithm can be seen in algorithm 1. X_{pool} denotes the pool, or group, of remaining potential regressors, and x_j denotes a specific regressor. Φ_{curr} denotes the current model regressors, and ϕ_j denotes a specific model regressor. F_{in}, F_{out}, and R^2_{in} are statistical thresholds for regressor inclusion or exclusion, and may be decided by the engineer. $\texttt{CalcF}_0(X_1, X_2)$ is a pseudocode function that calculates the F_0 statistic for comparison between two different models, where model 1 is a subset of model 2.

$\texttt{CalcR}^2(X_1, X_2)$ is a pseudocode function that calculates the change in the R^2 statistic between two different models. The pseudocode function `FindNextRegressor` finds the next regressor from the pool of available regressors, based on which regressor best predicts the residue left from the current model. This is outlined in more detail in a few paragraphs.

Algorithm 1 Stepwise Regression.

$\Phi_{\text{curr}} \leftarrow \{1\}$
for $X_{\text{pool}} \in \{X_{\text{linear}}, X_{\text{nonlinear}}\}$ **do**
  while Φ_{curr} is updated **do**
    $x_{\text{new}} \leftarrow \texttt{FindNextRegressor}(X_{\text{pool}})$ ▷ Forward selection step
    $F_0 \leftarrow \texttt{CalcF}_0(\Phi_{\text{curr}}, \Phi_{\text{curr}} \cup x_{\text{new}})$
    $\Delta R^2 \leftarrow \texttt{CalcR}^2(\Phi_{\text{curr}}, \Phi_{\text{curr}} \cup x_{\text{new}})$
    if $F_0 > F_{\text{in}}$ **and** $\Delta R^2 > R^2_{\text{in}}$ **then**
      $\Phi_{\text{curr}} \leftarrow \Phi_{\text{curr}} \cup x_{\text{new}}$
    end if
    while Φ_{curr} is updated **do** ▷ Backward elimination step
      for $\phi_j \in \Phi_{\text{curr}}$ **do**
        $F_{0,j} \leftarrow \texttt{CalcF}_0(\Phi_{\text{curr}} \setminus \phi_j, \Phi_{\text{curr}})$
      end for
      if $\min_j(F_{0,j}) < F_{\text{out}}$ **then**
        $\Phi_{\text{curr}} \leftarrow \Phi_{\text{curr}} \setminus x_j$
      end if
    end while
  end while
end for

Statistical metrics for regressor inclusion

In algorithm 1, various error metrics are calculated and used to determine whether to include or remove regressors. Those error metrics are the coefficient of determination R^2, and the performance of an *F-test*. Both of these metrics require calculation of the Total Sum of Squares (TSS) and Residual Sum of Squares (RSS) (also known as

the sum of squared errors in prediction), defined as

$$\text{TSS} \triangleq \sum_{i=1}^{N}(z[i] - \bar{z})^2 \tag{5.17}$$

$$\text{RSS}(\hat{\boldsymbol{\theta}}) \triangleq \sum_{i=1}^{N}(z[i] - \hat{y}(\hat{\boldsymbol{\theta}})[i])^2 \tag{5.18}$$

with $\bar{z} = \frac{1}{N}\sum_{i=1}^{N} z[i]$ being the mean of the measured outputs z, and $\hat{y}(\hat{\boldsymbol{\theta}})$ being the predicted output as a function of the estimated parameter vector $\hat{\boldsymbol{\theta}}$.

First, the coefficient of determination, denoted by R^2, is calculated as

$$R^2 \triangleq \left(1 - \frac{\text{RSS}}{\text{TSS}}\right) \times 100\% \tag{5.19}$$

where a score of 100% corresponds to a perfect fit. As seen in algorithm 1, a regressor is only included if it increases R^2 by a certain percentage. In [19] a minimum increase of $R^2_{\text{in}} = 0.5\%$ is proposed. In this thesis, a slightly higher value of 2% is used, as will be seen in chapter 6.

Next, it is possible to perform a partial F-test to see whether a more complex model significantly increases the fit. Let $\hat{\boldsymbol{\theta}}_1$ denote the parameters of model 1, and $\hat{\boldsymbol{\theta}}_2$ denote the parameters of model 2. Further, let model 1 be a subset of model 2, such that model 2 is the more complex model. By proposing the null hypothesis that model 2 does not have a significantly better fit than model 1, F_0 will have a F distribution with $(p_2 - p_1, N - p_2)$ degrees of freedom. The F-test can then be performed as

$$F_0 > F_{\text{in}} \tag{5.20}$$

where $F_{\text{in}} = F(\alpha; p_2 - p_1, N - p_2)$, and α is the selected significance level, and F_0 is the partial F statistics for null hypothesis. For the proposed null hypothesis, F_0 is

calculated as

$$F_0 \triangleq \frac{\left(\frac{\mathrm{RSS}(\hat{\boldsymbol{\theta}}_1)-\mathrm{RSS}(\hat{\boldsymbol{\theta}}_2)}{p_2-p_1}\right)}{\left(\frac{\mathrm{RSS}(\hat{\boldsymbol{\theta}}_2)}{N-p_2}\right)} \tag{5.21}$$

where $\hat{\boldsymbol{\theta}}_1$ is a subset of $\hat{\boldsymbol{\theta}}_2$, $p_i, i = 1, 2$ denotes the number of parameters in the models, and N is the total number of data points.

If the null hypothesis is rejected, in other words, if eq. (5.20) holds, model 2 gives a significantly better fit than model 1, with a confidence level of $(1-\alpha)\%$. For aircraft, it is common to choose $\alpha = 0.05$, such that a 95% confidence level is achieved. In this work, only the inclusion or exclusion of a single variable at a time is considered, such that $p_2 - p_1 = 1$ and $N - p_2 = N - p$, where p is the total number of regressors in the new model. When $N \gg 100$ and $p < 10$, the effect of p on $F(\alpha; 1, N-p)$ is small, and therefore one can employ $F_{\text{in}} = F_{\text{out}} = F(0.05; 1, N-p) \approx 4$ [19].

Picking the next regressor

In the forward selection step, the next candidate regressor is picked from the remaining regressor pool. If it passes the F-test and increases R^2 by a sufficient amount, the regressor is included in a model. In addition, there are some nuances to how the next candidate regressor should be picked, which are now covered.

In short, the next regressor is picked based on predicting the remaining information that is not yet captured by the current model, with information that is not yet used. This is done in two steps: first, the information already used by the model is removed. Then, the regressor with the strongest correlation to the output is selected as the next regressor. The first step is performed by making the remaining regressor pool orthogonal to the current model regressors. This is achieved through OLS, where one uses the current model regressors to predict the remaining regressors and subtracts this prediction from the remaining regressors. Then, the residuals are calculated based on the current model. Finally, to find the next regressor, the correlation coefficients between orthogonalized regressors and residue are computed for all regressors, and

the regressor with the highest value is picked. Note that the orthogonalization is only used to select the next regressor; the regressors used for prediction are provided at the beginning of the stepwise regression procedure. Intuitively, one may think of this as removing all the used information from the regressors and the output before picking the next candidate regressor.

The procedure for picking the next regressor index is outlined in algorithm 2. After the procedure, the next regressor x_{new} is then picked as the i-th regressor in X_{pool}. $\texttt{OLS}(\Phi, z)$ denotes standard OLS prediction of z, given the regressors Φ. The

Algorithm 2 Picking the next candidate regressor index.

$\hat{X}_{\text{pool}} = \texttt{OLS}(\Phi_{\text{curr}}, X_{\text{pool}})$
$X_{\text{orthogonal}} = X_{\text{pool}} - \hat{X}_{\text{pool}}$
$\boldsymbol{v} \leftarrow \texttt{residue}(\Phi_{\text{curr}})$
for $x_{\text{ort},j} \in X_{\text{orthogonal}}$ **do**
    $\sigma_j = \texttt{correlation}(x_{\text{ort},j}, v)$
end for
$i_{\text{new}} \leftarrow \text{argmax}\, \sigma_i$

residue is calculated as

$$\boldsymbol{v} = \boldsymbol{z} - \hat{\boldsymbol{y}} = \boldsymbol{z} - \boldsymbol{\Phi}^{\intercal}\hat{\boldsymbol{\theta}} \tag{5.22}$$

where $\hat{\boldsymbol{\theta}}$ is calculated from OLS as defined in section 5.4.2. The correlation is calculated as the correlation coefficient

$$\sigma(A, B) = \frac{\text{cov}(A, B)}{\sigma_A \sigma_B} \tag{5.23}$$

where σ_A and σ_B are the standard deviations of A and B, respectively.

5.5 Nonlinear Optimization with Output-Error

The Output-Error method is a maximum likelihood estimator commonly used to arrive at the final parameter estimates in system identification of fixed-wing aircraft. This method is used extensively, and various implementations can be seen in [19, 20,

36, 39, 40, 41, 68, 69]. The goal of the method is to minimize the simulation error, as presented very generally in eq. (5.2). However, the Output-Error is a maximum likelihood method that also accounts for measurements noise, and therefore the cost function will not be formulated exactly as in eq. (5.2). A summary of the algorithm as implemented in this work can be seen in algorithm 3. In this section, the details of this algorithm are covered.

In very general terms, the Output-Error methods seeks to minimize the simulation error, as shown in eq. (5.2). However, the Output-Error method is a maximum likelihood method, meaning that it maximizes the likelihood of observing the measured data $\boldsymbol{z}$, given a set of model parameters $\boldsymbol{\theta}$. In mathematical terms, this is expressed as

$$\hat{\boldsymbol{\theta}} = \max_{\theta} \mathbb{L}[\boldsymbol{z}; \theta] \tag{5.24}$$

where $\mathbb{L}$ is the likelihood function. From eq. (5.24), it is possible to arrive at the negative log-likelihood function [19] that is actually the cost function being minimized in the Output-Error method:

$$J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^{N} (\boldsymbol{z}[i] - \hat{\boldsymbol{y}}[i])^{\intercal} \hat{\boldsymbol{R}}^{-1} (\boldsymbol{z}[i] - \hat{\boldsymbol{y}}[i]) \tag{5.25}$$

where $\boldsymbol{z}$ are the obtained measurements corrupted with normally distributed measurement noise and $\hat{\boldsymbol{R}}$ is the estimated noise covariance. The noise covariance is estimated as

$$\hat{\boldsymbol{R}} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{v}[i] \boldsymbol{v}[i]^{\intercal} = \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{z}[i] - \hat{\boldsymbol{y}}[i])(\boldsymbol{z}[i] - \hat{\boldsymbol{y}}[i])^{\intercal} \tag{5.26}$$

where $\hat{\boldsymbol{y}}[i]$ is the predicted output at time step i. Further, it is assumed that $\boldsymbol{R}$ is diagonal, that is, that noise cross-covariances are zero.

Intuitively, one may think of eq. (5.25) being the same as eq. (5.2), but that measurement noise is now taken into account. The cost function in eq. (5.25) weights

the cost function by dividing each signal by the measurement noise variance. This means that prediction errors in signals with larger noise variance will be weighted less and that the signals with lower noise levels will be trusted more.

5.5.1 Solving the Optimization Problem

The cost function in eq. (5.25) is nonlinear, as it requires the simulation of the nonlinear system dynamics eq. (5.1). Therefore, there are no analytical solutions to minimizing the cost function, and computationally efficient methods like OLS cannot be applied. Instead, the cost function in eq. (5.25) is minimized with a numerical optimization algorithm called *Gauss-Newton* or *modified Newton-Raphson*. The procedure used in this work mostly follows the method as outlined in [19]; however, some aspects are changed slightly from the textbook to make the algorithm work satisfactory. In this subsection, the specific method used in this thesis is presented.

Second-Order Cost Function Approximation

First, a second-order Taylor expansion of the cost function in eq. (5.25) is performed around the parameter value $\boldsymbol{\theta}_0$

$$J(\boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta}) \approx J(\boldsymbol{\theta}_0) + \Delta\boldsymbol{\theta}^\intercal \frac{\partial J}{\partial \boldsymbol{\theta}}\bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} + \Delta\boldsymbol{\theta}^\intercal \frac{\partial^2 J}{\partial \boldsymbol{\theta}\partial\boldsymbol{\theta}^\intercal}\bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} \Delta\boldsymbol{\theta} \tag{5.27}$$

The necessary condition for $J(\boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta})$ to minimize J is

$$\frac{\partial}{\partial \boldsymbol{\theta}}[J(\boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta})] = \mathbf{0} \tag{5.28}$$

Combining the last two results gives an estimate for the vector of parameter changes $\Delta\boldsymbol{\theta}$

$$\Delta\hat{\boldsymbol{\theta}} = -\left[\frac{\partial^2 J}{\partial \boldsymbol{\theta}\partial\boldsymbol{\theta}^\intercal}\bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0}\right]^{-1} \frac{\partial J}{\partial \boldsymbol{\theta}}\bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} \tag{5.29}$$

From this, the updated parameter estimate $\hat{\boldsymbol{\theta}}$ can be computed from

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta} \tag{5.30}$$

where the update in eq. (5.30) has to be performed over multiple iterations, as the true cost is not really quadratic, as assumed in eq. (5.27)

The Gauss-Newton Step

In order to perform the update in eq. (5.30), both the gradient and Hessian matrix of J need to be known. First, by differentiation of eq. (5.25), the gradient is given as

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{i=1}^{N} \frac{\partial \boldsymbol{v}^{\mathsf{T}}[i]}{\partial \boldsymbol{\theta}} \hat{\boldsymbol{R}}^{-1} \boldsymbol{v}[i] = -\sum_{i=1}^{N} \frac{\partial \hat{\boldsymbol{y}}^{\mathsf{T}}[i]}{\partial \boldsymbol{\theta}} \hat{\boldsymbol{R}}^{-1} \boldsymbol{v}[i] \tag{5.31}$$

where the relation that $\boldsymbol{v} = \boldsymbol{z} - \hat{\boldsymbol{y}}$ is used to arrive at the final result.

Next, the Hessian is found as

$$\frac{\partial^2 J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{\mathsf{T}}} = \sum_{i=1}^{N} \frac{\partial \hat{\boldsymbol{y}}^{\mathsf{T}}[i]}{\partial \boldsymbol{\theta}} \hat{\boldsymbol{R}}^{-1} \frac{\partial \hat{\boldsymbol{y}}[i]}{\partial \boldsymbol{\theta}} - \sum_{i=1}^{N} \frac{\partial^2 \hat{\boldsymbol{y}}[i]}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{\mathsf{T}}} \hat{\boldsymbol{R}}^{-1} \boldsymbol{v}[i] \tag{5.32}$$

By neglecting the second term in eq. (5.32), we arrive at the Gauss-Newton method. There are several reasons for neglecting the second-order partial derivative: it is computationally expensive to obtain, and very susceptible to noise and numerical error. When the residue $\boldsymbol{v}[i]$ is small, this approximation is justifiable, as the second term is multiplied with the residue. Therefore, it is important that the Gauss-Newton method starts close to the true optimal values.

Define the Fischer information matrix $\boldsymbol{M}$ as

$$\boldsymbol{M} \triangleq \sum_{i=1}^{N} \frac{\partial \hat{\boldsymbol{y}}^{\mathsf{T}}[i]}{\partial \boldsymbol{\theta}} \hat{\boldsymbol{R}}^{-1} \frac{\partial \hat{\boldsymbol{y}}[i]}{\partial \boldsymbol{\theta}} \tag{5.33}$$

From this, the Gauss-Newton step is written as

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_0 - \boldsymbol{M}^{-1}_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} \left[\frac{\partial J(\boldsymbol{\theta}))}{\partial \boldsymbol{\theta}} \right]_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} \tag{5.34}$$

The Optimization Algorithm

The Gauss-Newton optimization algorithm is an iterative procedure that has to be repeated until convergence. Although $\hat{\boldsymbol{R}}$ is a function of the parameters $\hat{\boldsymbol{\theta}}$, $\hat{\boldsymbol{R}}$ and $\hat{\boldsymbol{\theta}}$ are not adjusted simultaneously. Instead, $J(\boldsymbol{\theta})$ is minimized for a fixed $\hat{\boldsymbol{R}}$, and then $\hat{\boldsymbol{R}}$ is recalculated, and $J(\boldsymbol{\theta})$ is minimized again. This is repeated until $\hat{\boldsymbol{R}}$ converges. For every step of minimizing $J(\boldsymbol{\theta})$, the Gauss-Newton step is applied as in eq. (5.34), until the cost, the cost gradient, or the parameters converge. In reality, alternating between changing $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{R}}$ is a relaxation of the problem. The reason for doing it this way is that the optimization hopefully will be more well-conditioned when this iterative approach is taken [19].

In [19], a full Gauss-Newton step is taken at each iteration of minimizing $J(\boldsymbol{\theta})$. However, in this work, a line search is used to find the optimal step length. This was done as it greatly increased the convergence properties of the algorithm. The line search was implemented by augmenting eq. (5.34) with a scaling parameter α:

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_0 - \alpha \boldsymbol{M}^{-1}_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} \left[\frac{\partial J(\boldsymbol{\theta}))}{\partial \boldsymbol{\theta}} \right]_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} \tag{5.35}$$

where $\alpha \in (0, 1]$.

The full Gauss-Newton optimization algorithm as implemented in this work can be seen in algorithm 3. $\epsilon_{\Delta \boldsymbol{R}}$, $\epsilon_{\Delta J(\boldsymbol{\theta})}$, $\epsilon_{\Delta \boldsymbol{\theta}}$, and $\epsilon_{\frac{\partial J}{\partial \boldsymbol{\theta}}}$ are convergence thresholds that are chosen by the engineer. In table 6.1, the values that are used for these parameters are presented.

Calculating the Gradients

The Gauss-Newton Algorithm as described in algorithm 3 requires the computation of gradients. More specifically, both the cost function gradient $\frac{J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$ and the Fischer

Algorithm 3 Gauss-Newton Algorithm.

while $\|\Delta\hat{\boldsymbol{R}}\| > \epsilon_{\Delta \boldsymbol{R}}$ **do**
  $\hat{\boldsymbol{R}} \leftarrow \frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{z}[i] - \hat{\boldsymbol{y}}(\hat{\boldsymbol{\theta}})[i])(\boldsymbol{z}[i] - \hat{\boldsymbol{y}}(\hat{\boldsymbol{\theta}})[i])^{\intercal}$
  while ($\|\Delta J(\boldsymbol{\theta})\| > \epsilon_{\Delta J(\boldsymbol{\theta})}$ **and** $\|\Delta\hat{\boldsymbol{\theta}}\| > \epsilon_{\Delta\boldsymbol{\theta}}$ **and** $\|\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\| > \epsilon_{\frac{\partial J}{\partial \boldsymbol{\theta}}}$) **do**
    $\Delta\boldsymbol{\theta} \leftarrow \boldsymbol{M}^{-1}_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}\left[\frac{\partial J(\boldsymbol{\theta}))}{\partial \boldsymbol{\theta}}\right]_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}$
    $\alpha \leftarrow \text{argmin}_{\alpha} J(\hat{\boldsymbol{\theta}} + \alpha\Delta\boldsymbol{\theta}), \quad \alpha \in (0, 1]$
    $\hat{\boldsymbol{\theta}} \leftarrow \hat{\boldsymbol{\theta}} - \alpha\Delta\boldsymbol{\theta}$
  end while
end while

information matrix $\boldsymbol{M}$ are dependent on the gradient of the output $\boldsymbol{y}$ with respect to the parameters $\boldsymbol{\theta}$, as is seen in eqs. (5.31) and (5.33). This matrix

$$\boldsymbol{S}[i] \triangleq \frac{\partial \boldsymbol{y}(\boldsymbol{\theta})[i]}{\partial \boldsymbol{\theta}} \in \mathbb{R}^{n_o \times n_p} \tag{5.36}$$

is more commonly referred to as the *sensitivity matrix*, where n_o is the number of outputs, and n_p is the number of parameters.

There are several ways to calculate the sensitivities. In this work, a central finite differences approach is taken. This is more accurate than the forward finite differences approach, at the cost of being more computationally expensive. The vectors in $\boldsymbol{S}$ are therefore calculated numerically as

$$\frac{\partial \boldsymbol{y}}{\partial \theta_j} = \frac{\boldsymbol{y}(\boldsymbol{\theta}_0 + \delta\boldsymbol{\theta}_j) - \boldsymbol{y}(\boldsymbol{\theta}_0 - \delta\boldsymbol{\theta}_j)}{2|\delta\boldsymbol{\theta}_j|} \tag{5.37}$$

where $\delta\boldsymbol{\theta}_j$ is a vector of zeros, except for the j-th location that is chosen as $\delta\theta = \xi\,\theta_{j,0}$, where $\theta_{j,0}$ denotes the initial guess for the j-th parameter, and ξ is a design parameter. In [19] a value of 0.01 is proposed, but this was found to give too large parameter changes at each iteration, and hence a smaller value of 0.001 is chosen.

Adding Regularization

In addition to the Gauss-Newton algorithm outlined until now, which was the standard Gauss-Newton algorithm augmented with a line search, it is possible to extend the method to include additional features. In this work, both regularization and out-

put weighting are tested to see if this gives better system identification results. Here, it is shown how to add regularization to the method, as this is not shown in classic references such as [19, 20]. In addition, it is shown how adding regularization may help make the optimization problem well-conditioned.

The cost function in eq. (5.25) can be augmented with a regularizing term in the following way:

$$J(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i=1}^{N}(\boldsymbol{z}[i] - \hat{\boldsymbol{y}}[i])^{\intercal}\hat{\boldsymbol{R}}^{-1}(\boldsymbol{z}[i] - \hat{\boldsymbol{y}}[i]) + \frac{1}{2}\lambda\|\Delta\boldsymbol{\theta}\|^2 \tag{5.38}$$

where $\Delta\boldsymbol{\theta} \triangleq \boldsymbol{\theta} - \boldsymbol{\theta}_0$, $\boldsymbol{\theta}_0$ are the initial parameter guesses, and λ is the regularization parameter. In addition to minimizing the simulation error, this cost function will also try to keep the parameter values close to the initial guesses. This can be useful to keep parameters from going to unrealistically large values, for example if a parameter does not significantly affect the output in the performed experiments. This may be the case if the experiments do not persistently excite all of the modes of the system, or if parameters are not uniquely identificable.

When augmenting the cost function with a regularization term, the gradient needs to be augmented as follows:

$$\frac{\partial J(\boldsymbol{\theta})}{\partial\boldsymbol{\theta}} = -\sum_{i=1}^{N}\frac{\partial\hat{\boldsymbol{y}}^{\intercal}[i]}{\partial\boldsymbol{\theta}}\hat{\boldsymbol{R}}^{-1}\boldsymbol{v}[i] + \lambda\Delta\boldsymbol{\theta} \tag{5.39}$$

where it can be seen how the regularization term will pull the parameters back towards the initial guesses at every step.

Likewise, the Ficher information matrix $\boldsymbol{M}$ is now

$$\boldsymbol{M} = \sum_{i=1}^{N}\frac{\partial\hat{\boldsymbol{y}}^{\intercal}[i]}{\partial\boldsymbol{\theta}}\hat{\boldsymbol{R}}^{-1}\frac{\partial\hat{\boldsymbol{y}}[i]}{\partial\boldsymbol{\theta}} + \lambda\boldsymbol{I} \tag{5.40}$$

where the regularization term actually improves the conditioning of $\boldsymbol{M}$, as it helps preventing it from becoming singular.

Adding Output Weighting

In this work, output weighting is used, as will be seen in chapter 6. Here, it is shown how to augment the Gauss-Newton with output weighting.

The cost function may readily be augmented with a diagonal weight matrix $\boldsymbol{W}$, in order to weight each of the outputs differently. This gives the following cost function:

$$J(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i=1}^{N}(\boldsymbol{z}[i] - \hat{\boldsymbol{y}}[i])^{\intercal}\boldsymbol{W}\hat{\boldsymbol{R}}^{-1}(\boldsymbol{z}[i] - \hat{\boldsymbol{y}}[i]) \tag{5.41}$$

This gives the following gradient

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = -\sum_{i=1}^{N}\frac{\partial \hat{\boldsymbol{y}}^{\intercal}[i]}{\partial \boldsymbol{\theta}}\boldsymbol{W}\hat{\boldsymbol{R}}^{-1}\boldsymbol{v}[i] \tag{5.42}$$

and the following Fisher information matrix

$$\boldsymbol{M} = \sum_{i=1}^{N}\frac{\partial \hat{\boldsymbol{y}}^{\intercal}[i]}{\partial \boldsymbol{\theta}}\boldsymbol{W}\hat{\boldsymbol{R}}^{-1}\frac{\partial \hat{\boldsymbol{y}}[i]}{\partial \boldsymbol{\theta}} \tag{5.43}$$

5.6 Model Validation through Residual Analysis

Residual analysis is a popular way of validating the performance of a model [19, 20, 21, 39, 40, 44]. In this work, the following metrics are used to determine the fit of a model: MAE, RMSE, GOF, and TIC. In chapter 6, these metrics will be employed to evaluate the fit of the final model.

Goodness-of-Fit (GOF) is determined for the i-th signal from the measured output $\boldsymbol{z}_i$, the initial value of the measured output z_{i_0}, and the estimated output $\boldsymbol{y}_i$:

$$\text{GOF}_i = 1 - \frac{(\boldsymbol{z}_i - \boldsymbol{y}_i)^{\intercal}(\boldsymbol{z}_i - \boldsymbol{y}_i)}{(\boldsymbol{z}_i - \boldsymbol{z}_{i_0})^{\intercal}(\boldsymbol{z}_i - \boldsymbol{z}_{i_0})} \tag{5.44}$$

GOF ranges from 0 to 1. A GOF near 1 indicates a good fit [20].

Theil's Inequality Coefficient (TIC) is defined as

$$\mathrm{TIC}_i = \frac{\sqrt{\frac{1}{N}(\boldsymbol{z}_i - \boldsymbol{y}_i)^\intercal(\boldsymbol{z}_i - \boldsymbol{y}_i)}}{\sqrt{\frac{1}{N}\boldsymbol{z}_i^\intercal \boldsymbol{z}_i} + \sqrt{\frac{1}{N}\boldsymbol{y}_i^\intercal \boldsymbol{y}_i}} \tag{5.45}$$

TIC ranges from 1 to 0, where a smaller value indicates a better fit. In general, a TIC of 0.25 - 0.3 indicates a good model [20, 44].

In addition, the Mean Absolute Error (MAE) and Root-Mean-Squared Error (RMSE) are defined as

$$\mathrm{MAE}_i = \frac{1}{N}\sum_{k=1}^{N} |z_i(k) - y_i(k)| \tag{5.46}$$

$$\mathrm{RMSE}_i = \sqrt{\frac{1}{N}\sum_{k=1}^{N} (z_i(k) - y_i(k))^2} \tag{5.47}$$

In order to compare the performance of the model on each signal, the error metrics are normalized as Normalized Mean Absolute Error (NMAE) and Normalized Root-Mean-Squared Error (nrmse):

$$\mathrm{NMAE}_i = \frac{\mathrm{MAE}_i}{\mathrm{range}(z_i)} \tag{5.48}$$

$$\mathrm{NRMSE}_i = \frac{\mathrm{RMSE}_i}{\mathrm{range}(z_i)} \tag{5.49}$$

Chapter 6

Identification Results

In this chapter, the complete final dynamic model developed for the VTOL UAV in question is presented and validated. The final model consists of the nonlinear rigid-body dynamics for a fixed-wing aircraft, a nonlinear aerodynamic model, and dynamic actuator models. The development and validation of the aerodynamic model is the primary focus of this chapter. However, the validation of the final aerodynamic model simulates both the aircraft actuator dynamics and the rigid-body dynamics as identified and presented in chapters 3 and 4, which is considered preliminary analysis of the aircraft.

The aerodynamic model identification follows the procedures outlined in chapter 5, and in this chapter, the results are presented, validated, and discussed. First, the baseline aerodynamic model developed through the use of VLM in section 4.5 is evaluated on validation data. As will be seen, this model does not accurately capture the dynamics of the aircraft. Next, the Equation-Error method from section 5.4 is used in combination with the stepwise regression procedure in section 5.4.3 to develop a suitable aerodynamic model structure based on the Flight-Test Data (FTD) and the prediction of the aerodynamic coefficients at each timestep. The model performance on one-step prediction of aerodynamic force and moment coefficients is shown, as well as the simulated model performance on the validation maneuvers. Next, the Output-Error method in section 5.5 is employed to improve the parameters from the Equation-Error method. Finally, the rudder-pitch coupling parameter is found with a

line search. Models from each step are inspected before the final aerodynamic model is presented to highlight the effect of the different techniques.

The final complete dynamic model for the aircraft predicts both the longitudinal and the lateral-directional motion well, in addition to coupled motion during the different maneuvers. As will be seen, the final aerodynamic model consists of parameters from the Output-Error method, augmented with the rudder-pitch coupling parameter. Through different measures for validation, it is concluded that the confidence in the final model is high. The model is validated through simulation on the validation maneuvers, evaluation of error metrics and residual analysis, an inspection of the model parameters and properties, and finally eigenmode analysis around trim conditions.

6.1 Aerodynamic Modeling Results

In this section, three separate aerodynamic models are presented. The models are from VLM, Equation-Error, and Output-Error, and all models are tested on a validation data set containing both 2-1-1 maneuvers for the elevators, ailerons, and rudders. Only a random subset of the maneuvers used for validation is plotted to improve this chapter's readability. However, the validation metrics used on the final model are calculated on the entire validation set for the relevant maneuvers. The final model is chosen as the parameters from the Output-Error method.

6.1.1 Baseline Aerodynamic Model from VLM

In the preliminary analysis in section 4.5, a baseline model is developed from a numerical approach, without the use of FTD. The validation results for the nonlinear VLM model can be seen in figs. 6-1 and 6-2. Although this method is convenient and much less time-consuming than obtaining experimental data, when validated on real data, it is clear that the parameters obtained through VLM do not generally comprise a good model. Results with the linear state-space model obtained through VLM are similar but perform worse around the largest perturbations from trim. As

the results obtained with the linear and nonlinear VLM are similar, only results from the nonlinear model are shown.

From the plots, it is clear that while the behavior of the aircraft seems to be captured in a general sense, the predicted state history is highly inaccurate, and there are large errors. From the plot of the longitudinal model in fig. 6-1, it is clear that the model overpredicts the influence of the elevator δ_e, for all of the longitudinal state variables. Next, for the plot of the lateral-directional model in fig. 6-2, the results are better. Surprisingly, the resulting model for lateral-directional motion performs well for the aileron deflections. The model performs worse for the rudder deflections. Despite some prediction error, the lateral-directional model obtained through VLM seems to capture the dynamics and aerodynamic forces and moments to some degree.

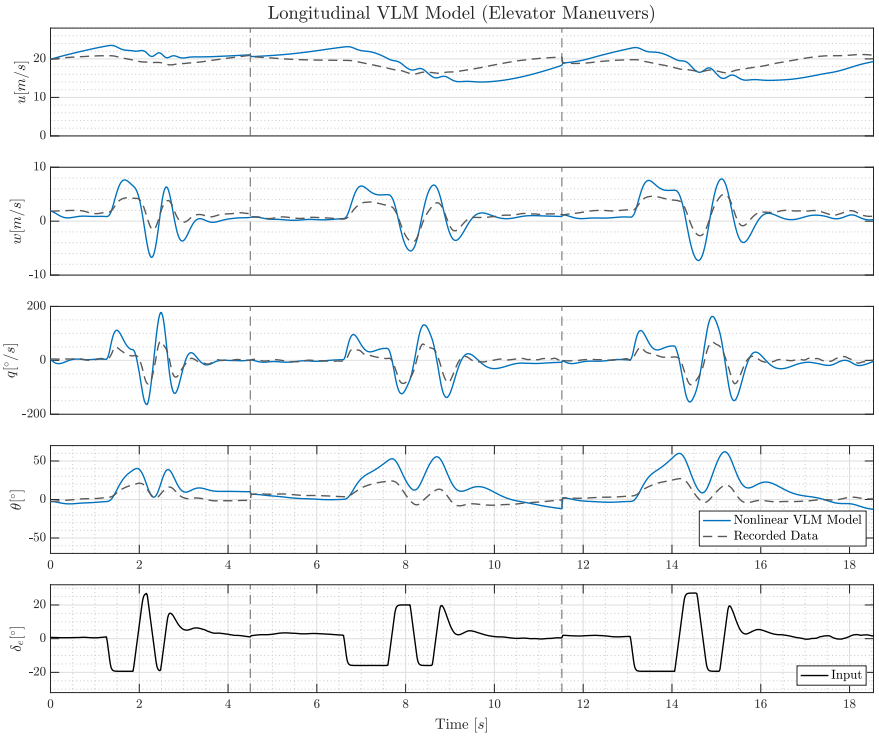


Figure 6-1: Validation of the longitudinal VLM model.

In summary, the model obtained through VLM does not perform satisfactorily. Significant errors are observed, especially in the longitudinal model, and it is clear that a purely numerical approach is insufficient to predict the aircraft's behavior accurately. In the following subsection, a new model is therefore developed from

FTD.

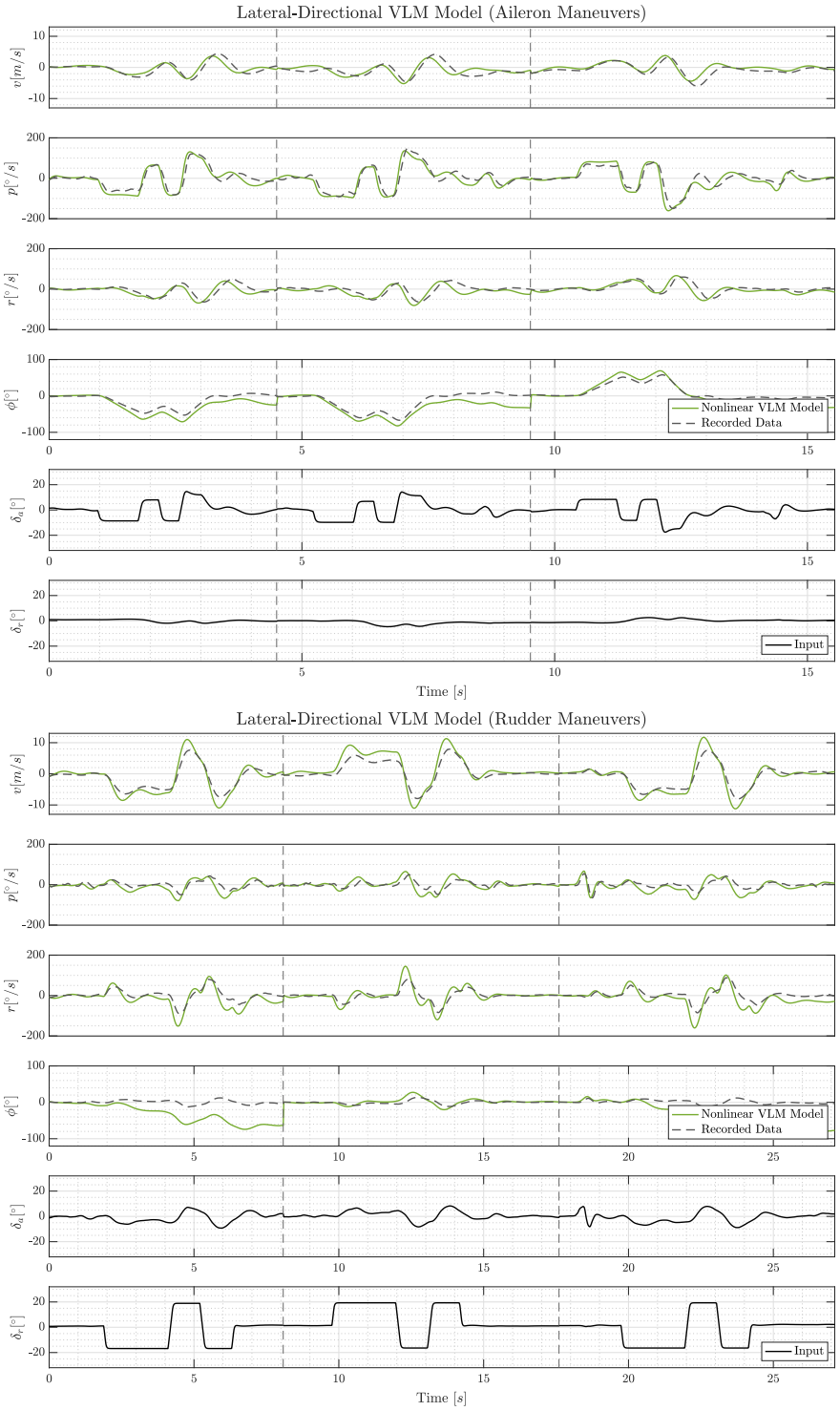


Figure 6-2: Validation of the lateral-directional VLM model.

6.1.2 Model Structure Determination with the Equation-Error Method

Next, the Equation-Error method as described in section 5.4 is used to establish a suitable aerodynamic model structure and to find estimates for the parameters. As described in section 5.4, the Equation-Error method aims only to predict the dependent variables at each timestep and treats every timestep as individual data points. The dependent variables to predict are the aerodynamic force and moment coefficients. The longitudinal aerodynamic coefficients c_L, c_D and c_m are predicted from elevator-deflection maneuvers, while the lateral-directional aerodynamic coefficients c_Y, c_l and c_n are predicted from both aileron and rudder-deflection maneuvers.

The stepwise regression procedure outlined in section 5.4.3 is used to determine which explanatory variables should be used, based on the available data. A regressor is only added if it improves the R^2 value with at least 2% and has an F-statistic of at least 4. Compared to [19] where it is suggested that terms which increase R^2 with less than 0.5% are insignificant, using a threshold of $R^2 \geq 2\%$ is quite conservative. This is done to prevent structural overfitting, as is discussed in section 6.3.

Longitudinal Equation-Error Model

The dependent variables for the longitudinal model are the force and moment coefficients c_L, c_D, and c_m. The independent variables are the longitudinal quantities α, and $\hat{q}$ and longitudinal input δ_e as defined in chapter 3. The throttle is not included as an independent variable, as the aerodynamic force in the x-direction is modeled as separate from the thrust force generated by the pusher propeller, as discussed in chapter 3.

The longitudinal aerodynamic coefficients are modelled as shown in eq. (3.31), and all the regressors in the proposed model structure are considered. Through the use of stepwise regression on the FTD, the model structure containing only the following

regressors was found to best explain the observed dynamics:

$$c_D = c_{D0} + c_{D\alpha}\alpha + c_{\alpha^2}\alpha^2 + c_{D\hat{q}}\hat{q} + (c_{D\delta_e} + c_{D\delta_e\alpha}\alpha)\delta_e \tag{6.1a}$$

$$c_L = c_{L0} + c_{L\alpha}\alpha + c_{L\alpha^2}\alpha^2 + c_{L\delta_e}\delta_e \tag{6.1b}$$

$$c_m = c_{m0} + c_{m\alpha}\alpha + c_{m\hat{q}}\hat{q} + c_{m\delta_e}\delta_e \tag{6.1c}$$

A plot showing the one-step prediction of the longitudinal coefficients on the validation data can be seen in fig. 6-3. As can be seen in the plot, the aerodynamic model obtained from Equation-Error explains the aerodynamic coefficients well. However, it is important to remember that this plot only shows the one-step prediction and that while these predictions are good, a dynamic model should predict the dynamics of a system over time. Therefore, the aerodynamic coefficients are inserted into the full nonlinear rigid-body model presented in chapter 3 and simulated. The simulated results compared with real validation data can be seen in fig. 6-5a, and various error metrics evaluated on the model can be seen in fig. 6-5b.

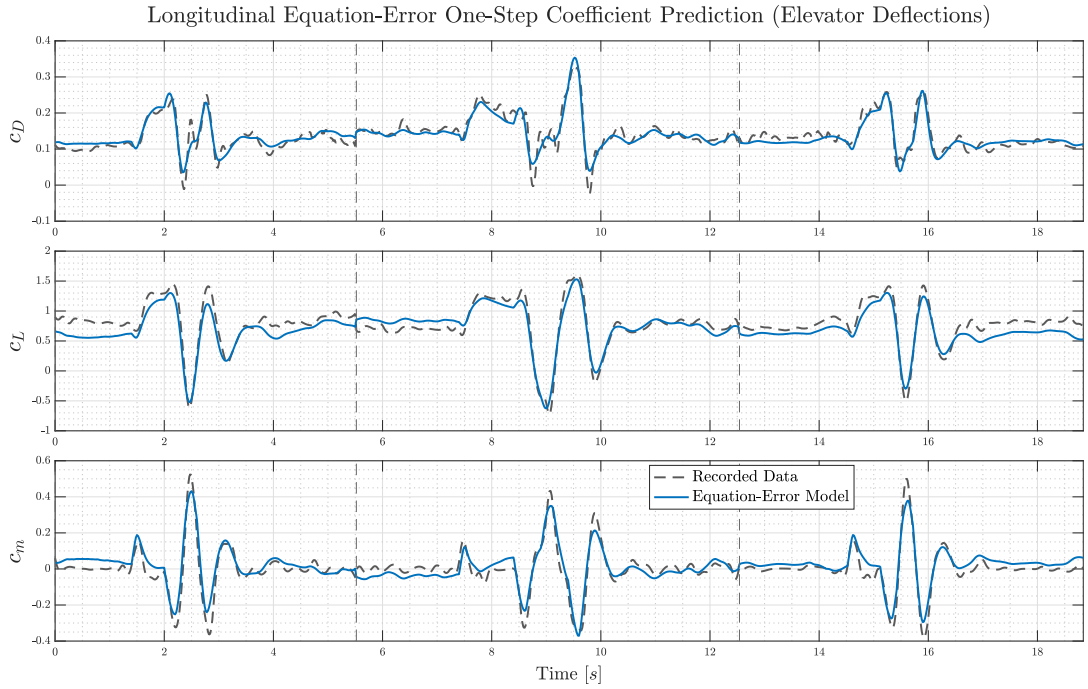


Figure 6-3: Longitudinal one-step coefficient predictions from the Equation-Error model.

Lateral-Directional Equation-Error Model

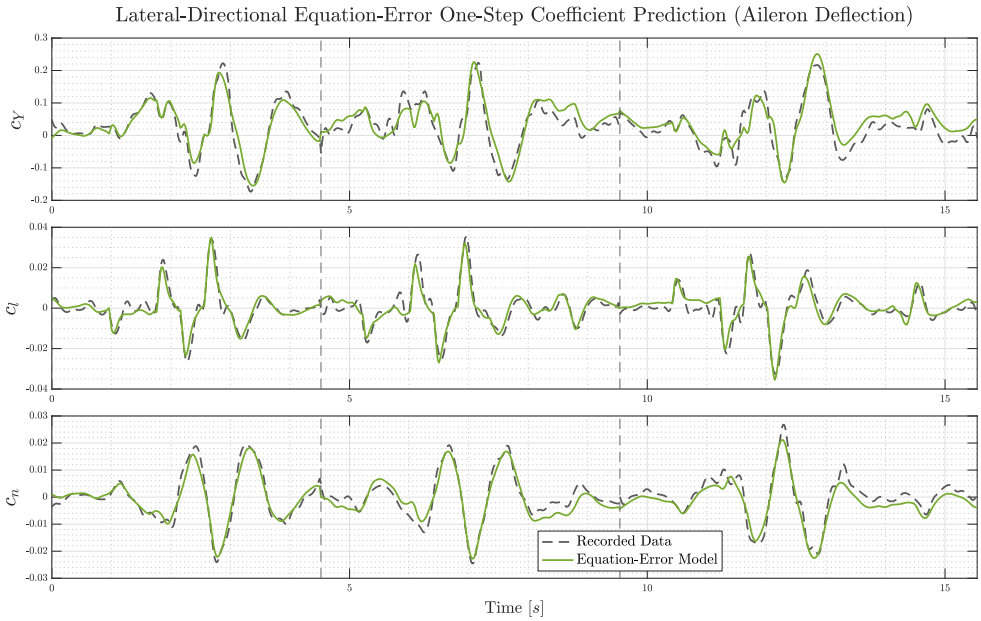
For the lateral directional model, the dependent variables are the force and moment coefficients c_Y, c_l, and c_n. The independent variables are the lateral-directional quantities, β, $\hat{p}$, $\hat{r}$ and lateral inputs δ_a and δ_r. The proposed model structure is presented in eq. (3.32) From the stepwise regression procedure, the model structure with the following regressors is chosen:

$$c_Y = c_{Y0} + c_{D\beta}\beta + c_{Y\hat{p}}\hat{p} + c_{Y\delta_a}\delta_a + c_{Y\delta_r}\delta_r \tag{6.2a}$$

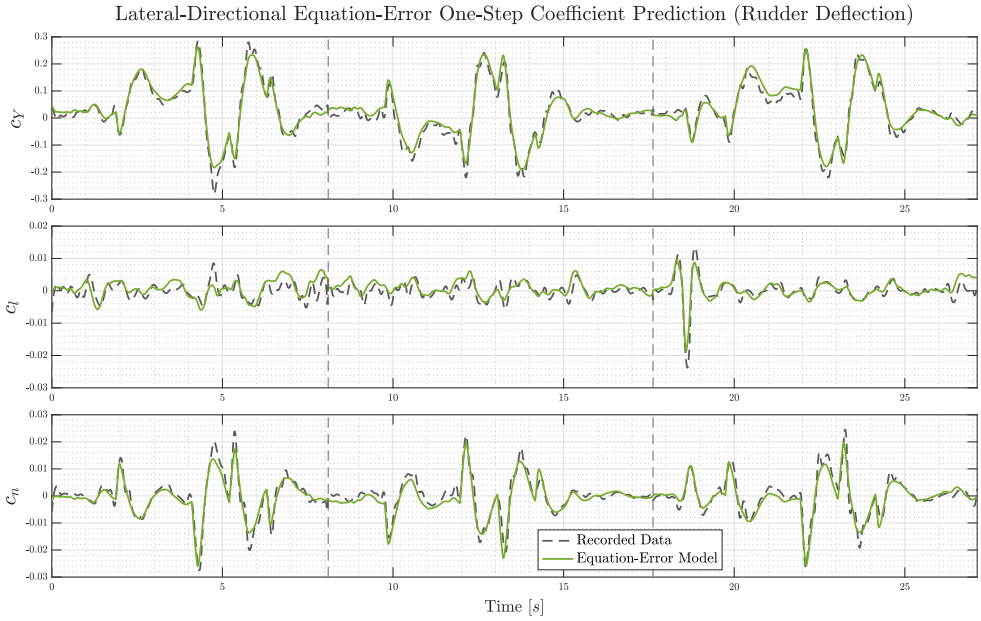
$$c_l = c_{l0} + c_{l\beta}\beta + c_{l\hat{p}}\hat{p} + c_{l\hat{r}}\hat{r} + c_{l\delta_a}\delta_a \tag{6.2b}$$

$$c_n = c_{n0} + c_{n\beta}\beta + c_{n\hat{p}}\hat{p} + c_{n\hat{r}}\hat{r} + c_{n\delta_r}\delta_r \tag{6.2c}$$

Nonlinear terms as described in eq. (3.32) were considered by the stepwise regression procedure, but not found to significantly improve the data prediction. The one-step predictions for lateral-directional coefficients for both aileron and rudder maneuvers can be seen in figs. 6-4a and 6-4b. The full nonlinear model with the coefficient estimates from the Equation-Error method is simulated on the validation data in fig. 6-7, and the error metrics can be seen in fig. 6-6.



(a) Aileron maneuvers.



(b) Rudder maneuvers.

Figure 6-4: Lateral-directional one-step coefficient predictions from the Equation-Error model.

6.1.3 Parameter Estimation with the Output-Error Method

In this section, a model is developed using the Output-Error method from section 5.5 to obtain final parameter values for the model structure formed with the Equation-Error method. This time, the unknown parameters are estimated through a nonlinear optimization scheme. The cost function is the sum of the squared errors of the simulated model response over all the maneuvers. This contrasts the sum of squared one-step prediction errors that is the cost function used by the Equation-Error method. Therefore, the Output-Error method yields different parameters values when compared to the Equation-Error method, as parameters that cause signal divergence are now penalized. In addition, the Output-Error accounts for measurement noise, which the Equation-Error method does not, making the Output-Error the preferred method.

As stated in [19] and used in [39, 40], the Equation-Error method helps provide the Output-Error method with good initial guesses for the parameter values. The Output-Error implemented with the Newton-Raphson optimization step converges if it starts close to the actual parameter values; otherwise, it is likely to diverge [19]. Thus, in this work, the parameter values from the Equation-Error method are used as initial guesses. All parameters are updated with the Output-Error method.

In table 6.1, the parameters used for convergence criterion and gradient calculation for the Output-Error method in this work can be seen. The same parameters are used for the longitudinal and the lateral-directional maneuvers.

Parameter	Value
$\epsilon_{\Delta \boldsymbol{R}}$	0.05
$\epsilon_{\Delta J(\boldsymbol{\theta})}$	0.001
$\epsilon_{\Delta \boldsymbol{\theta}}$	0.001
$\epsilon_{\frac{\partial J}{\partial \boldsymbol{\theta}}}$	0.05
ξ	0.001

Table 6.1: Parameters used for the Output-Error algorithm.

Longitudinal Output-Error Model

The longitudinal output errors are minimized for the Output-Error model are u, w, q and θ. For the longitudinal model, no regularization is used. However, a problem found during the optimization is that large errors in u are not penalized enough, which causes degraded simulation results. This is because the magnitude of u is large compared to the signals in w, p, and θ, which causes a disproportionally large covariance estimate, resulting in errors in u weighted less in the cost function.

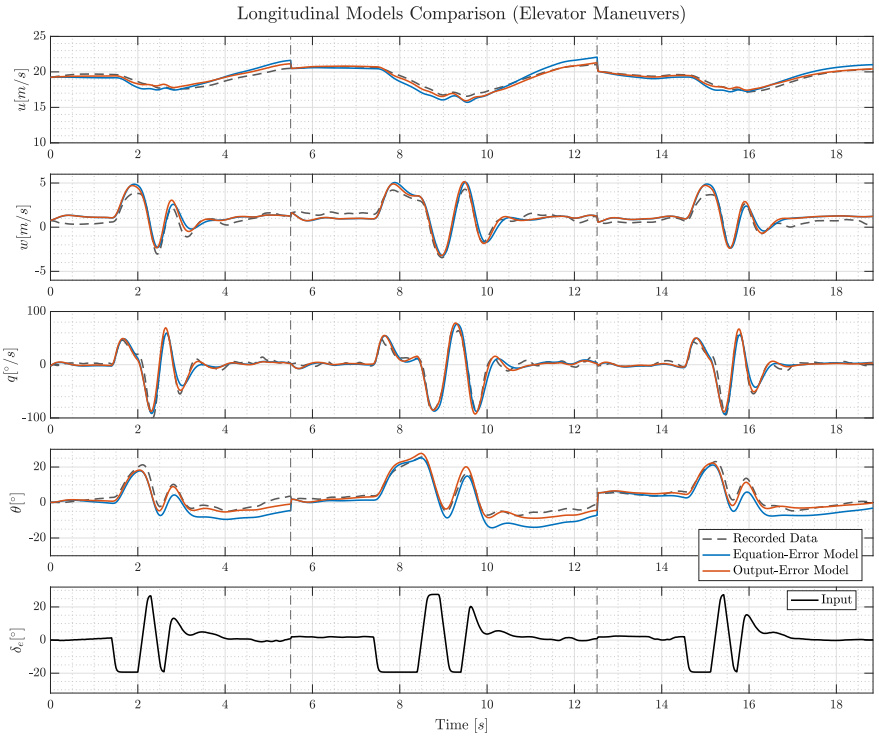
In order to obtain good results for the longitudinal model, the cost function is augmented in such a way that the signals are normalized before the cost is computed. To achieve this, the cost function is augmented with a diagonal weighting matrix, as described in section 5.5.1. To normalize the squared errors, the errors are divided by the squared nominal maximum values of each signal. Therefore, the following diagonal weight matrix is used for the longitudinal Output-Error optimization:

$$\boldsymbol{W} = \begin{bmatrix} \frac{1}{\max(u_0)^2} & 0 & 0 & 0 \\ 0 & \frac{1}{\max(w_0)^2} & 0 & 0 \\ 0 & 0 & \frac{1}{\max(q_0)^2} & 0 \\ 0 & 0 & 0 & \frac{1}{\max(\theta_0)^2} \end{bmatrix} \tag{6.3}$$

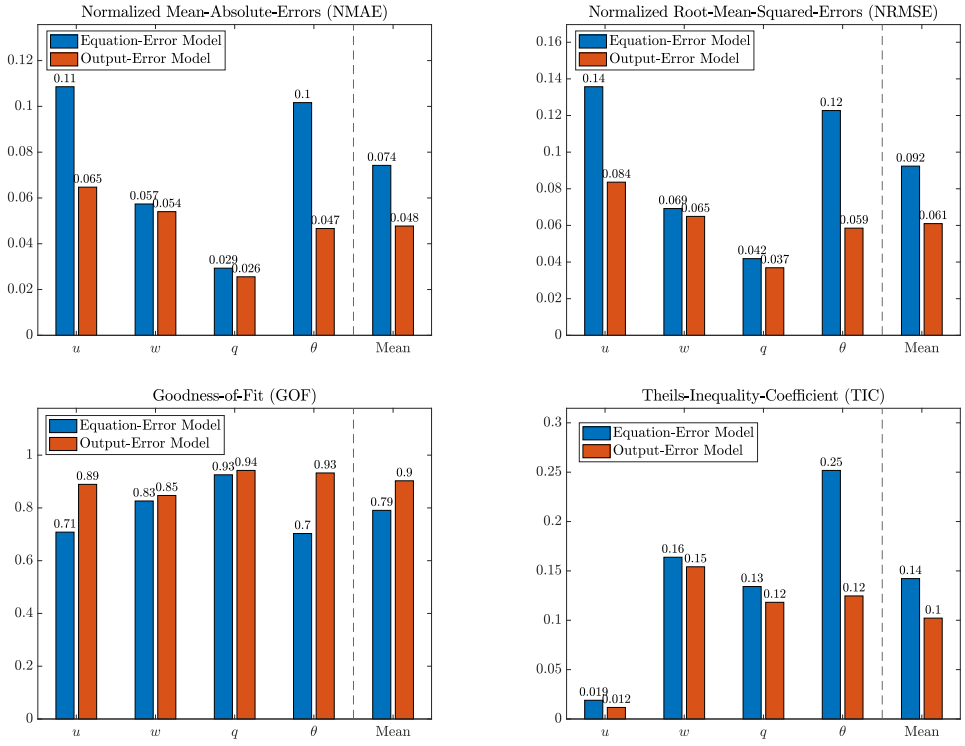
$$\max(u_0) = 21\text{ms}^{-1},\ \max(w_0) = 4\text{ms}^{-1},$$

$$\max(q_0) = 70^\circ\text{s}^{-1},\ \max(\theta_0) = 23^\circ$$

A plot showing the simulated performance of the longitudinal Output-Error model can be seen in fig. 6-5a. The error metrics for the model can be seen in fig. 6-5b. The results are discussed in detail in section 6.3.



(a) Simulated performance on validation data.



(b) Error metrics.

Figure 6-5: Validation plots and error metrics for the longitudinal models from the Equation-Error method and the Output-Error method.

Lateral-Directional Output-Error Model

The outputs errors to be minimized for the lateral-directional Output-Error model are v, p, r and ϕ. As ψ is an output that follows a purely kinematic relationship with the other states and does not feed back into the dynamics, it is not included as an output for the optimization.

For the lateral-directional model, no regularization or weighting is used. The simulated performances for the lateral-directional model on the validation data can be seen in fig. 6-7. The corresponding error metrics can be found in fig. 6-6. The results are discussed in section 6.3.

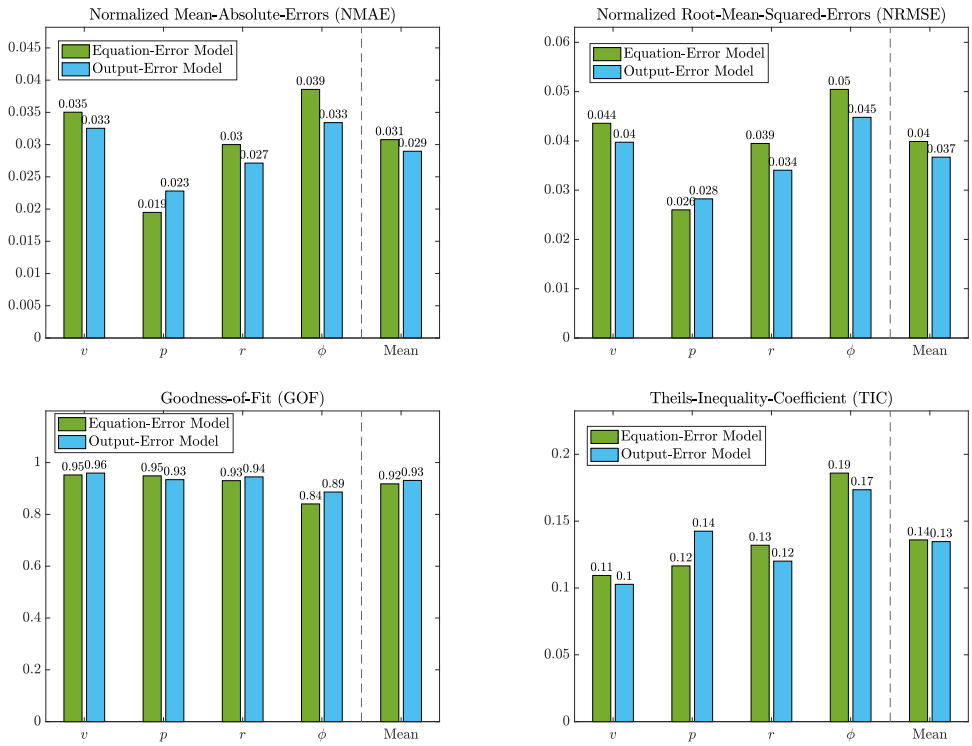


Figure 6-6: Error metrics for the lateral-directional models from the Equation-Error method and the Output-Error method.

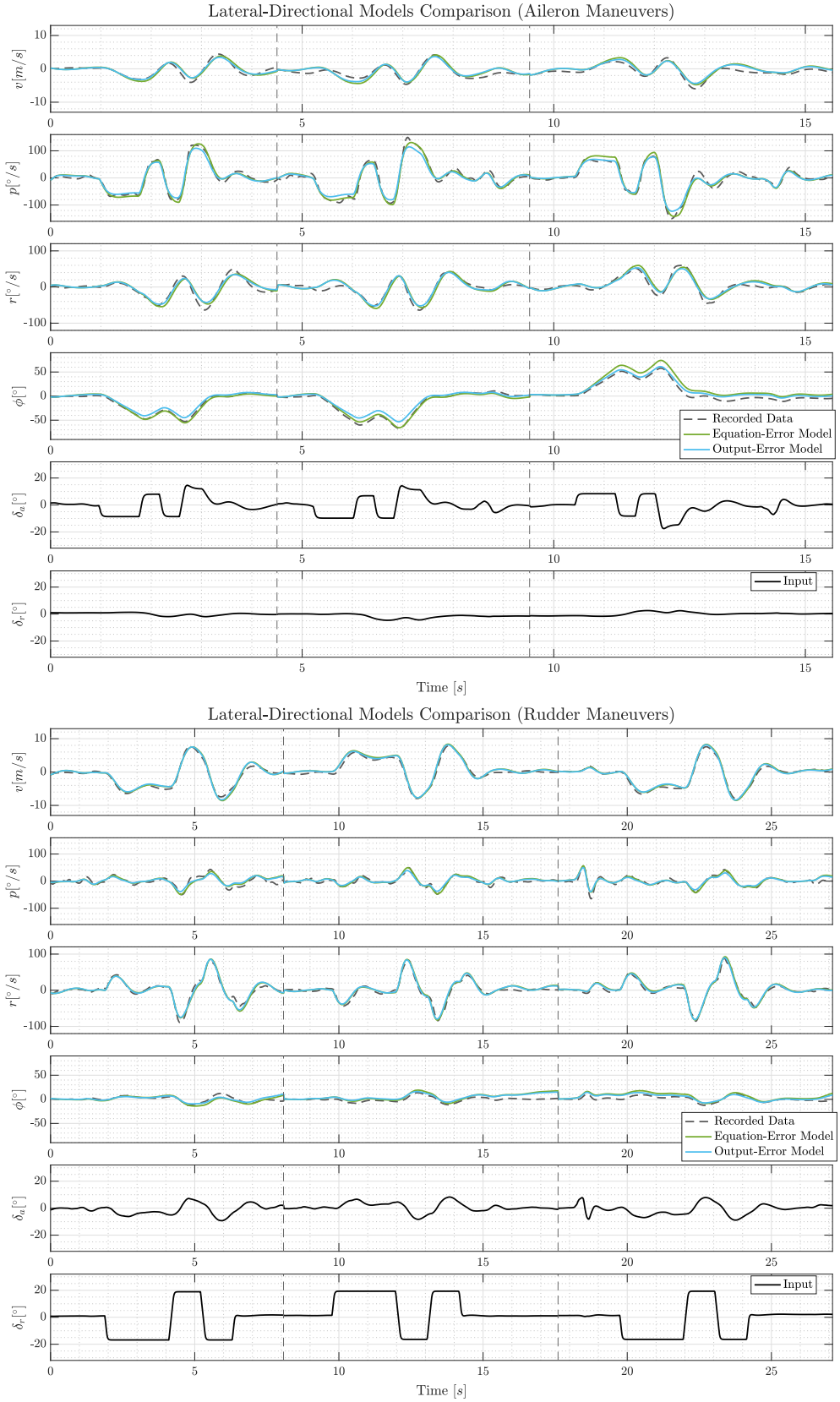


Figure 6-7: Validation plots for the lateral-directional models from the Equation-Error method and the Output-Error method.

6.1.4 Estimation of Rudder-Pitch Coupling

Finally, the last model parameter $c_{m\delta_r^2}$ is estimated. As this parameter is only visible as coupling in the pitch angle for large rudder deflections, this parameter is estimated separately from the 2-1-1 rudder maneuvers as a final step. The estimation procedure for this parameter is simplified compared to the rest of the model development.

First, the best-suited regressor is chosen as the regressor that is strongest correlated with the pitch coefficient residual, where the part that the current model already explains is removed, similar to the procedure outlined in section 5.4. For the potential regressors, δ_r^2, abs(δ_r), and abs($\sqrt{\delta_r}$) are considered, where δ_r^2 is found to have the strongest correlation with the pitch coefficient residual. Next, a simple line search is used to find the value of $c_{m\delta_r^2}$ which minimizes the squared simulation error over all the training maneuvers with yaw deflections. A plot of the line search is shown in fig. 6-8

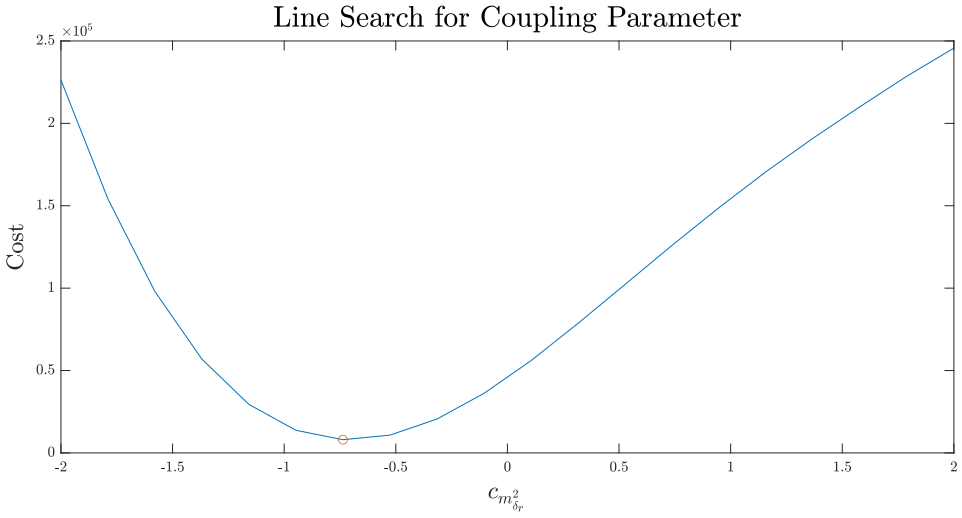


Figure 6-8: Line search to find $c_{m\delta_r^2}$.

6.2 Final Model for the Babyshark 260 VTOL UAV

In this section, the final model for the Babyshark 260 VTOL UAV is summarized as a stand-alone section, together with the resulting parameters and error metrics. The purpose of this is to faciliate for easy use of the model, by collecting all the information in one place. A ready-to-use implementation can be found at [1]. The complete model comprises multiple components, consisting of individual actuator models, rigid-body dynamics, and aerodynamic modeling based on flight-test data. Individual components of the model are presented in detail in previous chapters. The discussion of the final model is presented in the next section.

6.2.1 Final Equations-of-Motion

The entire state vector $\boldsymbol{x}$ and input vector $\boldsymbol{u}$ are given as

$$\boldsymbol{x} = [u,\, v,\, w,\, p,\, q,\, r,\, \phi,\, \theta,\, \psi,\, \delta_a,\, \delta_e,\, \delta_r]^\intercal \tag{6.4}$$

$$\boldsymbol{u} = [\delta_a^{sp},\, \delta_e^{sp},\, \delta_r^{sp},\, \delta_t,\, \delta_{\mathrm{MR},1},\, \delta_{\mathrm{MR},2},\, \delta_{\mathrm{MR},3},\, \delta_{\mathrm{MR},4}]^\intercal \tag{6.5}$$

and the nonlinear dynamics are formulated as

$$\dot{u} = rv - qw + \frac{1}{\mathsf{m}}(X + T - \mathsf{m}g\sin\theta) \tag{6.6a}$$

$$\dot{v} = pw - ru + \frac{1}{\mathsf{m}}(Y + \mathsf{m}g\sin\phi\cos\theta) \tag{6.6b}$$

$$\dot{w} = qu - pv + \frac{1}{\mathsf{m}}(Z - T_{\mathrm{MR}} + \mathsf{m}g\cos\phi\cos\theta) \tag{6.6c}$$

$$\dot{p} = \Gamma_1 pq - \Gamma_2 qr + \Gamma_3\tau_x + \Gamma_4\tau_z \tag{6.6d}$$

$$\dot{q} = \Gamma_5 pr - \Gamma_6(p^2 - r^2) + \frac{1}{J_{yy}}\tau_y \tag{6.6e}$$

$$\dot{r} = \Gamma_7 pq - \Gamma_1 qr + \Gamma_4\tau_x + \Gamma_8\tau_z \tag{6.6f}$$

$$\dot{\phi} = p + \tan\theta(q\sin\phi + r\cos\phi) \tag{6.6g}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{6.6h}$$

$$\dot{\psi} = \frac{q\sin\phi + r\cos\phi}{\cos\theta} \tag{6.6i}$$

where the control surface deflections are modelled separately as

$$\dot{\delta}_k = \text{sat}\left(\frac{-\frac{1}{T_{\text{servo}}}\delta_k + \frac{1}{T_{\text{servo}}}\delta_k^{sp}}{\dot{\delta}_{max}}\right)\dot{\delta}_{max}, \quad \text{sat}(x) = \begin{cases} x & \text{for } |x| \leq 1 \\ 1 & \text{for } x > 1 \end{cases}, \quad k = a, e, r$$

The aerodynamic forces are given as

$$X = \bar{q}Sc_X \tag{6.7a}$$

$$Y = \bar{q}Sc_Y \tag{6.7b}$$

$$Z = \bar{q}Sc_Z \tag{6.7c}$$

with the aerodynamic force coefficients

$$c_X = -c_D\cos\alpha + c_L\sin\alpha \tag{6.8a}$$

$$c_Y = c_{Y0} + c_{D\beta}\beta + c_{Y\hat{p}}\hat{p} + c_{Y\delta_a}\Delta\delta_a + c_{Y\delta_r}\Delta\delta_r \tag{6.8b}$$

$$c_Z = -c_D\sin\alpha - c_L\cos\alpha \tag{6.8c}$$

$$c_D = c_{D0} + c_{D\alpha}\alpha + c_{\alpha^2}\alpha^2 + c_{D\hat{q}}\hat{q} + (c_{D\delta_e} + c_{D\delta_e\alpha}\alpha)\Delta\delta_e \tag{6.8d}$$

$$c_L = c_{L0} + c_{L\alpha}\alpha + c_{L\alpha^2}\alpha^2 + c_{L\delta_e}\delta_e \tag{6.8e}$$

Similarly, the aerodynamic moments are modelled as

$$l = \bar{q}c_l Sb \tag{6.9a}$$

$$m = \bar{q}c_m S\bar{c} \tag{6.9b}$$

$$n = \bar{q}c_n Sb \tag{6.9c}$$

and the aerodynamic moment coefficients

$$c_l = c_{l0} + c_{l\beta}\beta + c_{l\hat{p}}\hat{p} + c_{l\hat{r}}\hat{r} + c_{l\delta_a}\Delta\delta_a \tag{6.10a}$$

$$c_m = c_{m0} + c_{m\alpha}\alpha + c_{m\hat{q}}\hat{q} + c_{m\delta_e}\Delta\delta_e + c_{m\delta_r^2}\delta_r^2 \tag{6.10b}$$

$$c_n = c_{n0} + c_{n\beta}\beta + c_{n\hat{p}}\hat{p} + c_{n\hat{r}}\hat{r} + c_{n\delta_r}\Delta\delta_r \tag{6.10c}$$

The non-dimensional angular rates are defined as

$$\hat{p} = \frac{bp}{2V_0}, \quad \hat{q} = \frac{\bar{c}q}{2V_0}, \quad \hat{r} = \frac{br}{2V_0} \tag{6.11a}$$

The dynamic pressure, aerodynamic angles, and airspeed are defined as

$$\bar{q} = \frac{1}{2}\rho V^2 \tag{6.12a}$$

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right) \tag{6.12b}$$

$$\beta = \sin^{-1}\left(\frac{v}{V}\right) \tag{6.12c}$$

$$V = \sqrt{u^2 + v^2 + w^2} \tag{6.12d}$$

The control surface deflection angles away from trim are defined as

$$\Delta\delta_a = \delta_a - \delta_a^* \tag{6.13a}$$

$$\Delta\delta_e = \delta_e - \delta_e^* \tag{6.13b}$$

$$\Delta\delta_r = \delta_r - \delta_r^* \tag{6.13c}$$

The total moments acting on the aircraft are defined as

$$\tau_x = l + \left(\sum_{i=1}^{4} -r_{i,y} T_{\mathrm{MR},i}\right) \tag{6.14a}$$

$$\tau_y = m + \left(\sum_{i=1}^{4} r_{i,x} T_{\mathrm{MR},i}\right) \tag{6.14b}$$

$$\tau_z = n + (Q_{\mathrm{MR},1} + Q_{\mathrm{MR},2} - Q_{\mathrm{MR},3} - Q_{\mathrm{MR},4}) \tag{6.14c}$$

Finally, the forces and moments generated by the propellers are modeled as

$$T = \rho D_{\mathrm{FW}}^4 c_{T_{\mathrm{FW}}} \delta_t \tag{6.15a}$$

$$T_{\mathrm{MR},i} = \rho D_{\mathrm{MR}}^4 c_{T_{\mathrm{MR}}} \delta_{\mathrm{MR},i} \tag{6.15b}$$

$$Q_{\mathrm{MR},i} = \rho D_{MR}^5 c_{Q_{\mathrm{MR}}} \delta_{\mathrm{MR},i} \tag{6.15c}$$

where the total multirotor propeller thrust is given as

$$T_{\mathrm{MR}} = \sum_{i=1}^{4} T_{\mathrm{MR},i} \tag{6.16a}$$

6.2.2 Final Parameters

There are several parameters in the presented model. Here, all of the parameters are collected. The static aircraft parameters and the actuator parameters can be seen in tables 6.2 and 6.3. The aerodynamic parameters are found in table 6.4. The trim conditions can be found in table 6.5.

Parameter	Value
b	2.5m
$\bar{c}$	0.242m
S	0.6617m
m	12.140 kg
J_{xx}	0.7316
J_{yy}	1.0664
J_{zz}	1.6917
J_{xz}	0.1277
Γ	$J_{xx}J_{zz} - J_{xz}^2$
Γ_1	$\frac{J_{xz}(J_{xx}-J_{yy}+J_{zz})}{\Gamma}$
Γ_2	$\frac{(J_{zz}(J_{zz}-J_{yy})+J_{xz}^2)}{\Gamma}$
Γ_3	$\frac{J_{zz}}{\Gamma}$
Γ_4	$\frac{J_{xz}}{\Gamma}$
Γ_5	$\frac{(J_{zz}-J_{xx})}{J_{yy}}$
Γ_6	$\frac{J_{xz}}{J_{yy}}$
Γ_7	$\frac{J_{xx}(J_{xx}-J_{yy})+J_{xz}^2}{\Gamma}$
Γ_8	$\frac{J_{xx}}{\Gamma}$

Table 6.2: Static model parameters.

Parameter	Value
T_{servo}	0.028s
$\dot{\delta}_{\max}$	$200^\circ\,\text{s}^{-1} = 3.491\ \text{rad}\,\text{s}^{-1}$
$\delta_{a,\max}$	25°
$\delta_{e,\max}$	25°
$\delta_{r,\max}$	22°
ρ	1.225kg/m^3
D_{FW}	0.3810m
D_{MR}	0.4064m
$c_{T_{\text{FW}}}$	0.0840
$c_{T_{\text{MR}}}$	0.0994
$c_{Q_{\text{MR}}}$	0.006338
$r_{1,x}$	0.353m
$r_{1,y}$	0.400m
$r_{2,x}$	-0.447m
$r_{2,y}$	-0.400m
$r_{3,x}$	0.353m
$r_{3,y}$	-0.400m
$r_{4,x}$	-0.447m
$r_{4,y}$	0.400m

Table 6.3: Actuator model parameters.

Coefficient	Value
c_{D0}	0.0820
$c_{D\alpha}$	0.272
$c_{D\alpha^2}$	1.810
$c_{D\hat{q}}$	10.102
$c_{D\delta_e}$	0.132
$c_{D\delta_e\alpha}$	0.450
c_{L0}	0.461
$c_{L\alpha}$	5.325
$c_{L\alpha^2}$	-3.969
$c_{L\delta_e}$	0.521
c_{m0}	0.0950
$c_{m\alpha}$	-1.495
$c_{m\hat{q}}$	-13.140
$c_{m\delta_e}$	-0.675
$c_{m\delta_r^2}$	-0.737

Coefficient	Value
c_{Y0}	0.0108
$c_{Y\beta}$	-0.731
$c_{Y\hat{p}}$	1.0778
$c_{Y\delta_a}$	-0.341
$c_{Y\delta_r}$	0.337
c_{l0}	0.000411
$c_{l\beta}$	-0.0354
$c_{l\hat{p}}$	-0.242
$c_{l\hat{r}}$	0.0953
$c_{l\delta_a}$	0.124
c_{n0}	0.00106
$c_{n\beta}$	0.0759
$c_{n\hat{p}}$	-0.0823
$c_{n\hat{r}}$	-0.0752
$c_{n\delta_r}$	-0.0537

Table 6.4: Aerodynamic model coefficients.

Parameter	Value
α^*	3°
V^*	$21\mathrm{m\,s^{-1}}$
δ_a^*	3.0309°
δ_e^*	-5.6436°
δ_r^*	0°
δ_t^*	${\eta_{\mathrm{FW}}^*}^2 = 125^2$
θ^*	3°
u^*	20.971m/s
w^*	1.099m/s

Table 6.5: Trim conditions.

6.2.3 Linearized Aircraft Model

The aircraft EOMs presented in section 6.2.1 can be linearized around trim conditions for the individual longitudinal and lateral-directional aircraft state as shown in section 3.4. Here, the linearized model is presented. The multirotor propellers are not included, and the control surface deflections are simulated separately with their nonlinear model as presented in section 6.2.1. The linearized state-space model can be seen in eqs. (6.17) and (6.18).

$$\boldsymbol{A}_{\text{lon}} = \begin{bmatrix} -0.0975 & 0.6862 & -1.9548 & -9.7965 \\ -0.8432 & -3.5172 & 20.9262 & -0.5138 \\ 0.2156 & -2.8796 & -3.0709 & 0 \\ 0 & 0 & 1.0000 & 0 \end{bmatrix}, \quad \boldsymbol{B}_{\text{lon}} = \begin{bmatrix} -1.8819 & 0.0002 \\ -7.7815 & 0 \\ -27.3955 & 0 \\ 0 & 0 \end{bmatrix} \tag{6.17}$$

$$\boldsymbol{A}_{\text{lat}} = \begin{bmatrix} -0.5125 & 2.0435 & -20.9710 & 9.7965 \\ -0.8731 & -9.1386 & 3.3002 & 0 \\ 0.8886 & -1.9841 & -0.9337 & 0 \\ 0 & 1.0000 & 0.0524 & 0 \end{bmatrix}, \quad \boldsymbol{B}_{\text{lat}} = \begin{bmatrix} -5.0270 & 4.9636 \\ 76.4902 & -2.5082 \\ 5.7709 & -14.3773 \\ 0 & 0 \end{bmatrix} \tag{6.18}$$

6.2.4 Final Model Performance

The performance of the final model on the individual validation data for longitudinal and lateral-directional maneuvers is summarized in table 6.6. In addition, a plot showing a simulation for the full model on a roll, pitch, and yaw validation maneuver can be seen in figs. 6-9 and 6-10, where all aircraft states are shown for all types of maneuvers. As shown in the error metrics and the plot, the model exhibits excellent results for predicting longitudinal motion, lateral-directional motion, and coupled motion during the maneuvers.

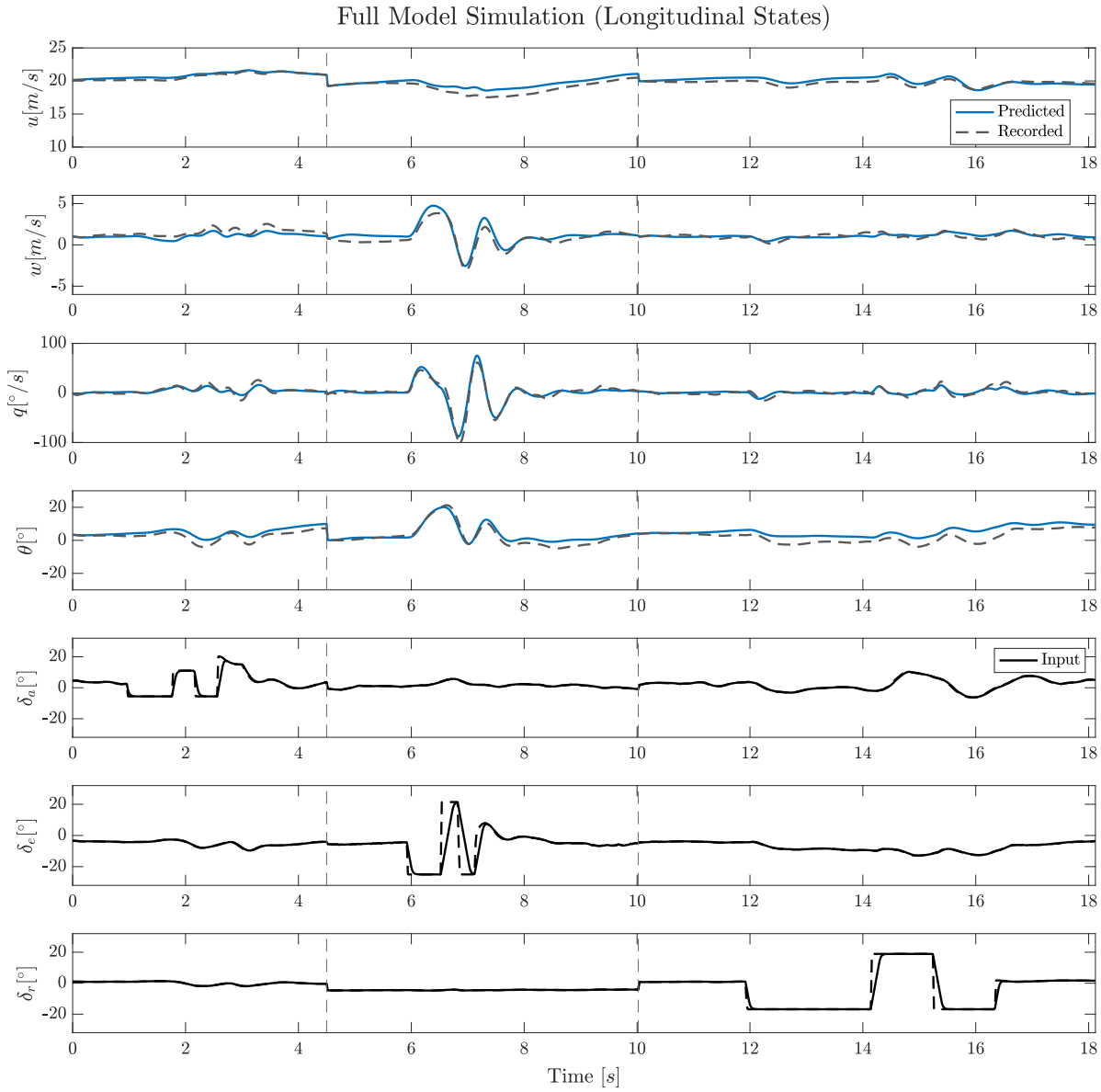


Figure 6-9: Longitudinal states for full nonlinear model simulation.

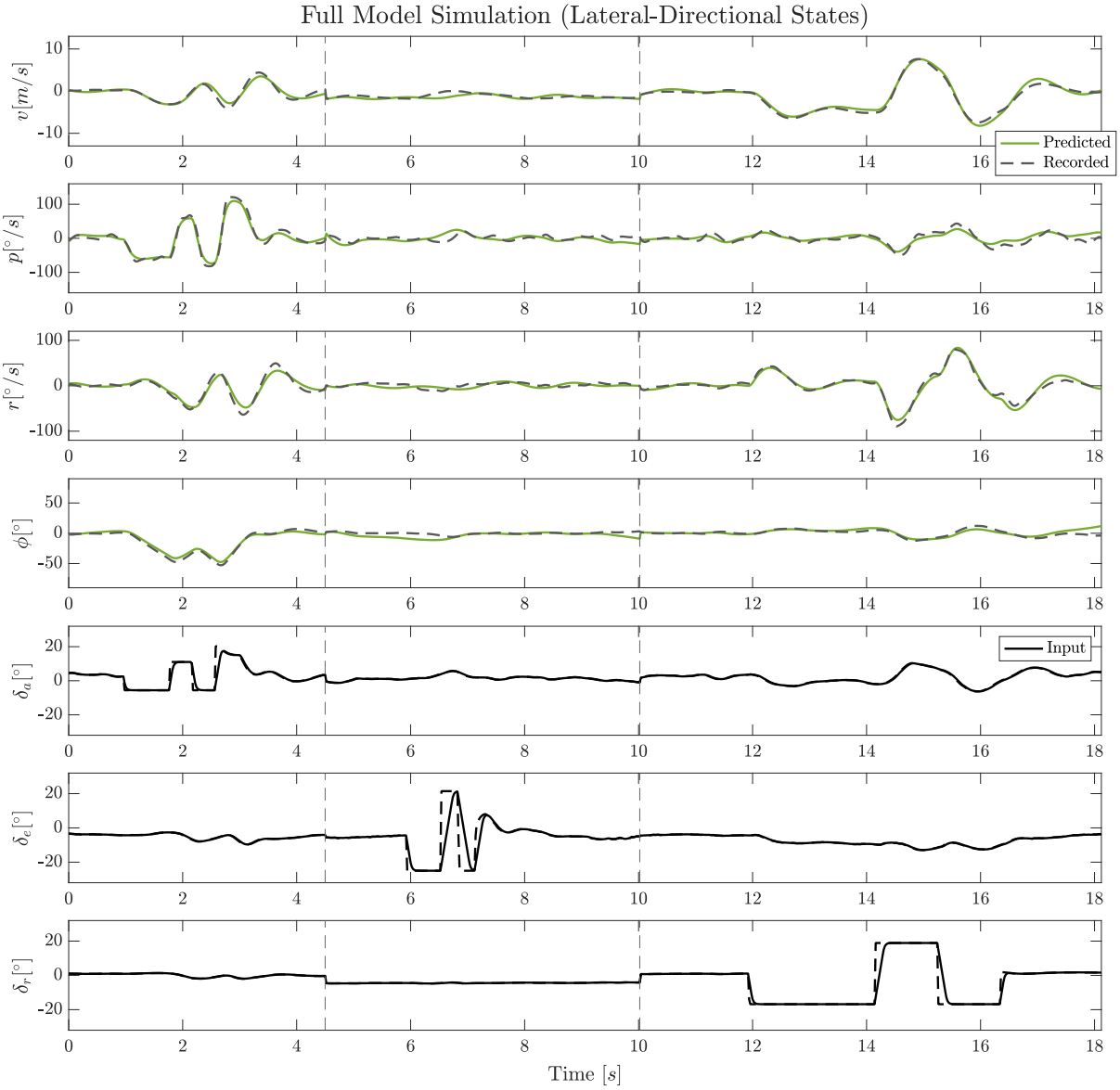


Figure 6-10: Lateral-Directional states for full nonlinear model simulation.

Signal	MAE	RMSE	GOF	TIC
u	0.347m/s	0.449m/s	0.89	0.012
w	0.462m/s	0.555m/s	0.85	0.15
q	4.425°/s	6.385°/s	0.94	0.12
θ	1.807°	2.266°	0.93	0.12
Mean	–	–	0.90	0.10
v	0.518m/s	0.640m/s	0.96	0.1
p	6.843°/s	8.848°/s	0.93	0.12
r	4.817°/s	6.234°/s	0.94	0.12
ϕ	4.166°	5.308°	0.89	0.17
Mean	–	–	0.93	0.13

Table 6.6: Error metrics for the final model.

6.3 Discussion and Validation of the Final Model

In this section, the results presented for the final model are validated and discussed. There are many different ways of validating a model. In this work, residual analysis through statistical error metrics is used, in combination with an evaluation of the models' physical properties, to gain confidence in the model. In particular, the Goodness-of-Fit (GOF), Theil's Inequality Coefficient (TIC),Mean Absolute Error (MAE), and Root-Mean-Squared Error (RMSE) are evaluated for the model. The model's static lift, drag, and pitch moment curves are analyzed and plotted with recorded data from the flight experiments. Then, the individual aerodynamic derivatives are analyzed, both the static and dynamic stability derivatives. Further, the nonlinear model is linearized around nominal flight conditions, and an eigenmode analysis of the dominant modes around trim conditions is performed. Finally, the developed model structure is evaluated, and potential pitfalls of an automated model development routine and how these are avoided in this thesis are highlighted.

6.3.1 Residual Analysis

In this subsection, a residual analysis of the final model is performed. The statistical metrics are defined as in section 5.6. In table 6.6, the error metrics for the final model are presented. These, along with the comparisons between the Equation-Error

method and the Output-Error presented in figs. 6-5 to 6-7, form the basis for the discussion in this subsection.

Residual Analysis of Longitudinal Model

For the longitudinal model, error metrics in fig. 6-5b show good results for all signals. As can be seen from the comparison, the Output-Error method increases the model performance for all signals.

First, the dimensional error metrics MAE and RMSE are evaluated, to give a sense of the magnitude of the expected error. For the velocities u and w, the RMSE and MAE are of the order ≈ 0.5m. This is considered low, compared to the nominal airspeed of $21\mathrm{m\,s^{-1}}$. For the pitch rate q the RMSE and MAE are on the order of $\approx 5^\circ\,\mathrm{s^{-1}}$, which is small compared to normal values for q which are on the order of $\approx 70^\circ\,\mathrm{s^{-1}}$ for the experiments. Finally, the RMSE and MAE are on the order $\approx 2^\circ$ for the pitch angle θ, which is considered low.

Next, the GOF and TIC are evaluated for the signals. All signals show GOF values above 0.85, and TIC values below 0.15. Compared to the guidelines from [20, 44], these are excellent results. The smallest GOF and TIC values are for prediction of w, which are 0.85 and 0.15, respectively. Although these values are far within limits for a good model, a comment on why the performance is likely to be low for w should be made. From chapter 3, w is measured on the assumption that there is no wind, an assumption that is impossible to fully justify when doing flight tests outside. Because of this, it is not surprising that the prediction of w shows the lowest score. Accurate measurement or estimation of AoA may potentially increase the prediction for w.

Further, by looking at the validation plots in fig. 6-5a, some of the qualitative differences between the Equation-Error method and the Output-Error method become clear. Specifically, the Output-Error method increased the model performance on θ. This highlights the difference between the two methods: the Equation-Error method minimizes the one-step prediction error, while the Output-Error method minimizes the simulation error. As already mentioned, the pitch θ follows a purely kinematic

relationship with the other states, and drift in θ is therefore not penalized by the Equation-Error method. However, it is punished in the Output-Error method, which shows an apparent increase in the pitch prediction.

In summary, residual analysis of the longitudinal model shows good model performance. As expected, the Output-Error yields a model that gives better simulation performance than the Equation-Error method, which is particularly visible in the improved prediction of θ. Although both methods result in models that perform well, the model with the best error metrics is chosen, the Output-Error model. By evaluating the RMSE, MAE, TIC, and RMSE of the final model, it is clear that the model performance is satisfactory.

Residual Analysis of Lateral-Directional Model

Similarly, for the lateral-directional model, the Output-Error method was found to generally increase the model performance. A comparison plot of the different error metrics can be seen in figs. 6-6 and 6-7. As described earlier, the Output-Error method is expected to increase performance, as the Output-Error method accounts for measurement noise and the Equation-Error method does not. Therefore, the Output-Error model is taken as the final lateral-directional model.

First, the error metrics MAE and RMSE are evaluated for the different lateral-directional signals. For the body velocity v, the errors are on the order of $\approx 0.5\mathrm{m\,s^{-1}}$, which is considered low compared to the nominal airspeed of $21\mathrm{m\,s^{-1}}$. Next, the error for the roll rate p is on the order of 5 to $10^\circ\,\mathrm{s^{-1}}$, which is the largest error amongst the angular rates. In the experiments, the roll rate reaches values on the order of $\approx 120^\circ\,\mathrm{s^{-1}}$, and in comparison, the error is not large. However, the model's relatively large error compared to other angular rates may be due to coupled aerodynamic effects resulting from large perturbations from trim, illustrated in fig. 5-2. Next, the error of the yaw rate r is on the order of $\approx 5^\circ\,\mathrm{s^{-1}}$, again a number that is considered low compared to the values reached throughout the experiments. Finally, the roll angle ϕ displays an error of approximately 5°. This is larger than the error for the pitch angle for the longitudinal model. Again, a potential cause for this may be unmodelled

coupled aerodynamics. Given that the error for the roll rate p is larger than errors for the other angular rates, it is expected that the error is also increased for the roll, as the latter follows a purely kinematic relationship with the angular rates. Nevertheless, the error is considered reasonable when compared to the maximum values reached in the maneuvers.

Next, have a look at the model performance metrics TIC and GOF. First, it is clear that all the signals show satisfiable results, with TIC and GOF values above 0.89 and below 0.17, respectively. It is also clear that the Output-Error method increases performance on all the signals, except for the roll rate p, which slightly decreases. Again, this highlights the effect of the Output-Error method compared to the Equation-Error method. The Output-Error method increases the overall simulation performance for the lateral-directional method while sacrificing some of the ability to predict the roll rate p accurately. However, the sacrifice is small compared to the increased performance for the other signals.

A surprising feature of the final model is that it shows a very good prediction of the y-velocity v. Similarly, for w, v is calculated on the assumption that there is no wind. However, this is an assumption that never holds entirely true. In addition, it is known that SSA estimation is challenging for small aircraft [52, 70, 71, 72]. Therefore, it is expected that the accuracy for v is not as high as that for the other signals. As the performance is high, this may indicate that there was, in fact, little horizontal wind at the altitude the experiments are performed at. In more wind, it is expected that prediction accuracy for v will drop. Again, if the SSA was measured directly or estimated in a more sophisticated manner, this will likely contribute to a better prediction of v in windy environments.

In summary, residual analysis of the lateral-directional model shows excellent performance. All errors are within reasonable bounds. Similar to the longitudinal model, the Output-Error method shows a general increase in the performance for the lateral-directional model, compared to the Equation-Error method. The model with the best metrics is selected, which is the model from the Output-Error method. However, there is a potential weakness in the model that should be kept in mind. Although

the model displays good predictions for v, it is expected that this prediction will be reduced during more windy conditions.

6.3.2 Analysis of Static Curves

As a part of the validation process of the final model, the static lift, drag, and pitch moment curves from the model are investigated. As the individual aerodynamic coefficients are recreated in the preprocessing step, comparison with the measured coefficients is possible. By verifying that the static curves match the estimated coefficients during the flight maneuvers, by comparing the obtained curves with theoretical values, and by confirming that signs make sense, the confidence in the model is further increased.

By assuming that the pitch rate is negligible $q \approx 0$, one obtains the following functions for the static curves from the final model used in this work:

$$c_D \approx c_{D0} + c_{D\alpha}\alpha + c_{\alpha^2}\alpha^2 + (c_{D\delta_e} + c_{D\delta_e\alpha}\alpha)\delta_e \tag{6.19a}$$

$$c_L \approx c_{L0} + c_{L\alpha}\alpha + c_{L\alpha^2}\alpha^2 + c_{L\delta_e}\delta_e \tag{6.19b}$$

$$c_m \approx c_{m0} + c_{m\alpha}\alpha + c_{m\delta_e}\delta_e \tag{6.19c}$$

The corresponding three-dimensional lift, drag and pitch moment surfaces can be seen plotted as functions of α and δ_e in fig. 6-11. Contour lines for different values of the elevator δ_e can be seen in figs. 6-12, 6-13 and 6-15, plotted together with lift, drag and moment coefficients as calculated from the measured data.

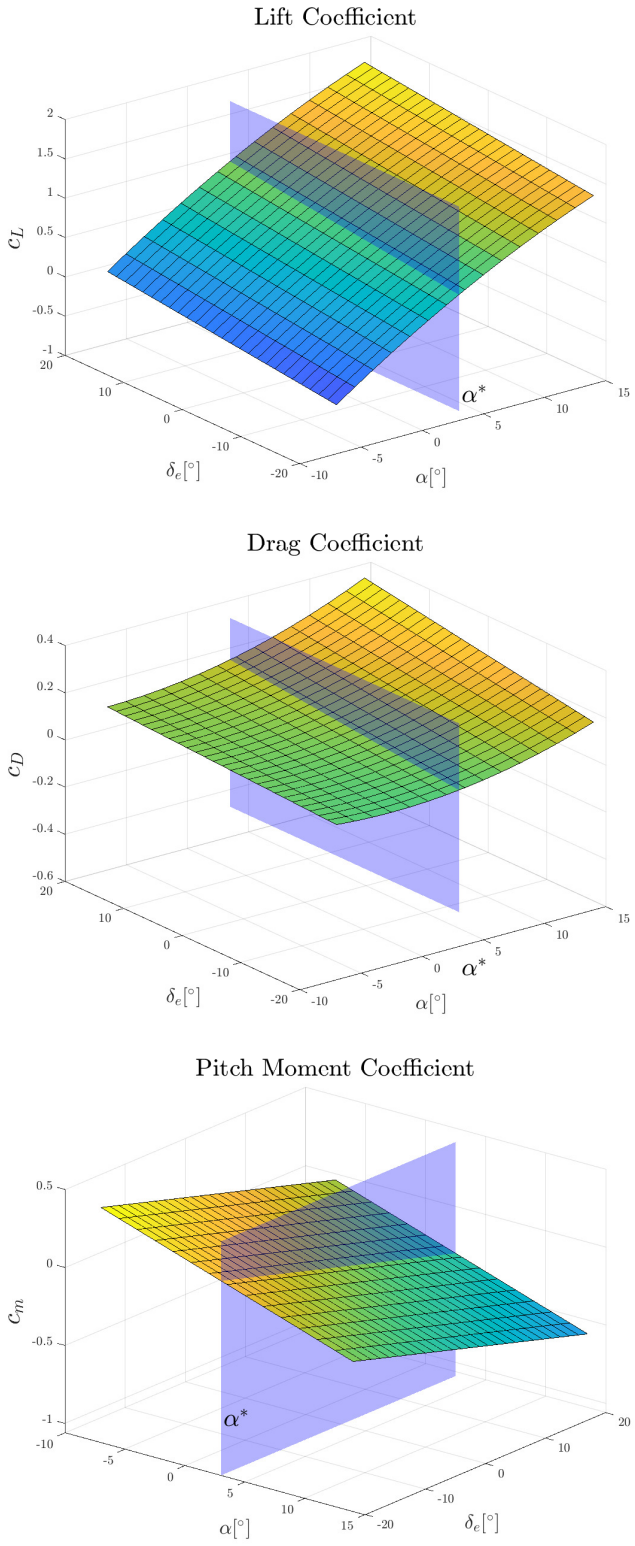


Figure 6-11: Static lift and drag curves plotted as three-dimensional surfaces.

Lift Curve Analysis

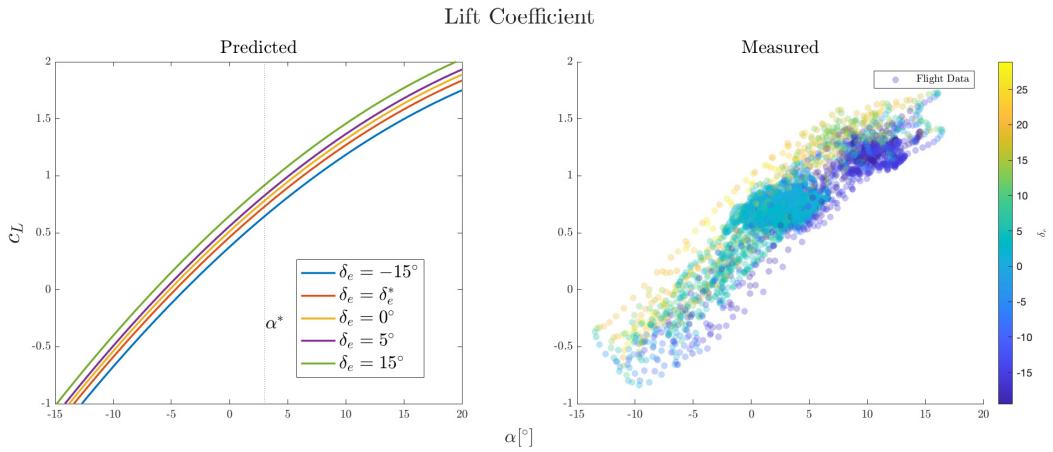


Figure 6-12: Lift curve for different elevator deflections.

First, the lift curves in fig. 6-12 are investigated. The stepwise regression procedure found that a quadratic model of the lift curve best fits the available data. By looking at the contour lines and comparing them with the data, it seems that a good model for the lift coefficient as a function of α and δ_e is found, as it matches the behavior of the observed data well. Close to the trim AoA α^*, the obtained lift model is close to linear. In reality, a linear approximation of lift is usually reasonable for small AoAs. This seems to be the case for the measured data, which is also captured in the resulting lift model.

As the aircraft approaches the stall angle, the lift will start to decrease rapidly [58, 60]. From the measured data, the lift is never seen to decline, but the increase seems to start slowing down around $\alpha \approx 10°$. This decrease in the slope appears to be captured in the quadratic lift model. Further, for most airfoils, the stall angle is usually between 15 to 20°. From the obtained data, it seems reasonable to assume that this aicraft exhibits similar behavior. However, enough data is not captured for the stall regime to accurately capture the stall angle, as a drastic decrease in the lift is not visible in the data points. Therefore, it is expected that the obtained quadratic lift model is invalid for AoAs in the stall regime.

Another property to note for the lift curve in fig. 6-12 is that the lift increases

with positive elevator deflections, that is, downwards deflections, and that the increase seems to match the data points. As pointed out in classical textbooks for aerodynamics, for example, in [58], a downward elevator deflection should generally increase the lift. In the final longitudinal model, $c_{L_{\delta_e}} > 0$, as it should be for all cambered airfoils [58]. Although not a sufficient condition for model validity, this is a necessary condition for model validity and, therefore, a good sanity check.

Finally, for the lift curve, aerodynamics give a theoretical approximations for the aerodynamic coefficient c_{L_α}. As pointed out in [54], an approximation is given as

$$\hat{c}_{L_\alpha} \approx \frac{\pi AR}{1 + \sqrt{1 + \left(\frac{AR}{2}\right)^2}} \tag{6.20}$$

With the measured airframe properties, one obtains $\hat{c}_{L_\alpha} = 5.092$. In the final longitudinal model presented in this work, the value is found to be $c_{L_\alpha} = 5.325$, a value that is very close to the theoretical approximation. Again, although this is not a sufficient condition for model validity, the fact that the obtained value is close to the theoretical approximation further builds confidence in the model.

Drag Curve Analysis

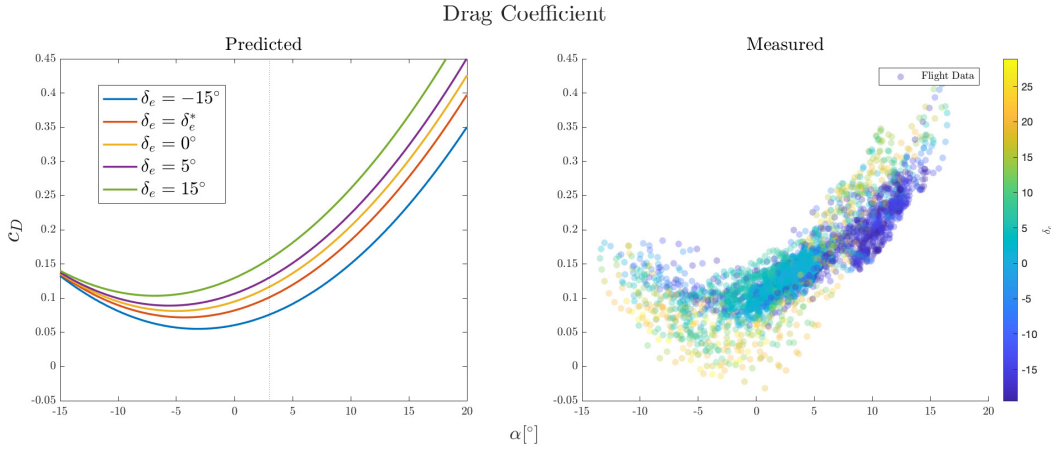


Figure 6-13: Drag curve for different elevator deflections.

Next, the drag curve in fig. 6-13 is investigated. Again, it seems that the obtained model matches the calculated drag coefficients well, especially in terms of dependency

on α. An important sanity check is that the drag should be strictly positive, a property that always holds for all aircraft. From the plot, it is clear that this is the case for the developed aerodynamic model. This is an important property to verify for a mathematical aerodynamic model, especially when it is developed from FTD using optimization, as the optimization scheme can just as well make the drag negative if this fits the data better.

Also, aerodynamics theory gives a theoretical approximation for the drag coefficient as a function of the lift coefficient for conventional fixed-wing aircraft. This theoretical relationship is found in example [8, 41, 54], and is given as

$$c_D = c_{D0} + \frac{c_L^2}{\pi e AR} \tag{6.21}$$

where e is the Oswald efficiency factor, usually in the range between 0.8 and 1.0 [54], and AR is the aspect ratio. By using the final lift model found in this work and setting $e = 0.8$, the theoretical drag curves are calculated and compared to the drag model obtained purely from FTD. A plot of this can be seen in fig. 6-14. The plot shows that the theoretical drag and the obtained drag model are very similar and are of the same magnitudes. The main difference is that the obtained drag model predicts more drag at higher AoA deflections from trim. This seems reasonable, as the aircraft is not a pure fixed-wing aircraft, but has large VTOL arms with propellers that causes drag as they are exposed to the airflow at higher AoAs. This comparison with theoretical values acts as yet another sanity check which further builds confidence in the model.

Further, the drag dependency on δ_e is more challenging to say something quantitative about, as this data is not as clear. At least, in the obtained model, $c_{D_{\delta_e \alpha}} > 0$, indicating that the drag increases when the aircraft has a positive AoA and a downward elevator deflection, and similarly the other way around. This makes intuitive sense, as the elevator is opposing the airflow in both of these situations compared to deflected to align with the airflow.

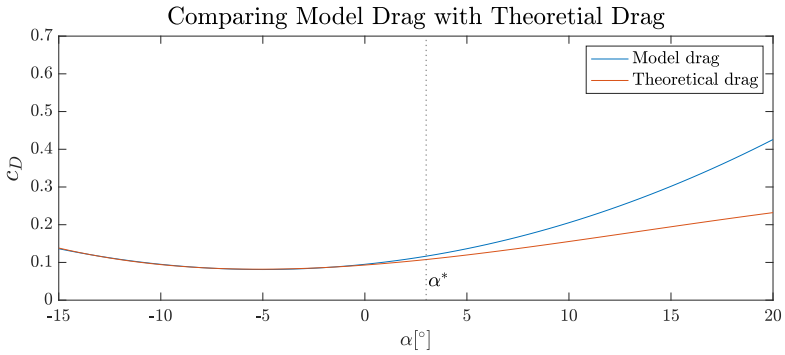


Figure 6-14: Theoretical drag compared with model drag.

Pitch Moment Curve Analysis

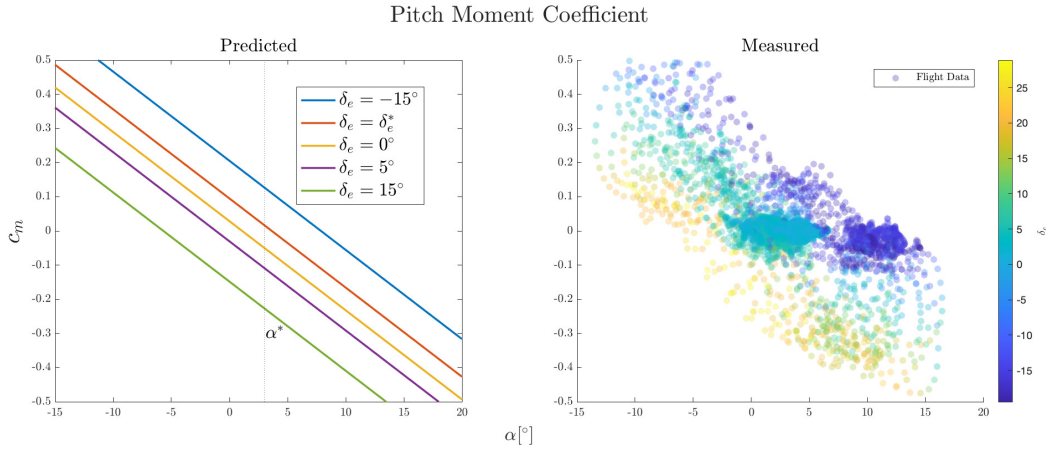


Figure 6-15: Pitch moment curve for different elevator deflections.

Finally, have a look at the pitch moment curve in fig. 6-15. First of all, from the plot, the obtained linear pitch moment model fits the data well. However, for large AoAs, it looks as the pitch moment coefficient starts to become nonlinear, as the slope seems to change for larger AoAs. This nonlinear behavior for large AoAs is expected for most aircraft, as the pitch moment curve, in reality, is a nonlinear function, similar to the real lift and drag. Despite this, the linear approximation seems to approximate the pitch coefficient reasonably well for the tested flight conditions. Another thing to note is the two apparent clusters of data points. As the flight maneuvers are 2-1-1 signals, these clusters correspond to the negative and positive position of the elevator deflections, respectively.

In order for any aircraft to be longitudinally statically stable, $c_{m0} > 0$ and $c_{m\alpha} < 0$ must hold, such that any AoA generates a restoring pitch moment, bringing the aircraft back to trim [54, 58, 60]. This depends on the airfoil's camber, the center of gravity compared to the center of pressure, and chosen trim conditions. As can be seen from the plot, the obtained aerodynamic model shows a longitudinally statically stable aircraft. This is very important for the model, as the aircraft is known to be longitudinally statically stable from manual flight tests.

Further, it is common to use the elevator to trim fixed-wing aircraft. It is used to trim the aircraft such that there is no pitching moment generated at trim conditions, meaning that $c_m(\alpha^*, \delta_e^*) = 0$ in the ideal case. As can be seen from the plot, this is approximately the case for the obtained model. However, to be exact, $c_m(\delta_e = \delta_e^*)$ is actually slightly higher than 0, which may be caused by a number of reasons. For example, there could be small inaccuracies in the pitch model, partly violated assumptions, or the model structure may be too simple. For instance, the fixed-wing propeller may not act directly on the center of mass and therefore cause a slight change in the pitching moment when δ_t is changed. Or, the experiments have small variations in the initial AoA, elevator deflection, and airspeed, changing the trim conditions slightly between maneuvers, making it hard to capture all the pitch coefficient terms accurately. Either way, the pitch moment curve seems to represent a reasonable model for the obtained data.

6.3.3 Analysis of Aerodynamic Parameters

Some simple additional sanity checks may be performed on the obtained model to verify that all the parameters make physical sense. For example, depending on the airframe, the signs of many of the coefficients are known a-priori. In this subsection, the signs and meaning of these parameters are introduced and checked briefly. For more details on the physical significance of these parameters, the reader is referred to [54, 58, 60].

Roll Static Stability

Whether the aircraft will return to $\phi = 0$ after a perturbation or not depends on the roll static stability. For aircraft which are statically stable, $c_{l\beta} < 0$, which will roll the aircraft away from the direction of sideslip, driving it to zero. From manual flight tests, it is known that the aircraft in question is statically roll stable. Correspondingly, the final aerodynamic parameters show that $c_{l\beta} < 0$, indicating that the roll stability is captured in the model.

Yaw Static Stability

If an aircraft is statically stable in yaw, it will always point in the direction of the wind. For aircraft with a tail, the tail will cause a restoring yaw moment when there is sideslip, driving the sideslip to zero [58]. This is reflected in the derivative $c_{n\beta}$, where a positive derivative gives a statically yaw-stable aircraft. Longer or larger tail will usually give a larger positive value for this derivative. The aircraft in question has a tail, and is additionally known to be statically stable in yaw from manual flight tests. For the obtained model $c_{n\beta} > 0$, indicating that the static yaw stability is captured in the model.

Dynamic Stability Derivatives

The dynamic stability derivatives are c_{mq}, c_{lp}, and c_{nr}. These are usually negative, as they function as damping coefficients for angular motion of the airframe [54, 60]. From table 6.4, all of the dynamic stability derivatives are negative, indicating normal second-order damping in for all the moments, which is common for most aircraft.

Control Derivatives

The sign of the control derivatives for the different moments are well known from intuition and physics. Wrong signs on the control derivatives would indicate a bad model. Following the aerodynamic model, all of the control derivatives in the final model have the correct signs, $c_{m\delta_e} < 0$, $c_{l\delta_a} > 0$, and $c_{n\delta_r} < 0$.

6.3.4 Eigenmode Analysis Around Trim

Finally, the nonlinear model is linearized around trim conditions as described in section 3.4, with the trim conditions as shown in table 4.5. The linearized model was shown in eqs. (6.17) and (6.18). A plot of the zeros of the system can be seen in fig. 6-16, and the table analyzing the characteristics of the mode can be seen in table 6.7. As stated in section 5.2, neither the phugoid nor the spiral mode is likely to be correct for the model, as the experiments do not excite or capture these modes as described earlier. However, for most applications, this is not a problem due to feedback control that tends to eliminate the effect of these modes [44].

Mode	Eigenvalue	ζ	f[cycles/s]	T_c[s]
Short-Period	-3.280 ± 7.790	0.388	1.345	0.305
Phugoid	$-0.0673 \pm 0.657i$	0.102	0.105	14.900
Roll	-8.82	$--$	$--$	0.113
Dutch roll	$-0.942 + 4.940i$	0.187	0.801	1.060
Spiral	0.116	$--$	$--$	-8.640

Table 6.7: Eigenmode analysis of the linearized longitudinal and lateral-directional model.

First, the longitudinal modes are as expected. They consist of a lightly damped oscillatory mode corresponding to the phugoid mode and another oscillatory mode corresponding to the short-period mode. The short-period mode has a time constant of $\approx$ 300ms, with damping of 0.393. This is very similar to the results presented from VLM in table 4.6 and is reasonable for an aircraft of the current size. The phugoid mode in the final model exhibits stronger damping and a shorter time constant than the one obtained from VLM; however, as noted, the phugoid mode is likely to be inaccurate for the obtained model.

Next, the lateral-directional model shows three modes: the roll mode, the Dutch roll mode, and the spiral mode. The roll mode is purely exponential and displays a time constant of $\approx$ 100ms, something that is reasonable and expected for an aircraft of the current size. Next, the Dutch roll mode is lightly damped, with a time-constant on the order of $\approx$ 1s. This is also as expected. Compared with the VLM analysis, the obtained model exhibits similar characteristics. However, the roll mode is slightly

slower in the final model, and the Dutch-roll mode is more damped. Finally, the spiral mode is found to be somewhat positive, with a similar magnitude as in VLM. As mentioned several times already, the spiral mode is likely to be inaccurate for the final model due to the experiments being too short to capture this mode.

In conclusion, the linearized model shows expected behavior for an aircraft of the given size. In addition, the eigenmode analysis of the final model resembles the results obtained from the purely numerical in section 4.5. The confidence is low in the two slowest modes, the phugoid, and the spiral mode. However, this is not an issue, as slow modes such as the spiral mode and the phugoid do not typically impact the model applications, and they are usually eliminated through feedback from the control systems.

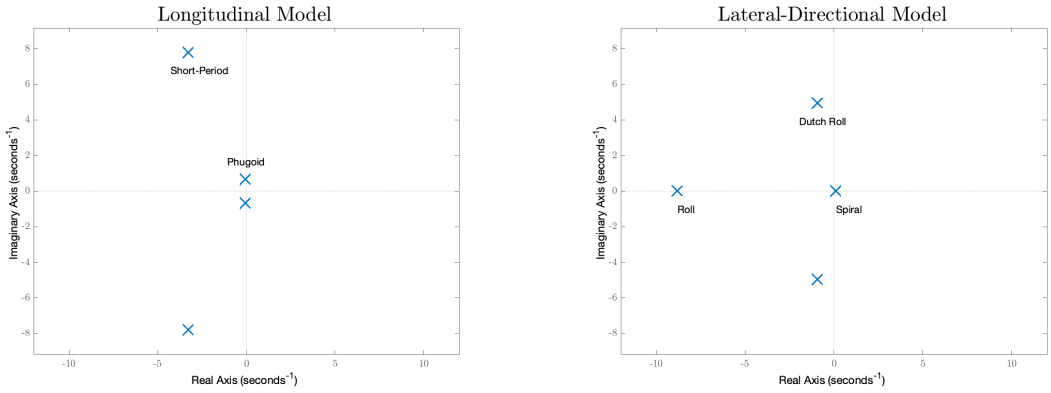


Figure 6-16: Pole plot for the linearized longitudinal and lateral-directional model.

6.3.5 Aerodynamic Model Structure

It seems that the final aerodynamic model structure is as simple as possible while explaining the obtained data well. As the aircraft is operating around trim conditions, it is expected that a linear aerodynamic model should capture the general behavior of the aircraft. The obtained model is almost linear, with a few nonlinear dependencies. These are all in the drag or in the lift model. In particular, the drag and lift are nonlinear in AoA, where the lift and induced drag is a function of the squared AoA, and the drag from the elevators increase with the AoA. These phenomena match aero-

dynamic theory, and as the maneuvers are relatively large perturbations performed around trim conditions, the nonlinearities seem to be well justified.

Through a purely data-driven approach, it is easy to fit the data well if one does not care about the number of regressors [19, 39, 40]. Therefore, the engineer must pay careful attention when developing the model. A simple model is always preferred, which is why the stepwise regression procedure in this work is only allowed to include a regressor if it significantly increases the prediction of the data and is removed or discarded otherwise. Still, intuition and knowledge of physics must be used when picking the regressors to avoid structural overfitting or the inclusion of regressors that do not make physical sense.

An excellent example of structural overfitting was found when it was attempted to add the squared elevator signal to the pool of potential regressors. In fact, the squared elevator signal significantly improved how well the models predict the pitch moment coefficient c_m. However, consider that due to safety reasons, all the pitching maneuvers are 2-1-1 elevator deflection signals that start with the aircraft pointing up. In addition, consider the physical meaning of the pitch moment to be a function of the squared elevator signal: the direction of the pitch moment is independent of the direction of the elevator deflection. This does not make physical sense. Therefore, it is concluded that such a functional relationship does not capture the actual dynamics of the aircraft, despite fitting the data well. It is simply a weakness in the gathered data that causes this regressor to predict c_m better. In addition, from [19], it is known that aircraft coefficients follow a linear relationship with the angular rates and control surfaces and that the nonlinear dependencies usually only occur in aerodynamic angles. This example illustrates why it is critical that the engineer monitors the selection process and carefully chooses which regressors to include in the pool of potential regressors.

Chapter 7

Future Work and Conclusion

In this final chapter, future work and conclusions are presented. First, promising directions for future work based on this thesis are presented, both for further expansion of the model and potential applications. Next, the work in this thesis is summarized, and conclusions are drawn.

7.1 Future Work

This work proposes the first dynamic model for the Foxtech Babyshark 260 VTOL. There are multiple promising directions for future work, both in terms of further modeling and model applications.

7.1.1 Expanding the Model

Linear Actuator Models

A natural extension of the model is to make a completely linear model of the fixed-wing regime that includes the control surfaces dynamics. Although a linear model is already proposed for the fixed-wing regime, it still uses the nonlinear model for the control surface dynamics. While the servos are known to be rate-limited in reality, this is not well suited for control design. Therefore, it may be desirable to develop a linear model for the servos depending on the intended application for the model.

In similar work, a simple second-order mass-spring-damper model is employed with success, such as in [8].

Near Stall Dynamics

Next, the model can be expanded to model close-to-stall behavior. It is well known that flow becomes unsteady for these regimes, which often requires nonlinear modeling [19, 49, 50, 54]. In this work, a quadratic lift model close to being linear gives the best fit on the obtained data. Although this seems to describe the behavior of the data obtained from the experiments, it is clear that the experiments do not capture data from the stall regime. For aggressive flight maneuvers, flight at larger AoAs may be desirable. Hence, the model will need to be expanded with nonlinear terms that capture this flight regime's complex dynamics. In that case, additional data should be gathered for these flight regimes, such that the stall angle of the aircraft, as well as stall behavior, may be identified. Using a combination between a linear and a flat-plate model such as proposed in [54] is expected to work well and can additionally be augmented with local function approximators such as in [50]. Radial basis functions, as proposed in [51] may be a good candidate basis function.

Multirotor Flight and Different Flight Envelopes

Although this work models the multirotor part of the system, flight in this regime should be verified on real data. As was mentioned in chapter 2, multirotor modeling usually deals with identifying the thrust and moment coefficients, as well as the inertial properties of the aircraft. Hence, it is expected that the multirotor model proposed in this work will work well during low speeds. However, this should be validated on data. In addition, there may be unmodelled coupling between the multirotor actuators and fixed-wing actuators if both actuators are used at high speed.

In addition, if the aircraft is to be flown at different flight envelopes, the model should be verified and expanded for these envelopes. Other flight envelopes could mean significantly different Mach numbers or altitudes. This is important before the model is used for flight conditions far from the trim conditions presented in this

thesis or where modifications to the aircraft mass or center of gravity are made. As the aircraft is flying at relatively low airspeeds, the generated propeller thrust may, for example, be dependent on the aircraft's airspeed, and the propeller model may have to be augmented. For this, using a lookup table generated from CFD such as in [36], or a more complex model using the advance ratio such as the ones from [8, 54] may be desirable.

Expanding the Model to Wind

Finally, the model assumes that there is no wind and that the aerodynamic angles AoA and SSA may be estimated from the body velocities. When there is wind, the aerodynamic angles cannot be estimated in this way, and if they are, the wind will start to introduce errors. This means that when there is a lot of wind, the model performance will likely begin to degrade.

Suppose it is expected that the aircraft will fly in significant wind, and the model is found to perform unsatisfactorily under such circumstances. In that case, this can be solved through estimation or measurement of the aerodynamic angles. In [36], comprehensive coverage of how an air probe capable of estimating the AoA and SSA may be built from cheap and readily available components. In addition, there are promising methods for estimating the aerodynamic angles without using additional sensors. For example, [52] presents an estimation framework for the estimation of aerodynamic angles from a sensor suite similar to the one on the Babyshark VTOL. Further, in [72], a complete estimation algorithm is proposed, where the AoA and SSA may be estimated through the inclusion of an aerodynamic model.

7.1.2 Model Applications

Improved Transition from Multirotor to Fixed-Wing Flight

The transition phase for VTOL aircraft is complex. When going from multirotor flight to fixed-wing flight, the vehicle will cover a large flight envelope, ranging from zero airspeed to trim conditions. In addition, the optimal combined use of multirotor

actuators and the fixed-wing actuators during this flight phase is not trivial. There are several works that deal with this for tailsitter VTOL UAVs [7, 9] and tiltrotor VTOL UAVs [11]. Relevant work on optimal transitioning for hybrid VTOL designs has not been found by the author. Through the use of a model-based approach, it may be possible to better optimize the transition phase of flight.

Currently, the control method for transition imposes limitations on the use of the aircraft. During this transition, the control system in PX4 interpolates between two separate control systems, one for the multirotor actuators and one for the fixed-wing actuators [28]. However, due to the complexities associated with the transition phase, this simple approach has drawbacks. As the current method does not account for the aircraft dynamics, the aircraft usually loses height during the transition and often starts to drift in either direction. Because of this, there is uncertainty related to the takeoff phase, and takeoffs require an unnecessarily large, open space. With a more precise transition, takeoffs would be possible in much smaller areas.

In order to optimize the transition, the first step is to generate dynamically feasible transition trajectories. One may use a model-based trajectory optimization algorithm to develop a full-state trajectory that minimizes height loss and drift to either direction. An excellent candidate algorithm for this is Direct Collocation, as first presented in [31], which was also mentioned at the beginning of this thesis. By using this method, multiple transition trajectories may be generated a-priori to develop a library of transition trajectories for different cases.

Next, the trajectory may be stabilized through the use of a nonlinear, model-based controller. One may, for example, stabilize it with a method as proposed in [9]. Here, a controller consisting of a feed-forward and a feedback term is proposed. The pre-computed trajectories are used as feed-forward inputs, and a simple feedback term with a proportional gain is used to account for minor tracking errors. Another approach is to employ a more sophisticated controller to stabilize the trajectory, such as Time-Varying Linear Quadratic Regulator (TVLQR).

Unified Attitude Controller

With the model, one may also develop a single attitude controller for both the multirotor and the fixed-wing flight regimes. As mentioned briefly, the current PX4 implementation for VTOL aircraft used two completely separate control systems for the control of multirotor actuators and fixed-wing actuators. In order to improve performance during both regimes and in-between, it is desirable to have only a single attitude controller. In order to achieve this, one may, for example, employ a geometric controller that calculates input torques and forces, similar to the ones presented in [7] or [17]. The control law may be very general, as it only operates with forces or moments. Then, the model can be used to calculate the input signals required to achieve the commanded forces and torques.

Another solution to improve aircraft performance during the transition and generally may be to implement Model Predictive Control (MPC) for control of the aircraft. This could provide better tracking while increasing system robustness to disturbances. Depending on the implementation, this may require either a linear or a nonlinear model. Examples and inspiration for this can be found in [24, 25].

Fault Detection and Handling

Different and promising use of the model is to develop a fault detection scheme. With the obtained model, it is now possible to predict the expected behavior of the aircraft. If the observed behavior starts deviating from the predicted behavior, something has likely changed in the system, which may be caused by a fault. The aircraft may either correct its behavior through adaptive control based on the model or initiate emergency procedures. Such a fault detection and handling system is highly desirable in the case of, for example, icing on the wings, which is known to decrease the lift generated by the wings or propellers rapidly.

Simulation

Finally, relevant for all of the above applications is that the obtained model allows for much more accurate simulation. For example, the model may expand the PX4 simulator with a more accurate physics simulation for the aircraft, which will simulate both the dynamics and all the software running on the aircraft. Alternatively, if the focus is purely on testing different methods and ideas, it may be more desirable to simulate the dynamic model on its own together with a standard integrator scheme. Such an implementation is provided at [1].

7.2 Conclusions

In this thesis, a novel dynamic model of the Vertical-TakeOff-and-Landing (VTOL) Unmanned Aerial Vehicle (UAV) named FoxTech Babyshark 260 VTOL is developed from Flight-Test Data (FTD), presented, and validated on real experimental data.

The final model is the result of multiple system identification steps. First, a preliminary analysis is performed on the aircraft and its actuators to identify all airframe parameters that can be obtained without flight tests. Static airframe properties such as inertia are found and estimated through measurements and accurate 3D modeling of the aircraft, and actuators models are proposed, where the propellers are modeled as static models and control surfaces are modeled with their own dynamics. All corresponding model parameters are found through actuator response measurements with different equipment, such as cameras and force sensors. Finally, trim conditions are identified through preliminary manual flight tests of the aircraft, and a baseline model of the aerodynamic model is obtained from numerical analysis in Vortex Lattice Method (VLM).

Next, actual flight experiments are performed to obtain Flight-Test Data (FTD). Through the combined use of an experienced pilot and preprogrammed maneuvers, 2-1-1 deflection signals with varying periods and amplitudes are input to the different control surfaces, designed to excite the relevant modes of the system. From the obtained data, all the required signals that will be used for identification are calcu-

lated. Finally, the signals are retrieved through an elaborate process of filtering and analytical derivation using piecewise splines, and Flight Path Reconstruction (FPR) is used to arrive at all the states and aerodynamic coefficients.

As the first step in the aerodynamic model development, the Equation-Error method is employed. Due to the data pre-processing steps, estimates of all of the aerodynamic coefficients during flight tests are available. This makes it possible to use Ordinary Least-Squares (OLS) to investigate different model structures rapidly. In order to determine the best-suited structure, a data-driven approach is taken to the model determination. A variation of stepwise regression is implemented and fully automated, resulting in an algorithm that automatically selects the best model structure from a pool of regressor candidates. Care is taken in the selection of the regressor pool in order to prevent structural overfitting of the data, which is a concern with fully automated procedures.

For arrival upon a final aerodynamic model, the Output-Error method is used to estimate the unknown parameters. The model determined by the automatic stepwise regression algorithm is used, and parameters are found through a nonlinear optimization scheme called the Gauss-Newton step. In order to improve the algorithm's convergence properties, a line search is added at every step of the procedure. The parameters from the Equation-Error method are used as initial guesses for the Output-Error method. Both the parameters from the Equation-Error method and the Output-Error method show good results on the validation data. Finally, the model is augmented with an additional term to capture rudder-pitch coupling caused by the V-tail configuration of the aircraft.

The entire model is presented in section 6.2. The best model is provided by the parameters from the Output-Error method. The result is a full, nonlinear model which accurately predicts both longitudinal and lateral-directional motion, as well as coupled motion during the maneuvers. The model parameters are discussed and evaluated through residual analysis, and analysis of static curves and comparison with measured aerodynamic coefficients is performed. Individual parameter are analyzed, and an eigenmode analysis performed around trim conditions.

In summary, the result is a nonlinear dynamic model which shows good performance. Validation on data not used for training shows excellent predictions of all aircraft states. In addition, the analysis of the model properties further builds confidence in the model. An open-source, ready-to-use implementation of the model is made available at [1]. The author hopes that the development of an accurate, dynamic model for the Babyshark 260 VTOL will contribute to research on VTOL UAVs in the future.

Bibliography

[1] Bernhard Paus Graesdal. *Dynamic Model of the Foxtech Babyshark 260 VTOL UAV*. URL: https://github.com/bernhardpg/babyshark_vtol_model.

[2] J. Senthilnath et al. "Application of UAV imaging platform for vegetation analysis based on spectral-spatial methods". In: *Computers and Electronics in Agriculture* 140 (Aug. 2017), pp. 8–24. ISSN: 01681699. DOI: 10.1016/j.compag.2017.05.027.

[3] Hesam Hamledari, Brenda McCabe, and Shakiba Davari. "Automated computer vision-based detection of components of under-construction indoor partitions". In: *Automation in Construction* 74 (Feb. 2017), pp. 78–94. ISSN: 09265805. DOI: 10.1016/j.autcon.2016.11.009.

[4] Francesco Nex and Fabio Remondino. *UAV for 3D mapping applications: A review*. 2014. DOI: 10.1007/s12518-013-0120-x.

[5] Philipp Oettershagen et al. "Robotic technologies for solar-powered UAVs: Fully autonomous updraft-aware aerial sensing for multiday search-and-rescue missions". In: *Journal of Field Robotics* 35.4 (2018), pp. 612–640. ISSN: 15564967. DOI: 10.1002/rob.21765.

[6] Jongseok Lee et al. "Towards Autonomous Stratospheric Flight: A Generic Global System Identification Framework for Fixed-Wing Platforms". In: *IEEE International Conference on Intelligent Robots and Systems* October (2018), pp. 6233–6240. ISSN: 21530866. DOI: 10.1109/IROS.2018.8594126.

[7] S. Verling et al. "Full Attitude Control of a VTOL tailsitter UAV". In: *Proceedings - IEEE International Conference on Robotics and Automation* 2016-June (2016), pp. 3006–3012. ISSN: 10504729. DOI: 10.1109/ICRA.2016.7487466.

[8] Carl Olsson et al. "Full Envelope System Identification of a VTOL Tailsitter UAV". In: (2021), pp. 1–14. DOI: 10.2514/6.2021-1054.

[9] Sebastian Verling et al. "Model-based transition optimization for a VTOL tailsitter". In: *Proceedings - IEEE International Conference on Robotics and Automation* (2017), pp. 3939–3944. ISSN: 10504729. DOI: 10.1109/ICRA.2017.7989454.

[10] Y. Demitrit et al. "Model-based wind estimation for a hovering VTOL tailsitter UAV". In: *Proceedings - IEEE International Conference on Robotics and Automation* (2017), pp. 3945–3952. ISSN: 10504729. DOI: 10.1109/ICRA.2017.7989455.

[11] David Rohr et al. “Attitude- and cruise control of a VTOL tiltwing UAV”. In: *arXiv* 4.3 (2019), pp. 2683–2690. ISSN: 23318422.

[12] FoxTech. *Foxtech Babyshark 260 VTOL*. URL: https://www.foxtechfpv.com/foxtech-baby-shark-vtol.html.

[13] Samir Bouabdallah and Roland Siegwart. “Full control of a quadrotor”. In: *IEEE International Conference on Intelligent Robots and Systems* October (2007), pp. 153–158. ISSN: 18172172. DOI: 10.1109/IROS.2007.4399042.

[14] J. Carino, H. Abaunza, and P. Castillo. “Quadrotor quaternion control”. In: *2015 International Conference on Unmanned Aircraft Systems, ICUAS 2015* June (2015), pp. 825–831. DOI: 10.1109/ICUAS.2015.7152367.

[15] Randal W. Beard. “Quadrotor Dynamics and Control”. In: *Brigham Young University* (2008), pp. 1–47. URL: http://image.ednchina.com/GROUP/uploadfile/201304/20130429210226589.pdf.

[16] Samir Bouabdallah, André Noth, and Roland Siegwart. “PID vs LQ control techniques applied to an indoor micro Quadrotor”. In: *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* 3.January (2004), pp. 2451–2456. DOI: 10.1109/iros.2004.1389776.

[17] Taeyoung Lee, Melvin Leok, and N. Harris McClamroch. “Geometric tracking control of a quadrotor UAV on SE(3)”. In: *Proceedings of the IEEE Conference on Decision and Control* (2010), pp. 5420–5425. ISSN: 01912216. DOI: 10.1109/CDC.2010.5717652.

[18] Daniel Mellinger and Vijay Kumar. “Minimum snap trajectory generation and control for quadrotors”. In: 2011. ISBN: 9781612843865. DOI: 10.1109/ICRA.2011.5980409.

[19] Vladislav Klein and Eugene A. Morelli. *Aircraft System Identification: Theory and Practice*. 2006. DOI: 10.2514/4.861505.

[20] Ravindra V. Jategaonkar. *Flight Vehicle System Identification: A Time Domain Methodology*. 2006. ISBN: 1563478366.

[21] Mark B. Tischler and Robert K. Remple. *Aircraft and Rotorcraft System Identification, Second Edition*. 2012. DOI: 10.2514/4.868207.

[22] Nathan V. Hoffer et al. “A survey and categorization of small low-cost unmanned aerial vehicle system identification”. In: *Journal of Intelligent and Robotic Systems: Theory and Applications* 74.1-2 (2014), pp. 129–145. ISSN: 09210296. DOI: 10.1007/s10846-013-9931-6.

[23] Mina Kamel et al. “Model predictive control for trajectory tracking of unmanned aerial vehicles using robot operating system”. In: *Studies in Computational Intelligence*. Vol. 707. Springer Verlag, May 2017, pp. 3–39. DOI: 10.1007/978-3-319-54927-9_1.

[24] Thomas Stastny and Roland Siegwart. "Nonlinear Model Predictive Guidance for Fixed-wing UAVs Using Identified Control Augmented Dynamics". In: *2018 International Conference on Unmanned Aircraft Systems, ICUAS 2018* (2018), pp. 432–442. DOI: 10.1109/ICUAS.2018.8453377. arXiv: 1802.02624.

[25] Gonzalo Andres Garcia, Shawn Shahriar Keshmiri, and Thomas Stastny. "Robust and Adaptive Nonlinear Model Predictive Controller for Unsteady and Highly Nonlinear Unmanned Aircraft". In: *IEEE Transactions on Control Systems Technology* 23.4 (2015), pp. 1620–1627. ISSN: 10636536. DOI: 10.1109/TCST.2014.2377711.

[26] J.J.E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice-Hall International Editions. Prentice-Hall, 1991. ISBN: 9780130400499. URL: https://books.google.no/books?id=HddxQgAACAAJ.

[27] K. S. Narendra and A. M. Annaswamy. *Stable Adaptive Systems*. USA: Prentice-Hall, Inc., 1989. ISBN: 0138399948.

[28] PX4 Developer Team. *PX4: Open-Source Autopilot*. URL: https://px4.io/ (visited on 09/11/2021).

[29] Lorenz Meier, Dominik Honegger, and Marc Pollefeys. "PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms". In: vol. 2015-June. 2015. DOI: 10.1109/ICRA.2015.7140074.

[30] Sanghyuk Park, John Deyst, and Jonathan P. How. "A new nonlinear guidance logic for trajectory tracking". In: *Collection of Technical Papers - AIAA Guidance, Navigation, and Control Conference*. Vol. 2. 2004, pp. 941–956. ISBN: 1563476703. DOI: 10.2514/6.2004-4900.

[31] C. Hargraves and Stephen Paris. "Direct Trajectory Optimization Using Nonlinear Programming and Collocation". In: *AIAA J. Guidance* 10 (July 1987), pp. 338–342. DOI: 10.2514/3.20223.

[32] Bernhard Paus Graesdal. *VTOL System Identification Github Repo*. URL: https://github.com/bernhardpg/vtol-system-identification.

[33] Peter G. Hamel and Ravindra V. Jategaonkar. "Evolution of flight vehicle system identification". In: *Journal of Aircraft* 33.1 (1996), pp. 9–28. ISSN: 15333868. DOI: 10.2514/3.46898.

[34] Q.P. Chu, J.A. Mulder, and J.K. Sridhar. "Decomposition of Aircraft State and Parameter Estimation Problems". In: *IFAC Proceedings Volumes* 27.8 (1994). ISSN: 14746670. DOI: 10.1016/s1474-6670(17)47843-8.

[35] J. A. Mulder, J.K. Sridhar, and J.H. Breeman. *Identification of Dynamic Systems-Applications to Aircraft Part 2: Nonlinear Analysis and Manoeuvre Design*. Vol. 3. May 1994. 1994.

[36] Ony Arifianto and Mazen Farhood. "Development and Modeling of a Low-Cost Unmanned Aerial Vehicle Research Platform". In: *Journal of Intelligent and Robotic Systems: Theory and Applications* 80.1 (Oct. 2015), pp. 139–164. ISSN: 15730409. DOI: 10.1007/s10846-014-0145-3.

[37] MW Green. *Measurement of the moments of inertia of full scale airplanes*. Tech. rep. NACA-TN-265. 1927.

[38] David J. Grymin and Mazen Farhood. "Two-Step System Identification and Trajectory Tracking Control of a Small Fixed-Wing UAV". In: *Journal of Intelligent and Robotic Systems: Theory and Applications* 83.1 (2016). ISSN: 15730409. DOI: 10.1007/s10846-015-0298-8.

[39] Benjamin M. Simmons, Hunter G. McClelland, and Craig A. Woolsey. "Nonlinear model identification methodology for small, fixed-wing, unmanned aircraft". In: *Journal of Aircraft* 56.3 (2019), pp. 1056–1067. ISSN: 15333868. DOI: 10.2514/1.C035160.

[40] Benjamin M Simmons et al. "System Identification of a Nonlinear Flight Dynamics Model for a System Identification of a Nonlinear Flight Dynamics Model for a Small , Fixed-Wing UAV". In: (2018).

[41] Lawrence E. Hale, Mayuresh Patil, and Christopher J. Roy. "Aerodynamic parameter identification and uncertainty quantification for small unmanned aircraft". In: *Journal of Guidance, Control, and Dynamics*. Vol. 40. 3. 2017. DOI: 10.2514/1.G000582.

[42] Lorenz Meier et al. "PIXHAWK: A system for autonomous flight using onboard computer vision". In: 2011. DOI: 10.1109/ICRA.2011.5980229.

[43] Gonzalo A. Garcia and Shahriar Keshmiri. "Online artificial neural network model-based nonlinear model predictive controller for the meridian UAS". In: *International Journal of Robust and Nonlinear Control* 23 (15 2013). ISSN: 10498923. DOI: 10.1002/rnc.3037.

[44] Andrei Dorobantu et al. "System identification for small, low-cost, fixed-wing unmanned aircraft". In: *Journal of Aircraft* 50.4 (2013), pp. 1117–1130. ISSN: 15333868. DOI: 10.2514/1.C032065.

[45] Andrei Dorobantu et al. "Frequency domain system identification for a small, low-cost, fixed-wing UAV". In: *AIAA Guidance, Navigation, and Control Conference 2011* (2011), pp. 1–13. DOI: 10.2514/6.2011-6719.

[46] Raghu Venkataraman and Peter Seiler. "System identification for a small, rudderless, fixed-wing unmanned aircraft". In: *Journal of Aircraft* 56.3 (2019), pp. 1126–1134. ISSN: 15333868. DOI: 10.2514/1.C035141.

[47] Harold Youngren Mark Drela. *Athena Vortex Lattice*. URL: http://web.mit.edu/drela/Public/web/avl/.

[48] Michael V. Cook. *Flight Dynamics Principles*. 2013. DOI: 10.1016/C2010-0-65889-5.

[49] Warren Hoburg and Russ Tedrake. "System identification of post stall aerodynamics for UAV perching". In: *AIAA Infotech at Aerospace Conference and Exhibit and AIAA Unmanned...Unlimited Conference*. 2009. DOI: 10.2514/6.2009-1930.

[50] Joseph Moore, Rick Cory, and Russ Tedrake. "Robust post-stall perching with a simple fixed-wing glider using LQR-Trees". In: *Bioinspiration and Biomimetics* 9.2 (2014). ISSN: 17483190. DOI: 10.1088/1748-3182/9/2/025013.

[51] Robert M. Sanner and Jean Jacques E. Slotine. "Gaussian Networks for Direct Adaptive Control". In: *IEEE Transactions on Neural Networks* 3 (6 1992). ISSN: 19410093. DOI: 10.1109/72.165588.

[52] Tor A. Johansen et al. "On estimation of wind velocity, angle-of-attack and sideslip angle of small UAVs using standard sensors". In: *2015 International Conference on Unmanned Aircraft Systems, ICUAS 2015* (2015), pp. 510–519. DOI: 10.1109/ICUAS.2015.7152330.

[53] Andreas Wenz, Tor Arne Johansen, and Andrea Cristofaro. "Combining model-free and model-based angle of attack estimation for small fixed-wing UAVs using a standard sensor suite". In: *2016 International Conference on Unmanned Aircraft Systems, ICUAS 2016* 1 (2016), pp. 624–632. DOI: 10.1109/ICUAS.2016.7502583.

[54] Randal W. Beard and Timothy W. McLain. *Small unmanned aircraft: Theory and practice*. 2012.

[55] Dinu Mihailescu-Stoica, Raul Acuna, and Jurgen Adamy. "High performance adaptive attitude control of a quadrotor". In: *2019 18th European Control Conference, ECC 2019* June (2019), pp. 3462–3469. DOI: 10.23919/ECC.2019.8796203.

[56] Emil Fresk and George Nikolakopoulos. "Full quaternion based attitude control for a quadrotor". In: *2013 European Control Conference, ECC 2013* (2013), pp. 3864–3869. DOI: 10.23919/ecc.2013.6669617.

[57] Kristofer Gryte et al. "Aerodynamic modeling of the Skywalker X8 Fixed-Wing Unmanned Aerial Vehicle". In: *2018 International Conference on Unmanned Aircraft Systems, ICUAS 2018* (2018), pp. 826–835. DOI: 10.1109/ICUAS.2018.8453370.

[58] John D Anderson. *Introduction to Flight, 3rd Edition*. 1989. ISBN: 0070016410.

[59] Tyto Robotics. *RCbenchmark Series 1585 Test Stand*. URL: https://www.tytorobotics.com/products/thrust-stand-series-1580 (visited on 09/11/2021).

[60] Richard M. Howard. "Dynamics of Flight: Stability and Control; Third Edition". In: *Journal of Guidance, Control, and Dynamics* 20.4 (1997), pp. 839–840. ISSN: 0731-5090. DOI: 10.2514/2.4126.

[61] Seyed Amin Bagherzadeh. "Nonlinear aircraft system identification using artificial neural networks enhanced by empirical mode decomposition". In: *Aerospace Science and Technology* 75 (2018). ISSN: 12709638. DOI: 10.1016/j.ast.2018.01.004.

[62] Tor Aleksander Johansen et al. "Nonlinear system identification of fixed wing UAV aerodynamics". In: *Proceedings of the IASTED International Conference on Modelling, Identification and Control*. Vol. 830. Acta Press, 2016, pp. 44–51. ISBN: 9780889869790. DOI: 10.2316/P.2016.830-026.

[63] Eugene A. Morelli. "Low-order equivalent system identification for the Tu-144LL supersonic transport aircraft". In: *Journal of Guidance, Control, and Dynamics* 26 (2 2003). ISSN: 15333884. DOI: 10.2514/2.5053.

[64] PX4 Developer Team. *PX4: Using the ECL EKF*. URL: https://docs.px4.io/master/en/advanced_config/tuning_the_ecl_ekf.html (visited on 09/11/2021).

[65] John D'Errico. *SLM - Shape Language Modeling*. URL: https://se.mathworks.com/matlabcentral/fileexchange/24443-slm-shape-language-modeling.

[66] M. A. Efroymson. "Multiple Regression Analysis". In: *Mathematical Methods for Digital Computer*. 1960.

[67] Jared A. Grauer and Eugene A. Morelli. "Generic global aerodynamic model for aircraft". In: *Journal of Aircraft*. Vol. 52. 1. 2015. DOI: 10.2514/1.C032888.

[68] Adrian Arus. "Development of a System Identification Tool for Subscale Flight Testing". In: (2019). URL: http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-157292.

[69] Joao Oliveira et al. "Output error method and two step method for aerodynamic model identification". In: *Collection of Technical Papers - AIAA Guidance, Navigation, and Control Conference*. Vol. 8. 2005. DOI: 10.2514/6.2005-6440.

[70] B. C. Chen and F. C. Hsieh. "Sideslip angle estimation using extended Kalman filter". In: vol. 46. 2008. DOI: 10.1080/00423110801958550.

[71] John Perry et al. "Estimating angle of attack and sideslip under high dynamics on small UAVs". In: vol. 1. 2008.

[72] Stefan Leutenegger et al. "Robust state estimation for small unmanned airplanes". In: 2014. DOI: 10.1109/CCA.2014.6981466.

Bernhard Paus Græsdal

Full Nonlinear System Identification for a Vertical-Takeoff-and-Landing Unmanned Aerial Vehicle

NTNU
Norwegian University of
Science and Technology