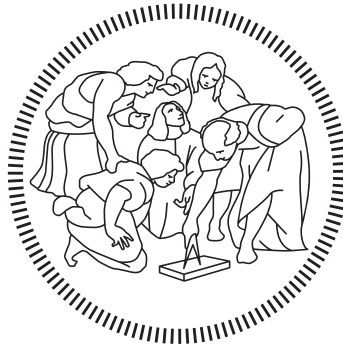


POLITECNICO DI MILANO
Scuola di Ingegneria Industriale e dell'Informazione
Corso di Laurea Magistrale in Ingegneria Aeronautica



Modelling and integration of an eVTOL UAV

Advisor: Prof. Marco LOVERA
Co-Advisor: Eng. Nicolò BATTAINI

Thesis by:
Niccolò Carlo MARTELLO Matr. 919766

Academic Year 2020 – 2021

Ai miei Genitori e alla mia Famiglia.

Acknowledgments

Vorrei innanzitutto ringraziare il mio relatore, Prof. Marco Lovera, per avermi dato l'opportunità di lavorare ad una tesi sperimentale durante la pandemia di COVID-19, e per la sua disponibilità mostrata durante questo lavoro. Un grande ringraziamento va a Nicolò, senza il quale questa tesi non esisterebbe: grazie al suo fantastico lavoro la continuazione del progetto del VTOL è stata decisamente più semplice e veloce. Lo ringrazio anche per l'ottimo lavoro di squadra che abbiamo portato avanti negli ultimi sette mesi sia dentro che fuori il laboratorio: è stato veramente un piacere concludere il mio percorso di studi in questo modo, avendo trovato un (quasi) omonimo amico. Concludendo i riconoscimenti accademici, vorrei ringraziare tutto lo staff del laboratorio *ASCL*, in particolare Mattia e Simone per il loro aiuto e Gabriele, Salvatore e Giovanni per le giornate spese in laboratorio: questi ultimi mesi sono volati con leggerezza anche grazie a voi.

Va da sé che senza la mia famiglia e miei amici che mi hanno supportato e sopportato nei momenti più difficili e con i quali ho condiviso tutti i momenti più belli non mi troverei di certo qui. Grazie quindi ai miei genitori, alle mie sorelle e ai miei fratelli, alla mia seconda famiglia di Varese, in particolare ad Ary e Cami, a Drew e Pier, a Pera, ai compagni di università "Ingegneri si diventa" e a tutti gli amici che ho incontrato durante il mio percorso. Senza di voi non sarei quello che sono adesso.

Abstract

The interest in Unmanned Aerial Vehicles (UAVs) has seen a dramatic increase in the past few years, due to their vast use cases and related economic advantages: infrastructure monitoring, aerial mapping, search and rescue, parcel delivery, law enforcement are just some examples. More recently, the potentiality of fixed wing eVTOL (electric Vertical Take Off and Landing) UAVs has emerged: such vehicles combine the ability to hover, take off and land vertically, with a high efficiency and long endurance and range typical of classic fixed wing aircraft.

The main purpose of this thesis is to actually build an eVTOL UAV completely designed from scratch within the *Aerospace Systems and Control Laboratory (ASCL)* of *Politecnico di Milano* in a previous master thesis, [1]. Such vehicle is capable of both vertical and forward flight, has a take off mass of around 6 kg, wing span of 2.25 m and an expected flight time of about 100 minutes. This experimental activity proved to be more challenging than it might sound, giving extremely valuable insights and lessons learned in the integration phase of a complex flight vehicle, the first one to be completely designed and produced inside the *Department of Aerospace Science and Technology (DAER)* long since. Each step of the building and integration phase has been thoroughly described and documented, to be used as reference for future works. Unfortunately, due to the many delays caused by the COVID-19 pandemic, the UAV could not be flight tested.

Along this primary objective, numerous other activities were performed on the project. The original aerodynamic model of the aircraft has been improved, including the fuselage, motor booms and motors and different software was used, *OpenVSP*. A motor test bench with accompanying software has been built from scratch, capable of estimating any arbitrary electric motor and propeller static and dynamic performance parameters quickly and with very little human intervention. A *Simulink* flight simulator has been built, using aircraft's data obtained from the aforementioned aerodynamic model and experimental motor identification, together with mass and inertia properties obtained with a precise *SolidWorks* CAD model and an accurate LiPo battery model. Such simulator is capable of simulating the UAV's three flight modes: vertical flight, transition and forward flight. The latter flight mode has been validated with linearized models and has been used to provide accurate estimates of the aircraft's flight dynamics and integral performance.

Sommario

L'interesse nei confronti degli Aeromobili a Pilotaggio Remoto (APR, o UAV - Unmanned Aerial Vehicles - in inglese) ha visto un drastico aumento negli ultimi anni, grazie ai numerosi casi applicativi e relativi vantaggi economici: monitoraggio di infrastrutture, cartografia aerea, ricerca e soccorso, consegna di pacchi e controllo da parte di forze dell'ordine sono solo alcuni esempi. Più recentemente sono emerse le potenzialità dei velivoli APR eVTOL (dall'inglese electric Vertical Take Off and Landing, ovvero elettrici a decollo ed atterraggio verticale) ad ala fissa: questi velivoli combinano la capacità di hover (ovvero stazionare in volo a punto fisso), decollo ed atterraggio verticale, con un'elevata efficienza e conseguenti lunghe distanze e tempi in volo tipici di velivoli classici ad ala fissa.

L'obiettivo principale di questa tesi è la costruzione di un eVTOL UAV, che è stato completamente progettato da zero all'interno del *Laboratorio di Sistemi e Controllo Aerospaziale (Aerospace Systems and Control Laboratory, ASCL)* del *Politecnico di Milano*, durante una precedente tesi magistrale, [1]. Questo velivolo è in grado di svolgere missioni sia in volo verticale, che in volo avanzato; ha una massa al decollo di circa 6 kg, apertura alare di 2,25 m e un tempo di volo stimato di circa 100 minuti. Questa attività sperimentale si è dimostrata essere più complessa di quanto possa apparentemente sembrare, producendo di fatto interessante esperienza e spunti di conoscenza relativi alla fase di costruzione ed integrazione di un velivolo complesso, il primo ad essere completamente progettato e costruito all'interno del *Dipartimento di Scienze e Tecnologie Aerospaziali (Department of Aerospace Science and Technology, DAER)* da molto tempo. Ogni fase dell'integrazione e costruzione è stata meticolosamente descritta e documentata, così che questo lavoro possa essere utile come riferimento per progetti futuri. Sfortunatamente, a causa dei numerosi ritardi dovuti alla pandemia di COVID-19, non è stato possibile provare il velivolo in volo.

Insieme a questo obiettivo primario, sono state eseguite numerose altre attività sul progetto. Il modello aerodinamico sviluppato nel lavoro originale è stato migliorato, includendo la fusoliera, i supporti motore e i motori e usando un software differente, *OpenVSP*. È stato costruito un banco prova motore con relativo software capace di stimare le prestazioni statiche e dinamiche di una qualsiasi combinazione di motore elettrico ed elica velocemente e con minima interazione umana. È stato inoltre creato un simulatore di volo in *Simulink*, usando i dati del

velivolo ottenuti dai già menzionati modello aerodinamico e dati sperimentali dei motori, insieme alle proprietà di massa e inerzia ottenute con un preciso modello CAD in *SolidWorks* e un accurato modello di una batteria LiPo. Il simulatore è in grado di simulare le tre differenti condizioni di volo del velivolo: hover e decollo e atterraggio verticale (VTOL), transizione e volo avanzato. Quest'ultima modalità di volo del simulatore è stata validata tramite confronto con modelli linearizzati ed è stata usata per fornire stime accurate della dinamica di volo del velivolo e delle sue prestazioni integrali.

Contents

Acknowledgments	I
Abstract	III
Sommario	V
List of figures	IX
List of tables	XV
Introduction	1
1 Aerodynamic analysis	7
1.1 Improved aerodynamic model in <i>OpenVSP</i>	7
1.2 <i>OpenVSP</i> aerodynamic analysis workflow	11
1.3 Validation of previous results	14
1.4 Elevator modification	19
1.5 Updated results	21
2 Propulsive unit experimental analysis	25
2.1 Theoretical background	25
2.2 Experimental setup	27
2.2.1 Hardware	28
2.2.2 Software	33
2.2.3 Calibration	33
2.3 Static performance identification	34
2.4 Dynamic response identification	36
2.5 Experimental Results	37
2.5.1 Validation	37
2.5.2 Experimental results	38
3 Fixed wing VTOL UAV dynamics	47
3.1 Frame of references and main hypothesis	48
3.2 Attitude descriptions	50

3.2.1	Euler angles	50
3.2.2	Quaternions	51
3.3	Kinematic equations	53
3.4	Dynamic equations	55
3.4.1	Translational motion	55
3.4.2	Rotational motion	56
3.5	External forces and moments	57
3.5.1	Common to all flight modes	57
3.5.2	Forward flight mode	58
3.5.3	VTOL mode	60
3.5.4	Transition flight	66
4	Simulink flight simulator	71
4.1	Simulator architecture	71
4.2	Validation of the aerodynamic derivatives interpolation scheme	77
4.3	Simulator validation with linearized response	79
4.4	Open loop forward flight dynamics and performance	83
5	Integration	91
5.1	Wing	91
5.1.1	Changes with respect to the original design	91
5.1.2	Building and integration	94
5.2	Tail	98
5.2.1	Changes with respect to the original design	98
5.2.2	Building and integration	101
5.3	Fuselage	105
5.3.1	Mockup	105
5.3.2	Avionics & Electronics	106
5.3.3	Integration	107
5.4	Motor booms	110
5.5	Final assembly of the UAV	114
	Conclusions	117
A	Additional test cases results from Chapter 1	125
B	Comparison of motor performance with manufacturer data	131
C	Determination of battery data from manufacturer datasheet	137
D	3D Printing technologies	143
D.1	Fused Deposition Modeling (FDM)	143
D.2	Selective Laser Sintering (SLS)	143

List of Figures

i.1	UAV build types: (a) multicopter; (b) fixed wing; (c) fixed wing VTOL.	2
i.2	eVTOL UAV (a) CAD model and (b) technical drawing, from [1].	4
1.1	<i>XFLR5</i> aerodynamic model developed in [1].	8
1.2	<i>OpenVSP</i> model used for the VLM analysis. The circles with arrows represent the positive rotation of the movable surfaces. . . .	10
1.3	Detail of the more complete <i>OpenVSP</i> model with motors, tail supports (not visible in the figure) and propellers.	11
1.4	Aerodynamic analysis flowchart.	12
1.5	<i>OpenVSP</i> and <i>XFLR5</i> lift coefficient against angle of attack curve comparison, case 2 (airspeed 15 m/s).	15
1.6	<i>OpenVSP</i> and <i>XFLR5</i> drag polar curve comparison, case 2 (airspeed 15 m/s).	16
1.7	<i>OpenVSP</i> and <i>XFLR5</i> single contributions to drag coefficient against angle of attack curve comparison, case 2 (airspeed 15 m/s).	16
1.8	<i>OpenVSP</i> and <i>XFLR5</i> trim elevator deflection comparison.	17
1.9	CAD model of the (a) new and (b) old elevator design.	20
1.10	Trim elevator deflection as a function of airspeed, new and old elevator design comparison.	21
1.11	Trimmed drag polar for the complete aircraft, from <i>OpenVSP</i> model aerodynamic analysis and preliminary estimate from [1].	22
1.12	Trim elevator deflection as a function of airspeed, final design. . .	22
2.1	(a) CAD model and (b) picture of the developed motor test bed.	28
2.2	Load cell and <i>HX711</i> acquisition chip.	29
2.3	Load cells setup on the motor test bed.	30
2.4	Photodiode and laser used.	31
2.5	Power module, used as a voltage and current sensor.	32
2.6	<i>Arduino</i> data acquisition and control board.	32
2.7	Calibration of the thrust load cell.	34
2.8	Motor angular velocity against throttle, motor test bed validation, <i>TMotor F20II KV3250</i>	37

2.9	Motor thrust against angular velocity, motor test bed validation, <i>TMotor F20II KV3250</i>	38
2.10	Thrust against angular velocity, motor powered from battery at 50% SOC, motor for forward flight <i>KDE2304XF-2350</i> , 5x4.5 inch, 2 blades bullnose propeller.	39
2.11	Angular velocity against throttle, motor powered from battery at 50% SOC, motor for forward flight <i>KDE2304XF-2350</i> , 5x4.5 inch, 2 blades bullnose propeller.	40
2.12	Electrical and mechanical power against angular velocity, motor powered from battery at 50% SOC, motor for forward flight <i>KDE2304XF-2350</i> , 5x4.5 inch, 2 blades bullnose propeller.	40
2.13	Motor dynamic response, measured and identified response, motor powered from DC power supply at 14.8 V, motor for forward flight <i>KDE2304XF-2350</i> , 5x4.5 inch, 2 blades bullnose propeller.	41
2.14	Thrust against angular velocity, motor powered from battery at 50% SOC, motor for vertical flight <i>KDE2315XF-2050</i> , 5x4.5 inch, 2 blades bullnose propeller.	42
2.15	Angular velocity against throttle, motor powered from battery at 50% SOC, motor for vertical flight <i>KDE2315XF-2050</i> , 5x4.5 inch, 2 blades bullnose propeller.	43
2.16	Electrical and mechanical power against angular velocity, motor powered from battery at 50% SOC, motor for vertical flight <i>KDE2315XF-2050</i> , 5x4.5 inch, 2 blades bullnose propeller.	43
2.17	Motor dynamic response, measured and identified response, motor powered from DC power supply at 14.8 V, motor for vertical flight <i>KDE2315XF-2050</i> , 5x4.5 inch, 2 blades bullnose propeller.	44
3.1	Typical VTOL UAV mission profile, from [1]. Phase A-B is a vertical take off and climb, B-C a cruise in forward flight mode, C-D is hovering phase using VTOL engines, D-E is a cruise in forward flight mode and E-F is vertical descent and landing.	47
3.2	Frames of reference used for the description of aircraft dynamics.	49
3.3	Schematics for the derivation of the kinematic equations.	54
3.4	Aircraft top view, with relevant dimensions named, propellers numbering and propellers spinning directions.	61
3.5	Flowchart of the possible strategies for the transition manoeuvre.	68
3.6	Possible strategies for the transition manoeuvre. For all the four cases, the transition starts on the left and ends on the right. The aircraft trajectory is shown by the dashed line.	69
4.1	<i>Simulink</i> model overall architecture.	73
4.2	<i>Simulink</i> model, airframe block.	74

4.3	<i>Simulink</i> model, aerodynamic subsystems: (a) forward flight mode and (b) VTOL mode.	75
4.4	<i>Simulink</i> model, environment block.	76
4.5	$C_{N_{\delta_r}}$ coefficient as a function of airspeed, filtered and unfiltered data.	77
4.6	C_{S_q} coefficient as a function of airspeed, filtered and unfiltered data.	78
4.7	C_{L_r} coefficient as a function of airspeed, interpolated model.	78
4.8	$C_{N_{\delta_a}}$ coefficient as a function of airspeed, interpolated model.	79
4.9	Time histories of the aircraft's longitudinal response after elevator doublet, <i>Simulink</i> non linear model and linear model.	81
4.10	Time histories of the aircraft's lateral-directional response after aileron doublet, <i>Simulink</i> non linear model and linear model.	82
4.11	Time histories of the aircraft's lateral-directional response after rudder doublet, <i>Simulink</i> non linear model and linear model.	83
4.12	UAV's eigenvalues, longitudinal (a) and lateral-directional (b) , as a function of airspeed. The arrows indicate the direction of increasing airspeed.	85
4.13	Trajectory of the aircraft after an elevator singlet excitation. The initial phugoid and successive descending spiral motions are clearly visible.	86
4.14	Time histories of the aircraft's longitudinal response after elevator singlet.	87
4.15	Time histories of the aircraft's lateral-directional response after elevator singlet.	88
5.1	Original design of the wing, from [1].	92
5.2	CAD models of the (a) new and (b) original wing designs. Colors are not representative of the real materials used.	93
5.3	Wing load carrying structure, both wings.	94
5.4	Detail of the aileron area of the load carrying structure, left wing.	94
5.5	Wing structure with complete servomotor and wiring, right wing.	95
5.6	Detail of the Pitot area of the wing with complete wiring, left wing.	95
5.7	Wing coverage procedure.	96
5.8	Detail of the aileron control mechanism, right wing.	97
5.9	Detail of the wing's Pitot tube, left wing.	97
5.10	Completed wings, ready to be installed.	97
5.11	Test of the original control mechanism for rudders and elevator.	99
5.12	Test of the new control mechanism for rudders and elevator.	99
5.13	CAD drawing of the (a) new, (b) intermediate and (c) original tail design. The movable surfaces (elevator and rudders) are coloured in blue.	100
5.14	Tail mounting procedure.	103
5.15	Complete tail, ready to be assembled on the rear booms.	104

5.16	Details of the mounting screws and nuts for the (a) vertical and (b) horizontal stabilizer.	104
5.17	Fuselage mockup, with top covers (a) open and (b) closed.	105
5.18	Avionics components. a) FCU; b) FCC; c) RX; d) Telemetry & Antenna; e) GPS; f) UBEC; g) PDB; h) Battery.	107
5.19	UAV's wiring scheme. Black represents ground wires, red represents positive voltage wires (either at battery voltage or 5 Volts), orange represents PWM signal wires and green represent more complex connections.	108
5.20	Final version of the fuselage, (a) before and (b) after the three lower parts were joined together permanently and avionics were inserted.	110
5.21	CAD model of the motor booms.	111
5.22	Forward motor and support mounting procedure.	111
5.23	Vertical motor and support mounting procedure.	112
5.24	VTOL motor and support mounted on boom.	113
5.25	Completed motor booms, ready to be installed.	113
5.26	Locking mechanism for the motor booms and wings. The carbon fibre tubes are placed inside this annular component and then are secured by tightening the screw.	114
5.27	Final integration of the UAV.	116
5.28	Complete assembled eVTOL UAV.	116
A.1	<i>OpenVSP</i> and <i>XFLR5</i> lift coefficient against angle of attack curve comparison, case 1 (airspeed 11 m/s).	126
A.2	<i>OpenVSP</i> and <i>XFLR5</i> drag polar curve comparison, case 1 (airspeed 11 m/s).	126
A.3	<i>OpenVSP</i> and <i>XFLR5</i> single contributions to drag coefficient against angle of attack curve comparison, case 1 (airspeed 11 m/s).	127
A.4	<i>OpenVSP</i> and <i>XFLR5</i> lift coefficient against angle of attack curve comparison, case 3 (airspeed 22 m/s).	127
A.5	<i>OpenVSP</i> and <i>XFLR5</i> drag polar curve comparison, case 3 (airspeed 22 m/s).	128
A.6	<i>OpenVSP</i> and <i>XFLR5</i> single contributions to drag coefficient against angle of attack curve comparison, case 3 (airspeed 22 m/s).	128
B.1	Thrust against angular velocity, motor powered from DC power supply, motor for forward flight <i>KDE2304XF-2350</i> , 5x4.5 inch, 2 blades bullnose propeller.	132
B.2	Angular velocity against throttle, motor powered from DC power supply, motor for forward flight <i>KDE2304XF-2350</i> , 5x4.5 inch, 2 blades bullnose propeller.	132

B.3	Electrical and mechanical power against angular velocity, motor powered from DC power supply, motor for forward flight <i>KDE2304XF-2350</i> , 5x4.5 inch, 2 blades bullnose propeller.	133
B.4	Thrust against angular velocity, motor powered from DC power supply, motor for vertical flight <i>KDE2315XF-2050</i> , 5x4.5 inch, 2 blades bullnose propeller.	134
B.5	Angular velocity against throttle, motor powered from DC power supply, motor for vertical flight <i>KDE2315XF-2050</i> , 5x4.5 inch, 2 blades bullnose propeller.	134
B.6	Electrical and mechanical power against angular velocity, motor powered from DC power supply, motor for vertical flight <i>KDE2315XF-2050</i> , 5x4.5 inch, 2 blades bullnose propeller.	135
B.7	Voltage against throttle, motor powered from DC power supply, motor for vertical flight <i>KDE2315XF-2050</i> , 5x4.5 inch, 2 blades bullnose propeller. The voltage drop at higher throttle setting is clearly visible.	135
C.1	Overlander 8500 mAh, 14.8 V, 20C datasheet (from [2]). Relevant data highlighted in red.	138
C.2	Battery voltage against depth of discharge at constant temperature of 23 °C, varying discharge currents (C-rates).	139
C.3	Battery voltage against depth of discharge at constant current of 4.25 A (0.5C), varying temperatures.	139
C.4	Battery voltage against current for different depths of discharge. Extrapolation to zero current shown in dashed lines; open circuit voltages represented by circles.	140
C.5	Battery schematics electrical diagram.	141
C.6	Battery resistance as a function of the normalized depth of discharge, temperature of 23 °C.	142
D.1	Schematics of 3D printing technologies: (a) Fused Deposition Modeling (FDM) and (b) Selective Laser Sintering (SLS).	144
D.2	FDM printers used for the construction of the eVTOL UAV: (a) <i>Ultimaker S3</i> and (b) <i>Stratasys Fortus 450 MC</i>	144

List of Tables

i.1	eVTOL UAV main design parameters and performance requirements, from [1].	3
1.1	Summary of aerodynamic test cases flight conditions.	15
1.2	<i>OpenVSP</i> and <i>XFLR5</i> aerodynamic coefficients and derivatives comparison, case 2 (airspeed 15 m/s).	18
1.3	<i>OpenVSP</i> and <i>XFLR5</i> neutral point position comparison.	19
2.1	Motor static coefficients identification results, motor for forward flight <i>KDE2304XF-2350</i> , 5x4.5 inch, 2 blades bullnose propeller. .	41
2.2	Motor dynamic response identification results, motor for forward flight <i>KDE2304XF-2350</i> , 5x4.5 inch, 2 blades bullnose propeller. .	42
2.3	Motor static coefficients identification results, motor for vertical flight <i>KDE2315XF-2050</i> , 5x4.5 inch, 2 blades bullnose propeller. .	44
2.4	Motor dynamic response identification results, motor for vertical flight <i>KDE2315XF-2050</i> , 5x4.5 inch, 2 blades bullnose propeller. .	45
3.1	Drag coefficients of the VTOL parts; values from [3].	63
4.1	Eigenvalues of the aircraft's dynamic modes during the forward flight phase, original model from [1] and improved model.	84
4.2	Frequency and damping of the aircraft's dynamic modes during the forward flight phase, original model from [1] and improved model. .	84
4.3	Endurance and range for different airspeeds.	88
5.1	Rudders and elevator measured maximum deflections.	101
A.1	<i>OpenVSP</i> and <i>XFLR5</i> aerodynamic coefficients and derivatives comparison, case 1 (airspeed 11 m/s).	129
A.2	<i>OpenVSP</i> and <i>XFLR5</i> aerodynamic coefficients and derivatives comparison, case 3 (airspeed 22 m/s).	130
B.1	Motor static coefficients comparison, motor for forward flight <i>KDE2304XF-2350</i> , 5x4.5 inch, 2 blades bullnose propeller.	133
B.2	Motor static coefficients comparison, motor for vertical flight <i>KDE2315XF-2050</i> , 5x4.5 inch, 2 blades bullnose propeller.	136

Introduction

Unmanned Aerial Vehicles (UAVs) have seen a dramatic growth in the past few years. The broadest definition of a UAV is an aircraft without a human pilot on board. This encompasses a huge array of vehicles, from the Piaggio P.1HH HammerHead, a 6 tonnes UAV developed from the Piaggio P180, to the Axis Vidius, a 250 grams drone equipped with a camera. Focusing the discussion on smaller and lighter vehicles, mainly three UAV's build types have emerged. The first one is the multicopter (Figure i.1a), which is the first thing that comes in mind when talking about drones. It has the advantages of being highly maneuverable, versatile, inexpensive, low weight and easy to build, deploy and recover. On the other hand, these vehicles can carry limited payloads, are susceptible to high winds and are not efficient, consuming a lot of battery power for relatively short flight times. The second type is the fixed wing (Figure i.1b), which is basically a scaled version of a conventional fixed wing aircraft. The main advantages are its long flight time and endurance, derived from a much higher flight efficiency compared to multicopters. The requirement to have a runway from which to take off and land, or some other supports (such as catapults for taking off and nets for landing (or more appropriately catching)), and the lower maneuverability compared to multicopters are the main drawbacks of such solution. The third type (Figure i.1c) is a mix of the previous two categories, and can generally be referred to as a fixed wing VTOL UAV (VTOL standing for Vertical Take Off and Landing). Those aircraft feature some sort of fixed wing configuration, allowing them to fly in the forward flight mode just like a normal airplane, and some mechanism that allows them to transition to vertical flight and hover, take off and land vertically, like a multicopter. It can be easily seen how such solution combines the advantages of multicopters and fixed wing aircraft: easy deployment without the need of any launching or recovery solution and hover capabilities, with the benefits of being able to fly much more efficiently, providing drastic improvements to flight time and range.

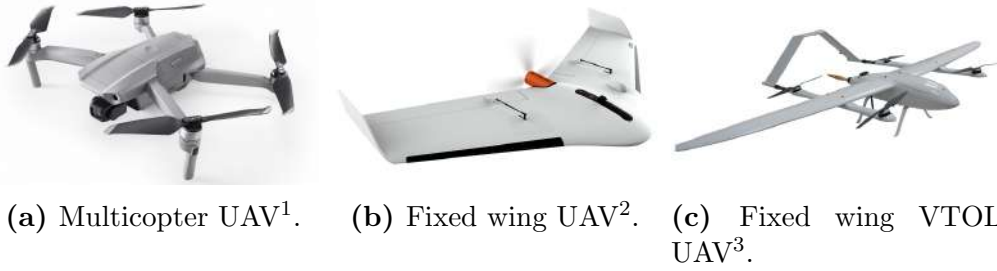


Figure i.1: UAV build types: (a) multicopter; (b) fixed wing; (c) fixed wing VTOL.

A report from DHL on the current and near future use cases of UAVs ([7]) helps to get an idea of the potentiality of fixed wing VTOL aircraft.

- **Infrastructure monitoring.** Using UAVs for long or remote infrastructure surveillance and monitoring, such as power lines, oil & gas pipelines, offshore wind farms, drastically cuts the operations costs, since helicopters are currently used for such activities ([8]).
- **Agriculture and forestry.** Real time data on crops, animal tracking, wild fires spotting and mapping are all activities which are well suited for VTOL UAVs ([9]).
- **Environmental protection.** UAVs can be effectively used to safeguard endangered species: they are already being used in Africa to fight rhinoceros poaching ([10]).
- **Emergency response and police.** Thanks to the vast array of sensors which can be installed on UAVs, they can be used in disaster relief situations, to gather essential data otherwise not obtainable, as well as being suitable for Search And Rescue (SAR) missions and for law enforcement ([11]).
- **Logistics and delivery.** VTOL UAVs can play an important role both for last miles delivery, as well as vital parcels dispatch to remote areas ([12]).

A huge number of VTOL UAVs currently exist on the market, with famous names such as Google, DHL, UPS and Amazon building and prototyping their own vehicles as well. Academic institutions are also showing interest in the topic, with much experience gained by Southampton University in the field of fixed wing UAVs ([13]).

¹DJI Mavic Air 2, picture from [4].

²Trimble fixed wing drone, picture from [5].

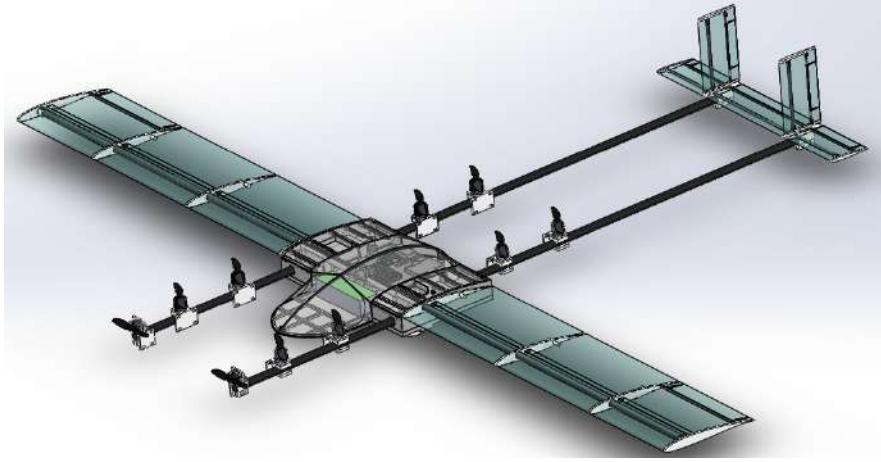
³Baby Shark VTOL 250, picture from [6].

Background

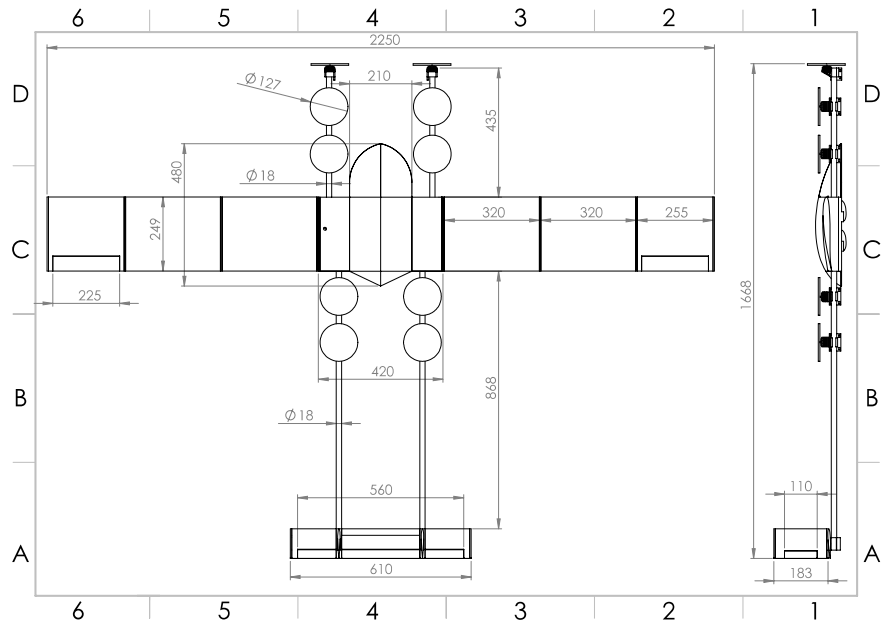
The present thesis is a continuation of the work started by Nicolò Battaini, in his master thesis titled "Design and dynamic modeling of a VTOL UAV" ([1]). The goal of the original work was to design completely from scratch, build and fly an eVTOL UAV within the *ASCL* Laboratory of *Politecnico di Milano*. However, due to the COVID-19 pandemic, only the detailed design of the vehicle could be carried out, which is the starting point of the present work. Figure i.2 shows the original CAD model and technical drawings of the vehicle from [1] and Table i.1 summarizes the aircraft's main design parameters and performance requirements. It is worthy to recall the original main driver and requirement of the whole project: the aircraft has to exist and fly. Another important high level objective was that the vehicle had to be modular, so that different configurations could be easily swapped and studied in future works: in this sense the vehicle can be considered a research-focused machine.

Table i.1: eVTOL UAV main design parameters and performance requirements, from [1].

Type	Parameter	Value
Performance requirements	Stall speed	11 m/s
	Cruise speed	15 m/s
	Maximum speed	22 m/s
	Rate of climb	2 m/s
	Cruise altitude	120 m AGL
	Hover time	5 min
	Cruise endurance	85 min
	Payload mass	200 g
Design parameters	Take off mass	4.8 kg
	Wing span	2.25 m
	Wing surface	0.56 m ²
	Wing airfoil	Selig 2046
	Number of FWD flight motors	2
	Number of VTOL motors	8
	Battery capacity	8500 mAh



(a) CAD model.



(b) Technical drawing, scale 1:10. Measurements in mm.

Figure i.2: eVTOL UAV (a) CAD model and (b) technical drawing, from [1].

Thesis objectives

The main objective of this thesis is to actually build and test the aircraft. As it was for [1], this work is the first prototyping experience of a fixed-wing aircraft at the *Department of Aerospace Science and Technology (DAER)* of *Politecnico di Milano* long since.

Besides this primary objective, another important goal of this project is to improve the preliminary models of the vehicle and to build a flight simulator for

the aircraft, which will be heavily used for the study and prototyping of the UAV's control laws.

As already previously mentioned, this machine will be used as a flying test bed in the field of fixed-wing eVTOL aircraft. Apart from the design itself, much study can be done with this machine regarding the transition phase from vertical flight to forward flight, which is a relative new subject which poses numerous challenges and opportunities as well.

Unfortunately, the delays caused by the COVID-19 pandemic did not allow to actually test the aircraft in flight, which will be the next activity to be performed on the vehicle.

Thesis structure

The thesis is organized as follows.

- In Chapter 1, an improved aerodynamic model of the eVTOL UAV is built, using the software *OpenVSP*. The new model is first validated against the original one, developed in [1] using another software, and then results, in terms of drag polar, trim elevator deflections and aerodynamic and stability derivatives as a function of airspeed are obtained.
- Chapter 2 is dedicated to the experimental identification of the UAV's forward and vertical electric motors performance. A custom design motor test bench and accompanying software have been developed, which have been used to obtain accurate estimates of the motors' static and dynamic performance. This test bench is ready to use for future tests on other motors, should the need arise.
- In Chapter 3 the dynamic equations of the aircraft are derived, for its three different modes of operations: vertical flight, transition and forward flight. Possible transitions strategies are described as well.
- In Chapter 4 the *Simulink* flight simulator is described. Such simulator is first validated with the linearized aircraft response and is then used to estimate the aircraft's open loop flight dynamics and integral performance data.
- Chapter 5 describes the vehicle's building and integration. For each main component (wings, tail, fuselage, motor booms) the design changes which had to be done with respect to the original design are described and each step in the construction is thoroughly described and documented.

Chapter 1

Aerodynamic analysis

In the original work on the VTOL developed in [1], an aerodynamic model in the Vortex Lattice Method (VLM) software *XFLR5* was developed, consisting only in the aircraft's lifting surfaces (wings, horizontal and vertical tail). Hence, there was the need to improve this model, including the whole aircraft's geometry, to assess more accurately the vehicle's flight characteristics. Another, more comprehensive software has been used for this purpose: *OpenVSP*.

In this chapter the improvements with respect to the original model and the results obtained are presented. In more detail, in the first section the upgraded aerodynamic model is described, detailing the advantages of *OpenVSP* over *XFLR5*. In the second section, the *MATLAB* routines for the aerodynamic analysis are presented. Then, the results of the simulations of the new model, considering only its lifting surfaces, are compared to the results obtained in [1], for validation purposes. As the aircraft's integration proceeded, there was the need to modify the geometry of the aircraft's elevator, which is described in the fourth section. Finally the updated aerodynamic results for the full model are presented. Further results are provided in Appendix A.

1.1 Improved aerodynamic model in *OpenVSP*

One of the suggestions presented in [1] to further improve the UAV was to increase the modeling accuracy of the aircraft aerodynamics, by performing a Computational Fluid Dynamics (CFD) analysis. However, after some considerations, it was deemed unnecessarily complicated and long to simulate the drone's aerodynamics using CFD and hence it was decided to improve the modelling using simpler tools, which make use of the Vortex Lattice Method (VLM).

The original aerodynamic model was developed in the software *XFLR5* ([14]), an open-source aerodynamic tool which implements a VLM solver. It was made up of the main wing and the complete tail (both horizontal and vertical surfaces). The original model, developed in [1], is shown in Figure 1.1.

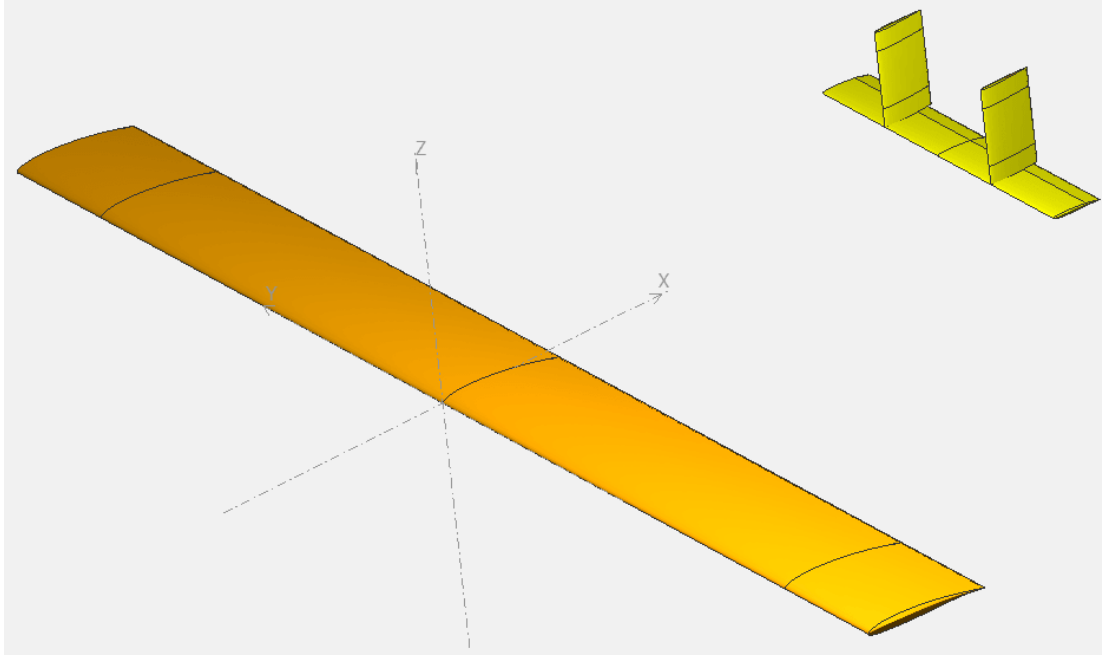


Figure 1.1: *XFLR5* aerodynamic model developed in [1].

The choice for the modelling software of the improved model fell on *OpenVSP* ([15]), which is a parametric aircraft geometry tool, developed originally by *NASA* in the early 1990's and is still being actively updated and supported. The improvements on the already existing aerodynamic model of the drone can be summarized as follows.

- **Improved estimation of the complete aircraft's viscous drag coefficient.** As reported in Chapter 5 of [1], the viscous drag coefficient was originally calculated increasing by 20% the viscous drag of similar UAVs, because the other available empirical formulas were developed for other types of aircraft (*i.e.* large jet/turboprop engines). With the new, more complete model, a more accurate estimation of the drag coefficient can be obtained, by using the parasitic drag tool of *OpenVSP*.
- **Improved stability derivatives estimate.** Thanks to the added modeling of the fuselage and motor booms (both forward and aft), the estimation of the stability derivatives is more accurate.

The new software *OpenVSP* has numerous advantages and added features compared to *XFLR*.

- **Improved viscous drag estimation.** *OpenVSP*'s parasitic drag tool allows for a more accurate estimation of the viscous drag contribution, by

considering the actual wetted surface of each modelled component, and then applying empirical formulas based on the skin friction coefficient, namely

$$C_{D_0} = \frac{S_{wet} \cdot Q \cdot C_f \cdot FF}{S_{ref}}, \quad (1.1)$$

where S_{wet} and S_{ref} are the wetted and reference areas, Q is the interference factor, that is a scale factor applied to the drag coefficient, C_f is the skin friction coefficient and FF is the form factor. More information on the formulas implemented in the software can be found in [16]. Whilst this approach is still based on empirical formulations, it has the advantages of taking into account the real and complete three dimensional geometry of the aircraft.

- **Estimation of unsteady aerodynamic derivatives.** The unsteady aerodynamic derivatives (which are the derivatives with respect to the angle of attack and angle of sideslip derivatives) can be calculated using *OpenVSP*, whereas *XFLR5* does not implement unsteady aerodynamics. This allows to improve the fidelity of any aircraft dynamic model which will be developed based on the present aerodynamic analysis.
- **Choice of either VLM or panel method solvers.** *OpenVSP* supports both panel and Vortex Lattice Method (VLM) solvers and although only the latter has been used, in a future work the former could be studied as well and the two methods could also be compared. The choice to use only a VLM method was driven by the fact that it was computationally faster than the panel method. This is important since the complete calculation of the stability derivatives in the flying envelope already took quite some time (in the order of hours). Furthermore, the panel method does not provide significant improvements (see [17]). Lastly, the panel method solver presents serious convergence problems with the modelled geometry, whereas VLM does not.
- **Propeller modelling.** *OpenVSP* also supports the study of the propeller aerodynamic interaction using an actuator disk model. Although this has not been investigated in the present work, it could be an interesting development for future studies.

The final *OpenVSP* model, used in the VLM aerodynamic analysis and shown in Figure 1.2, consists of the main wing, complete tail, fuselage, engine booms and movable surfaces; it was modelled with the actual dimensions directly taken from the *SolidWorks* CAD model developed in [1]. Furthermore, an improved version of this model has been developed, considering also the motors, tail supports and the propellers themselves (modelled as actuator disks), which is shown in Figure 1.3. This model only missed the four small fuselage feet, which were more complicated

to model and only account for a small contribution to the viscous drag. The first, simpler model was used in the actual VLM analysis, since neither lift nor lift-induced drag are affected by the motors and tail supports. The latter elements were included in the separate calculation of the viscous drag, which however did not take into consideration the propellers' drag, since the software did not allow for this calculation.

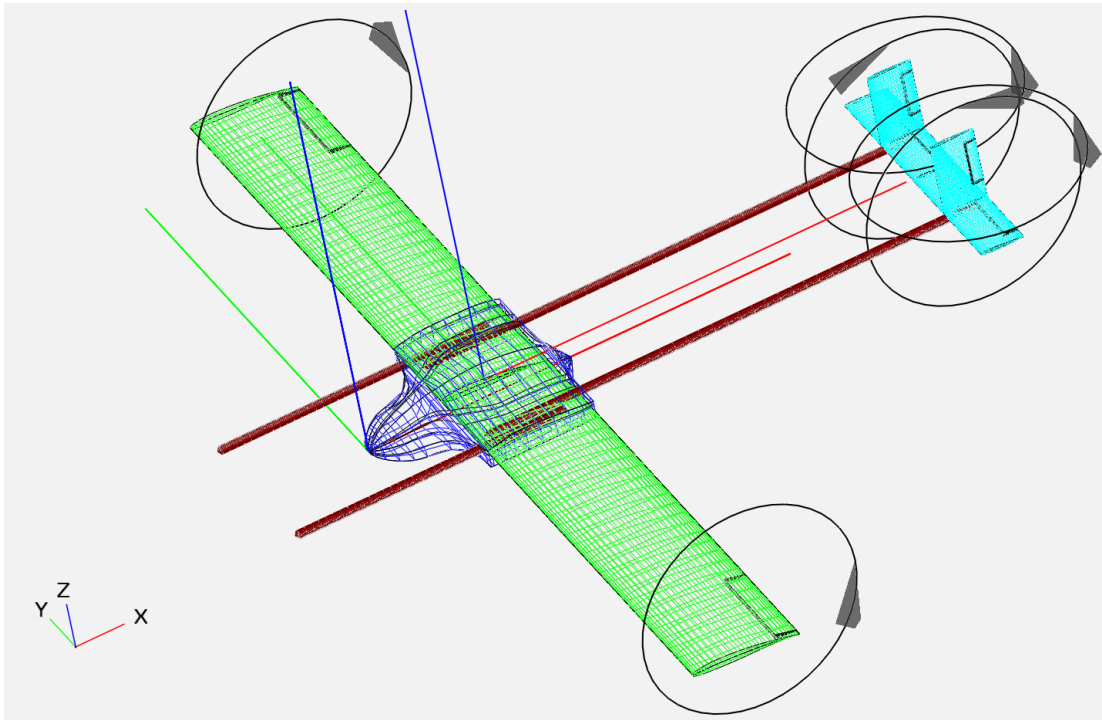


Figure 1.2: *OpenVSP* model used for the VLM analysis. The circles with arrows represent the positive rotation of the movable surfaces.

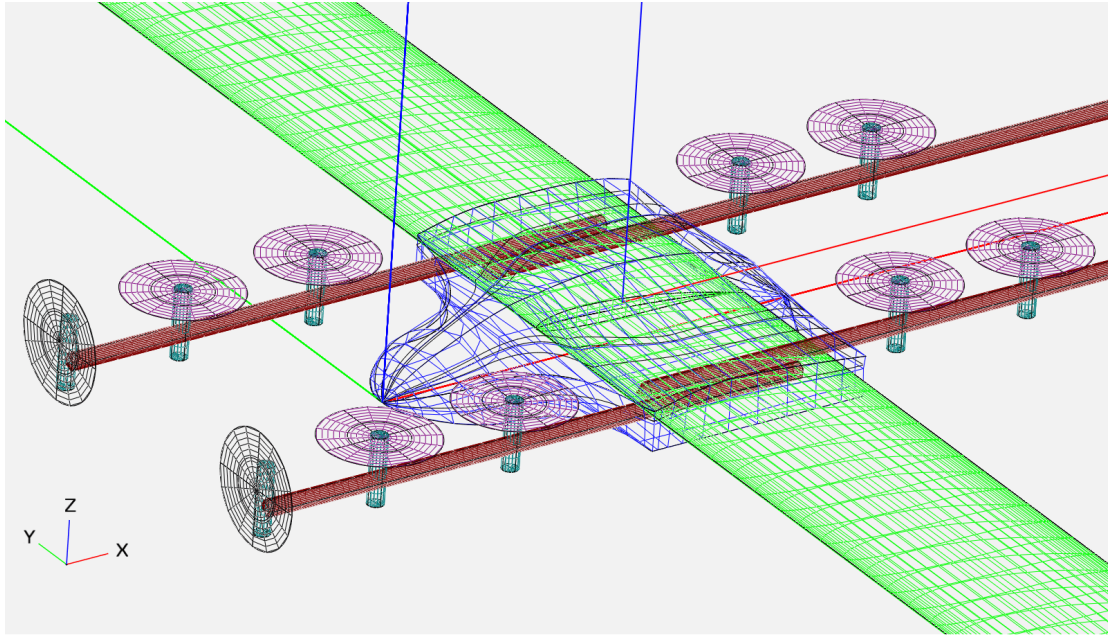


Figure 1.3: Detail of the more complete *OpenVSP* model with motors, tail supports (not visible in the figure) and propellers.

1.2 *OpenVSP* aerodynamic analysis workflow

After the development of the new aerodynamic model in *OpenVSP*, the various results presented in [1] were first validated against the new model, as it will be shown Section 1.3, and then updated with the more complete model, which results will be shown in Section 1.5. The analyses performed consist in the trimmed drag polar, the elevator deflection range to trim the aircraft within its flight envelope and the calculation of all the stability and control derivatives for a number of trimmed flight conditions also within the aircraft's envelope.

All those analyses require the calculation of the trim state, namely the values of angle of attack and elevator deflection, for any given aircraft weight and flight condition (airspeed and altitude). Whilst *XFLR5* automatically calculates such trim conditions, *OpenVSP* does not yet implement this feature, so a custom procedure had to be developed. The approach chosen was to use the aerodynamic analysis capabilities of *VSPAero*, which is the aerodynamic solver of *OpenVSP*, while writing the main routines in *MATLAB*. Figure 1.4 shows the flow chart schematics of the three main *MATLAB* routines developed for this purpose.

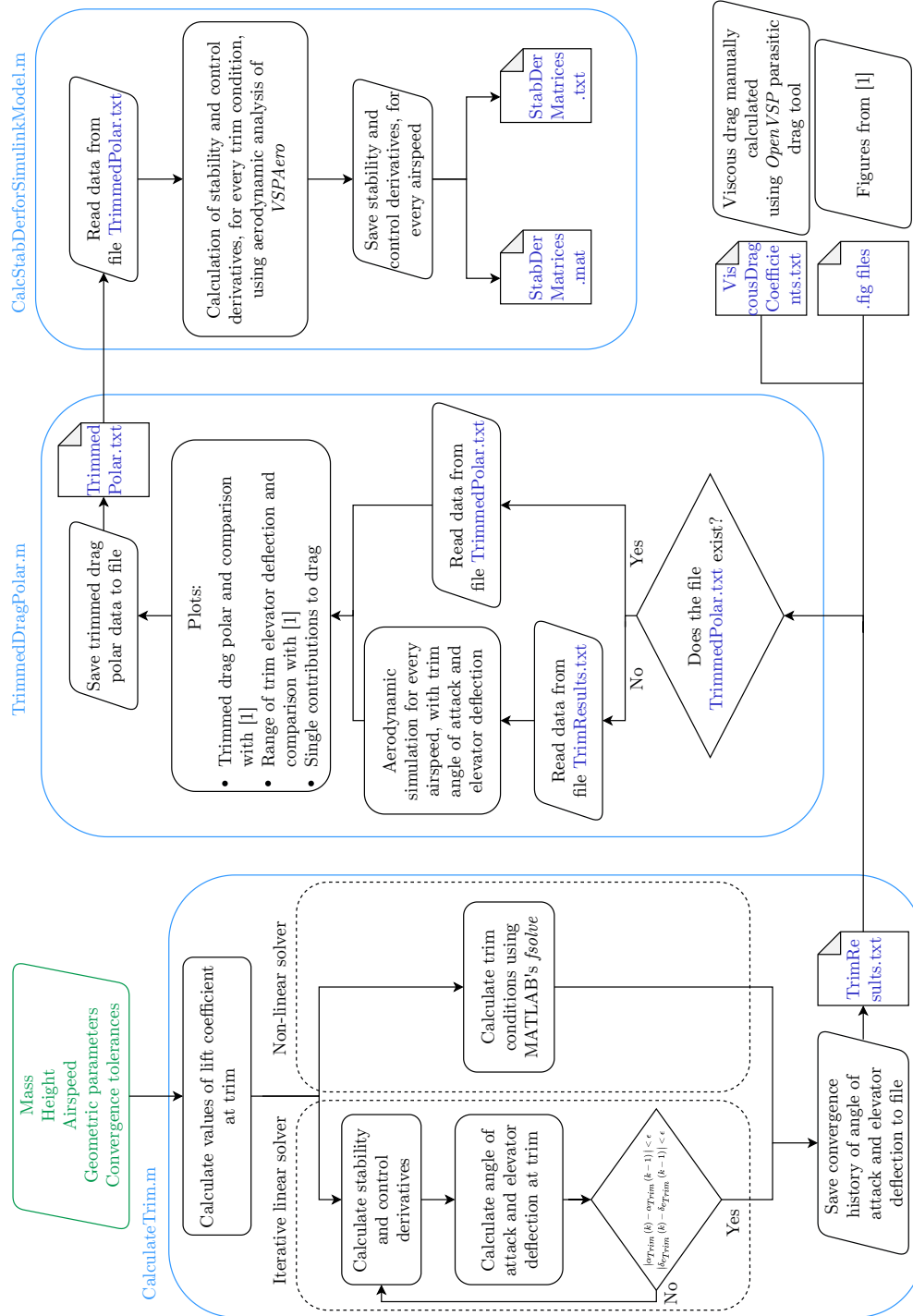


Figure 1.4: Aerodynamic analysis flowchart.

The three main scripts will be hereby described.

- *CalculateTrim.m*.

This routine calculates the values of the angle of attack and elevator deflection at trim for any given weight and flight condition. Two different methods have been implemented. The first iteratively solves the linear system of two equations which describes the trim condition (lift equals weight and null pitching moment with respect to the center of gravity), in its non-dimensional form:

$$\begin{cases} C_{L_{Trim}} &= C_{L_\alpha} \alpha_{Trim} + C_{L_{\delta_e}} \delta_{e_{Trim}} + C_{L_0} \\ 0 &= C_{M_\alpha} \alpha_{Trim} + C_{M_{\delta_e}} \delta_{e_{Trim}} + C_{M_0} \end{cases} \quad (1.2)$$

where $C_{L_{Trim}}$ is the trim lift coefficient, C_{L_α} , C_{M_α} , $C_{L_{\delta_e}}$, $C_{M_{\delta_e}}$ are the derivatives of the lift and moment coefficients with respect to the angle of attack and elevator deflection, C_{L_0} and C_{M_0} are the lift and moment coefficients at zero angle of attack and elevator deflection, and α_{Trim} , $\delta_{e_{Trim}}$ are the angle of attack and elevator deflection at trim. Since the values of the coefficients C_{L_α} , C_{M_α} , $C_{L_{\delta_e}}$, $C_{M_{\delta_e}}$, C_{L_0} and C_{M_0} depend on the flight condition, at each iteration they are calculated again with *VSPAero* with the updated values of α_{Trim} and $\delta_{e_{Trim}}$ as reference condition. The stopping condition of the iterative procedure takes into account the difference between the last two iterations in the value of both the angle of attack and elevator deflection:

$$\max \left(\left| \alpha_{Trim}(k) - \alpha_{Trim}(k-1) \right|, \left| \delta_{e_{Trim}}(k) - \delta_{e_{Trim}}(k-1) \right| \right) < \epsilon, \quad (1.3)$$

where $\alpha_{Trim}(k)$, $\delta_{e_{Trim}}(k)$ indicate the k^{th} iteration of the calculated trim angle of attack and elevator deflection and ϵ is an arbitrary tolerance. Even though this method converged, its results were not satisfactory due to the relatively large values of the pitching moment coefficient for the final calculated trim condition (up to an order of magnitude of 10^{-2}).

Hence, a second routine has been implemented, based on the non-linear system of equations solver *fsolve* built in *MATLAB*. Using this method, the non-linear system which describes the trim condition given by

$$\begin{cases} 0 &= C_L(\alpha_{Trim}, \delta_{e_{Trim}}) - C_{L_{Trim}} \\ 0 &= C_M(\alpha_{Trim}, \delta_{e_{Trim}}) \end{cases} \quad (1.4)$$

is solved coupling *fsolve* and *VSPAero*, the latter being used for the estimation of the aerodynamic coefficients fed into *fsolve*. This method provides satisfactory results (pitching moment coefficients lower than 10^{-3}) in few iterations of the solver (always lower than nine for the analyzed cases), and hence it has been used for all the aerodynamic analyses.

- *TrimmedDragPolar.m*

This script reads the output *.txt* file produced by the *CalculateTrim.m* routine, containing the trim angle of attack and elevator deflection for all the specified flight conditions. For each one of those conditions it runs a *VSPAero* aerodynamic simulation with the aircraft in trim conditions. It then outputs the trimmed drag polar, the trim elevator deflection range and a decomposition of the single contributions to the drag (induced and viscous); finally it saves all the relevant results in a *.txt* file.

- *CalcStabDerforSimulinkModel.m*

This script reads the output *.txt* file produced by the *TrimmedDragPolar.m* routine and for each of the flight conditions calculates all the stability and control derivatives, which are useful for the development of a flight simulator of the aircraft. The results are then saved both in a *.txt* file and *.mat* file.

1.3 Validation of previous results

A comparison of the obtained results was performed with the *XFLR5* model already developed in [1], in order to build confidence in the *OpenVSP* model and in its aerodynamic solver. The comparison was done on a simplified version of the *OpenVSP* model, consisting in only the lifting surfaces of the UAV (wing, horizontal and vertical tail), to match the *XFLR5* model.

Three flight conditions were analyzed: at the stall speed of 11 m/s, at the cruise speed of 15 m/s and at the maximum speed of 22 m/s. Different aerodynamic results of interest were compared:

- the trimmed drag polar, obtained from an angle of attack sweep from -6 degrees to +6 degrees with all control surfaces in neutral position;
- the elevator deflection needed to trim the aircraft within its flight envelope;
- all the static stability and control derivatives, obtained from trimmed conditions (found using *XFLR5* software, since *OpenVSP* does not implement an algorithm to find the aircraft trim condition, as already previously said) at the three already mentioned airspeeds.

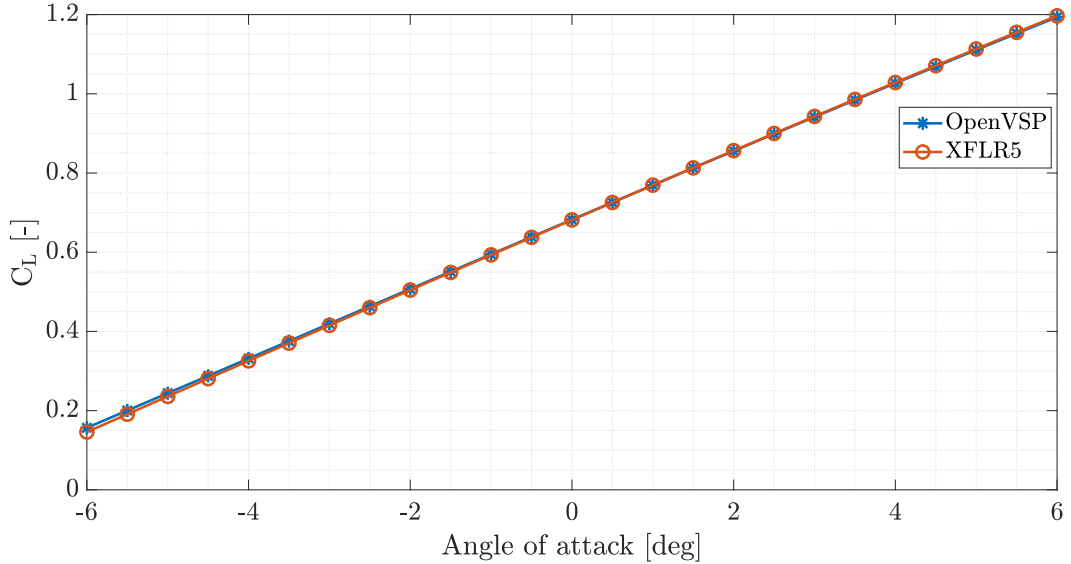
Table 1.1 summarizes the flight conditions for the three test cases analyzed.

Table 1.1: Summary of aerodynamic test cases flight conditions.

	Case 1	Case 2	Case 3
Airspeed [m/s]	11	15	22
Angle of attack [deg]	5.8	-0.6	-4.5
Elevator deflection ¹ [deg]	-2.0	0.0	1.5

¹ Positive deflection for trailing edge down.

The results for case 2 (airspeed of 15 m/s) are hereby presented, whilst the results for the other two cases can be found in Appendix A. Figure 1.5, Figure 1.6 and Figure 1.7 show the lift coefficient against angle of attack curve, the drag polar of the aircraft and the single contributions to the drag coefficient (viscous, induced, and total value) against the angle of attack, for the two *OpenVSP* and *XFLR5* models. Note that the *XFLR5* model also contains the contribution of the viscous drag, calculated by using an approach similar to the strip theory ([18]). In this method the lifting surfaces are considered as a sum of quasi two dimensional strips; the value of the two dimensional lift coefficient on each profile is obtained and the drag coefficient is calculated from the viscous drag polar of the profile. The former is then integrated over the wing span to obtain the total lifting surface viscous drag coefficient.

**Figure 1.5:** *OpenVSP* and *XFLR5* lift coefficient against angle of attack curve comparison, case 2 (airspeed 15 m/s).

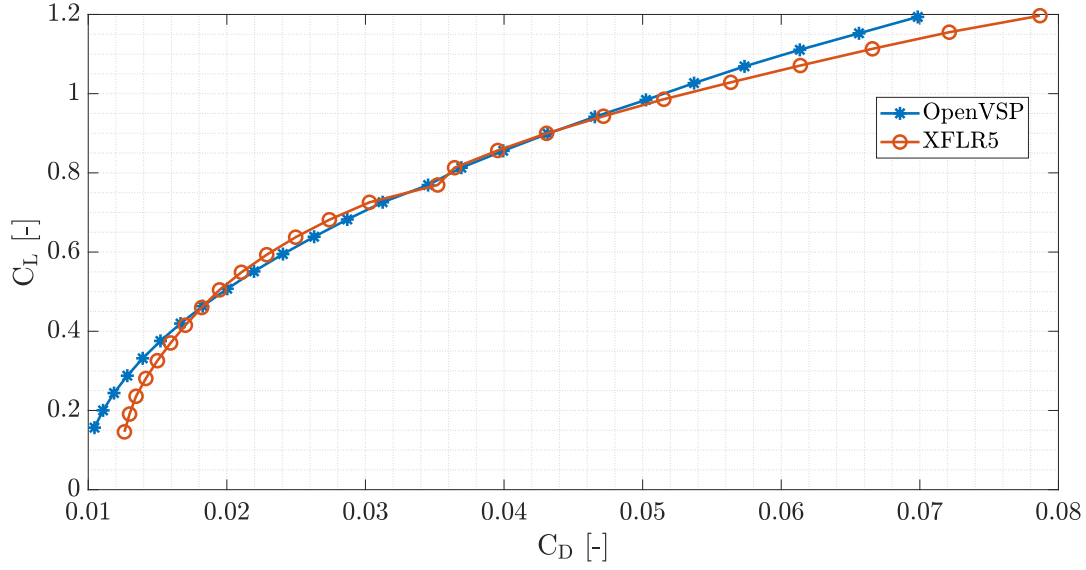


Figure 1.6: *OpenVSP* and *XFLR5* drag polar curve comparison, case 2 (airspeed 15 m/s).

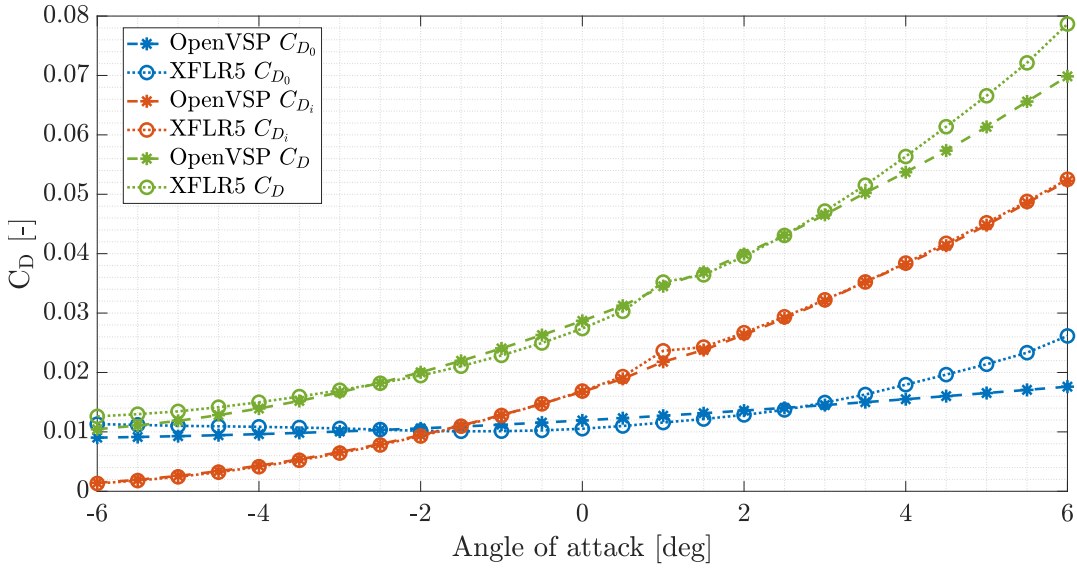


Figure 1.7: *OpenVSP* and *XFLR5* single contributions to drag coefficient against angle of attack curve comparison, case 2 (airspeed 15 m/s).

As it can be seen from Figure 1.5, the results in term of the lift coefficient almost perfectly match. The mismatch in the polar curve, shown in Figure 1.6, is due to the difference in the drag coefficient against angle of attack curve, most specifically only in the viscous contribution of drag, which can be seen in the blue curves of Figure 1.7. This can be attributed to the different methods used by the two software in the calculations of the viscous drag: *OpenVSP* uses empirical

formulas based on skin friction coefficients, whereas *XFLR5* uses a strip-like approach. The induced drag coefficient perfectly matches between the two software.

Figure 1.8 shows the comparison of the elevator deflection needed to trim the aircraft for the two models, obtained not from a single case analysis as previously described, but by a sweep of trim airspeed from the stall airspeed of 11 m/s to the maximum airspeed of 22 m/s.

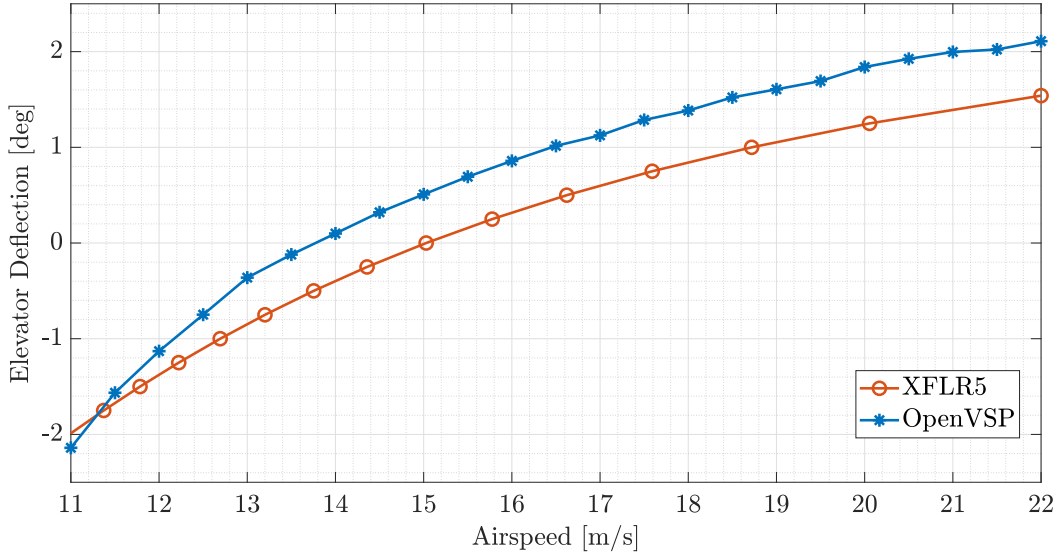


Figure 1.8: *OpenVSP* and *XFLR5* trim elevator deflection comparison.

The two curves are similar both in their shape and in their absolute value, with the maximum difference between the two being approximately 0.5 degrees, showing good agreement between the trim results from the two software.

The comparison of all the stability and control derivatives between the two software is presented in Table 1.2.

Both the aerodynamic coefficients, the stability and the control derivatives show a really good agreement, with the sole exception of the coefficient of the X force in the body axes. This is because the X force is almost aligned with the drag force (which is, as previously said, estimated differently between the two software) due to the smallness of the angle of attack in the analyses. Hence this difference reflects in the less similar values of the C_X coefficient and of its derivatives. Apart for this, the match between the results from the two software is more than satisfactory, with differences for the most important stability and control derivatives all below 20%, and most of them below 10%. As an example, the least accurate match is between the derivatives of the C_N coefficient, whereas the most accurate one is between both the C_Z and C_M coefficients.

Similar results were obtained for the other two test cases, and can be found in Appendix A.

Table 1.2: *OpenVSP* and *XFLR5* aerodynamic coefficients and derivatives comparison, case 2 (airspeed 15 m/s).

	<i>Tot</i>	α	β	p	q	r	δ_a	δ_e	δ_r
C_X	<i>XFLR5</i>	-0.0313	0 ¹	0 ¹	-0.452	0 ¹	$4.94 \cdot 10^{-3}$	-0.174	$-1.13 \cdot 10^{-5}$
	<i>OpenVSP</i>	-0.0151	$2.64 \cdot 10^{-3}$	0.0726	-0.0850	$9.49 \cdot 10^{-3}$	$-3.25 \cdot 10^{-4}$	-0.0133	$-1.19 \cdot 10^{-4}$
	% Diff.	51.7%	-163.2%	Na ²	81.2%	Na ²	Na ³	92.3%	Na ³
C_Y	<i>XFLR5</i>	0.00	0 ¹	0.0313	0 ¹	0.157	$-2.12 \cdot 10^{-3}$	$2.09 \cdot 10^{-15}$	0.0525
	<i>OpenVSP</i>	$-1.00 \cdot 10^{-7}$	$-5.8 \cdot 10^{-6}$	0.0323	$1.22 \cdot 10^{-3}$	0.158	$8.19 \cdot 10^{-3}$	$-7.00 \cdot 10^{-7}$	0.0766
	% Diff.	Na ³	12.3%	-3.3%	Na ²	-0.9%	Na ³	Na ³	-46%
C_Z	<i>XFLR5</i>	-0.629	-5.15	0 ¹	-7.32	0 ¹	$2.60 \cdot 10^{-3}$	-0.321	$-4.77 \cdot 10^{-5}$
	<i>OpenVSP</i>	-0.626	-4.96	-2.52 · 10 ⁻³	-7.33	-0.0221	$6.68 \cdot 10^{-4}$	-0.300	$-3.29 \cdot 10^{-5}$
	% Diff.	0.4%	3.7%	Na ²	-0.1%	Na ²	Na ³	6.5%	Na ³
C_L	<i>XFLR5</i>	0.00	0 ¹	-1.30 · 10 ⁻³	0 ¹	0.173	0.147	$5.71 \cdot 10^{-15}$	$9.86 \cdot 10^{-4}$
	<i>OpenVSP</i>	0.00	$5.10 \cdot 10^{-6}$	$-6.92 \cdot 10^{-3}$	-4.04 · 10 ⁻⁴	0.170	0.127	$-5.00 \cdot 10^{-7}$	$2.54 \cdot 10^{-3}$
	% Diff.	0%	Na ²	Na ³	Na ²	1.7%	13.8%	Na ³	Na ³
C_M	<i>XFLR5</i>	$-1.08 \cdot 10^{-3}$	-0.453	0 ¹	-15.4	0 ¹	$6.62 \cdot 10^{-4}$	-1.24	$-1.66 \cdot 10^{-4}$
	<i>OpenVSP</i>	0.0102	-0.463	$-6.05 \cdot 10^{-3}$	-16.2	-0.0340	$1.47 \cdot 10^{-4}$	-1.23	$-9.16 \cdot 10^{-5}$
	% Diff.	Na ³	-2.2%	Na ²	-5.5%	Na ²	Na ³	1.0%	Na ³
C_N	<i>XFLR5</i>	0.00	0 ¹	0.0769	0 ¹	-0.0677	$9.36 \cdot 10^{-4}$	$1.98 \cdot 10^{-16}$	-0.0240
	<i>OpenVSP</i>	0.00	$2.80 \cdot 10^{-6}$	0.0799	-5.68 · 10 ⁻⁴	-0.0780	$-5.32 \cdot 10^{-3}$	$-3.00 \cdot 10^{-7}$	-0.0364
	% Diff.	0%	Na ²	-3.9%	Na ²	-15.2%	Na ³	Na ³	-51.5%

¹ Not calculated explicitly by *XFLR5*.² Not available because coefficient by *XFLR5* is not calculated explicitly.³ Not calculated because one of the values is 0 or really close to 0 (smaller than $1 \cdot 10^{-2}$).

The last comparison which can be drawn is between the neutral points calculated by the two software, which are presented in Table 1.3.

Table 1.3: *OpenVSP* and *XFLR5* neutral point position comparison.

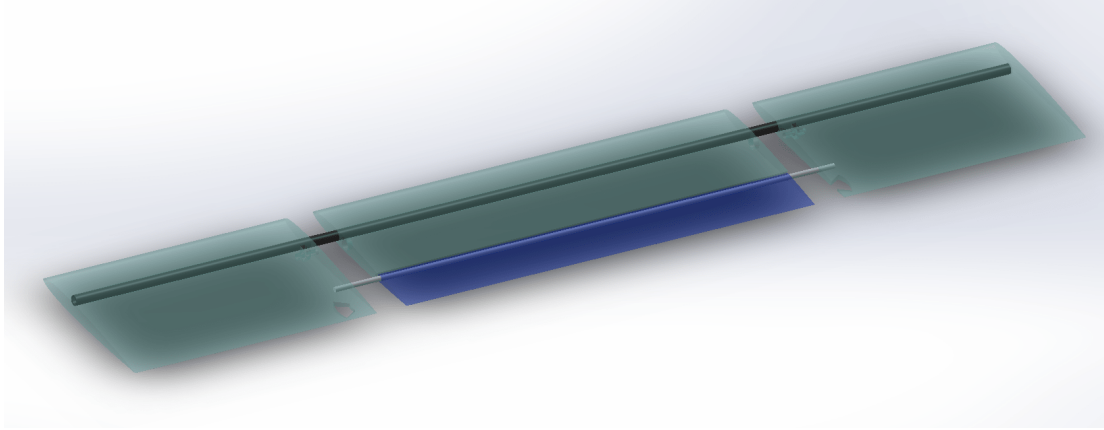
	Case 1	Case 2	Case 3
<i>XFLR5</i>	0.116 m	0.124 m	0.128 m
<i>OpenVSP</i>	0.131 m	0.123 m	0.127 m
% Diff.	-12.9%	0.8%	0.8%

As it can be seen, the results are accurate, apart for case 1 (airspeed 11 m/s). This might be caused by the fact that this case is run at the stall speed, and hence the aircraft is in a high angle of attack attitude, with incipient stall conditions. Since VLM methods are not able to model the stall and high angle of attack behaviour, the results for such conditions are not extremely reliable, and might explain the difference visible for such case.

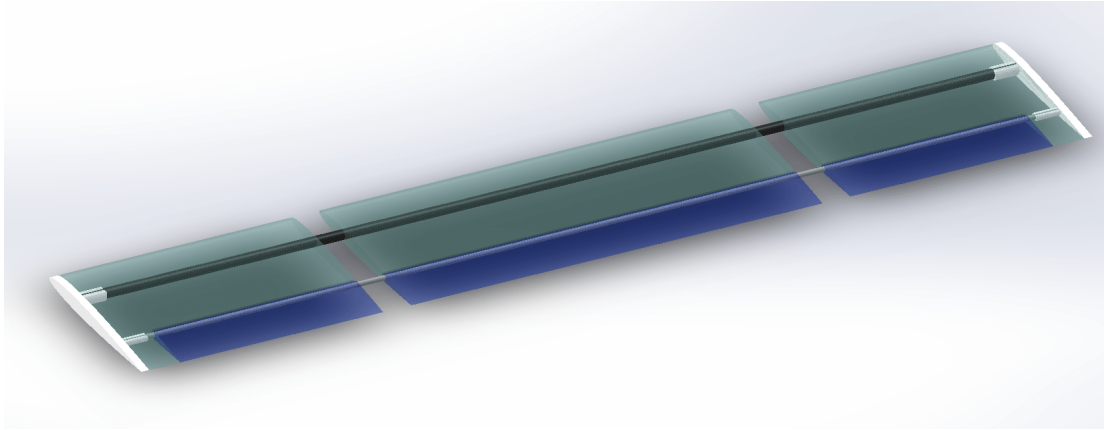
To conclude, this analysis not only validated the new *OpenVSP* model, but also grew confidence in the results of the preliminary design of the vehicle of [1].

1.4 Elevator modification

During the design review of each component for the actual production of the UAV, some changes were made to different elements of the aircraft, as it will be thoroughly described in Chapter 5. One such change was made to the elevator. Originally, it was designed to be made by three parts: a central one, between the two vertical stabilizers, and two external ones, to the left and to the right of the left and right vertical stabilizer, respectively. When studying the positioning of the servomotors for the actuation of the elevator, it was decided, for easiness of installation and construction, to eliminate the two external parts, and to keep only the central one. This has the direct effect of reducing the elevator area and hence, for a given flight condition, the deflection required to trim the aircraft will be higher, compared to the original design. The reduction of the elevator area was also directly suggested in [1], which states that "[...] the proposed elevator design can be considered oversized with respect to the operative conditions. For future versions of the UAV a reduced elevator can be considered." The old and new elevators are shown in Figure 1.9.



(a) New elevator design.



(b) Old elevator design.

Figure 1.9: CAD model of the (a) new and (b) old elevator design.

A study of the elevator deflections needed to trim the aircraft within its flight envelope with the smaller elevator was already made in [1] (but not explicitly reported in the final manuscript), and has been also studied with the new updated *OpenVSP* model in the present work. Such envelope, for the old and new design, is shown in Figure 1.10.

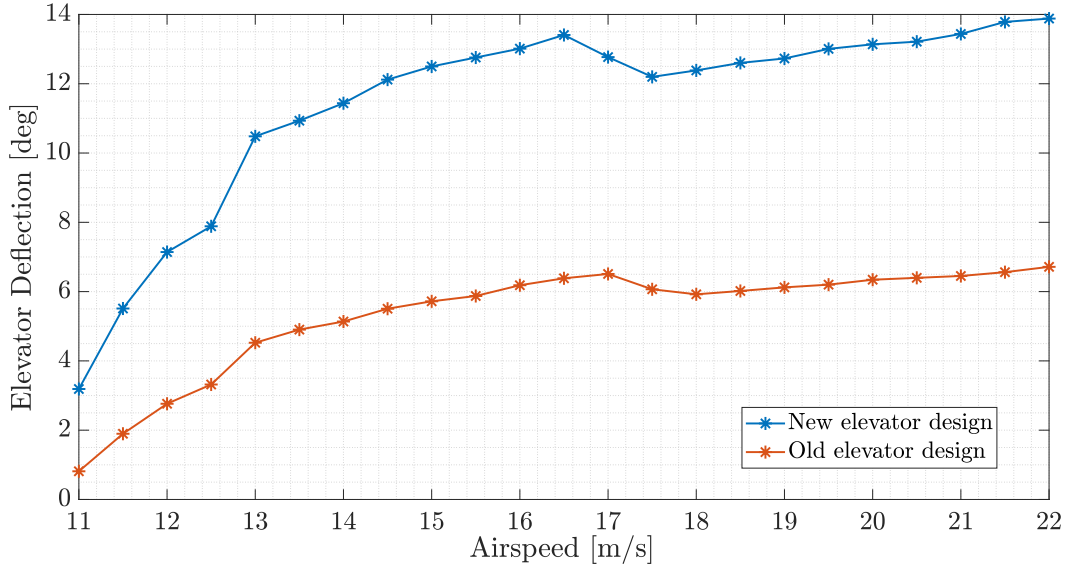


Figure 1.10: Trim elevator deflection as a function of airspeed, new and old elevator design comparison.

It can be seen that also for the new elevator the maximum deflection required to trim the aircraft is more than acceptable, with the maximum value reaching approximately 14 degrees (the maximum deflection limit reachable by the elevator being dictated by either the stall of the horizontal surface or by the maximum torque which the servomotor can provide). Incidentally, the higher deflections to trim the aircraft given by the new elevator have another advantage. In the original design, the elevator deflections values were between 1 and 7 degrees, which meant that the servomotor needed to have a high angular position accuracy to allow for a precise trim of the aircraft within its envelope. By lowering the elevator's surface area, and hence increasing the deflection needed to trim the aircraft, the angular displacement of the control surface to trim the aircraft within its envelope is increased, and hence the servomotor does not need to be as accurate as it was required by the original design.

1.5 Updated results

Figure 1.11 shows the trimmed drag polar for the updated *OpenVSP* complete model, together with the initial estimate provided in [1]. It is recalled that the latter was obtained from preliminary design requirements and by computing the viscous drag coefficient C_{D_0} as 120% the value obtained from similar competitors calculated using the procedure reported in [19].

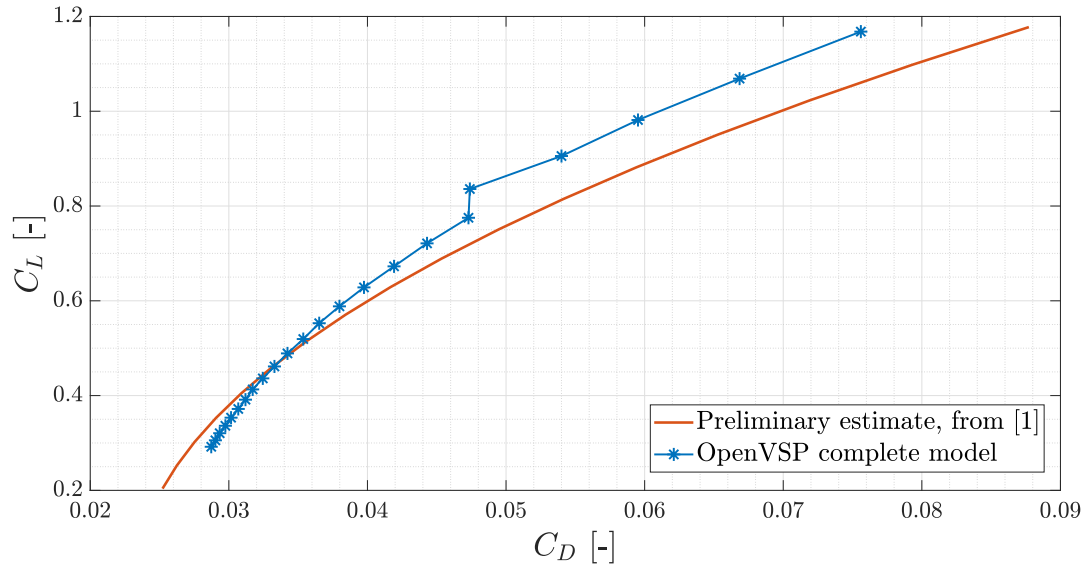


Figure 1.11: Trimmed drag polar for the complete aircraft, from *OpenVSP* model aerodynamic analysis and preliminary estimate from [1].

As it can be seen, the drag polars are reasonably similar to each other. At this stage, the one computed in the present work has to be considered the most reliable. Nevertheless, the similarity of the two results validates the methodology for the preliminary estimation done in [1] and grows confidence in the overall aircraft design.

In Figure 1.12 the trimmed elevator deflection within the flight envelope is presented, as already shown by the blue curve of Figure 1.10.

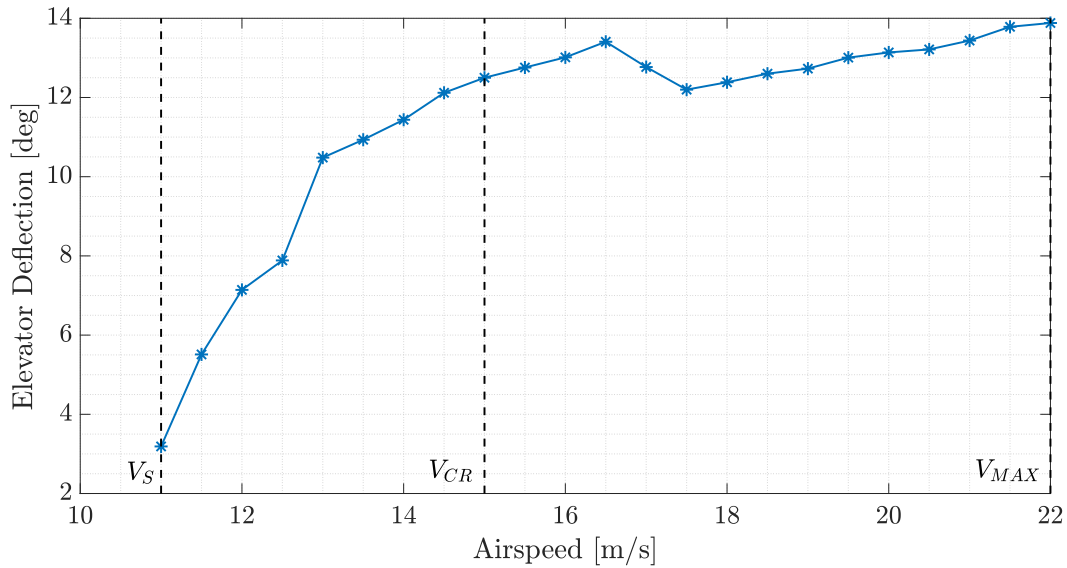


Figure 1.12: Trim elevator deflection as a function of airspeed, final design.

The rougher behaviour of the curve is due to the effect of the non-lifting bodies of the fuselage and the motor booms, since it is not present in the analysis performed on only the lifting bodies (see red curve of Figure 1.8).

With the new design and with the improved aerodynamic analysis, the elevator deflection is always positive (trailing edge down), and is around 12 degrees at cruising airspeed, whereas the original tail was designed such that at cruise the elevator deflection would be zero. This could be solved by changing the tail incidence. However this modification would influence the stability parameters and flying qualities of the aircraft, hence potentially requiring other re-designs. Hence it is considered to be a better approach to keep the UAV as is and then group all the major modifications after a proper flight test campaign on the actual aircraft is performed.

Chapter 2

Propulsive unit experimental analysis

It is important, for the development of an accurate flight simulator, to gather experimental data of the aircraft's real propulsive unit, even if manufacturer data is already available, since the test conditions (atmospheric, power supply, etc) might be different. This chapter is devoted to the description of the developed motor test bed, which was designed with the aim of being self contained, both in terms of hardware and software, and being readily reusable to execute further tests on different motors in an easy and straightforward way. The final aim is to experimentally obtain the thrust, torque and power coefficients C_T , C_Q and C_P of the motors, the relations between throttle percentage, motor angular velocity and thrust and the actual electrical power absorbed by the propulsive unit (composed by the ESC, motor and propeller). The final aim is to have a better estimate of the actual expected performance of the aircraft.

In the first section the theory behind the non-dimensional coefficients which will be identified is shortly presented. In the second section the actual test bed is described in detail, both in its hardware and software parts. Then, the methodology used for the static performance identifications, namely a fitting in the least square sense, is presented. In the fourth section, the motor dynamic response identification procedure is explained. Finally, in the last section, the experimental results are shown.

2.1 Theoretical background

The flow generated by a rotating propeller is complex to accurately model, even in its simplest form, such as during a stationary hover of an helicopter, drone or VTOL. Its three-dimensional characteristics, viscous effects and other behaviours generate a complex vortical wake structure, which makes the development of accurate models for performance predictions complicated. Hence, a simpler approach

was developed, named the *Rankine-Froude momentum theory*. The main hypothesis for this theory are:

- one-dimensional flow, parallel to the axis of rotation of the propeller;
- stationary flow;
- incompressible flow;
- inviscid flow;
- actuator disk hypothesis: the propeller is modelled as being infinitely thin and as having an infinite number of blades.

Following the procedure detailed in [20], the Buckingham Π theorem is used for the derivation of the non-dimensional thrust, power and torque coefficients.

Theorem 1 (Buckingham Π theorem) *Let $q_1, q_2, q_3, \dots, q_n$ be n dimensional variables that are physically relevant in a given problem and that are inter-related by an (unknown) dimensionally homogeneous set of equations. These can be expressed via a functional relationship of the form*

$$F(q_1, q_2, q_3, \dots, q_n) = 0 \quad \text{or equivalently} \quad q_1 = f(q_2, q_3, \dots, q_n). \quad (2.1)$$

If k is the number of fundamental dimensions required to describe the n variables, then there will be k primary variables and the remaining $j = (n - k)$ variable can be expressed as $(n - k)$ dimensionless and independent quantities or " Π groups", $\Pi_1, \Pi_2, \Pi_3, \dots, \Pi_{n-k}$. The functional relationship can thus be reduced to the much more compact form:

$$\Phi(\Pi_1, \Pi_2, \Pi_3, \dots, \Pi_{n-k}) = 0 \quad \text{or equivalently} \quad \Pi_1 = \phi(\Pi_2, \Pi_3, \dots, \Pi_{n-k}). \quad (2.2)$$

■

The induced velocity v_i through the rotor can be written as a function of the thrust T , a reference area A , usually the disk area, a reference velocity V , usually the blade tip speed V_{Tip} and the air density ρ of the flow:

$$v_i = f(T, A, V_{Tip}, \rho) \quad \text{or equivalently} \quad F(T, A, V_{Tip}, \rho, v_i) = 0. \quad (2.3)$$

Applying the Buckingham Π theorem with $n = 5$ variables, $k = 3$ fundamental dimensions (mass, length, time), two variables can be expressed as Π groups, and the functional relationship rewritten according to equation (2.2). Choosing the linearly independent variables ρ , A and V_{Tip} , the two Π groups can be obtained:

$$\Pi_1 = \rho^{-1} A^{-1} V_{Tip}^{-2} T = \frac{T}{\rho A V_{Tip}^2} = C_T, \quad (2.4)$$

$$\Pi_2 = \rho^0 A^0 V_{Tip}^{-1} v_i = \frac{v_i}{V_{Tip}} = \lambda_i, \quad (2.5)$$

where C_T is a thrust coefficient and λ_i is an inflow coefficient, and hence

$$\lambda_i = \phi(C_T) = \frac{v_i}{\Omega R} = \frac{1}{\Omega R} \sqrt{\frac{T}{2\rho A}} = \sqrt{\frac{T}{2\rho A (\Omega R)^2}} = \sqrt{\frac{C_T}{2}}, \quad (2.6)$$

where the last equalities come from the definition of the inflow coefficient and from the relation between thrust and the inflow velocity, here not reported for brevity.

Therefore, by knowing that the tip speed can be calculated as $V_{Tip} = \Omega R$, with R being the rotor radius, the thrust coefficient is defined as

$$C_T = \frac{T}{\rho A \Omega^2 R^2}. \quad (2.7)$$

The power coefficient is then defined as

$$C_P = \frac{P}{\rho A V_{Tip}^3} = \frac{P}{\rho A \Omega^3 R^3} \quad (2.8)$$

and can be directly related to the thrust coefficient:

$$C_P = \frac{P}{\rho A \Omega^3 R^3} = \frac{T v_i}{\rho A \Omega^3 R^3} = \left(\frac{T}{\rho A \Omega^2 R^2} \right) \left(\frac{v_i}{\Omega R} \right) = C_T \lambda_i = \frac{C_T^{3/2}}{\sqrt{2}}. \quad (2.9)$$

From the simple relation between power and rotor shaft torque $P = \Omega Q$ the torque coefficient is readily obtained:

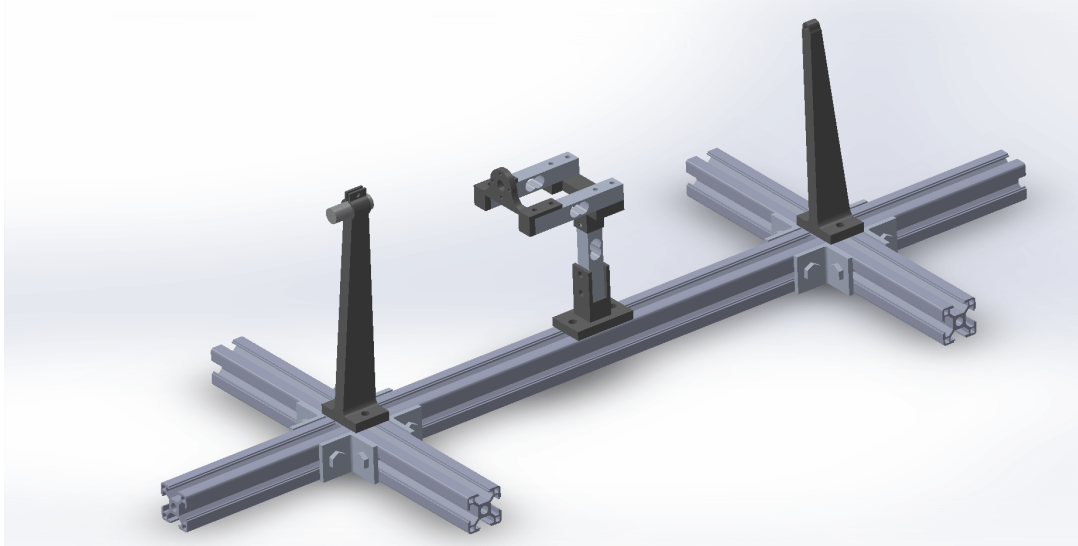
$$C_Q = \frac{Q}{\rho A V_{Tip}^2 R} = \frac{Q}{\rho A \Omega^2 R^3} = C_P. \quad (2.10)$$

2.2 Experimental setup

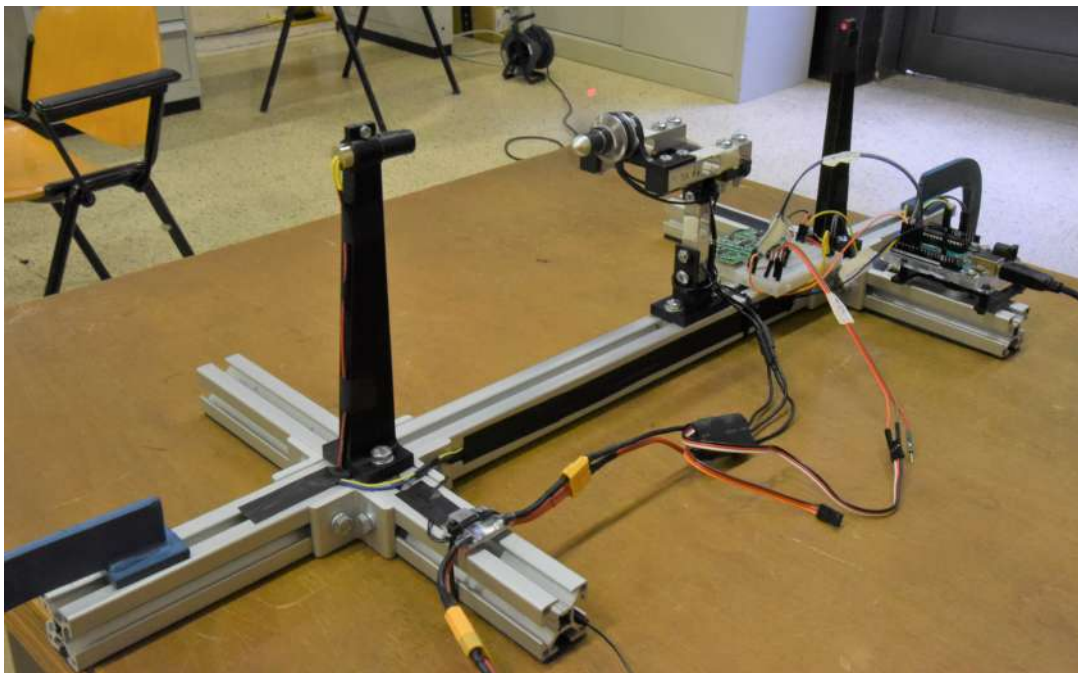
The motor test bed was designed with the aim of being fully reusable for future tests of other motors, in the simplest possible way. This resulted in the development of both the hardware setup, as well as a small software suite, written in *MATLAB*, for both the data acquisition and its post-processing. Inspiration was taken from two previous master thesis developed within the *ASCL* Laboratory, [21] and [22].

2.2.1 Hardware

Figure 2.1 shows the CAD three dimensional view of the complete platform, as well as a picture of its final realisation during an identification test.



(a) CAD model of the motor test bed.



(b) Picture of the test bed during an identification test.

Figure 2.1: (a) CAD model and (b) picture of the developed motor test bed.

Main frame

The main frame of the motor test bed is a simple, yet solid structure made of standard 30 mm x 30 mm aluminum profiles, on which it is possible to screw in position the various components, namely the towers which hold the laser and photodiode, the load cells and the *Arduino*, together with its breadboard. During the tests, the main frame is clamped on both ends to the working table.

Load cells

The load cells selected are readily available online and are of the strain gauge type. For this type of load cells, strain gauges are attached to a piece of metal: as a force is applied to the cell, the strain gauges deform, hence changing their resistance. This results in a change of voltage, which is sensed and amplified, and which is then correlated to the applied force. The amplifier chip comes with the load cell, and is a *HX711* chip, which features a 24 bit analog-to-digital converter with selectable gain and sampling frequency of either 10 or 80 samples per second (the chip datasheet can be found at [23]). However, in order to achieve the 80 Hz sampling frequency, some soldering work had to be made on the chip, which was deemed too difficult. Hence the sampling frequency of the overall system is limited to 10 Hz, but with the possibility of increasing it up to 80 Hz in the future, if needed. The load cells chosen can measure up to 5 kg. Figure 2.2 shows the load cell and the *HX711* chip.

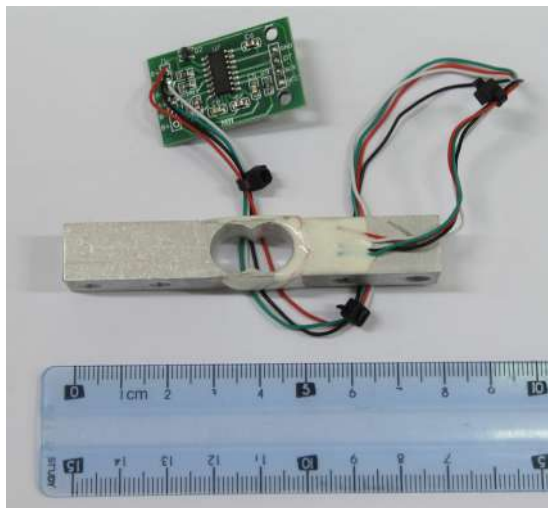


Figure 2.2: Load cell and *HX711* acquisition chip.

Three load cells in total were needed, together with three *HX711* chips, which were connected all together sharing the same clock, so that their data acquisition would be synchronized. Figure 2.3 shows a detail of the load cells disposition on

the motor test bed main frame. The vertical load cell measures the thrust produced by the motor, whereas the two horizontal cells measure the torque exerted by the motor, by providing two separate force measurements, which can then be used to easily calculate the total torque. This set up was directly inspired by other commercial motor test beds, such as [24]. The clock signal and the three digital output are connected to pins D4 to D7 of the *Arduino* board.

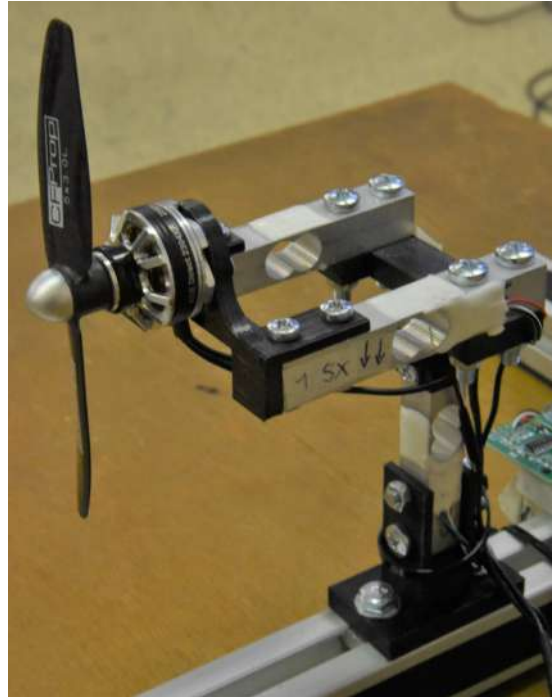


Figure 2.3: Load cells setup on the motor test bed.

Optical tachometer

The digital tachometer is used to measure the motor's angular velocity, and works on the same principle as the one described in [21]. A small laser light, installed on a tower on one side of the test bed, is shone onto a photodiode, which is installed on the other tower on the opposite side with respect to the laser, with the motor being between the two. The photodiode is a component which acts as short circuit if enough light is shone onto it, whereas it behaves as an open circuit otherwise. Hence, whenever a blade passes between the laser and photodiode, the light beam from the laser is interrupted, and the photodiode does not conduct anymore, until the blade has passed. By wiring the photodiode appropriately to pin D2 of the *Arduino*, it generates a square wave as the blades rotate, triggering an Interrupt Service Routine (ISR) in the *Arduino* microprocessor. An ISR is a part of code in the processor which execution is triggered by an external event: in this set-up the external event is set to be the voltage on pin D2 transitioning from high (5

V) to low (0 V). When this happens, the time at which the event was triggered is recorded, $T_{Int}(k)$, and by subtracting this value from the previous time at which the interrupt occurred, $T_{Int}(k-1)$, the blade period T_{Blade} is obtained:

$$T_{Blade} = T_{Int}(k) - T_{Int}(k-1). \quad (2.11)$$

From this value, it is easy to obtain the period of the whole propeller and its angular frequency, by knowing the number of blades N_{Blades} :

$$\Omega_{Prop} = \frac{2 \cdot \pi}{T_{Prop}} = \frac{2 \cdot \pi}{N_{Blades} \cdot T_{Blade}}. \quad (2.12)$$

Figure 2.4 shows the photodiode and laser used.

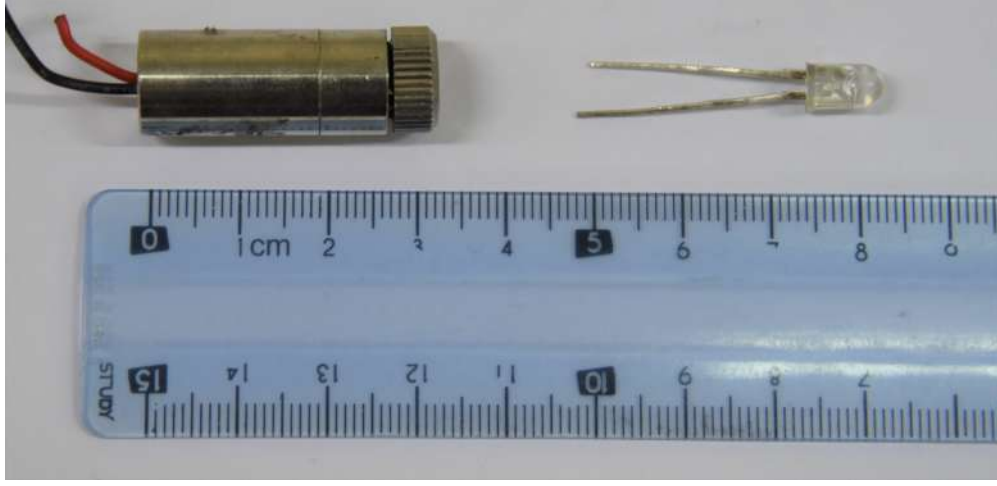


Figure 2.4: Photodiode and laser used.

Voltage and current sensor

A handy component which can be used to measure the electrical quantities is the power module. A power module is used in drones to power the on-board electronics boards (avionics, telemetry, etc): it draws electrical power directly from the battery, and delivers a stabilized 5 V DC power supply, while providing also measurements of the current drawn from the battery and of its voltage. Hence, by putting this small component between the DC power supply or battery and the power cables feeding into the motor's ESC, the current drawn by the motor and its driving voltage are directly obtained. The analog outputs are then sampled by analog pins A0 and A1 of the *Arduino* board. One interesting note: in order to have proper measurements, the ground of the *Arduino* has to be connected to the ground of the power supply, otherwise the voltage and current measurements just fluctuate between their maximum and minimum value, since they would not have a common reference ground signal with the acquisition board. Figure 2.5 shows the power module component used in this experimental setup.

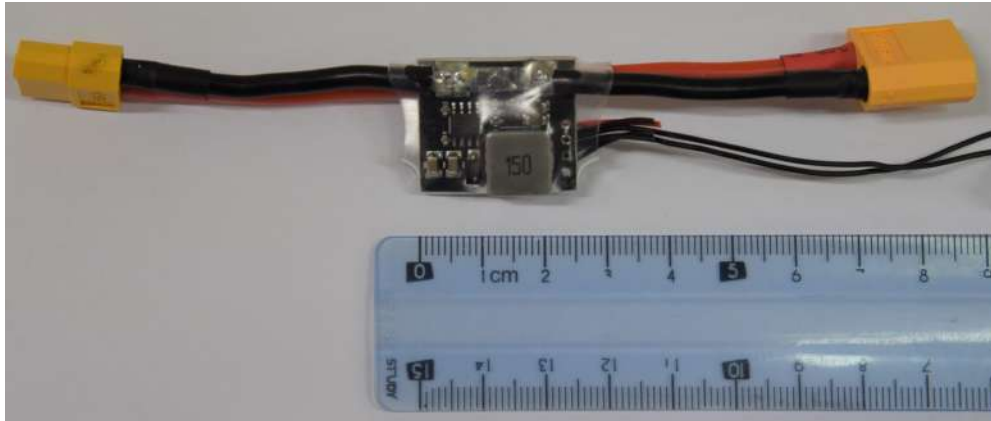


Figure 2.5: Power module, used as a voltage and current sensor.

Acquisition and control board

The acquisition and motor control board is a low cost *Arduino Uno*, which is a microcontroller board based on the *Atmega328P* chip. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator and a USB connection. It can be programmed directly from the computer. Figure 2.6 shows the acquisition board.

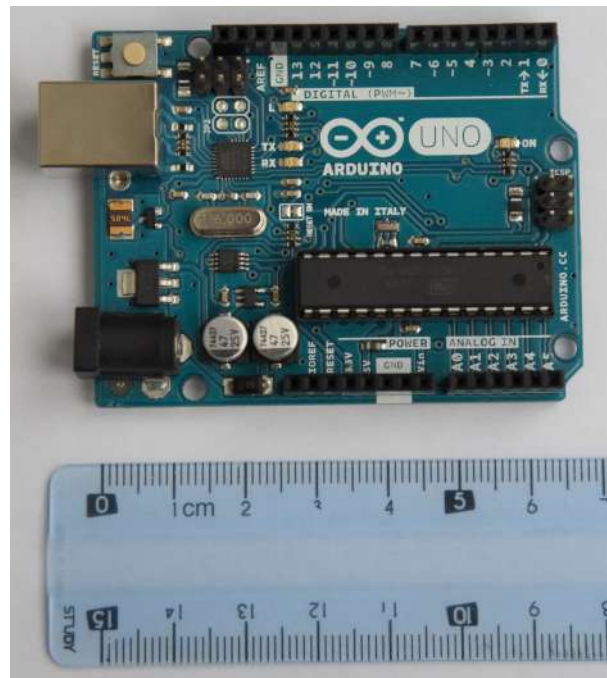


Figure 2.6: *Arduino* data acquisition and control board.

2.2.2 Software

MATLAB scripts have been developed for the operation of the motor test bed. The data acquisition script deals with the actual data acquisition. Some settings, namely the motor PWM signal (which is equivalent to the motor throttle set) can be changed in the *Arduino* code, which controls the motor and sends the recorded data over the serial port to the computer on which the *Arduino* is connected, data which is then read by *MATLAB*. Several live plots are made while the experiment is running: motor RPM, thrust, torque, mechanical power, current, voltage, electrical power, efficiency, percentage throttle and motor PWM. After the test has finished, all the raw data from the sensors, the calibrated measurements and other additional test information (environmental data, date and time of the test, motor and propeller model, etc.) are saved.

Another script deals with the data post-processing. Firstly, saved recorded measurements can be filtered, and non-physical spikes are removed. Then, the single experiments at different throttle settings are loaded, and for each experiment the user on the recorded motor RPM plot interactively selects a time interval from which take the measurements for the following data processing. Then all the relevant estimations and plots are made, as they will be presented in the following sections. If manufacturer data is provided the results are compared with those reference values. Results are saved in a *.txt* files and in figures generated by *MATLAB*.

2.2.3 Calibration

Calibration is made via a dedicated *MATLAB* script which, given a recorded data file of the calibration experiment and the value of the known calibration quantity, calculates the calibration constant, or checks it against an already obtained value.

Calibrations for the three load cells have been done on each cell individually, applying a known mass of 250 g and then checking the result against another known mass of 1 kg. Calibrations for the voltage and current sensors have been done using multimeters, with different conditions, also for verification purposes. Figure 2.7 shows, just as an example, the calibration of the thrust load cell. The test bed is put vertically because the calibration mass has to act in the same direction as the thrust generated by the motor.

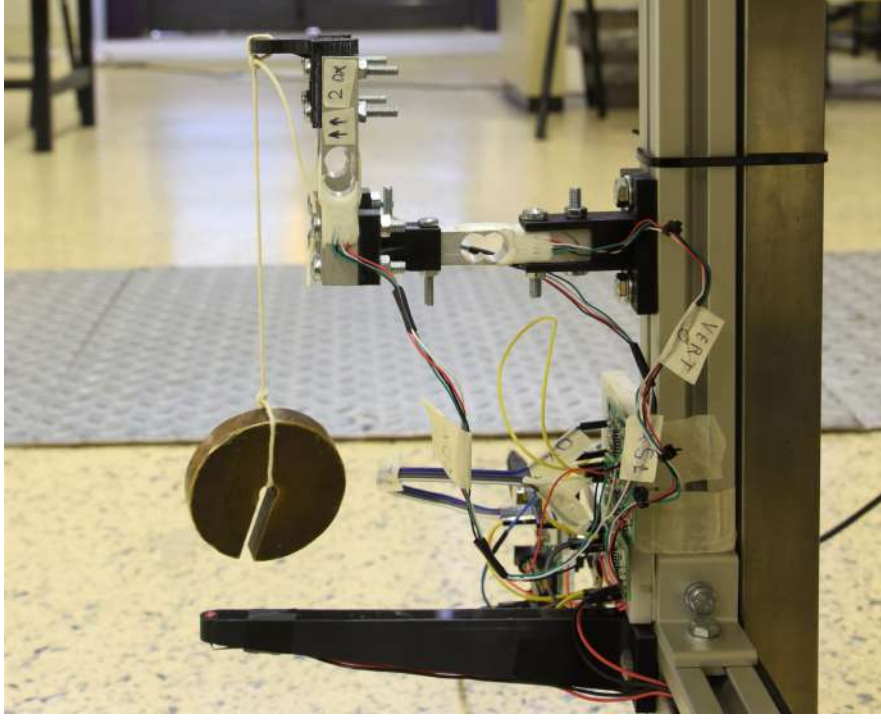


Figure 2.7: Calibration of the thrust load cell.

2.3 Static performance identification

As derived in Section 2.1, the thrust, torque and power coefficients depend on the square (the first two) and on the cube (the last) of the angular velocity. Hence, by measuring those four quantities the performance coefficients can be calculated. Another important relationship which is useful to gather experimentally is the one between the throttle percentage set (which translates in a PWM pulse width sent to the motor) and the motor angular velocity, which should theoretically be linear.

To summarize, the four relationships which are determined experimentally are the following:

$$T = \rho A R^2 C_T \Omega^2 = K_T \Omega^2, \quad (2.13)$$

$$Q = \rho A R^3 C_Q \Omega^2 = K_Q \Omega^2, \quad (2.14)$$

$$P = \rho A R^3 C_P \Omega^3 = K_P \Omega^3, \quad (2.15)$$

$$\Omega = m \cdot Thr_{\%} + q, \quad (2.16)$$

where $Thr_{\%}$ is the percentage throttle set.

In order to deal with the intrinsic uncertainties of the experimental identification, it is useful to gather a number of different samples, and then to estimate the required coefficients by using a least square fit. In this technique the overall solution to the over-determined system minimizes the sum of the squares of the residuals for each equation, a residual being the difference between an observed value and a fitted value calculated from the identified model.

In a general sense, an over-determined linear system of equations can be written as

$$\mathbf{X}\boldsymbol{\beta} = \mathbf{y}, \quad (2.17)$$

with

$$\mathbf{X} = \begin{bmatrix} X_{11} & X_{12} & \dots & X_{1n} \\ X_{21} & X_{22} & \dots & X_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ X_{m1} & X_{m2} & \dots & X_{mn} \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}, \quad (2.18)$$

and $m > n$. The least square approach solves the problems by minimizing a cost function J :

$$\hat{\boldsymbol{\beta}} = \min_{\boldsymbol{\beta}} J(\boldsymbol{\beta}), \quad (2.19)$$

with the cost function defined as the sum of the squared residuals:

$$J(\boldsymbol{\beta}) = \sum_{i=1}^m \left| y_i - \sum_{j=1}^n X_{ij}\beta_j \right|^2. \quad (2.20)$$

It can be proven that this problem has a unique solution, which is given by equation (2.21), provided that the n columns of the matrix $\mathbf{X}$ are linearly independent:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}. \quad (2.21)$$

Hence, by performing various experiments at different throttle settings, the coefficients K_T , K_Q , K_P , m and q can be identified with the least squared approach. From the identified values, written with the hat $\widehat{(\cdot)}$, it is easy to obtain the thrust, torque and power coefficients:

$$\widehat{C_T} = \frac{\widehat{K_T}}{\rho A R^2}, \quad \widehat{C_Q} = \frac{\widehat{K_Q}}{\rho A R^3}, \quad \widehat{C_P} = \frac{\widehat{K_P}}{\rho A R^3}. \quad (2.22)$$

For example, if m experiments are performed, the over-determined system which describes the relationship between throttle and angular velocity is the following:

$$\begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \vdots \\ \Omega_m^2 \end{bmatrix} \cdot \widehat{K_T} = \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_m \end{bmatrix}. \quad (2.23)$$

Thus the coefficient $\widehat{K}_T$ can be calculated using equation (2.21), with

$$\mathbf{X} = \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \vdots \\ \Omega_m^2 \end{bmatrix} \quad \text{and} \quad \mathbf{y} = \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_m \end{bmatrix} \quad (2.24)$$

and similarly for $\widehat{K}_Q$ and $\widehat{K}_P$, with the appropriate modification of matrices $\mathbf{X}$ and $\mathbf{y}$. The identification of the coefficients m and q of equation (2.16) leads to the following system:

$$\begin{bmatrix} Thr_{\%1} & 1 \\ Thr_{\%2} & 1 \\ \vdots & \vdots \\ Thr_{\%m} & 1 \end{bmatrix} \cdot \begin{bmatrix} \widehat{m} \\ \widehat{q} \end{bmatrix} = \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \vdots \\ \Omega_m \end{bmatrix} \quad (2.25)$$

with

$$\mathbf{X} = \begin{bmatrix} Thr_{\%1} & 1 \\ Thr_{\%2} & 1 \\ \vdots & \vdots \\ Thr_{\%m} & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{y} = \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \vdots \\ \Omega_m \end{bmatrix}. \quad (2.26)$$

2.4 Dynamic response identification

The dynamic response of the propulsive system was also obtained with the test bed. The interest was in identifying the relationship between the input throttle percentage and the propeller angular velocity as a function of time. A grey box identification method was used, assuming a first order transfer function between the throttle percentage set $Thr_{\%}$ and the angular velocity Ω of the type

$$G(s) = \frac{\Omega(s)}{Thr_{\%}(s)} = \frac{\mu}{1 + s\tau} \quad (2.27)$$

Step like inputs were set to the throttle, and the propulsive unit response recorded. However, direct measurement of the angular velocity with the tachometer were not usable in the identification process, since in the short transient phase they were highly discontinuous. Hence, it was decided to derive the motor angular velocity from the measurement of thrust and from the identified value of the $\widehat{K}_T$ coefficient, using the inverse of equation (2.13):

$$\Omega = \sqrt{\frac{T}{\widehat{K}_T}} \quad (2.28)$$

Finally, steps that started from 0% throttle were not used, since around the low throttle regions non linear behaviours could be observed, and the identification did not provide satisfactory results. Hence, in an effort to remain around a linearity region of the motor's dynamic response, small steps of 10% throttle were set, starting from values of throttle around the hover region of main interest during normal flight, namely from 40% to 70% .

2.5 Experimental Results

In this section, the experimental results are presented. First, a validation against already known data was performed. Then, the two motor types used in the VTOL UAV are tested.

2.5.1 Validation

In order to validate the overall design and software of the test bed, a smaller motor was tested, using a different test bed present in the laboratory. This test bench was different from the developed one both in the hardware and software parts. The motor was a *TMotor F20II KV3250*, together with a 3 bladed propeller of 3 inch diameter and with a 4 inch pitch. Static tests data were collected, with measurements of thrust and angular velocity, from 9% throttle all the way up to 99% throttle.

Figure 2.8 shows the motor angular velocity against throttle, whereas Figure 2.9 shows the thrust against motor angular velocity curve.

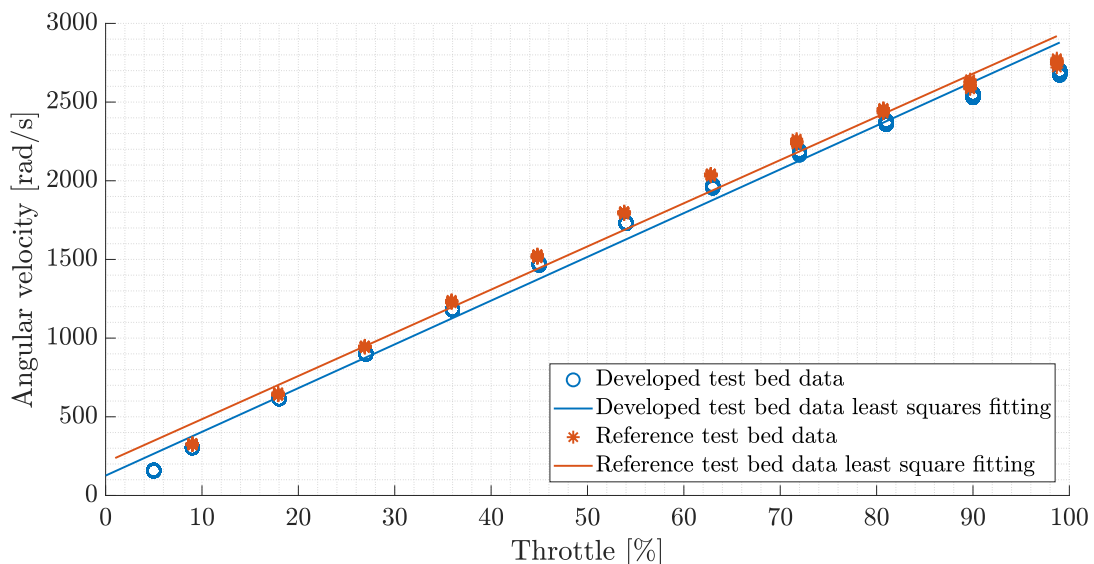


Figure 2.8: Motor angular velocity against throttle, motor test bed validation, *TMotor F20II KV3250*.

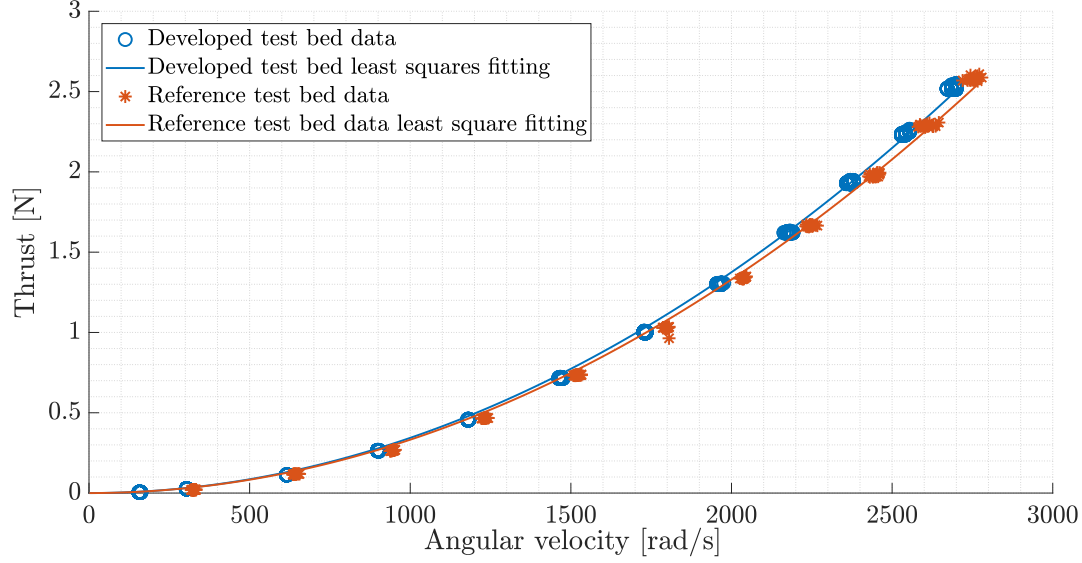


Figure 2.9: Motor thrust against angular velocity, motor test bed validation, *TMotor F20II KV3250*.

The thrust coefficient C_T obtained with the developed test bed and the reference one are, respectively, 0.00559 and 0.00540. The motor angular velocity as function of throttle percentage with the developed test bed and the reference one are, respectively, $\Omega = 27.8 \cdot Thr\% + 126$ and $\Omega = 27.4 \cdot Thr\% + 210$.

As it can be seen, the results between the two platforms are extremely similar, and hence the developed design is effectively validated.

2.5.2 Experimental results

Four tests have been conducted for each motor:

- using the laboratory DC power supply, at a voltage equal to the one used by the manufacturer for their tests and published in the motor data sheet, in order to compare the obtained results with the declared performance;
- using the laboratory DC power supply, at 14.8 V, which is the rated nominal voltage of the battery;
- using the VTOL's battery (8500 mAh, 50C, 4S1P battery) at 100% state of charge;
- using the VTOL's battery (8500 mAh, 50C, 4S1P battery) at 50% state of charge.

For the dynamic identification, only the second test has been performed. In Appendix B the results for the first test are presented, whereas in the following

sections the results from the last test will be shown. Note that a voltage drop at higher throttles is experienced when powering the motor from the power supply. This voltage drop was compensated in the first test, but not in the second one. When operating from the battery, such voltage drop is not seen.

The plots of thrust against angular velocity, angular velocity against throttle percentage set, electrical and mechanical power against angular velocity and the motor transient response are presented in Figure 2.10 to Figure 2.17, whereas the identified static and dynamic coefficients are shown in Table 2.1 to Table 2.4. For the coefficients C_Q and C_P , "direct fitting" means that the coefficient is derived from a direct least square fitting of the experimental data, whereas "theoretical derivation" means that the coefficient is calculated from the C_T coefficient, using equation (2.9) and equation (2.10).

Motor for forward flight

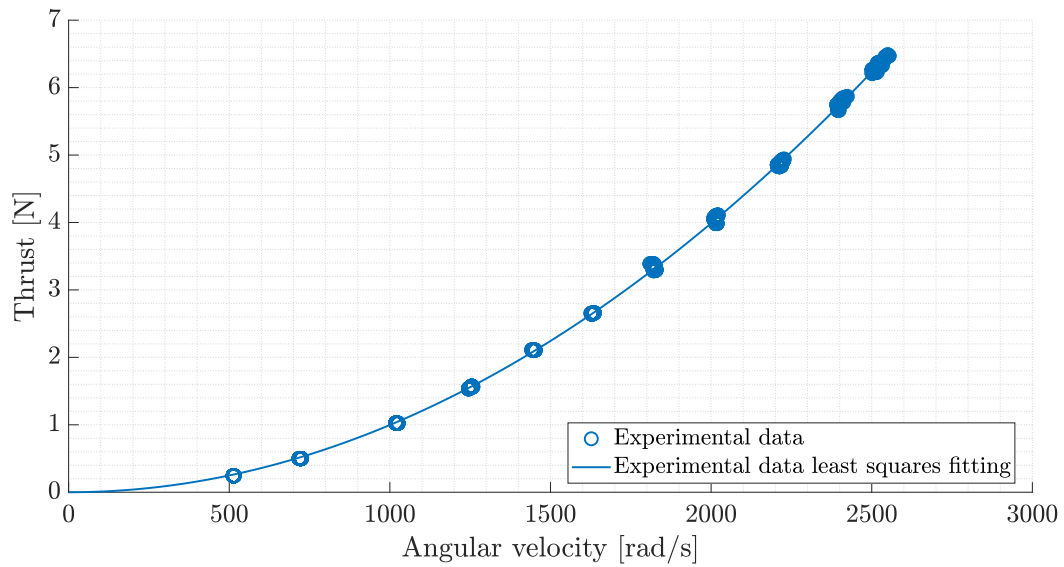


Figure 2.10: Thrust against angular velocity, motor powered from battery at 50% SOC, motor for forward flight *KDE2304XF-2350*, 5x4.5 inch, 2 blades bull-nose propeller.

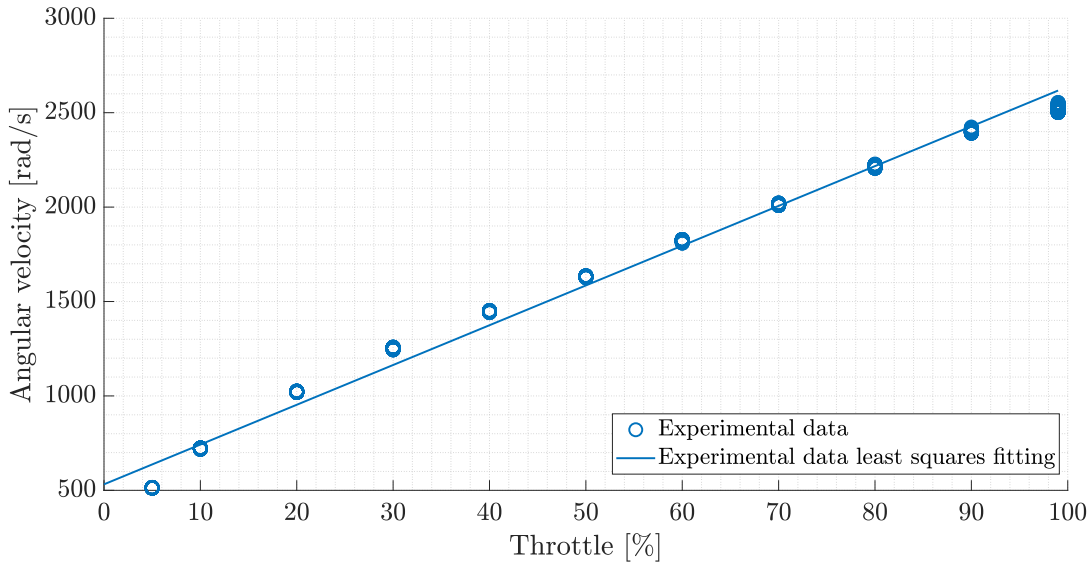


Figure 2.11: Angular velocity against throttle, motor powered from battery at 50% SOC, motor for forward flight *KDE2304XF-2350*, 5x4.5 inch, 2 blades bullnose propeller.

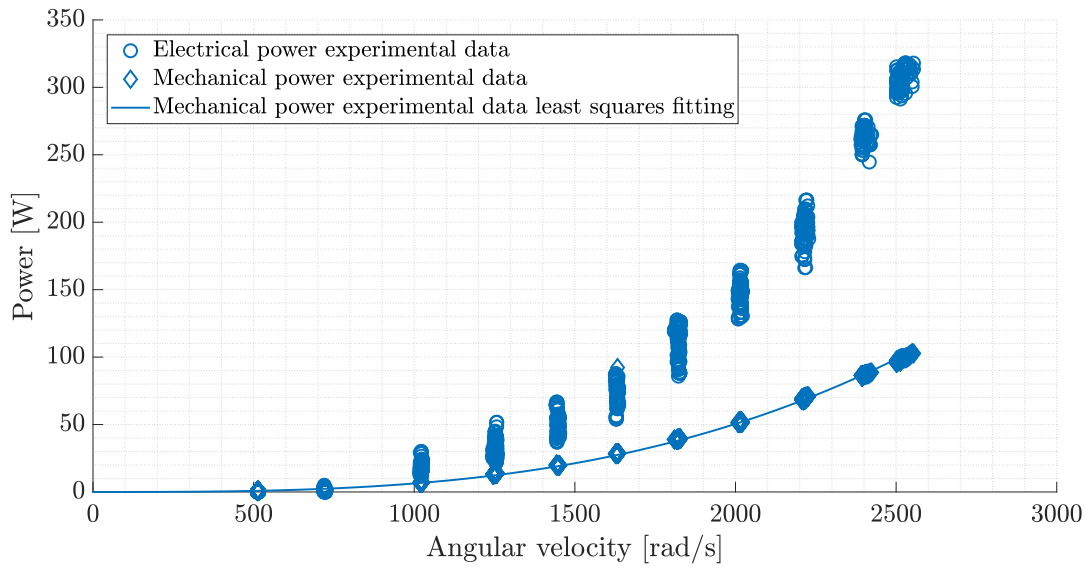


Figure 2.12: Electrical and mechanical power against angular velocity, motor powered from battery at 50% SOC, motor for forward flight *KDE2304XF-2350*, 5x4.5 inch, 2 blades bullnose propeller.

Table 2.1: Motor static coefficients identification results, motor for forward flight *KDE2304XF-2350*, 5x4.5 inch, 2 blades bullnose propeller.

	Direct fitting	Theoretical derivation
Thrust coefficient C_T	0.0163	-
Torque coefficient C_Q	0.00163	0.00147
Power coefficient C_P	0.00162	0.00147

The motor angular velocity as function of throttle percentage is:

$$\Omega = 21.1 \cdot Thr\% + 532. \quad (2.29)$$

Figure 2.13 shows, as an example, the measured and identified dynamic response of the motor for a 10% throttle step starting from 50% throttle. Table 2.2 shows the summary of the tests done and the identified gain μ and time constant τ of the first order transfer function from throttle percentage to motor angular velocity (see equation (2.27)).

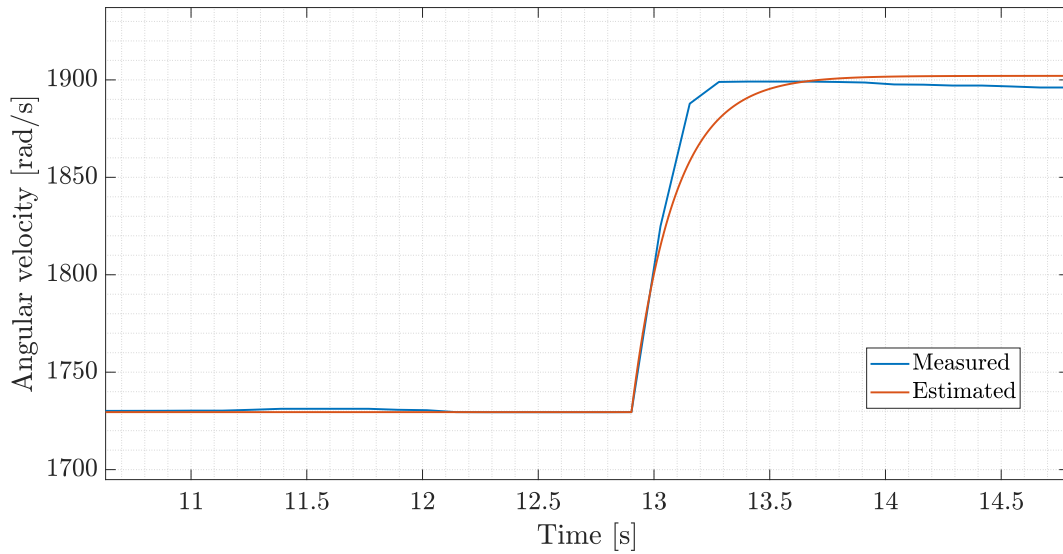


Figure 2.13: Motor dynamic response, measured and identified response, motor powered from DC power supply at 14.8 V, motor for forward flight *KDE2304XF-2350*, 5x4.5 inch, 2 blades bullnose propeller.

Table 2.2: Motor dynamic response identification results, motor for forward flight *KDE2304XF-2350*, 5x4.5 inch, 2 blades bullnose propeller.

Start throttle	End throttle	Gain μ	Time constant τ	Fit percentage
40%	50%	18.9	0.24	79%
50%	60%	17.3	0.18	83%
60%	70%	17.8	0.25	80%
70%	80%	16.7	0.25	81%

As it can be seen, the fits are quite satisfactory; the various responses share a really similar time constant, but a slightly different gain. This can be explained by the fact that the angular velocity against throttle percentage relationship is linear, but not quite exactly (see Figure 2.11).

Motor for vertical flight

For this motor, with the battery as power source, the maximum throttle had to be limited to 70%, because above that limit the motor and its supporting structure started to vibrate excessively.

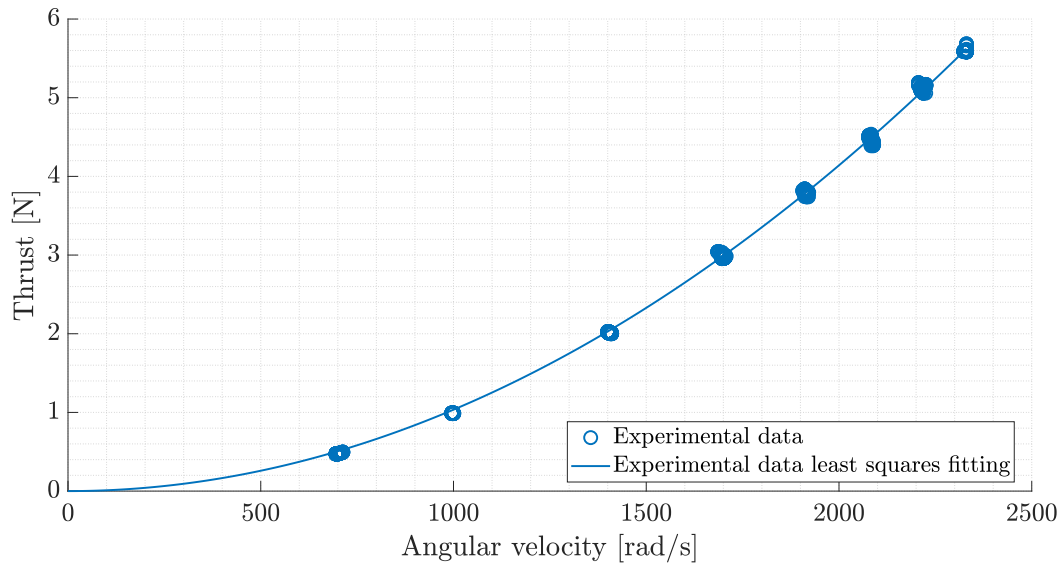


Figure 2.14: Thrust against angular velocity, motor powered from battery at 50% SOC, motor for vertical flight *KDE2315XF-2050*, 5x4.5 inch, 2 blades bullnose propeller.

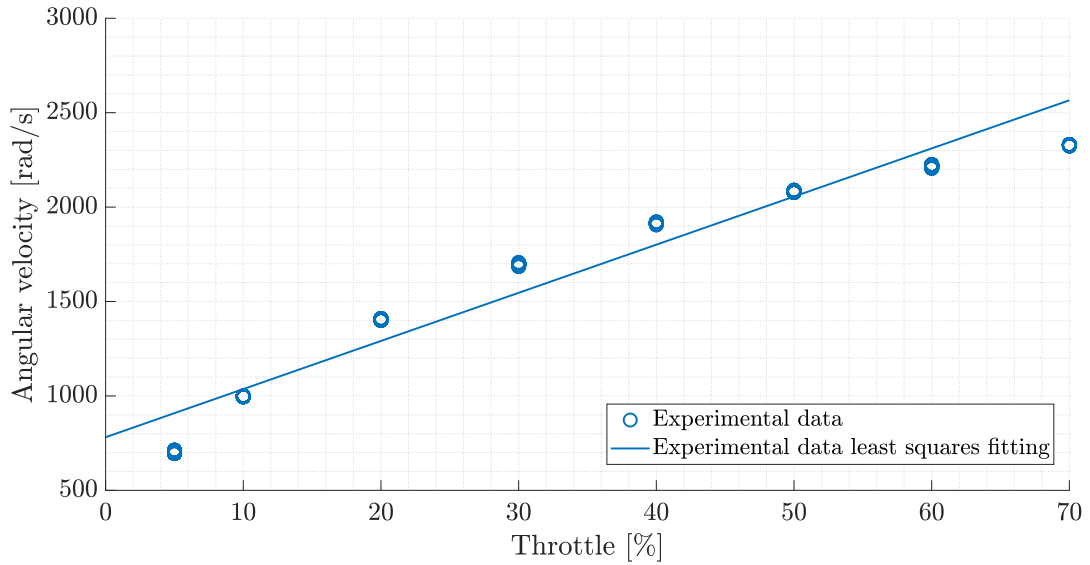


Figure 2.15: Angular velocity against throttle, motor powered from battery at 50% SOC, motor for vertical flight *KDE2315XF-2050*, 5x4.5 inch, 2 blades bullnose propeller.

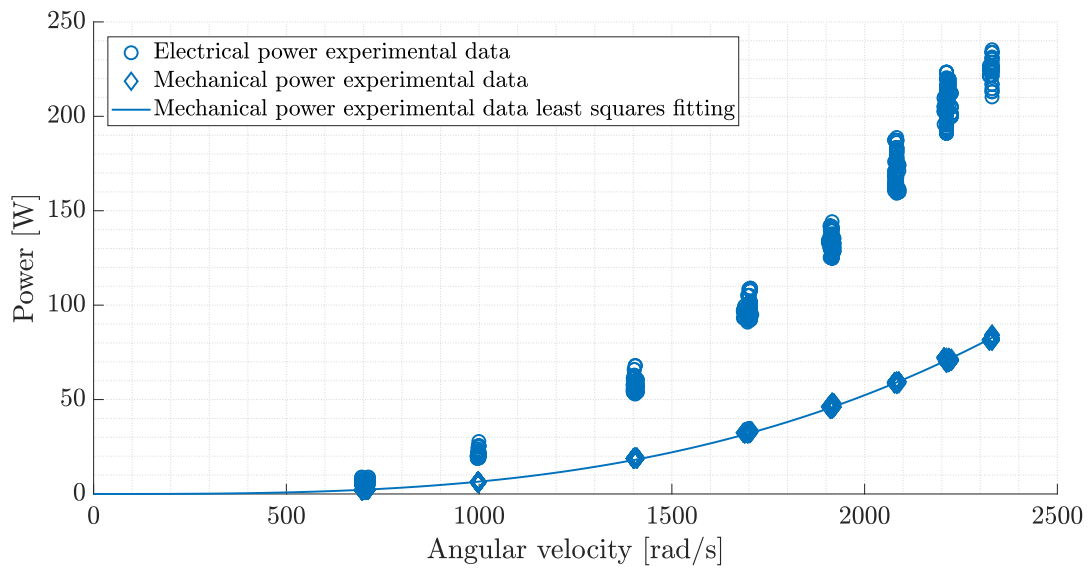


Figure 2.16: Electrical and mechanical power against angular velocity, motor powered from battery at 50% SOC, motor for vertical flight *KDE2315XF-2050*, 5x4.5 inch, 2 blades bullnose propeller.

Table 2.3: Motor static coefficients identification results, motor for vertical flight *KDE2315XF-2050*, 5x4.5 inch, 2 blades bullnose propeller.

	Direct fitting	Theoretical derivation
Thrust coefficient C_T	0.0168	-
Torque coefficient C_Q	0.00168	0.00154
Power coefficient C_P	0.00168	0.00154

The motor angular velocity as function of throttle percentage is:

$$\Omega = 25.6 \cdot Thr_{\%} + 776. \quad (2.30)$$

Figure 2.17 shows, as an example, the measured and identified dynamic response of the motor for a 10% throttle step starting from 70% throttle. Table 2.4 shows the summary of the tests done and the identified gain μ and time constant τ of the first order transfer function from throttle percentage to motor angular velocity (see equation (2.27)).

Similar conclusions to the ones explained in the previous section can be drawn.

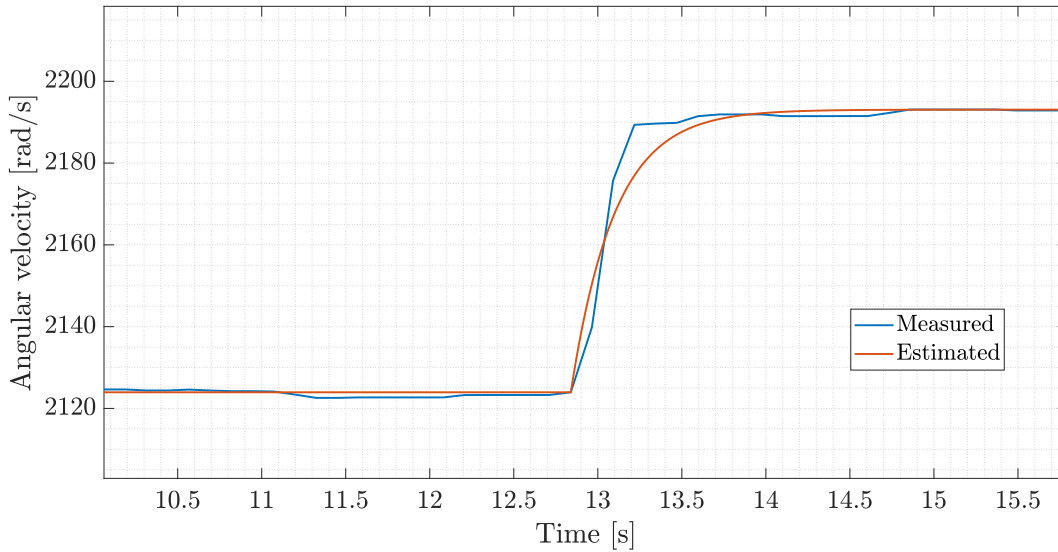


Figure 2.17: Motor dynamic response, measured and identified response, motor powered from DC power supply at 14.8 V, motor for vertical flight *KDE2315XF-2050*, 5x4.5 inch, 2 blades bullnose propeller.

Table 2.4: Motor dynamic response identification results, motor for vertical flight *KDE2315XF-2050*, 5x4.5 inch, 2 blades bullnose propeller.

Start throttle	End throttle	Gain μ	Time constant τ	Fit percentage
40%	50%	15.8	0.27	86%
50%	60%	11.1	0.23	85%
60%	70%	9.2	0.23	88%
70%	80%	6.9	0.26	87%

Chapter 3

Fixed wing VTOL UAV dynamics

A dynamic model of the VTOL UAV for all its different flight modes is crucial for the development of a flight simulator, which in turn is needed for building and studying the UAV's control laws. Since the aircraft operates in different flight modes, they all have to be implemented in the model. Figure 3.1 shows a typical mission profile for the aircraft, which was at the bases of its design.

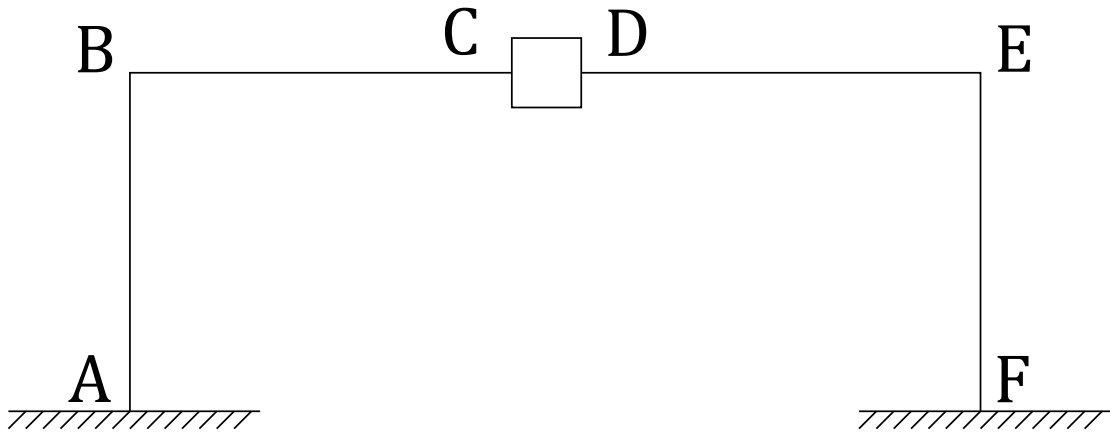


Figure 3.1: Typical VTOL UAV mission profile, from [1]. Phase A-B is a vertical take off and climb, B-C a cruise in forward flight mode, C-D is hovering phase using VTOL engines, D-E is a cruise in forward flight mode and E-F is vertical descent and landing.

During the terminal phases (take off, landing) and hovering the aircraft effectively flies like a multicopter; during the cruise phase the eVTOL is a conventional wing and tail aircraft, whereas during the transition phase it has the characteristics of both types. Modelling and control for either the multicopter-like or airplane-like behaviour is usually well developed and available in current literature; much less work is present for the transition phase though.

This chapter is organized as follows. In the first section, an introduction to the frames of reference used for the description of aircraft's dynamics and the main hypothesis used are presented. Then, the two most used aircraft's attitude description are presented: Euler angles and quaternions. In the third and fourth sections, the kinematic and dynamic equations for the aircraft are derived, starting from first principles. In the last section, the external forces and moments which act on the aircraft are thoroughly described, for each of the three flight modes: conventional aircraft, multicopter and transition phase. The *Simulink* flight simulator will be described in Chapter 4.

3.1 Frame of references and main hypothesis

In order to write the aircraft's equations of motion it is important to firstly define the various frames of reference which will be useful in the derivation of the equations. The main ones are the following: the navigation frame; the local vertical reference frame (or NED, North-East-Down frame); the body frame, the stability frame and the wind frame. All of those frames are right-handed. Figure 3.2 graphically shows those frames, which will now be described. Vectors expressed in a certain coordinate system will be denoted with an apex: $\mathbf{v}^{\mathcal{N}}$ represent vector $\mathbf{v}$ in the navigation frame, $\mathbf{v}^{\mathcal{B}}$ represent vector $\mathbf{v}$ in the body frame, and so on.

Navigation frame $\overline{\mathcal{N}}$

The navigation frame has origin in a generic point N on Earth's surface (usually the aircraft's starting position), and it can be approximately considered as an inertial frame. The first unit vector $\overline{\mathbf{n}}_1$ is directed towards the local North pole; $\overline{\mathbf{n}}_2$ points towards the local East and $\overline{\mathbf{n}}_3$ points to the centre of the Earth.

Local vertical reference frame (NED) $\mathcal{N}$

The local vertical reference frame (or NED, North-East-Down frame) has origin in the centre of gravity of the aircraft and thus it is a non-inertial frame. $\mathbf{n}_1$ points to the local North pole; $\mathbf{n}_2$ is aligned with the local East and $\mathbf{n}_3$ points to the centre of the Earth.

Body frame $\mathcal{B}$

The body frame has origin in the centre of gravity of the aircraft and thus it is a non-inertial frame. $\mathbf{b}_1$ is aligned with the fuselage and points forward, towards the nose of the aircraft; $\mathbf{b}_2$ points to the right wing and $\mathbf{b}_3$ points to the aircraft's underside, according to the right-hand rule.

Stability frame $\mathcal{S}$

The stability frame has origin in the centre of gravity of the aircraft and thus it is a non-inertial frame. $\mathbf{s}_1$ points forward (towards the nose of the aircraft) and is aligned with the projection of the relative wind velocity in the aircraft's symmetry plane; $\mathbf{s}_2$ is aligned with $\mathbf{b}_2$ and $\mathbf{s}_3$ forms a right-hand triad with $\mathbf{s}_1$ and $\mathbf{s}_2$.

Wind frame $\mathcal{W}$

The wind frame has origin in the centre of gravity of the aircraft and thus it is a non-inertial frame. The unit vector $\mathbf{w}_1$ is aligned with the relative wind speed, but in opposite direction; $\mathbf{w}_3$ lies within the aircraft's longitudinal symmetry plane and is perpendicular to $\mathbf{w}_1$, pointing downwards; $\mathbf{w}_2$ forms a right-hand triad with $\mathbf{w}_1$ and $\mathbf{w}_3$.

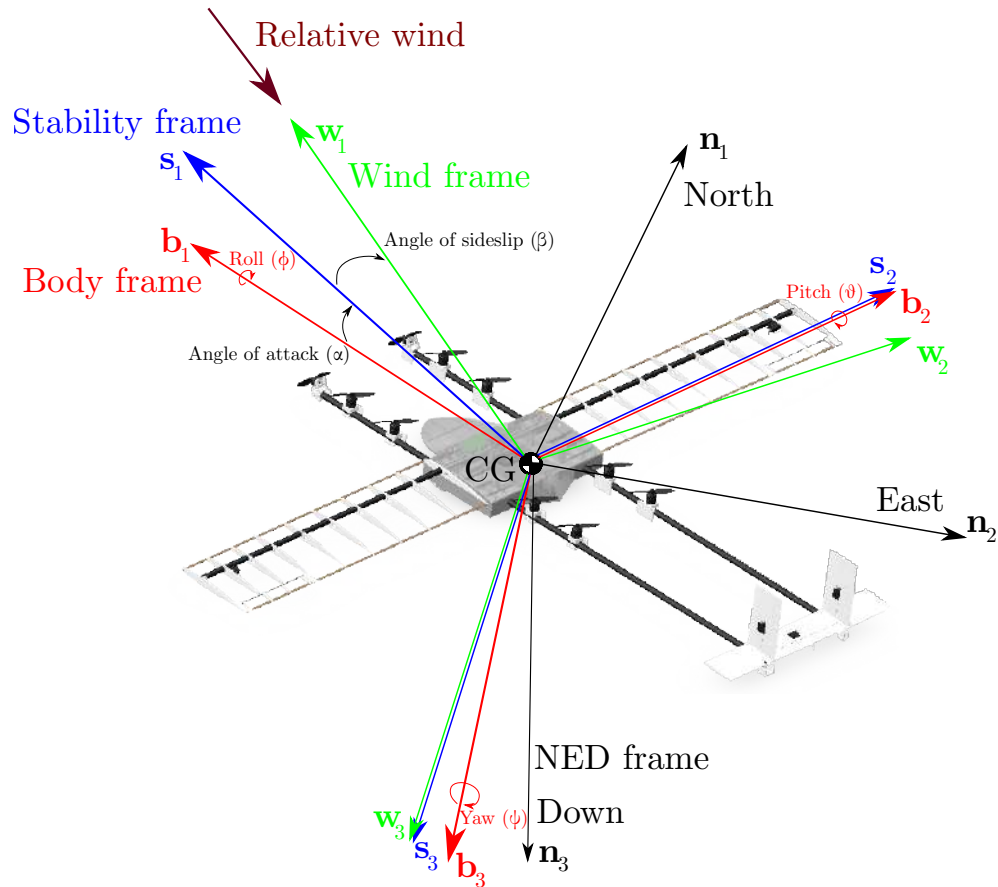


Figure 3.2: Frames of reference used for the description of aircraft dynamics.

The main hypothesis for the derivations which follow are:

- the aircraft is assumed to be a rigid body;
- the Earth is flat and non-rotating, and hence any frame attached to it is inertial.

3.2 Attitude descriptions

The aircraft's angular orientation in space is defined as its attitude. Two main sets of parameters can be used to unambiguously define the aircraft's attitude using the least possible number of parameters: the Euler angles representation and the quaternions representation. Euler angles have the advantage of being immediately related to the aircraft's position in space, but face the problem of gimbal lock. Quaternions, on the other hand, do not have an immediate physical representation, but can be used to describe the aircraft's attitude without discontinuities, hence eliminating the gimbal lock problem.

3.2.1 Euler angles

In order to obtain a certain attitude, vectors can be rotated in space any number of times, around any arbitrary direction. However, any arbitrary attitude can also be achieved by the combination of three consecutive rotations around each axes of an orthogonal frame of reference. By considering a generic orthogonal frame of reference, composed of an x , y and z axes, the rotation about each axes can be expressed by the following rotation matrices $\mathbf{R}_x(\phi)$, $\mathbf{R}_y(\theta)$, $\mathbf{R}_z(\psi)$:

$$\mathbf{R}_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix}, \quad \mathbf{R}_y(\theta) = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix}, \quad (3.1)$$

$$\mathbf{R}_z(\psi) = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

and the generic equation transforming a vector $\mathbf{v}^{\mathcal{I}}$ in the original frame $\mathcal{I}$ to the rotated frame $\mathcal{J}$, $\mathbf{v}^{\mathcal{J}}$, with the rotation matrix $\mathbf{R}^{\mathcal{I} \rightarrow \mathcal{J}}$ from $\mathcal{I}$ to $\mathcal{J}$:

$$\mathbf{v}^{\mathcal{J}} = \mathbf{R}^{\mathcal{I} \rightarrow \mathcal{J}} \mathbf{v}^{\mathcal{I}}. \quad (3.2)$$

In aircraft dynamics the generic attitude of the vehicle is usually obtained by the so-called rotation sequence 321, which means that the attitude of the body axes (always aligned with the aircraft), with respect to a non-rotating frame of reference, is obtained by a first rotation about the z axis of an angle ψ , then a

rotation about the y axis of an angle θ , and finally a rotation about the x axes of an angle ϕ . The final rotation matrix is obtained by the ordered multiplication of the previous rotation matrices, and has the following expression:

$$\mathbf{R}_{321}^{\mathcal{I} \rightarrow \mathcal{B}} = \begin{bmatrix} \cos \psi \cos \theta & \sin \psi \cos \theta & -\sin \theta \\ \cos \psi \sin \theta \sin \phi - \sin \psi \cos \phi & \sin \psi \sin \theta \sin \phi + \cos \psi \cos \phi & \cos \theta \sin \phi \\ \cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi & \sin \psi \sin \theta \sin \phi - \cos \psi \sin \phi & \cos \theta \cos \phi \end{bmatrix}. \quad (3.3)$$

The Euler angles ϕ , θ and ψ are grouped in the vector $\mathbf{e}_{321}$, which univocally describes the aircraft's attitude in space. The bank angle ϕ is the angle between the aircraft's y body axis $\mathbf{b}_2$ and the horizontal plane of the NED frame; the pitch angle θ is the angle between the aircraft's x body axis $\mathbf{b}_1$ and the horizontal plane of the NED frame; the yaw angle ψ is the angle between the aircraft's x body axis $\mathbf{b}_1$ and the North direction ($\mathbf{n}_1$) of the NED frame. The angular velocities measured in body frame $\boldsymbol{\omega}^{\mathcal{B}}$ are the roll rate p , pitch rate q and yaw rate r :

$$\mathbf{e}_{321} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, \quad \boldsymbol{\omega}^{\mathcal{B}} = \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \quad (3.4)$$

The rate of change of Euler angles is related to the angular velocity in the body frame by the following equation:

$$\dot{\mathbf{e}}_{321} = \mathcal{S}_{321}^{\mathcal{B}^{-1}} \boldsymbol{\omega}^{\mathcal{B}}, \quad (3.5)$$

with $\mathcal{S}_{321}^{\mathcal{B}^{-1}}$ being the inverse of matrix $\mathcal{S}_{321}^{\mathcal{B}}$, whose expressions are given by:

$$\mathcal{S}_{321}^{\mathcal{B}} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \cos \theta \sin \phi \\ 0 & -\sin \phi & \cos \theta \cos \phi \end{bmatrix}, \quad \mathcal{S}_{321}^{\mathcal{B}^{-1}} = \begin{bmatrix} 1 & \tan \theta \sin \phi & \tan \theta \cos \phi \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sec \theta \sin \phi & \sec \theta \cos \phi \end{bmatrix}. \quad (3.6)$$

It can be seen that matrix $\mathcal{S}_{321}^{\mathcal{B}^{-1}}$ is singular for pitch angles of $\pm 90^\circ$: the angular rate in terms of Earth axis is not defined for those values of pitch. This is the so-called gimbal lock.

3.2.2 Quaternions

Quaternions representation of attitude is useful because it avoids the problem of gimbal lock. With reference to [25], a quaternion is a four-dimensional complex number that can be used to describe the attitude of a rigid body in space, as the Euler angles do as well. By considering two frames $\mathcal{I}$ and $\mathcal{J}$, any arbitrary orientation of frame $\mathcal{J}$ with respect to frame $\mathcal{I}$ can be achieved through a rotation of an angle θ around an axis ${}^{\mathcal{I}}\mathbf{r}$ defined in frame $\mathcal{I}$. By denoting the components

of such vector ${}^{\mathcal{I}}\mathbf{r}$ as ${}^{\mathcal{I}}r_x$, ${}^{\mathcal{I}}r_y$ and ${}^{\mathcal{I}}r_z$, the quaternion describing the orientation of frame $\mathcal{J}$ relative to frame $\mathcal{I}$ is denoted as ${}^{\mathcal{I}}_{\mathcal{J}}\mathbf{q}$, where the notation follows the one adopted in [26]: a leading superscript denotes the frame with respect to which a certain quaternion orientation is referenced to, and a leading subscript denotes the actual frame of reference which orientation is being described. The quaternion components are defined as follows:

$${}^{\mathcal{I}}_{\mathcal{J}}\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ -{}^{\mathcal{I}}r_x \sin\left(\frac{\theta}{2}\right) \\ -{}^{\mathcal{I}}r_y \sin\left(\frac{\theta}{2}\right) \\ -{}^{\mathcal{I}}r_z \sin\left(\frac{\theta}{2}\right) \end{bmatrix}. \quad (3.7)$$

It is usually required for quaternions to be normalized to unit length, in order to arithmetically operate on them.

The quaternion conjugate, denoted as $(\cdot)^*$, is used to swap the frames described by an orientation. To follow the previous example, the orientation of frame $\mathcal{I}$ with respect to frame $\mathcal{J}$, described by the quaternion ${}^{\mathcal{J}}_{\mathcal{I}}\mathbf{q}$, can be expressed by the conjugate of quaternion ${}^{\mathcal{I}}_{\mathcal{J}}\mathbf{q}$, defined as:

$${}^{\mathcal{J}}_{\mathcal{I}}\mathbf{q} = {}^{\mathcal{I}}_{\mathcal{J}}\mathbf{q}^* = \begin{bmatrix} q_1 \\ -q_2 \\ -q_3 \\ -q_4 \end{bmatrix}. \quad (3.8)$$

The quaternion product, denoted as $\otimes$, is used to describe successive rotations. The orientation of a frame $\mathcal{K}$ with respect to a frame $\mathcal{I}$, described by the quaternion ${}^{\mathcal{I}}_{\mathcal{K}}\mathbf{q}$, composed of two successive rotation involving an intermediate frame $\mathcal{J}$, described by the quaternions ${}^{\mathcal{I}}_{\mathcal{J}}\mathbf{q}$ and ${}^{\mathcal{J}}_{\mathcal{K}}\mathbf{q}$, can be expressed as the quaternion product of the latter two quaternions:

$${}^{\mathcal{I}}_{\mathcal{K}}\mathbf{q} = {}^{\mathcal{J}}_{\mathcal{K}}\mathbf{q} \otimes {}^{\mathcal{I}}_{\mathcal{J}}\mathbf{q}. \quad (3.9)$$

The quaternion product $\otimes$ between two quaternions $\mathbf{a}$ and $\mathbf{b}$ is determined using the Hamilton rule, as expressed by equation (3.10), and is not commutative:

$$\mathbf{a} \otimes \mathbf{b} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \otimes \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = \begin{bmatrix} a_1b_1 - a_2b_2 - a_3b_3 - a_4b_4 \\ a_1b_2 + a_2b_1 + a_3b_4 - a_4b_3 \\ a_1b_3 - a_2b_4 + a_3b_1 + a_4b_2 \\ a_1b_4 + a_2b_3 - a_3b_2 + a_4b_1 \end{bmatrix}. \quad (3.10)$$

A three dimensional vector $\mathbf{v}$ can be rotated from a frame $\mathcal{I}$ to a frame $\mathcal{J}$ by using equation (3.11) and by appending a zero as a first element of the vector, to make it dimensionally consistent with quaternions:

$${}^{\mathcal{J}}\mathbf{v} = {}^{\mathcal{I}}_{\mathcal{J}}\mathbf{q} \otimes {}^{\mathcal{I}}\mathbf{v} \otimes {}^{\mathcal{I}}_{\mathcal{J}}\mathbf{q}^*. \quad (3.11)$$

The rotation matrix from frame $\mathcal{I}$ to frame $\mathcal{J}$ (the analogous of the one expressed in equation (3.3) in terms of Euler angles) has the following expression:

$$\mathbf{R}_{321}^{\mathcal{I} \rightarrow \mathcal{B}} = \begin{bmatrix} 2q_1^2 - 1 + 2q_2^2 & 2(q_2q_3 + q_1q_4) & 2(q_2q_4 - q_1q_3) \\ 2(q_2q_3 - q_1q_4) & 2q_1^2 - 1 + 2q_3^2 & 2(q_3q_4 + q_1q_2) \\ 2(q_2q_4 + q_1q_3) & 2(q_3q_4 - q_1q_2) & 2q_1^2 - 1 + 2q_4^2 \end{bmatrix}. \quad (3.12)$$

The Euler angles ϕ , θ and ψ can be obtained from the elements of the quaternion expressing the aircraft's orientation with respect to a non-rotating frame ${}_{\mathcal{J}}^{\mathcal{I}}\mathbf{q} = [q_1 \ q_2 \ q_3 \ q_4]^T$:

$$\begin{aligned} \phi &= \tan^{-1} \left(\frac{2q_3q_4 - 2q_1q_2}{2q_1^2 - 1 + 2q_4^2} \right), \\ \theta &= -\sin^{-1} (2q_2q_4 + 2q_1q_3), \\ \psi &= \tan^{-1} \left(\frac{2q_2q_3 - 2q_1q_4}{2q_1^2 - 1 + 2q_2^2} \right). \end{aligned} \quad (3.13)$$

Finally, the quaternion derivative of the rate of change of the orientation of the body frame relative to the NED frame ${}_{\mathcal{B}}^{\mathcal{N}}\dot{\mathbf{q}}$ can be related to the vector of the angular velocities in body frame $\boldsymbol{\omega}^{\mathcal{B}}$ by the following equation:

$${}_{\mathcal{B}}^{\mathcal{N}}\dot{\mathbf{q}} = \frac{1}{2} {}_{\mathcal{B}}^{\mathcal{N}}\mathbf{q} \otimes \boldsymbol{\omega}^{\mathcal{B}}, \quad (3.14)$$

where $\boldsymbol{\omega}^{\mathcal{B}}$ has been appended with a zero at the beginning, to make it a four dimensional vector. Resolving equation (3.14) leads to

$${}_{\mathcal{B}}^{\mathcal{N}}\dot{\mathbf{q}} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}. \quad (3.15)$$

3.3 Kinematic equations

The aircraft's kinematic equations are used to express the aircraft's velocity and acceleration with respect to an inertial frame $\mathcal{I}$, and will be later used in the derivation of the vehicle's dynamics equations.

Considering a generic inertial frame $\mathcal{I}$ and the non-inertial body frame $\mathcal{B}$, and using the convention that the velocity (or acceleration) of a point P with respect to a frame $\mathcal{I}$ is written as $\mathbf{v}_{P/\mathcal{I}}$, with reference to Figure 3.3, the velocity of a generic point Q belonging to the aircraft, measured with respect to the inertial frame $\mathcal{I}$ can be expressed as

$$\mathbf{v}_{Q/\mathcal{I}} = \mathbf{v}_{P/\mathcal{I}} + \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}} \times \mathbf{r}_{PQ}, \quad (3.16)$$

where point P is another generic point belonging to the aircraft, origin of the body frame $\mathcal{B}$ (which here is kept unspecified for generality, but for later derivations will be identified as the centre of gravity of the aircraft), and $\mathbf{r}_{PQ}$ is the vector between point P and Q . Note that the term $\mathbf{v}_{Q/\mathcal{B}}$, which in general should be present, is null, since the aircraft is considered to be a rigid body.

The equation for the acceleration of point Q measured in the inertial frame $\mathcal{I}$ is found by calculating the derivative of equation (3.16) with respect to time:

$$\mathbf{a}_{Q/\mathcal{I}} = \frac{{}^{\mathcal{I}}d\mathbf{v}_{Q/\mathcal{I}}}{dt} = \frac{{}^{\mathcal{I}}d\mathbf{v}_{P/\mathcal{I}}}{dt} + \frac{{}^{\mathcal{I}}d\boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}}{dt} \times \mathbf{r}_{PQ} + \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}} \times \frac{{}^{\mathcal{I}}d\mathbf{r}_{PQ}}{dt}, \quad (3.17)$$

where $\frac{{}^{\mathcal{I}}d(\cdot)}{dt}$ represents the derivative of a vector with respect to frame $\mathcal{I}$.

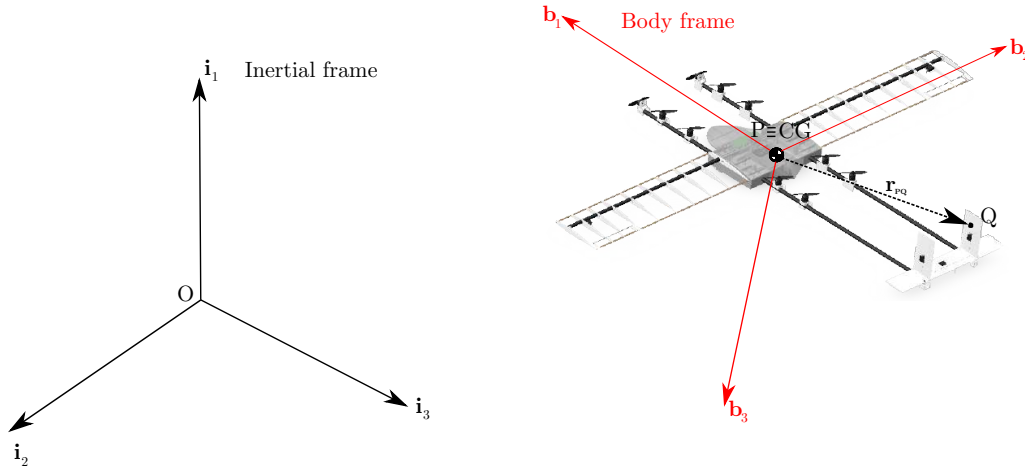


Figure 3.3: Schematics for the derivation of the kinematic equations.

All the derivatives in equation (3.17) measured in the inertial frame $\mathcal{I}$ have to be expressed in the body frame $\mathcal{B}$. This can be achieved with the moving axes theorem¹, and leads to the following expressions:

$$\begin{aligned} \frac{{}^{\mathcal{I}}d\mathbf{v}_{P/\mathcal{I}}}{dt} &= \frac{{}^{\mathcal{B}}d\mathbf{v}_{P/\mathcal{I}}}{dt} + \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}} \times \mathbf{v}_{P/\mathcal{I}}; \\ \frac{{}^{\mathcal{I}}d\boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}}{dt} &= \frac{{}^{\mathcal{B}}d\boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}}{dt} + \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}} \times \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}} = \frac{{}^{\mathcal{B}}d\boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}}{dt}; \\ \frac{{}^{\mathcal{I}}d\mathbf{r}_{PQ}}{dt} &= \frac{{}^{\mathcal{B}}d\mathbf{r}_{PQ}}{dt} + \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}} \times \mathbf{r}_{PQ} = \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}} \times \mathbf{r}_{PQ}; \end{aligned} \quad (3.18)$$

¹The moving axes theorem states that for any two frames of reference $\mathcal{I}$ and $\mathcal{J}$, using the notation previously introduced, for any generic vector $\mathbf{b}$ the following holds true:

$$\frac{{}^{\mathcal{I}}d\mathbf{b}}{dt} = \frac{{}^{\mathcal{J}}d\mathbf{b}}{dt} + \boldsymbol{\omega}_{\mathcal{J}/\mathcal{I}} \times \mathbf{b}.$$

where $\frac{{}^B d\mathbf{r}_{PQ}}{dt}$ is null since the aircraft is assumed to be a rigid body.

Substituting equation (3.18) in equation (3.17) one gets

$$\mathbf{a}_{Q/I} = \frac{{}^B d\mathbf{v}_{P/I}}{dt} + \boldsymbol{\omega}_{B/I} \times \mathbf{v}_{P/I} + \frac{{}^B d\boldsymbol{\omega}_{B/I}}{dt} \times \mathbf{r}_{PQ} + \boldsymbol{\omega}_{B/I} \times (\boldsymbol{\omega}_{B/I} \times \mathbf{r}_{PQ}). \quad (3.19)$$

3.4 Dynamic equations

3.4.1 Translational motion

By integrating Newton's first law for a particle of mass dm located in a generic point Q :

$$d\mathbf{F}_Q = \mathbf{a}_{Q/I} dm \quad (3.20)$$

over the whole aircraft volume $\mathcal{V}$, one gets the equation for translational motion for the whole aircraft:

$$\int_{\mathcal{V}} d\mathbf{F}_Q = \int_{\mathcal{V}} \mathbf{a}_{Q/I} dm. \quad (3.21)$$

By substituting equation (3.19) to the right hand side of equation (3.21), one gets

$$\begin{aligned} \mathbf{F} &= \int_{\mathcal{V}} \left[\frac{{}^B d\mathbf{v}_{P/I}}{dt} + \boldsymbol{\omega}_{B/I} \times \mathbf{v}_{P/I} + \frac{{}^B d\boldsymbol{\omega}_{B/I}}{dt} \times \mathbf{r}_{PQ} + \boldsymbol{\omega}_{B/I} \times (\boldsymbol{\omega}_{B/I} \times \mathbf{r}_{PQ}) \right] dm \\ &= \int_{\mathcal{V}} \left[\frac{{}^B d\mathbf{v}_{P/I}}{dt} + \frac{{}^B d\boldsymbol{\omega}_{B/I}}{dt} \times \mathbf{r}_{PQ} + \boldsymbol{\omega}_{B/I} \times (\mathbf{v}_{P/I} + \boldsymbol{\omega}_{B/I} \times \mathbf{r}_{PQ}) \right] dm \\ &= \int_{\mathcal{V}} \left[\frac{{}^B d\mathbf{v}_{P/I}}{dt} - \mathbf{r}_{PQ} \times \frac{{}^B d\boldsymbol{\omega}_{B/I}}{dt} + \boldsymbol{\omega}_{B/I} \times (\mathbf{v}_{P/I} - \mathbf{r}_{PQ} \times \boldsymbol{\omega}_{B/I}) \right] dm, \end{aligned} \quad (3.22)$$

where $\mathbf{F}$ is the sum of all the external forces acting on the aircraft. By observing that the quantities $\boldsymbol{\omega}_{B/I}$, $\mathbf{v}_{P/I}$ and their time derivatives do not depend on the volume integral, and that

$$m = \int_{\mathcal{V}} dm, \quad \mathbf{S}_P := \int_{\mathcal{V}} dm \mathbf{r}_{PQ} \times \quad \text{and} \quad \mathbf{S}_P^T := - \int_{\mathcal{V}} dm \mathbf{r}_{PQ} \times, \quad (3.23)$$

with m being the total mass of the aircraft and $\mathbf{S}_P$ the static moment tensor reduced to point P , which is skew-symmetric (and hence $\mathbf{S}_P^T = -\mathbf{S}_P$), equation (3.22) can be rewritten as

$$\mathbf{F} = m \frac{{}^B d\mathbf{v}_{P/I}}{dt} + \mathbf{S}_P^T \frac{{}^B d\boldsymbol{\omega}_{B/I}}{dt} + \boldsymbol{\omega}_{B/I} \times (m\mathbf{v}_{P/I} + \mathbf{S}_P^T \boldsymbol{\omega}_{B/I}). \quad (3.24)$$

3.4.2 Rotational motion

For what concerns rotational motion, Newton's second law states that for a particle of mass located in point Q , with moments taken with respect to point P , the following equation holds true:

$$d\mathbf{M}_P = \mathbf{r}_{PQ} \times d\mathbf{F}_Q = dm \mathbf{r}_{PQ} \times \mathbf{a}_{Q/I}. \quad (3.25)$$

Integrating equation (3.25) over the entire aircraft's volume $\mathcal{V}$, and substituting the expression of $\mathbf{a}_{Q/I}$ of equation (3.19) leads to

$$\begin{aligned} \mathbf{M}_P &= \int_{\mathcal{V}} dm \mathbf{r}_{PQ} \times \left[\frac{{}^B d\mathbf{v}_{P/I}}{dt} - \mathbf{r}_{PQ} \times \frac{{}^B d\boldsymbol{\omega}_{B/I}}{dt} + \boldsymbol{\omega}_{B/I} \times (\mathbf{v}_{P/I} - \mathbf{r}_{PQ} \times \boldsymbol{\omega}_{B/I}) \right] \\ &= \int_{\mathcal{V}} \left[dm \mathbf{r}_{PQ} \times \frac{{}^B d\mathbf{v}_{P/I}}{dt} + dm \mathbf{r}_{PQ} \times (\boldsymbol{\omega}_{B/I} \times \mathbf{v}_{P/I}) \right. \\ &\quad \left. - dm \mathbf{r}_{PQ} \times \left(\mathbf{r}_{PQ} \times \frac{{}^B d\boldsymbol{\omega}_{B/I}}{dt} \right) - dm \mathbf{r}_{PQ} \times (\boldsymbol{\omega}_{B/I} \times (\mathbf{r}_{PQ} \times \boldsymbol{\omega}_{B/I})) \right]. \end{aligned} \quad (3.26)$$

By defining the inertia tensor $\mathbf{J}_P$ about a generic pole P as

$$\mathbf{J}_P := - \int_{\mathcal{V}} dm \mathbf{r}_{PQ} \times \mathbf{r}_{PQ} \times \quad (3.27)$$

and by using the Jacobi identity² twice, so that

$$\begin{aligned} \mathbf{r}_{PQ} \times (\boldsymbol{\omega}_{B/I} \times \mathbf{v}_{P/I}) &= \boldsymbol{\omega}_{B/I} \times (\mathbf{r}_{PQ} \times \mathbf{v}_{P/I}) - \mathbf{v}_{P/I} \times (\mathbf{r}_{PQ} \times \boldsymbol{\omega}_{B/I}) \\ \mathbf{r}_{PQ} \times (\boldsymbol{\omega}_{B/I} \times (\mathbf{r}_{PQ} \times \boldsymbol{\omega}_{B/I})) &= -\boldsymbol{\omega}_{B/I} \times ((\mathbf{r}_{PQ} \times \boldsymbol{\omega}_{B/I}) \times \mathbf{r}_{PQ}) \\ &\quad - (\mathbf{r}_{PQ} \times \boldsymbol{\omega}_{B/I}) \times (\mathbf{r}_{PQ} \times \boldsymbol{\omega}_{B/I}) \\ &= \boldsymbol{\omega}_{B/I} \times (\mathbf{r}_{PQ} \times (\mathbf{r}_{PQ} \times \boldsymbol{\omega}_{B/I})), \end{aligned} \quad (3.28)$$

equation (3.26) can be rewritten as

$$\begin{aligned} \mathbf{M}_P &= \int_{\mathcal{V}} \left[dm \mathbf{r}_{PQ} \times \frac{{}^B d\mathbf{v}_{P/I}}{dt} + dm \boldsymbol{\omega}_{B/I} \times (\mathbf{r}_{PQ} \times \mathbf{v}_{P/I}) \right. \\ &\quad \left. - dm \mathbf{v}_{P/I} \times (\mathbf{r}_{PQ} \times \boldsymbol{\omega}_{B/I}) - dm \mathbf{r}_{PQ} \times \left(\mathbf{r}_{PQ} \times \frac{{}^B d\boldsymbol{\omega}_{B/I}}{dt} \right) \right. \\ &\quad \left. - dm \boldsymbol{\omega}_{B/I} \times (\mathbf{r}_{PQ} \times (\mathbf{r}_{PQ} \times \boldsymbol{\omega}_{B/I})) \right]. \end{aligned} \quad (3.29)$$

²The Jacobi identity states that for any three vectors $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, the following holds true: $\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) + \mathbf{b} \times (\mathbf{c} \times \mathbf{a}) + \mathbf{c} \times (\mathbf{a} \times \mathbf{b}) = 0$.

By recalling the definitions of $\mathbf{S}_P$ and $\mathbf{J}_P$, equation (3.29) can be further simplified, leading to

$$\begin{aligned} \mathbf{M}_P = & \mathbf{S}_P \frac{{}^B d\mathbf{v}_{P/I}}{dt} + \mathbf{J}_P \frac{{}^B d\boldsymbol{\omega}_{B/I}}{dt} + \mathbf{v}_{P/I} \times (m\mathbf{v}_{P/I} + \mathbf{S}_P^T \boldsymbol{\omega}_{B/I}) \\ & + \boldsymbol{\omega}_{B/I} \times (\mathbf{S}_P \mathbf{v}_{P/I} + \mathbf{J}_P \boldsymbol{\omega}_{B/I}). \end{aligned} \quad (3.30)$$

By taking point P coincident with the center of gravity G of the aircraft, the static moment tensor $\mathbf{S}_G$ (and its transpose as well) is null. For easiness of notation, from here on the velocity of the center of gravity measured with respect to the inertial frame $\mathbf{v}_{G/I}$ will be denoted as $\mathbf{v}_G$, the angular velocity measured in the inertial frame $\boldsymbol{\omega}_{B/I}$ will be denoted as $\boldsymbol{\omega}$ and the time derivative as measured in the body frame will be written with the Newton's dot notation. Hence, the 6 degrees of freedom equations of motion with respect to the centre of gravity are as follows:

$$\begin{cases} \mathbf{F} = m\dot{\mathbf{v}}_G + \boldsymbol{\omega} \times m\mathbf{v}_G \\ \mathbf{M}_G = \mathbf{J}_G \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{J}_G \boldsymbol{\omega}. \end{cases} \quad (3.31)$$

In the simulator, the right hand side of equation (3.31) is computed inside the *6DOF Euler angles* block, already built in the *Aerospace Blockset* of *Simulink*, as it will be further described in Chapter 4. Since no assumption on the flight mode of the aircraft has been introduced so far, equation (3.31) holds true for any mode of operation of the aircraft. What changes are the external forces and moments which act on the vehicle. Hence, in Section 3.5 the forces and moments which need to be fed into the integration block will be described, for each of the three flight phases of the aircraft.

3.5 External forces and moments

For all the three flight phases, the forces and moments which act on the VTOL are of gravitational, aerodynamic and propulsive type:

$$\begin{cases} \mathbf{F}^B = \mathbf{F}_{\text{Grav}}^B + \mathbf{F}_{\text{Aero}}^B + \mathbf{F}_{\text{Prop}}^B \\ \mathbf{M}_G^B = \mathbf{M}_{\text{Grav}}^B + \mathbf{M}_{\text{Aero}}^B + \mathbf{M}_{\text{Prop}}^B. \end{cases} \quad (3.32)$$

It is convenient to write equation (3.31) in the body frame, and hence all the forces and moments will have to be written in this frame.

3.5.1 Common to all flight modes

Gravitational forces and moments

Gravitational forces and moments are the same for all the three flight phases of the vehicle. In the NED frame, the gravitational force has only a z component towards

the centre of the Earth. To convert it to body frame, it has to be multiplied by the transformation matrix from NED to body frame:

$$\mathbf{F}_{\text{Grav}}^{\mathcal{N}} = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}, \quad \mathbf{F}_{\text{Grav}}^{\mathcal{B}} = \mathbf{R}_{321}^{\mathcal{N} \rightarrow \mathcal{B}} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}, \quad (3.33)$$

where $\mathbf{R}_{321}^{\mathcal{N} \rightarrow \mathcal{B}}$ is the transformation matrix for the Euler angles sequence 321 from NED to body frame, as already described in Section 3.2.1:

$$\mathbf{R}_{321}^{\mathcal{N} \rightarrow \mathcal{B}} = \begin{bmatrix} \cos \psi \cos \theta & \sin \psi \cos \theta & -\sin \theta \\ \cos \psi \sin \theta \sin \phi - \sin \psi \cos \phi & \sin \psi \sin \theta \sin \phi + \cos \psi \cos \phi & \cos \theta \sin \phi \\ \cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi & \sin \psi \sin \theta \sin \phi - \cos \psi \sin \phi & \cos \theta \cos \phi \end{bmatrix}, \quad (3.34)$$

By definition of centre of gravity, the moment due to the gravitational force about this point is null:

$$\mathbf{M}_{\text{Grav}}^{\mathcal{B}} = \mathbf{0}. \quad (3.35)$$

3.5.2 Forward flight mode

Aerodynamic forces and moments

Aerodynamic forces and moments, in the wind frame, can be written as follows:

$$\begin{bmatrix} D \\ Q \\ L \end{bmatrix} = \frac{1}{2} \rho V^2 S \begin{bmatrix} C_D \\ C_Q \\ C_L \end{bmatrix}, \quad \begin{bmatrix} \mathcal{L} \\ \mathcal{M} \\ \mathcal{N} \end{bmatrix} = \frac{1}{2} \rho V^2 S \begin{bmatrix} bC_{\mathcal{L}} \\ cC_{\mathcal{M}} \\ bC_{\mathcal{N}} \end{bmatrix}. \quad (3.36)$$

where ρ is the air density, V is the absolute value of airspeed, S is the wing's reference surface, c is the mean aerodynamic chord, b is the wingspan, D , Q , L are drag, sideforce, lift and $\mathcal{L}$, $\mathcal{M}$, $\mathcal{N}$ are pitching moment, rolling moment and yawing moment. The coefficients C_D , C_Q , C_L , $C_{\mathcal{L}}$, $C_{\mathcal{M}}$, $C_{\mathcal{N}}$ are, in the most general sense, function of the angle of attack and its derivative α and $\dot{\alpha}$, angle of sideslip and its derivative β and $\dot{\beta}$, body angular rates p , q , r , airspeed V , aileron, elevator and rudder deflections δ_a , δ_e , δ_r . The most common relationship is a linear one and the derivatives with respect to the angle of attack and angle of sideslip derivatives are usually negligible, leading to the following expressions:

$$\begin{cases} C_D = C_{D_0} + C_{D_\alpha} \alpha + C_{D_\beta} \beta + C_{D_p} p + C_{D_q} q + C_{D_r} r + C_{D_V} V + C_{D_{\delta_a}} \delta_a + C_{D_{\delta_e}} \delta_e + C_{D_{\delta_r}} \delta_r \\ C_Q = C_{Q_0} + C_{Q_\alpha} \alpha + C_{Q_\beta} \beta + C_{Q_p} p + C_{Q_q} q + C_{Q_r} r + C_{Q_V} V + C_{Q_{\delta_a}} \delta_a + C_{Q_{\delta_e}} \delta_e + C_{Q_{\delta_r}} \delta_r \\ C_L = C_{L_0} + C_{L_\alpha} \alpha + C_{L_\beta} \beta + C_{L_p} p + C_{L_q} q + C_{L_r} r + C_{L_V} V + C_{L_{\delta_a}} \delta_a + C_{L_{\delta_e}} \delta_e + C_{L_{\delta_r}} \delta_r \\ C_{\mathcal{L}} = C_{\mathcal{L}_0} + C_{\mathcal{L}_\alpha} \alpha + C_{\mathcal{L}_\beta} \beta + C_{\mathcal{L}_p} p + C_{\mathcal{L}_q} q + C_{\mathcal{L}_r} r + C_{\mathcal{L}_V} V + C_{\mathcal{L}_{\delta_a}} \delta_a + C_{\mathcal{L}_{\delta_e}} \delta_e + C_{\mathcal{L}_{\delta_r}} \delta_r \\ C_{\mathcal{M}} = C_{\mathcal{M}_0} + C_{\mathcal{M}_\alpha} \alpha + C_{\mathcal{M}_\beta} \beta + C_{\mathcal{M}_p} p + C_{\mathcal{M}_q} q + C_{\mathcal{M}_r} r + C_{\mathcal{M}_V} V + C_{\mathcal{M}_{\delta_a}} \delta_a + C_{\mathcal{M}_{\delta_e}} \delta_e + C_{\mathcal{M}_{\delta_r}} \delta_r \\ C_{\mathcal{N}} = C_{\mathcal{N}_0} + C_{\mathcal{N}_\alpha} \alpha + C_{\mathcal{N}_\beta} \beta + C_{\mathcal{N}_p} p + C_{\mathcal{N}_q} q + C_{\mathcal{N}_r} r + C_{\mathcal{N}_V} V + C_{\mathcal{N}_{\delta_a}} \delta_a + C_{\mathcal{N}_{\delta_e}} \delta_e + C_{\mathcal{N}_{\delta_r}} \delta_r \end{cases} \quad (3.37)$$

The aerodynamic coefficients derivatives were calculated with the VLM software *OpenVSP*, and scheduled as a function of the airspeed, as described in Chapter 1. Hence, for every airspeed ranging from the stall speed of 11 m/s to the maximum airspeed of 22 m/s, with a step of 0.5 m/s, the complete set of the aerodynamic derivatives of equation (3.37) were computed. In the simulation, the values between the airspeed anchor points were interpolated, effectively constructing a stitched, or quasi-, Linear Parameter Varying (qLPV) aerodynamic model. A further description of this aspect and of the interpolation scheme implemented can be found in Chapter 4.

After having calculated the forces and moments in the wind frame, they have to be expressed in body axes for the successive integration of the equations of motion. This is achieved by multiplying them with the transformation matrix from the wind frame to body axes $\mathbf{R}^{\mathcal{W} \rightarrow \mathcal{B}}$:

$$\mathbf{R}^{\mathcal{W} \rightarrow \mathcal{B}} = \begin{bmatrix} \cos \alpha \cos \beta & -\cos \alpha \sin \beta & -\sin \alpha \\ \sin \beta & \cos \beta & 0 \\ \sin \alpha \cos \beta & -\sin \alpha \sin \beta & \cos \alpha \end{bmatrix}. \quad (3.38)$$

where α is the angle of attack, and β is the angle of sideslip. Hence

$$\mathbf{F}_{\text{Aero}}^{\mathcal{B}} = \mathbf{R}^{\mathcal{W} \rightarrow \mathcal{B}} \begin{bmatrix} -D \\ -L \\ -Q \end{bmatrix}, \quad \mathbf{M}_{\text{G}_{\text{Aero}}}^{\mathcal{B}} = \mathbf{R}^{\mathcal{W} \rightarrow \mathcal{B}} \begin{bmatrix} \mathcal{L} \\ \mathcal{M} \\ \mathcal{N} \end{bmatrix}, \quad (3.39)$$

where the negative sign in front of the forces comes from the definition of drag, lift and sideforce.

Propulsive forces and moments

The propulsive forces are calculated starting from the results of the experimental motor identification described in Chapter 2. After having identified the thrust coefficient $\overline{C}_T$ and the linear relationship between throttle percentage and angular velocity, it is easy to obtain the propulsive forces:

$$\Omega = \overline{m} \cdot Thr_{\%} + \overline{q}, \quad (3.40)$$

$$T = \overline{K}_T \Omega^2 = \rho A R^2 \overline{C}_T \Omega^2. \quad (3.41)$$

The propulsive force is aligned with the aircraft x body axis, and hence has always only an x component in the body frame:

$$\mathbf{F}_{\text{Prop}}^{\mathcal{B}} = \begin{bmatrix} 2T \\ 0 \\ 0 \end{bmatrix}, \quad (3.42)$$

where the multiplication by two is explained by the fact that the motors for horizontal flight are two.

The moment generated by the forward flight propellers has only a pitch component, since the two motors spin in opposite directions, hence effectively producing a null yawing and rolling moment. Hence, by denoting as d_{vert} the vertical (along the body z axis) distance between the motors and the center of gravity, the moment generated by the propulsive unit, in body axes, is given by the following expression:

$$\mathbf{M}_{\mathbf{G}_{\text{Prop}}}^B = \begin{bmatrix} 0 \\ -2d_{vert}T \\ 0 \end{bmatrix}, \quad (3.43)$$

where the multiplication by two is explained by the fact that the motors for horizontal flight are two, and the negative sign due to the fact that, since the motors are above the center of gravity, the pitching moment is negative (pitch down). For this simulator no yawing moment due to thrust asymmetries is considered, since the aim of this model is not to investigate the aircraft behaviour during components' failures.

3.5.3 VTOL mode

During the VTOL flight phase, the aircraft is assumed to climb, descend and hover vertically, with no horizontal component of velocity. However the aircraft, during this flight phase, can manoeuvre as well, by individually adjusting the thrust of each of the eight vertical flight motors, just like a normal multicopter would do.

Firstly, the direction of rotation of the VTOL motors has to be decided. It has been chosen that the two forward left and rear right motors (1a, 1b, 3a, 3b, with reference to Figure 3.4) spin clockwise, when viewed from above, and that the two forward right and rear left ones (2a, 2b, 4a, 4b) spin counterclockwise. It is important that the overall number of motors that spin clockwise and counterclockwise is the same, so that the net yawing moment generated is null. The rest of the decision is mainly arbitrary, with the choice to spin two adjacent motors in the same direction justified by the fact that with this configuration the VTOL aircraft could be approximated as being a quadrotor multicopter.

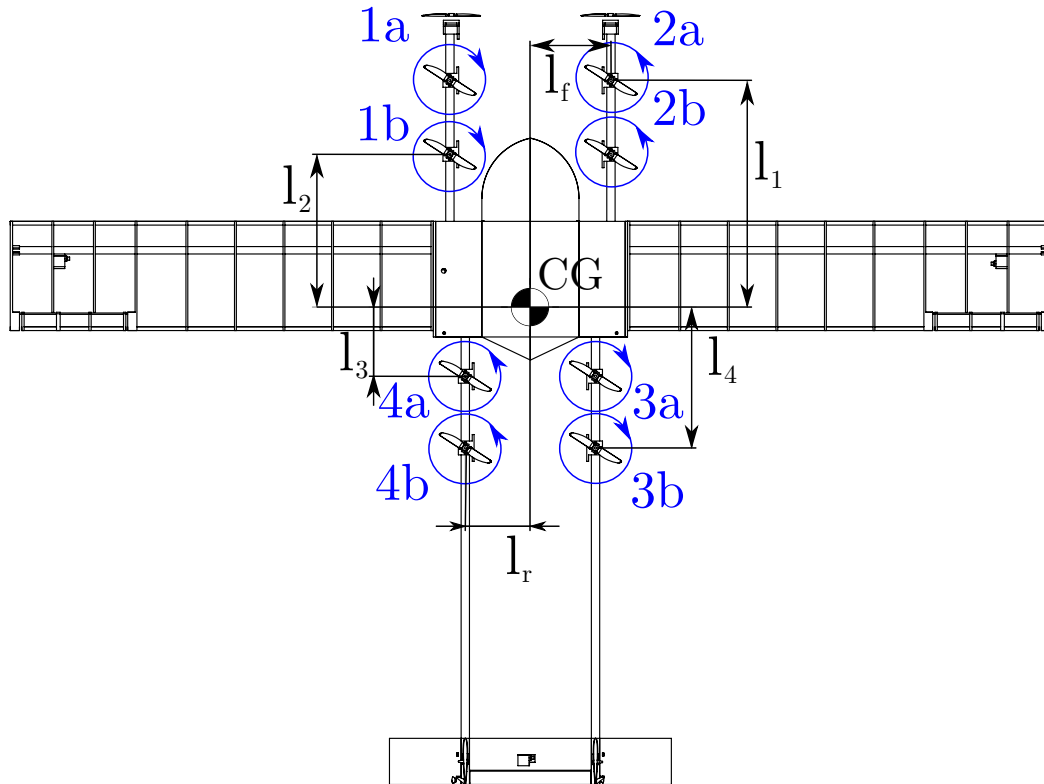


Figure 3.4: Aircraft top view, with relevant dimensions named, propellers numbering and propellers spinning directions.

In order to move forward, the thrust on the forward VTOL propellers (number 1a, 1b, 2a, 2b, with reference to Figure 3.4) is decreased, whereas it is increased by the same amount on the rear ones (number 3a, 3b, 4a, 4b). Hence, the UAV pitches forward and a component of thrust in the forward direction is also generated, thus allowing the vehicle to move forward. Similarly, to move the aircraft sideways, thrust is increased on the propellers on one side (for instance, to move to the left, on propellers 2a, 2b, 3a, 3b) and decreased on the propellers on the other side (1a, 1b, 4a, 4b). The aircraft hence banks and starts to move laterally. Finally, to turn the aircraft about its z (yaw) axis, the thrust on diametrically opposite propellers is increased (for instance 1a, 1b, 3a, 3b) and simultaneously decreased on the remaining propellers (2a, 2b, 4a, 4b). Hence, the total torque provided by the VTOL motors is not zero anymore, and a resulting moment about the z axis is generated, while the other forces and moments remain the same as during the hover condition.

Hence, during the VTOL phase, the airplane's control surfaces (ailerons, rudders and elevator) are not used at all (because they are ineffective). The propellers for horizontal flight mode could be used to move the aircraft forward without increasing its pitch, contrary to what was described before. This redundant possibil-

ity for forward movement shows how the aircraft is over-actuated, hence providing a great deal of flexibility during its usage. This will be also shown during the description of some of the possible transition manoeuvres in Section 3.5.4.

Aerodynamic forces and moments

The two main contributions for the aerodynamic forces and moments during the VTOL phase are the aerodynamic drag coming from big, flat components (namely the wing, fuselage and tail) and the the aerodynamic damp caused by rotating propellers moving through the air:

$$\begin{cases} \mathbf{F}_{\text{Aero}}^{\mathcal{B}} = \mathbf{F}_{\text{Drag}}^{\mathcal{B}} \\ \mathbf{M}_{\text{GAero}}^{\mathcal{B}} = \mathbf{M}_{\text{GDrag}}^{\mathcal{B}} + \mathbf{M}_{\text{GDamp}}^{\mathcal{B}} \end{cases} \quad (3.44)$$

Aerodynamic drag

During the vertical takeoff, landing and hovering phases the main aerodynamic forces are given by the drag exerted by the various aircraft components. The main contributions which have been considered are the ones from the main wing, fuselage, horizontal tail (horizontal stabilizer and elevator) and vertical tail (vertical stabilizers and rudders), while the effects from the motor booms have been neglected.

As already explained, during the VTOL operations the UAV has a non zero (or zero while hovering) w body velocity along the z axis, whereas the other two components of velocities u and v are small. Hence, the wing, fuselage and horizontal tail exert a drag force along the z direction. Since the component of the airflow which contributes to this drag is the one normal to the surfaces, the drag force can be expressed as follows:

$$\begin{aligned} D_{\text{Wing}} &= -\text{sgn}(w_{\text{Wing}}) \frac{1}{2} \rho w_{\text{Wing}}^2 S_{\text{Wing}} C_{D_{\text{Wing}}}, \\ D_{\text{Fus}} &= -\text{sgn}(w_{\text{Fus}}) \frac{1}{2} \rho w_{\text{Fus}}^2 S_{\text{Fus}} C_{D_{\text{Fus}}}, \\ D_{\text{HTail}} &= -\text{sgn}(w_{\text{HTail}}) \frac{1}{2} \rho w_{\text{HTail}}^2 S_{\text{HTail}} C_{D_{\text{HTail}}}. \end{aligned} \quad (3.45)$$

Note that the sign function is needed to ensure that drag always acts in an opposite direction with respect to the component's airspeed.

The vertical tail, on the other hand, exerts a drag force in the y direction, proportional to the square of the y component of the vertical tail velocity v_{VTail} :

$$D_{\text{VTail}} = -\text{sgn}(v_{\text{VTail}}) \frac{1}{2} \rho v_{\text{VTail}}^2 S_{\text{VTail}} C_{D_{\text{VTail}}}. \quad (3.46)$$

Hence, the overall aerodynamic force in body axes is given by

$$\mathbf{F}_{\text{Drag}}^{\mathcal{B}} = \begin{bmatrix} 0 \\ D_{\text{VTail}} \\ D_{\text{Wing}} + D_{\text{Fus}} + D_{\text{HTail}} \end{bmatrix}. \quad (3.47)$$

Note that in equation (3.45) and equation (3.46) there is the need to calculate the velocities of the various components which produce drag. For instance, if the aircraft is hovering, and yawing about its z axis, the y velocity of the center of gravity is zero, but the velocity of the vertical surfaces of the tail is non-zero, and hence they will produce a drag, which in turn will also produce both a yawing and a rolling moment (the last one due to the fact that the center of pressure of the vertical surfaces is in a different position along the z direction with respect to the center of gravity of the aircraft). Hence, there is the need to calculate the velocities of the centres of pressure of the surfaces which produce drag. For simplicity, the center of pressure is considered to be coincident to the geometric centre of the rectangular surfaces which produce drag, and their velocity is easily calculated with equation (3.16), here reported for convenience:

$$\mathbf{v}_{\text{CP}} = \mathbf{v}_{\text{CG}} + \boldsymbol{\omega} \times \mathbf{r}_{\text{CP}-\text{CG}}, \quad (3.48)$$

where the subscript $(\cdot)_{\text{CP}}$ refers to the centre of pressure of any of the drag producing components.

The reference surfaces are known from the CAD model of the vehicle, and the value of the drag coefficient is taken from [3], which reports experimental results for drag coefficients of flat rectangular plates normal to the airflow, as a function of their aspect ratio. For the sake of simplicity and without introducing any relevant error, also the fuselage planform shape is considered to be rectangular. Table 3.1 reports the drag coefficients used for the various drag-producing components.

Table 3.1: Drag coefficients of the VTOL parts; values from [3].

Component	Aspect ratio	Drag coefficient
Wing	9	1.27
Fuselage	~ 2	1.15
Horizontal tail	7	1.22
Vertical tail	2	1.15

Once each individual drag force is calculated, the moment produced by the drag force on each each single i^{th} surface is easily obtained as follows:

$$\mathbf{M}_{\text{GDrag}_i}^{\mathcal{B}} = \mathbf{r}_{\text{CG}-\text{CP}_i}^{\mathcal{B}} \times \mathbf{D}_i^{\mathcal{B}}, \quad (3.49)$$

where $\mathbf{r}_{\text{CG}-\text{CP}_i}^{\mathcal{B}}$ is the vector from the center of gravity to the center of pressure of the i^{th} surface. The total moment is obtained by summing up the moment contributions from all surfaces:

$$\mathbf{M}_{\text{GDrag}}^{\mathcal{B}} = \mathbf{M}_{\text{GDragWing}}^{\mathcal{B}} + \mathbf{M}_{\text{GDragFus}}^{\mathcal{B}} + \mathbf{M}_{\text{GDragHTail}}^{\mathcal{B}} + \mathbf{M}_{\text{GDragVTail}}^{\mathcal{B}}. \quad (3.50)$$

Aerodynamic damp

The aerodynamic damp is expressed as a moment only. As reported in [21] and [27], two main assumptions are made for modelling the aerodynamic damp:

- the aerodynamic damp is proportional to the angular velocity ω ;
- the aerodynamic damp moments are decoupled: the moment relative to an axis is proportional only to the rotational speed about that axis.

Hence, the aerodynamic damp moment can be expressed by the following equation:

$$\mathbf{M}_{\mathbf{G}_{\text{Damp}}}^{\mathcal{B}} = \begin{bmatrix} \frac{\partial M_x}{\partial p} & 0 & 0 \\ 0 & \frac{\partial M_y}{\partial q} & 0 \\ 0 & 0 & \frac{\partial M_z}{\partial r} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \quad (3.51)$$

The stability derivatives $\frac{\partial M_x}{\partial p}$, $\frac{\partial M_y}{\partial q}$ and $\frac{\partial M_z}{\partial r}$ can be calculated using analytical formulas, taken from the realm of helicopter's dynamics (see [28], [29]). As a rough approximation, the derivatives $\frac{\partial M_x}{\partial p}$ and $\frac{\partial M_y}{\partial q}$ can be considered to be equal, due to the VTOL's geometry, and have the following expression:

$$\frac{\partial M_x}{\partial p} = -4\rho A R^2 \Omega^2 \frac{\partial C_T}{\partial p} \sqrt{(l_1^2 + l_f^2)}, \quad (3.52)$$

with

$$\frac{\partial C_T}{\partial p} = \frac{C_{L_\alpha}}{8} \frac{\sigma}{R\Omega} \sqrt{(l_1^2 + l_f^2)} \quad (3.53)$$

and σ being the solidity ratio, defined as the ratio between the total blade area A_b and the disk area A :

$$\sigma = \frac{A_b}{A}. \quad (3.54)$$

The lift coefficient derived with respect to the angle of attack C_{L_α} of the propeller can be approximated as being equal to 2π , from the results of analytical aerodynamics, since no accurate geometric data is known for the propeller.

Note that, due to the fact that not all propellers have the same distance from the centre of gravity, the value of $\frac{\partial M_x}{\partial p}$ is an approximation, which however has a negligible impact on the simulation, since the difference in distance is small.

Finally, due to the small rotational speed around the z axis compared to the other axis, the derivative $\frac{\partial M_z}{\partial r}$ can be neglected.

Propulsive forces and moments

Each of the propellers exerts a thrust and torque proportional to the square of the angular velocity, according to equation (3.40), equation (3.41), and equation (3.55):

$$Q = \overline{K_Q} \Omega^2 = \rho A R^3 \overline{C_Q} \Omega^2. \quad (3.55)$$

The overall forces and moments generated by the VTOL propulsion, in body axes, are given by the following equations:

$$\begin{bmatrix} F_{Prop_x} \\ F_{Prop_y} \\ F_{Prop_z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -(K_T \Omega_{1a}^2 + K_T \Omega_{1b}^2 + K_T \Omega_{2a}^2 + K_T \Omega_{2b}^2 + K_T \Omega_{3a}^2 + K_T \Omega_{3b}^2 + K_T \Omega_{4a}^2 + K_T \Omega_{4b}^2) \end{bmatrix}, \quad (3.56)$$

$$\begin{bmatrix} M_{Prop_x} \\ M_{Prop_y} \\ M_{Prop_z} \end{bmatrix} = \begin{bmatrix} K_T l_f (\Omega_{1a}^2 + \Omega_{1b}^2) - K_T l_f (\Omega_{2a}^2 + \Omega_{2b}^2) - K_T l_r (\Omega_{3a}^2 + \Omega_{3b}^2) + K_T l_r (\Omega_{4a}^2 + \Omega_{4b}^2) \\ K_T l_1 (\Omega_{1a}^2 + \Omega_{2a}^2) + K_T l_2 (\Omega_{1b}^2 + \Omega_{2b}^2) - K_T l_3 (\Omega_{4a}^2 + \Omega_{3a}^2) - K_T l_4 (\Omega_{4b}^2 + \Omega_{3b}^2) \\ K_Q (\Omega_{1a}^2 + \Omega_{1b}^2) - K_Q (\Omega_{2a}^2 + \Omega_{2b}^2) + K_Q (\Omega_{3a}^2 + \Omega_{3b}^2) - K_Q (\Omega_{4a}^2 + \Omega_{4b}^2) \end{bmatrix}, \quad (3.57)$$

where the distances l_f , l_r , l_1 , l_2 , l_3 , l_4 are shown in Figure 3.4. Hence

$$\mathbf{F}_{\mathbf{Prop}}^{\mathcal{B}} = \begin{bmatrix} F_{Prop_x} \\ F_{Prop_y} \\ F_{Prop_z} \end{bmatrix}, \quad \mathbf{M}_{\mathbf{G}_{\mathbf{Prop}}}^{\mathcal{B}} = \begin{bmatrix} M_{Prop_x} \\ M_{Prop_y} \\ M_{Prop_z} \end{bmatrix}. \quad (3.58)$$

It is usually convenient to group the forces and moments in the so-called "mixer matrix" χ , which relates the forces and moments generated by the propellers to the square of the angular velocities:

$$\begin{bmatrix} T \\ M_{Prop_x} \\ M_{Prop_y} \\ M_{Prop_z} \end{bmatrix} = \begin{bmatrix} K_T & K_T & K_T & K_T & K_T & K_T & K_T & K_T \\ K_T l_f & K_T l_f & -K_T l_f & -K_T l_f & -K_T l_r & -K_T l_r & K_T l_r & K_T l_r \\ K_T l_1 & K_T l_2 & K_T l_1 & K_T l_2 & -K_T l_3 & -K_T l_4 & -K_T l_3 & -K_T l_4 \\ K_Q & K_Q & -K_Q & -K_Q & K_Q & K_Q & -K_Q & -K_Q \end{bmatrix} \begin{bmatrix} \Omega_{1a}^2 \\ \Omega_{1b}^2 \\ \Omega_{2a}^2 \\ \Omega_{2b}^2 \\ \Omega_{3a}^2 \\ \Omega_{3b}^2 \\ \Omega_{4a}^2 \\ \Omega_{4b}^2 \end{bmatrix}, \quad (3.59)$$

where T is the total thrust, which has opposite sign with respect to the force in the z direction, due to the orientation of the body frame (z axis pointing downward). Hence

$$\chi = \begin{bmatrix} K_T & K_T & K_T & K_T & K_T & K_T & K_T & K_T \\ K_T l_f & K_T l_f & -K_T l_f & -K_T l_f & -K_T l_r & -K_T l_r & K_T l_r & K_T l_r \\ K_T l_1 & K_T l_2 & K_T l_1 & K_T l_2 & -K_T l_3 & -K_T l_4 & -K_T l_3 & -K_T l_4 \\ K_Q & K_Q & -K_Q & -K_Q & K_Q & K_Q & -K_Q & -K_Q \end{bmatrix}. \quad (3.60)$$

3.5.4 Transition flight

The transition is the short flight phase during which the VTOL from a hover or near hover condition starts accelerating forward, to transition to the forward flight mode. There a number of different transition maneuvers which can be executed, and two main distinctions have been identified. The first distinction is whether the transition starts from a hover condition or from a climb/descend condition. The second distinction is how the forward acceleration is achieved.

Starting condition

In order to simplify the problem, without impacting on the operatibility of the vehicle, at the start of the transition phase the VTOL is supposed to have a zero (or near zero) horizontal velocity. However, the aircraft can start transitioning either from a hover condition, and hence have also a null vertical velocity, or can start the transition during a vertical climb/descend, and maintain its vertical speed during the transition. The second case should provide more flexibility to the aircraft, since it can combine both the ascend/descend phase from/to the target final position and the forward flight towards that position. However also the first type might be useful as well. Imagine the VTOL operating in a dense urban environment, full of nearby tall obstacles (trees, street poles, buildings): it is safer for the aircraft to vertically climb to a safe altitude, clear the vertical obstacles, and then start the transition to forward flight.

For the purpose of simulation, the transition starting from a hover condition is the only one implemented in this work. That is because for the other type of transition, the lifting surfaces (wing and horizontal tail) experience high angles of attack (starting from around ± 90 degrees, when the aircraft has near-zero horizontal velocity, and gradually decreasing as the aircraft accelerates). The aerodynamic model developed so far is only valid for small angles of attack, since it based on a VLM solver. For future works, either a CFD model or wind tunnel testing covering a wide range of high angles of attack (from -90 degrees to 90 degrees) is needed to accurately simulate and study also the second transition type.

Forward acceleration strategy

The forward acceleration can be achieved in different ways. A first possibility is the following: maintain a level attitude (null pitch angle); turn on the forward flight motors and as the aircraft accelerates gradually reduce the thrust of the vertical motors, as the wing starts producing lift. When the aircraft reaches the speed for which lift equals weight with the zero pitch angle attitude, the vertical motors can be turned off.

A second option is to start accelerating horizontally by pitching the aircraft forward, using only the VTOL motors, like a normal multicopter would do (as

explained in Section 3.5.3), and after a certain speed is achieved turn on the forward flight motors.

A third option is to combine the previous two cases: simultaneously pitch forward and turn on the horizontal flight motors.

There are some disadvantages for the second and third options. As the aircraft is pitched forward (negative pitch angle), the wing has a negative angle of attack, and hence generates a negative lift. So, the vertical motors need to produce more thrust compared to the first case, because they have to balance both the weight and the negative lift of the wing. For the third option a positive z component (towards the ground) of force is also generated by the thrust of the forward flight propellers, which also needs to be balanced by the vertical flight propellers.

Hence, to summarize, Figure 3.5 depicts the possible combinations of transitions which have been identified, and Figure 3.6 shows a graphical schematics of the possible strategies. In the present work only the first type (starting from hover and accelerating with a level attitude) has been implemented in the simulator. Future work can be dedicated to a detailed study of this complex manoeuvre. Optimization strategies can be implemented to develop the best transition profile to minimize a defined objective function, which could take into account, for instance, the time taken by the transition, for time-critical missions, or the overall energy used during the transition, for long endurance missions, or any other mission related critical parameter.

The external forces and moments which act during the transition phase are a combination of the ones already detailed for the forward flight mode and the VTOL mode.

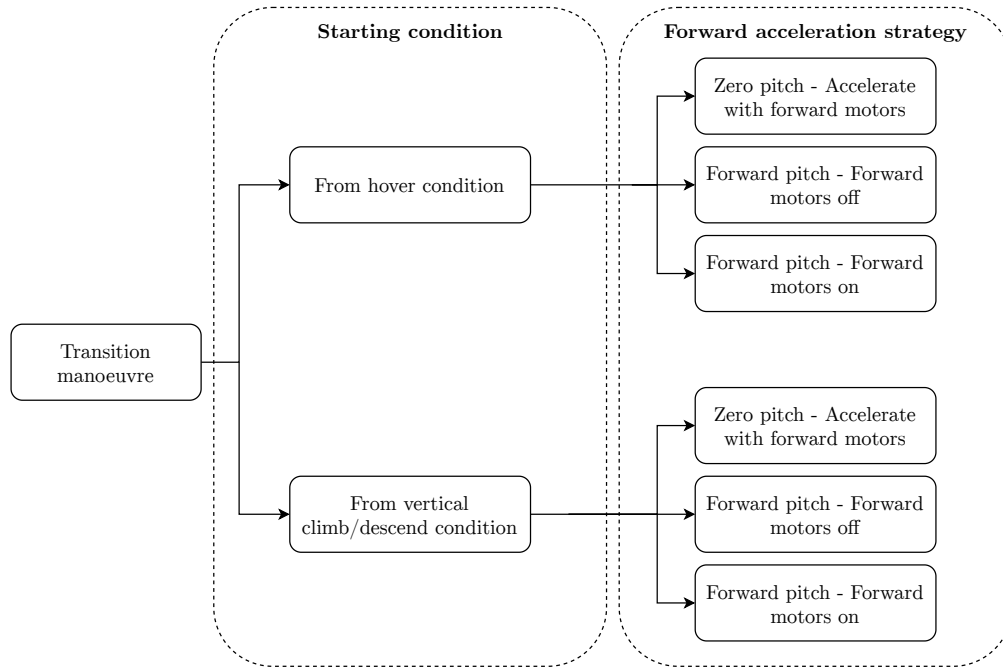
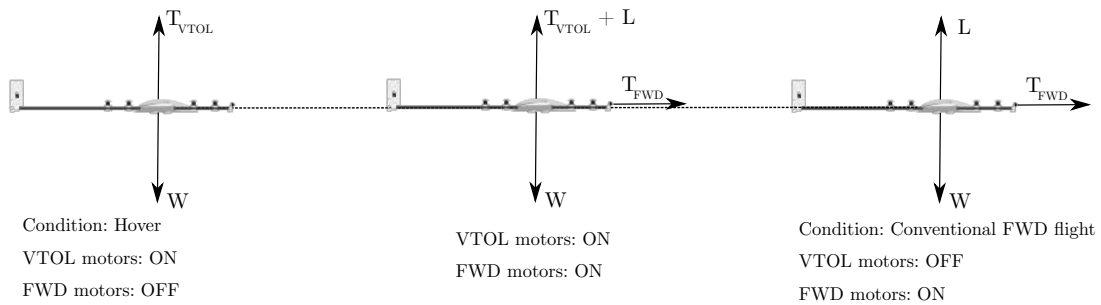
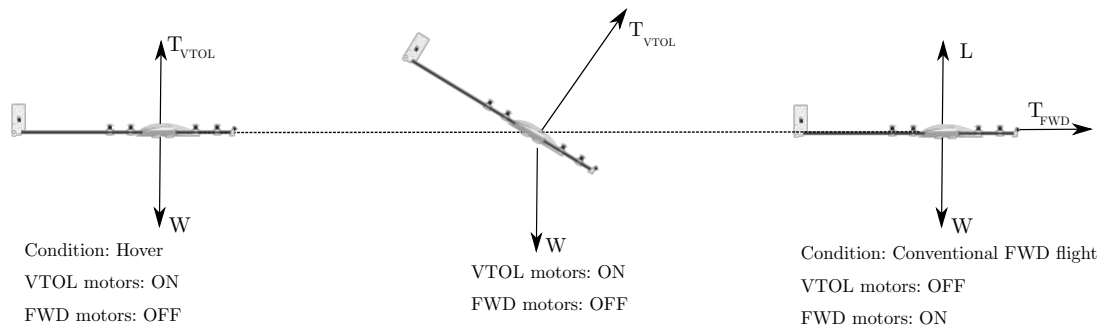


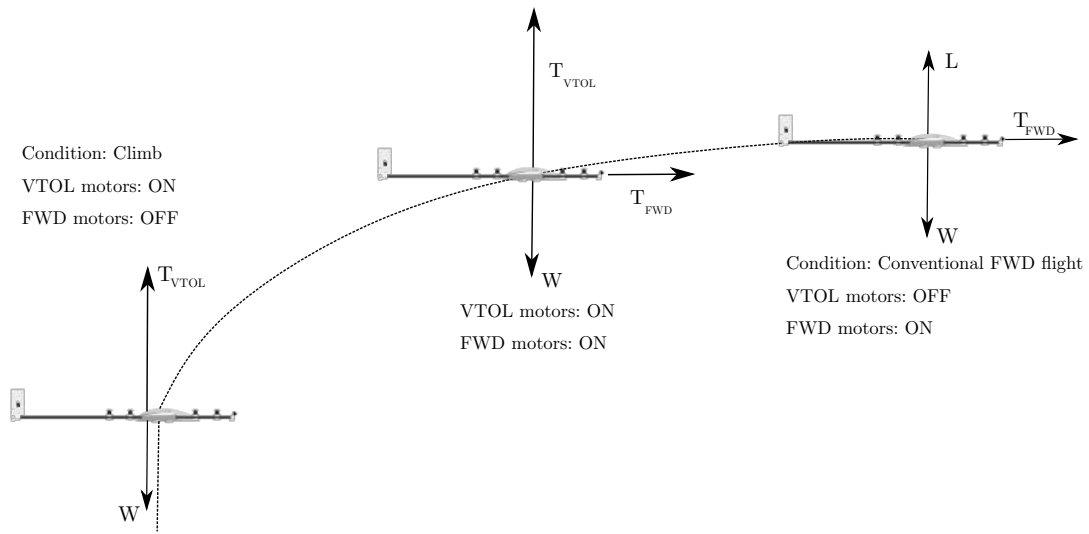
Figure 3.5: Flowchart of the possible strategies for the transition manoeuvre.



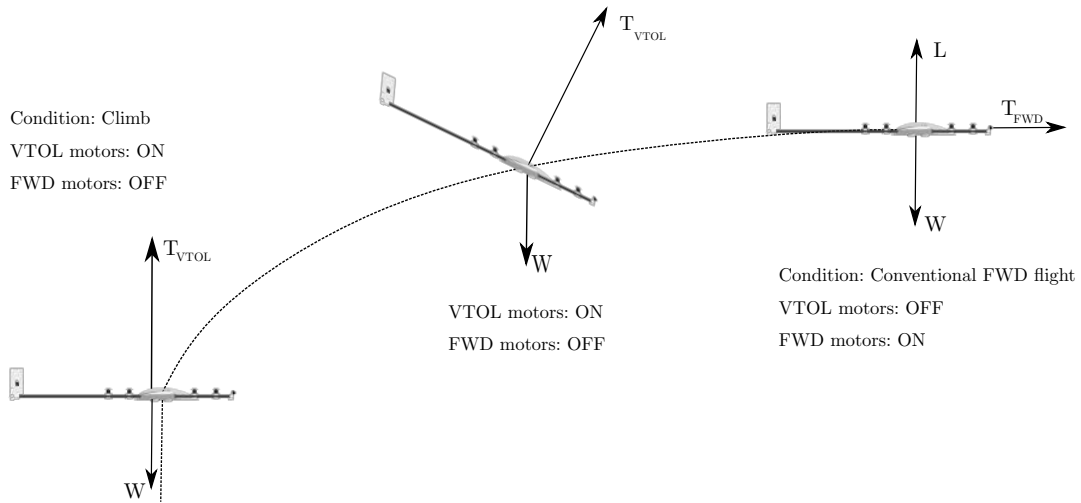
(a) Starting condition: hover. Forward acceleration strategy: zero pitch.



(b) Starting condition: hover. Forward acceleration strategy: forward pitch.



(c) Starting condition: climb. Forward acceleration strategy: zero pitch.



(d) Starting condition: climb. Forward acceleration strategy: forward pitch.

Figure 3.6: Possible strategies for the transition manoeuvre. For all the four cases, the transition starts on the left and ends on the right. The aircraft trajectory is shown by the dashed line.

Aerodynamic forces and moments

The type of aerodynamic forces and moments depends on the type of transition which is executed. If the transition is of the kind starting from an hover condition, the aerodynamic forces which act on the aircraft are the ones described in Section 3.5.2. In order to obtain aerodynamic data for this condition as well, the following approach has been used.

First, a "reduced" weight of the aircraft is calculated, for every airspeed rang-

ing from 0 m/s to the airspeed at which the aircraft can aerodynamically sustain its own weight with a zero pitch attitude (from this moment on this airspeed is going to be called the end of transition airspeed). The "reduced" weight is equal to the lift that the aircraft can generate for a given airspeed in this range:

$$W_{Red}(V) = \frac{1}{2}\rho V^2 SC_L(\theta = 0) < W \quad , \quad (3.61)$$

where W_{Red} is the "reduced" weight, and W is the aircraft's weight.

Then, aerodynamic simulations with the software *OpenVSP*, as described in Chapter 1, are run for airspeeds ranging from 0 m/s to the end of transition airspeed, with the aircraft weight equal to its reduced weight. With this approach, it is straightforward to calculate the total amount of vertical thrust that the VTOL propellers need to generate, as a function of airspeed, during the transition phase, to keep the aircraft at a constant altitude:

$$F_{Prop}(V) = W - \frac{1}{2}\rho V^2 SC_L(\theta = 0) = W - W_{Red} \quad . \quad (3.62)$$

Hence, during this type of transition it is as if the aircraft, due to the additional thrust of the VTOL propellers, weighted less.

Note that in this case lift has been approximated as acting along the z body axis. This is not the real case, since lift is tilted by the angle of attack of the wing. However this angle is small for a null pitch attitude of the aircraft and hence the aforementioned approximation can be reasonably assumed.

As previously said, for the modelling of the transition phase during a climb or descend a high angle of attack characterization of the aircraft is needed, which is not present in the current work and thus this case will not be further developed.

Propulsive forces and moments

Propulsive forces and moments are just the linear combination of the ones coming from the forward flight propellers (described in Section 3.5.2) and the VTOL propellers (described in Section 3.5.3).

Chapter 4

Simulink flight simulator

In order to more precisely analyze the VTOL's flight performance and to have a simulation platform on which to develop and test the control laws, a flight simulator of the aircraft was developed. Such simulator covers all of the UAV's flight envelope and its three modes of operations: vertical flight, transition and forward flight. *Simulink* was chosen as the main frame for the simulation, since it allows for an easy implementation of the aircraft's dynamic equations and control laws.

In this chapter the *Simulink* flight simulator is developed and analyzed. In the first section, the model is described, focusing on its main building blocks. Next, the interpolation scheme used for the aerodynamic stability and control derivatives is reported and validated. In the third section the aircraft's response to control inputs, computed with the non-linear *Simulink* simulator, is compared to the classical linearized longitudinal and lateral-directional model, in order to validate the simulator and to appreciate the difference between a complete and a linearized aircraft model. Finally, the aircraft's flight dynamics, modes and performance is evaluated, using the flight simulator.

4.1 Simulator architecture

The simulator developed follows a model stitching approach, building a stitched, or quasi Linear Parameter Varying (qLPV) model for the aircraft. This technique consists in combining together individual linear models for discrete flight conditions to produce a continuous, full flight-envelope simulation model (see [30] and [31]). The aerodynamic data is built by interpolating linear models, which vary with airspeed, while the rest of the simulator is built around the non-linear equations of motion of the vehicle.

Figure 4.1 shows the general scheme of the *Simulink* flight simulator. Note that the current simulation frame is open-loop, hence no control has been implemented

yet. Manual time histories of the control inputs (control surfaces, VTOL and forward flight motors) can be manually set in the "Inputs" section, together with the flight mode (either VTOL or airplane) and the wind and turbulence model can be globally toggled on or off using the "wind switch".

In the airframe block, detailed in Figure 4.2, the VTOL's dynamics is simulated, as described in Chapter 3. Aerodynamic, propulsive and gravitational forces and moments are calculated and the six degrees of freedom equations of motion are integrated, using *MATLAB*'s *Aerospace Blockset* six degrees of freedom dynamic equations integrator block, which can be set to either use Euler angles or quaternion description for the aircraft attitude. Figure 4.3 shows the two blocks which calculate aerodynamic forces and moments for the forward flight mode (Figure 4.3a) and for the VTOL flight mode (Figure 4.3b). The battery is simulated as well, to provide estimates for the total current drawn during the various aircraft operations and to have an accurate estimate of the energy consumption of the vehicle. Actual LiPo battery data was used, from a 8500 mAh 20C data (from [2]), instead of the actual 8500 mAh 50C battery installed on the aircraft, since no experimental discharge curve was found for the latter. In Appendix C the detailed procedure for the computation of the required data to be fed in the *Simulink* block is described.

The environment block, shown in Figure 4.4, calculates accurate gravitational data, based on the aircraft's position, as well as the atmospheric conditions from the ISA atmosphere model. Wind and turbulence data are also generated in this block, consisting in the combination of a gust profile with a single cosine form, a wind shear profile and a turbulence model, which can be selected to either be of Dryden or Von Karman type. Wind and turbulence can be globally switched on or off in the simulator.

The scopes block features numerous live scopes which monitor all the aircraft states and other relevant quantities during the simulation.

The FlightGear video interface block allows the connection of the *Simulink* model to the freeware and open-source flight simulator *FlightGear*. It enables the live view of the aircraft's position and orientation in space, thus helping to get an immediate visual overview of the aircraft's position and attitude.

The simulation is automatically stopped by the STOP Simulation block, when either the aircraft's altitude is zero, or when the battery state of charge drops below a level which can be manually set before the simulation.

The VTOL's mass, center of gravity and moments of inertia with respect to the center of gravity are computed on the most recent and updated CAD model in *SolidWorks* software; constants and initial states are initialized using a *MATLAB* function.

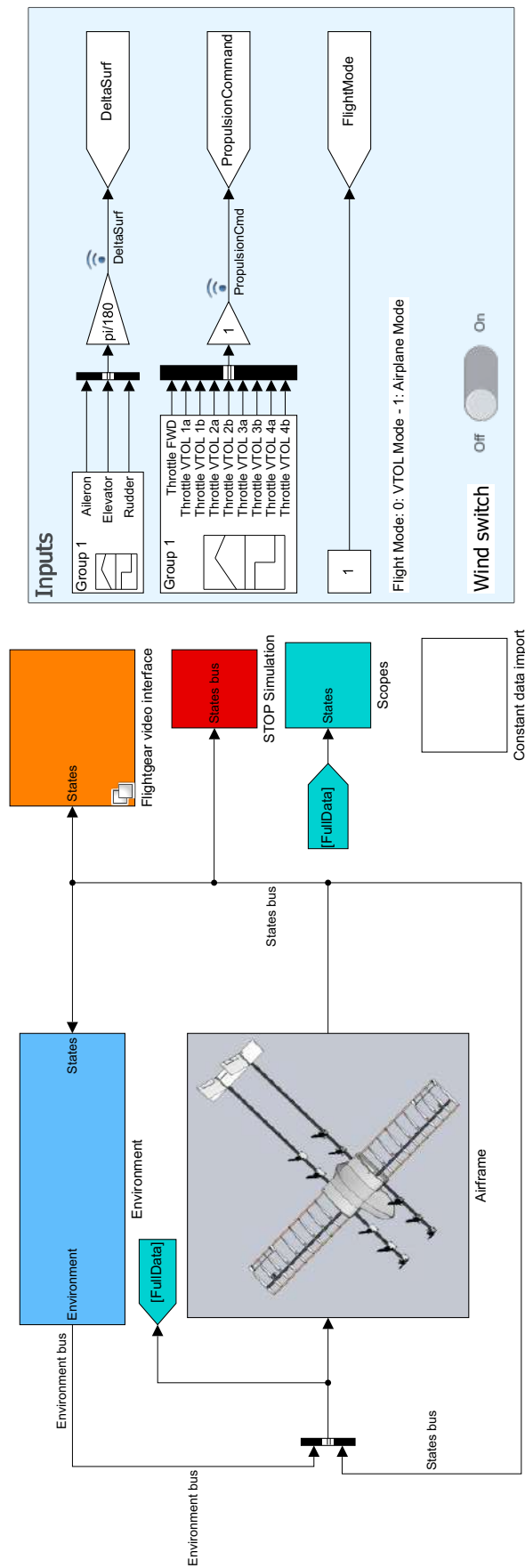


Figure 4.1: Simulink model overall architecture.

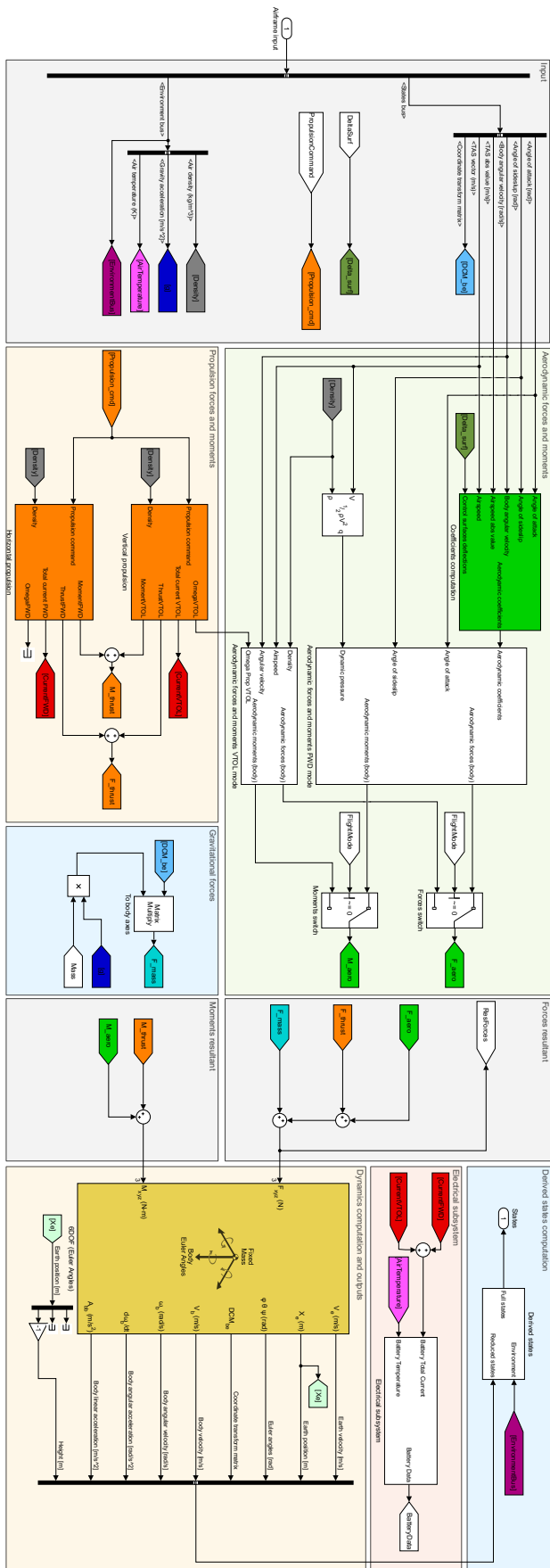
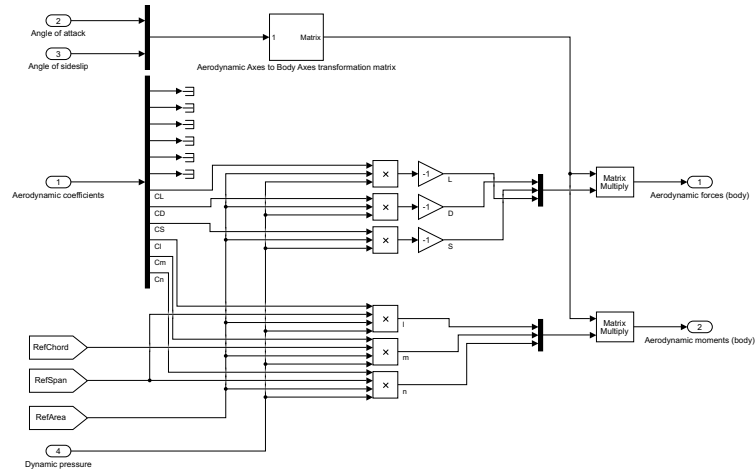
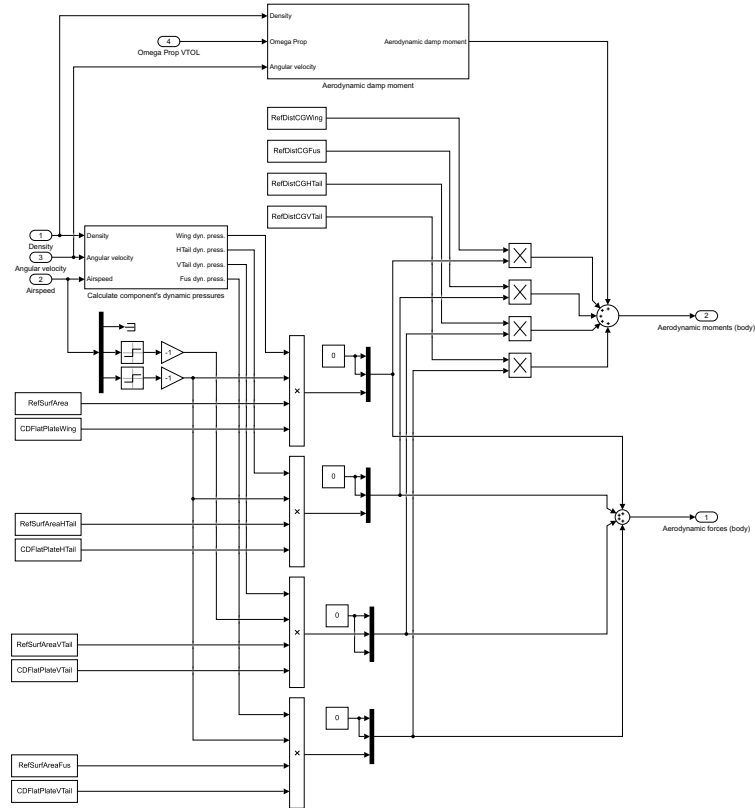


Figure 4.2: *Simulink* model, airframe block.



(a) Forward flight mode aerodynamic subsystem.



(b) VTOL flight mode aerodynamic subsystem.

Figure 4.3: *Simulink* model, aerodynamic subsystems: (a) forward flight mode and (b) VTOL mode.

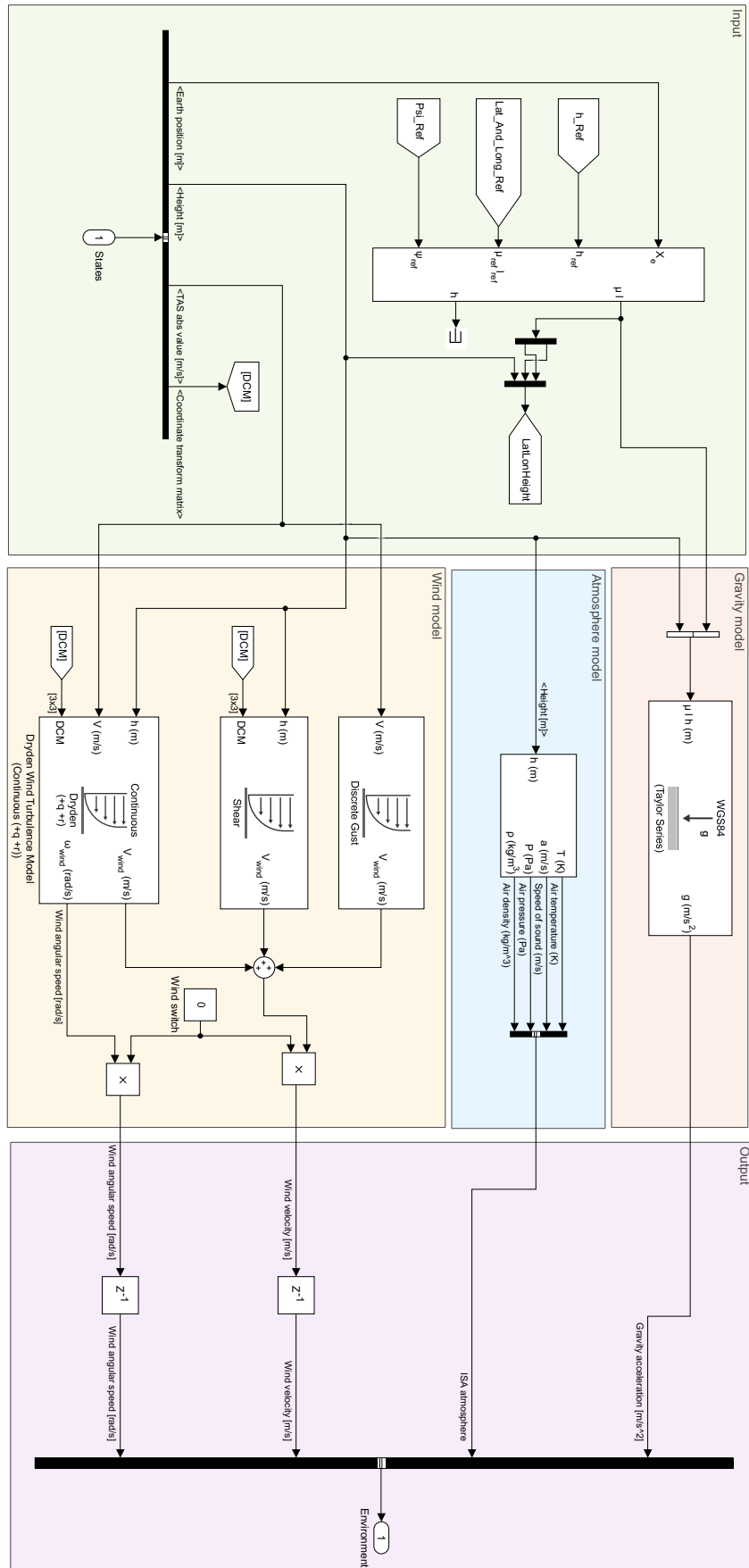


Figure 4.4: Simulink model, environment block.

4.2 Validation of the aerodynamic derivatives interpolation scheme

A shape-preserving piece-wise cubic interpolation scheme has been used for the interpolation of the stability and control derivatives. In order to validate it, the following approach has been used. For the full aerodynamic dataset used in the flight simulator model the aerodynamic analyses were done for airspeeds ranging from 11 m/s (the stall speed) to 22 m/s (the maximum airspeed), with a step of 0.5 m/s. For the validation of the interpolation scheme, this dataset was divided into two subsets:

- a dataset used as anchor points to construct the interpolating curve, with airspeeds ranging from 11 m/s to 22 m/s with a step of 1 m/s;
- the rest of the dataset used as validation points, to see how much did the interpolated curve differ from the actual computed coefficients in the validation points. For this purpose airspeeds ranging from 11.5 m/s to 19.5 m/s were used, with a step of 1 m/s.

In this process, numerical noise from the computed *OpenVSP* model was observed, with non-physical peaks present in otherwise smooth derivatives. Hence, the raw stability and control derivatives computed from *OpenVSP* were filtered with a third order median filter. The order was empirically adjusted so that the peaks were effectively removed, while the rest of the derivatives were not changed. Figure 4.5 and Figure 4.6 show, as an example, the results of the filtering process for the $C_{N_{\delta_r}}$ and C_{S_q} derivatives, whereas Figure 4.7 and Figure 4.8 show the interpolated model for the $C_{\mathcal{L}_r}$ and $C_{N_{\delta_a}}$ derivatives.

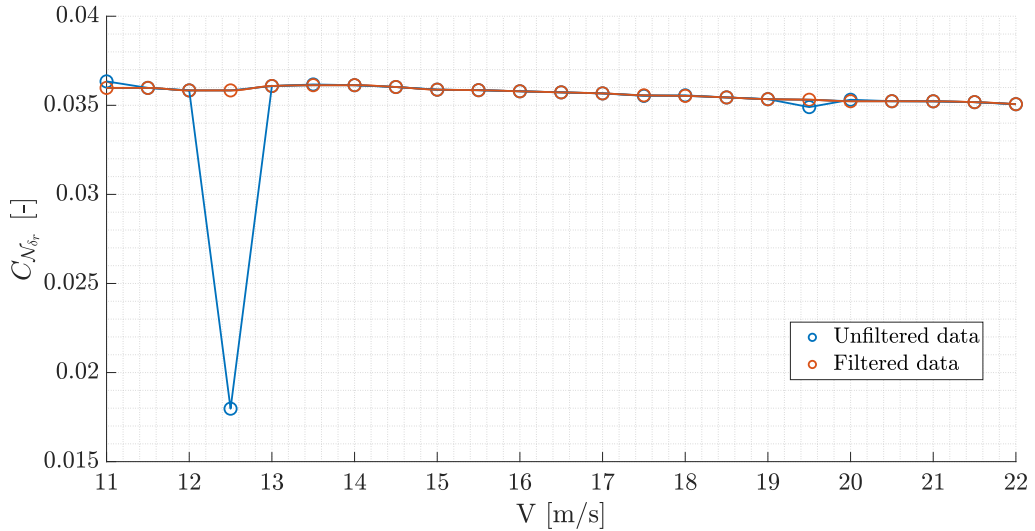


Figure 4.5: $C_{N_{\delta_r}}$ coefficient as a function of airspeed, filtered and unfiltered data.

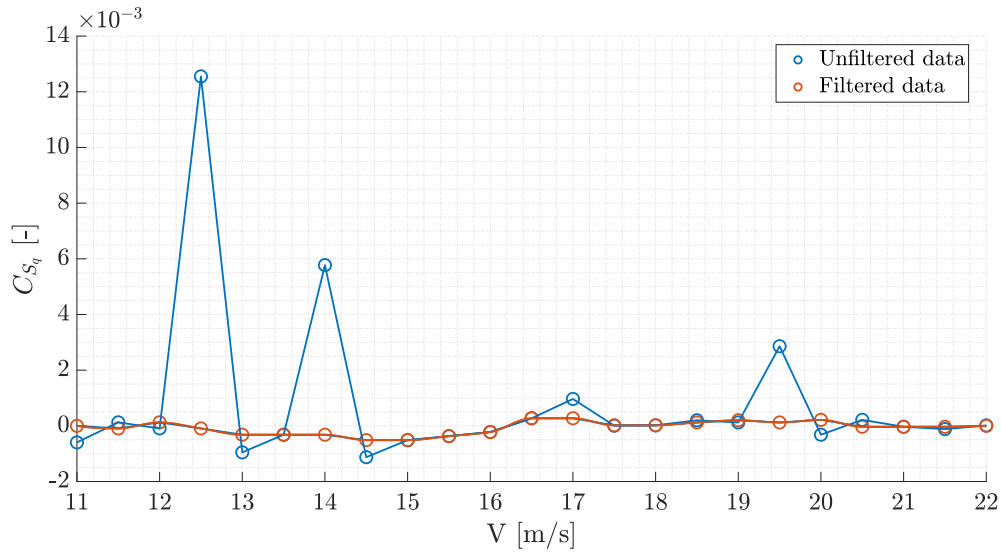


Figure 4.6: C_{S_q} coefficient as a function of airspeed, filtered and unfiltered data.

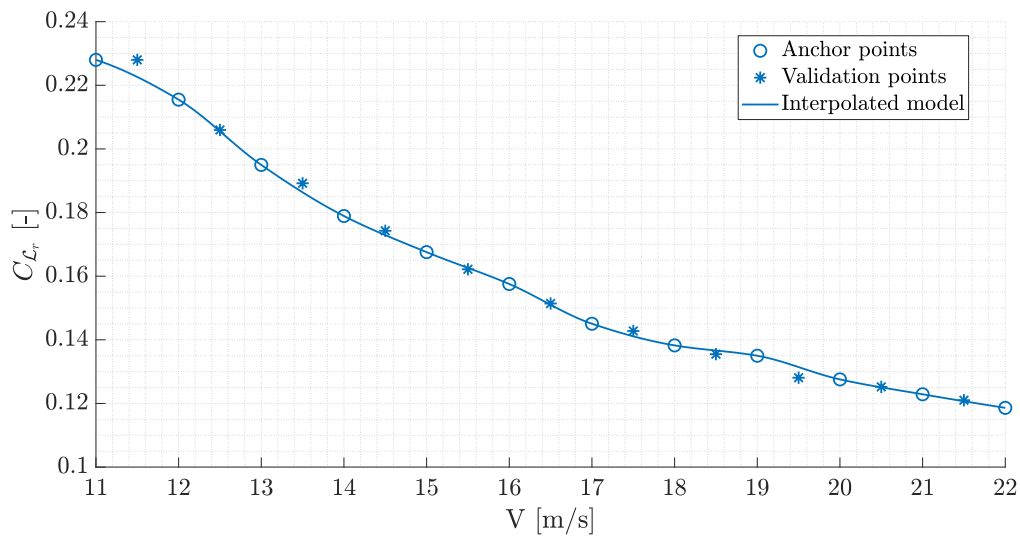


Figure 4.7: C_{L_r} coefficient as a function of airspeed, interpolated model.

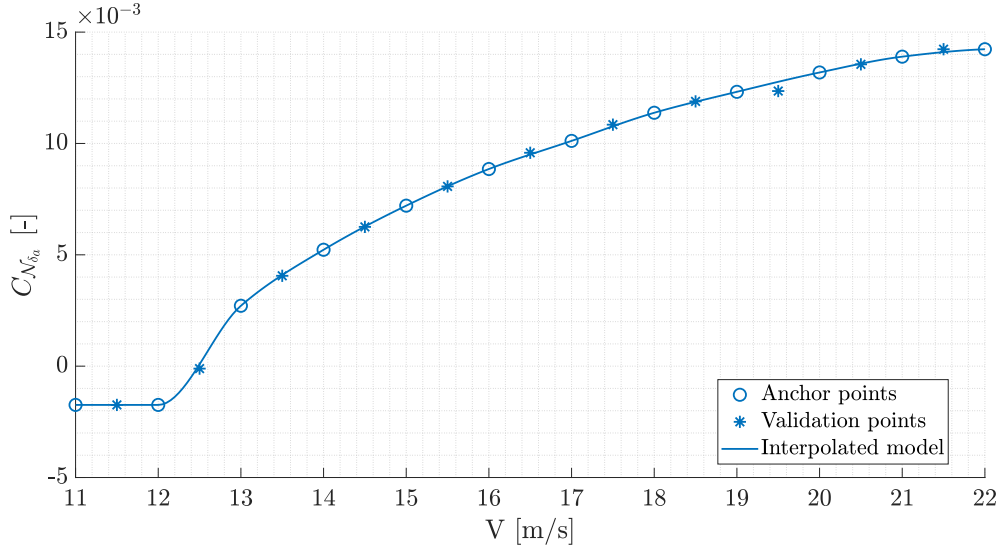


Figure 4.8: $C_{N_{\delta_a}}$ coefficient as a function of airspeed, interpolated model.

The interpolated model fits very well the validation dataset, hence confirming the validity of the interpolation method. It has to be kept in mind, though, that in the actual simulation all the available aerodynamic dataset is used as anchor points for the simulator, hence providing even more accuracy in the aerodynamic estimation.

4.3 Simulator validation with linearized response

In order to validate the simulator, a comparison between the response of the non-linear dynamic model in *Simulink* and the classical linearized model of the aircraft's longitudinal and lateral-directional dynamics (given by equation (4.1) and equation (4.2)) has been done.

$$\mathbf{M}_{\text{LON}} \dot{\mathbf{x}}_{\text{LON}} + \mathbf{K}_{\text{LON}} \mathbf{x}_{\text{LON}} = \mathbf{U}_{\text{LON}} \mathbf{u}_{\text{LON}}, \quad (4.1)$$

$$\mathbf{M}_{\text{LAT}} \dot{\mathbf{x}}_{\text{LAT}} + \mathbf{K}_{\text{LAT}} \mathbf{x}_{\text{LAT}} = \mathbf{U}_{\text{LAT}} \mathbf{u}_{\text{LAT}}, \quad (4.2)$$

with

$$\mathbf{x}_{\text{LON}} = \begin{bmatrix} u \\ \Delta\alpha \\ \Delta q \\ \Delta\theta \end{bmatrix}, \quad \mathbf{u}_{\text{LON}} = \begin{bmatrix} \Delta\delta_{Thr} \\ \Delta\delta_e \end{bmatrix}, \quad \mathbf{x}_{\text{LAT}} = \begin{bmatrix} \Delta\beta \\ \Delta p \\ \Delta r \\ \Delta\phi \\ \Delta\psi \end{bmatrix}, \quad \mathbf{u}_{\text{LAT}} = \begin{bmatrix} \Delta\delta_a \\ \Delta\delta_r \end{bmatrix},$$

$$\mathbf{M}_{\text{LON}} = \begin{bmatrix} m_1 & -c_1 C_{X_{\dot{\alpha}}} & 0 & 0 \\ 0 & -c_1 C_{Z_{\dot{\alpha}}} + m_1 & 0 & 0 \\ 0 & -c_1 C_{M_{\dot{\alpha}}} & J_{y_1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{U}_{\text{LON}} = \begin{bmatrix} C_{X_{\delta_{Thr}}} & C_{X_{\delta_e}} \\ C_{Z_{\delta_{Thr}}} & C_{Z_{\delta_e}} \\ C_{M_{\delta_{Thr}}} & C_{M_{\delta_e}} \\ 0 & 0 \end{bmatrix},$$

$$\begin{aligned}
\mathbf{K}_{\text{LON}} &= \begin{bmatrix} -C_{X_u} & -C_{X_\alpha} + m_1 q & -c_1 C_{X_q} + m_1 \frac{W_0}{U_0} & m_1 \frac{g}{U_0} \cos \theta_0 \\ -C_{Z_u} - m_1 q_0 & -C_{Z_\alpha} & -c_1 C_{Z_q} - m_1 & m_1 \frac{g}{U_0} \sin \theta_0 \\ -C_{M_u} & -C_{M_\alpha} & -c_1 C_{M_q} & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix}, \\
\mathbf{M}_{\text{LAT}} &= \begin{bmatrix} -b_1 C_{Y_{\dot{\beta}}} + m_1 & 0 & 0 & 0 & 0 \\ -b_1 C_{\mathcal{L}_{\dot{\beta}}} & J_{x_1} & -J_{xz_1} & 0 & 0 \\ -b_1 C_{N_{\dot{\beta}}} & -J_{xz_1} & J_{z_1} & 0 & 0 \\ 0 & 0 & 0 & 1 & -\sin \theta_0 \\ 0 & 0 & 0 & 0 & \cos \theta_0 \end{bmatrix}, \quad \mathbf{U}_{\text{LAT}} = \begin{bmatrix} C_{Y_{\delta_a}} & C_{Y_{\delta_r}} \\ C_{\mathcal{L}_{\delta_a}} & C_{\mathcal{L}_{\delta_r}} \\ C_{N_{\delta_a}} & C_{N_{\delta_r}} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \\
\mathbf{K}_{\text{LAT}} &= \begin{bmatrix} -C_{Y_\beta} & -b_1 C_{Y_p} - m_1 \frac{W_0}{U_0} & -b_1 C_{Y_r} + m_1 & -m_1 \frac{g}{U_0} \cos \theta_0 & 0 \\ -C_{\mathcal{L}_\beta} & -b_1 C_{\mathcal{L}_p} - q_0 J_{xz_1} & -b_1 C_{\mathcal{L}_r} + q_0 (J_{z_1} - J_{y_1}) & 0 & 0 \\ -C_{N_\beta} & -b_1 C_{N_p} + q_0 (J_{y_1} - J_{x_1}) & -b_1 C_{N_r} + q_0 J_{xz_1} & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix}, \\
m_1 &= \frac{m}{\frac{1}{2}\rho U_0 S}, \quad b_1 = \frac{b}{2U_0}, \quad c_1 = \frac{c}{2U_0}, \quad J_{x_1} = \frac{J_x}{\frac{1}{2}\rho U_0^2 S b}, \quad J_{y_1} = \frac{J_y}{\frac{1}{2}\rho U_0^2 S c}, \\
J_{z_1} &= \frac{J_z}{\frac{1}{2}\rho U_0^2 S b}, \quad J_{xz_1} = \frac{J_{xz}}{\frac{1}{2}\rho U_0^2 S b}.
\end{aligned} \tag{4.3}$$

Singlets¹ and doublets² excitations were given to each control input (elevator, ailerons and rudders) and the aircraft's response was observed. In Figure 4.9, Figure 4.10 and Figure 4.11 the aircraft's responses to doublets to the elevator, ailerons and rudders respectively are shown, obtained with the *Simulink* non-linear simulator (solid lines) and with the linearized model of equation (4.1) and equation (4.2) (dashed lines).

¹A singlet refers to a half square cycle input on the control: neutral-positive-neutral.

²A doublet refers to a full square cycle input: neutral-positive-negative-neutral.

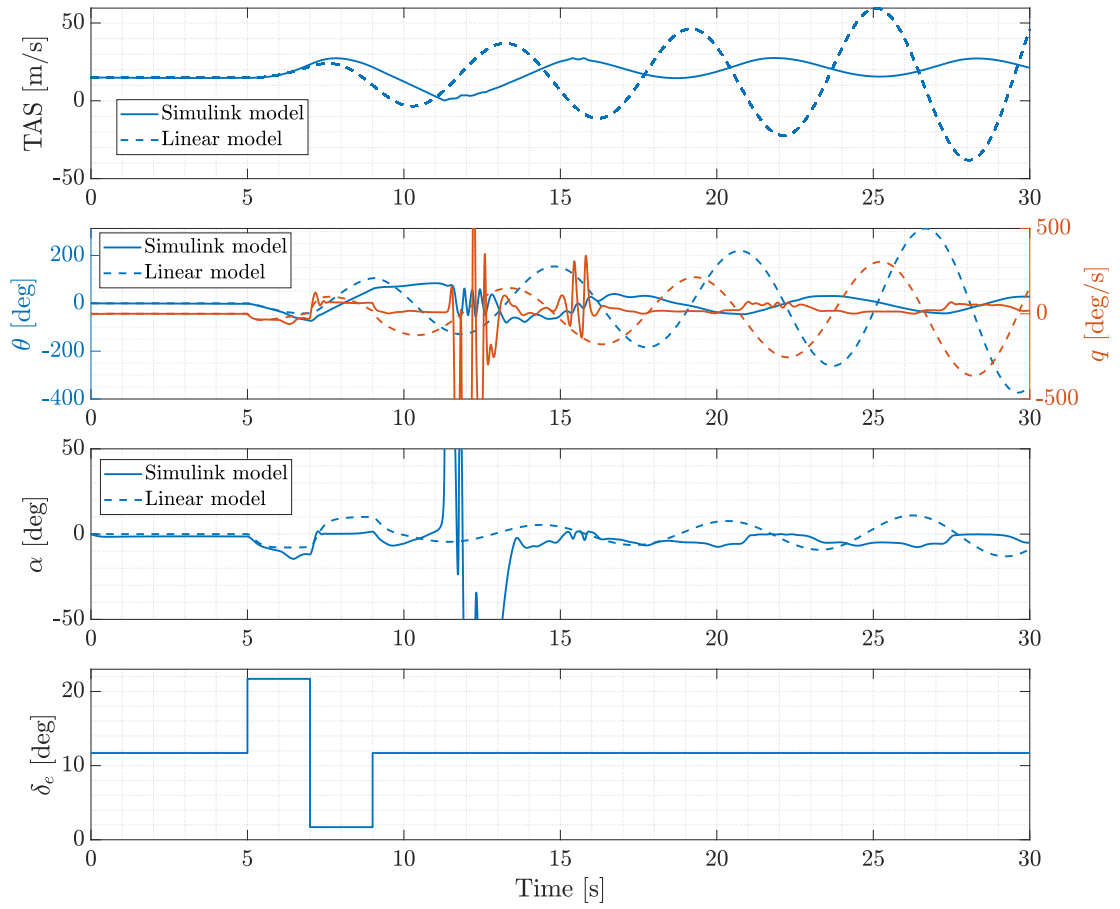


Figure 4.9: Time histories of the aircraft's longitudinal response after elevator doublet, *Simulink* non linear model and linear model.

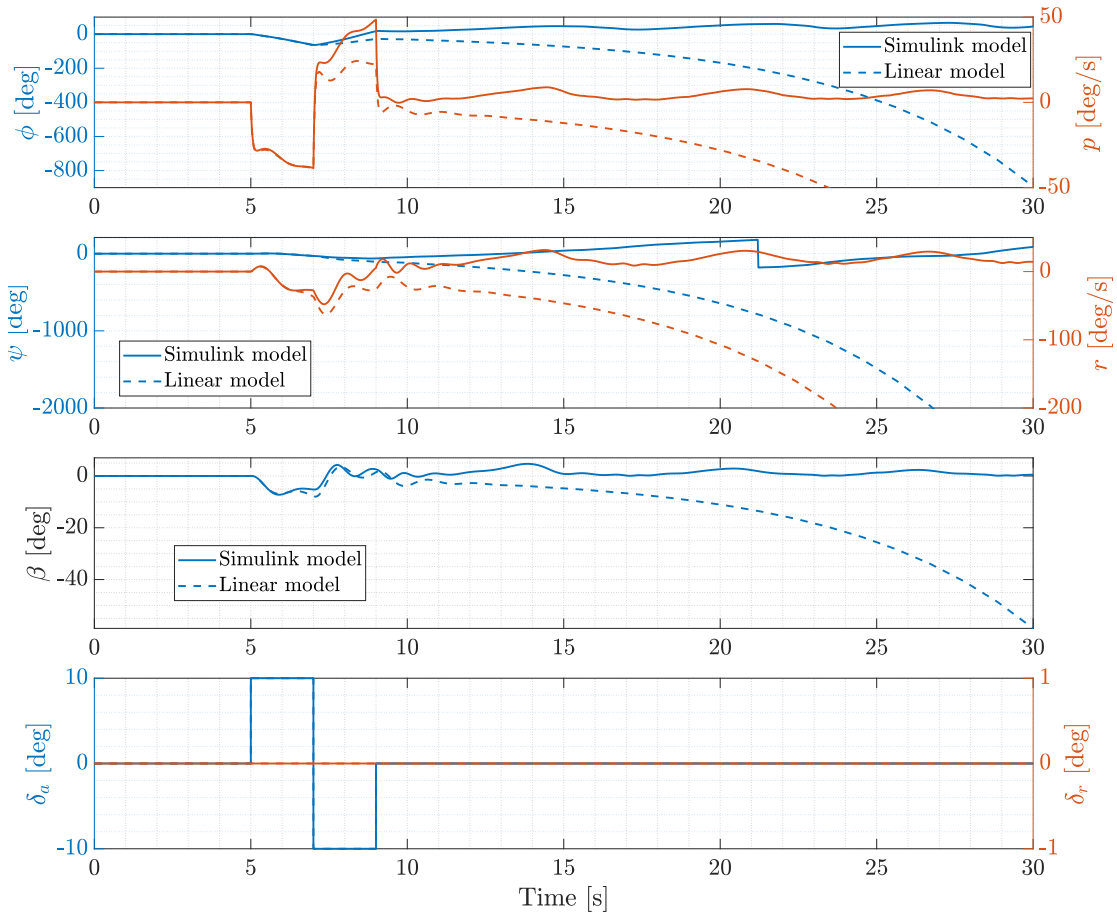


Figure 4.10: Time histories of the aircraft's lateral-directional response after aileron doublet, *Simulink* non linear model and linear model.

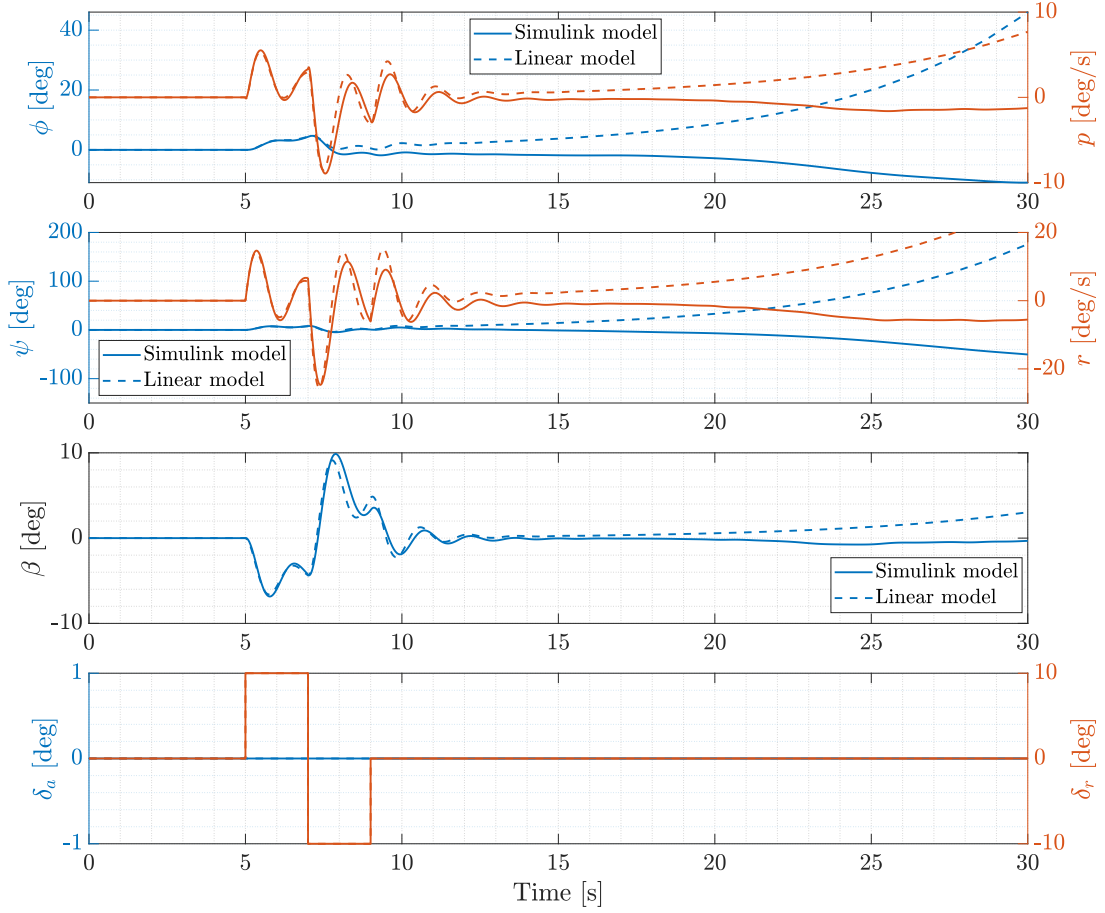


Figure 4.11: Time histories of the aircraft's lateral-directional response after rudder doublet, *Simulink* non linear model and linear model.

The short period motions (such as the short period and the dutch roll) are almost overlapping between the two models. The long period evolutions however show significant difference, since in the linear model the aircraft is unstable and the parameters start diverging, whereas in the non-linear model a stable condition is eventually reached. Due to the more detailed and accurate model of the *Simulink* simulator, it will be used as the sole reference for the following analyses.

4.4 Open loop forward flight dynamics and performance

During the original design work of the UAV in [1], the aircraft's flight characteristics were evaluated, using the *XFLR5* aerodynamic model consisting of only the lifting surfaces. Thanks to the improved model developed in the present work (described in Chapter 1), it was possible not only to increase the accuracy

of the preliminary estimates for the aircraft's flight dynamics, but also to have the parameters of the dynamic modes (eigenvalues, damping and frequency) as a function of airspeed, within the UAV's flight envelope.

Table 4.1 summarizes the eigenvalues of the original analysis and the ones obtained with the new improved aerodynamic model, at the cruise speed of 15 m/s. Table 4.2 draws the same comparison but reporting the modes' frequency and damping and the associated rating according to the MIL F 8785 C³ specification⁴ ([32]). Figure 4.12 shows the aircraft's eigenvalues as a function of airspeed.

Table 4.1: Eigenvalues of the aircraft's dynamic modes during the forward flight phase, original model from [1] and improved model.

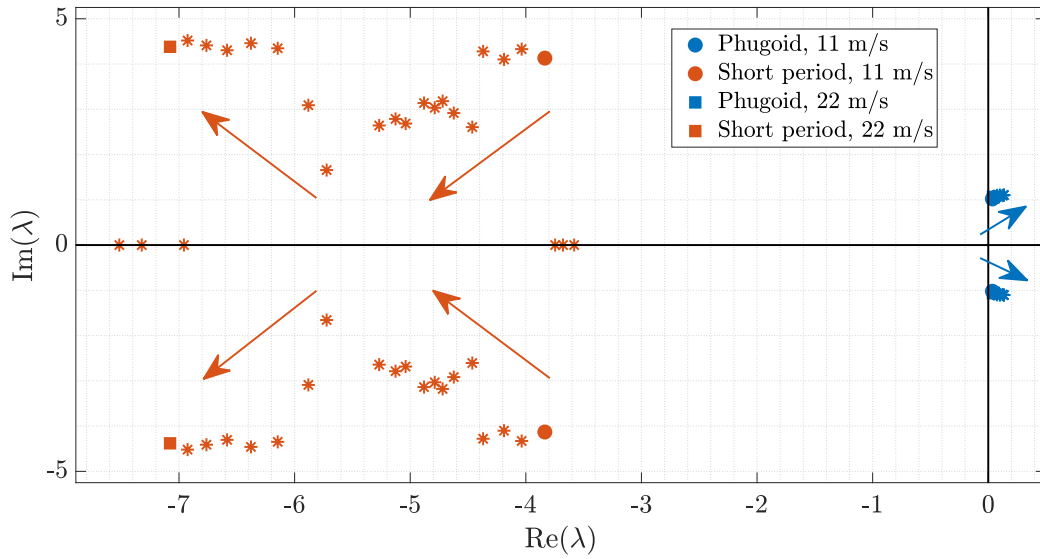
Mode	Original model of [1]		Improved model	
	Real part	Imaginary part	Real part	Imaginary part
Phugoid	-0.0053	0.59	0.06	1.0
Short period	-5.3	4.3	-4.9	3.1
Dutch roll	-1.0	4.1	-0.93	4.1
Spiral	0.18	0.0	0.18	0.0
Roll	-21.2	0.0	-20	0.0

Table 4.2: Frequency and damping of the aircraft's dynamic modes during the forward flight phase, original model from [1] and improved model.

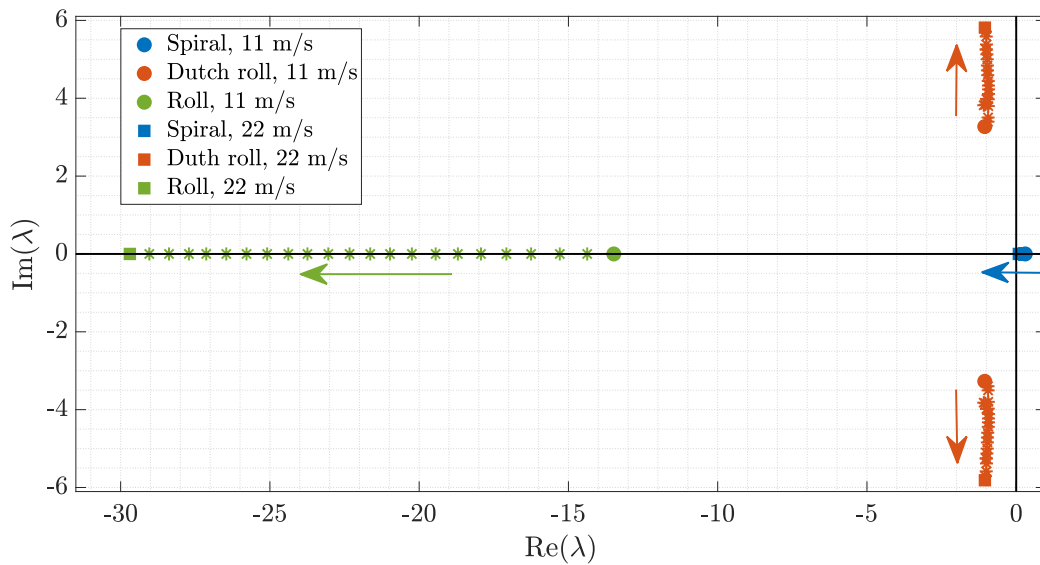
Mode	Original model of [1]		Improved model		
	Frequency [Hz]	Damping	Frequency [Hz]	Damping	Rating
Phugoid	0.57	0.01	1.06	-0.056	Level 3
Short period	6.79	0.78	5.80	0.84	Level 1
Dutch roll	4.21	0.24	4.21	0.22	Level 1
Spiral	0.19	-1.00	0.17	-1.00	Level 3
Roll	21.17	1.00	20.24	1.00	Level 1

³A rating of Level 1 describes the aircraft's flying qualities as being clearly adequate for the mission; Level 2 rating requires increased workload to achieve the desired mission and for Level 3 the aircraft can be controlled safely but with excessive pilot workload, with inadequate mission effectiveness. The reference aircraft is class 1 (small, light airplanes) with flight phase B (non-terminal flight phases with gradual maneuvers).

⁴Note that this regulation is actually out of context for a light and small UAV. However, it is the only standard available for the evaluation of the aircraft's flying qualities and hence it can provide useful information anyway.



(a) Longitudinal eigenvalues.



(b) Lateral-directional eigenvalues.

Figure 4.12: UAV's eigenvalues, longitudinal (a) and lateral-directional (b), as a function of airspeed. The arrows indicate the direction of increasing airspeed.

The results from the new model are similar to the preliminary estimates: the short period, dutch roll and roll modes are all stable. The spiral mode remains unstable, as already noted out in [1], together with the phugoid mode, which passes from being slightly stable in the original analysis to being slightly unstable with the new, improved model. The change in frequency and damping for the long period modes (phugoid and spiral) with respect to airspeed is small, with the

phugoid becoming more unstable for increasing airspeeds and the spiral becoming more stable for increasing airspeeds. The variation of the characteristics of the short period modes (short period, dutch roll and roll) with airspeed is more significant, with all the three modes increasing their frequency as airspeed increases.

All the aircraft's modes are well captured by the *Simulink* simulator. Just as an example, the following scenario has been used to visualize the phugoid and short period modes. The aircraft, starting from a precise trim condition, manually found, at the cruise speed of 15 m/s, after 5 seconds is excited with an elevator singlet of 10 degrees amplitude and of 1 second duration. Figure 4.13 shows the aircraft's trajectory, whereas Figure 4.14 and Figure 4.15 show the aircraft's longitudinal and lateral-directional response. The initial phugoid motion can be clearly seen, with the successive descending spiral. The phugoid mode appears to be actually stable, since its oscillations die out by the end of the simulation (of 3 minutes duration). Hence, the phugoid mode is actually slightly stable.

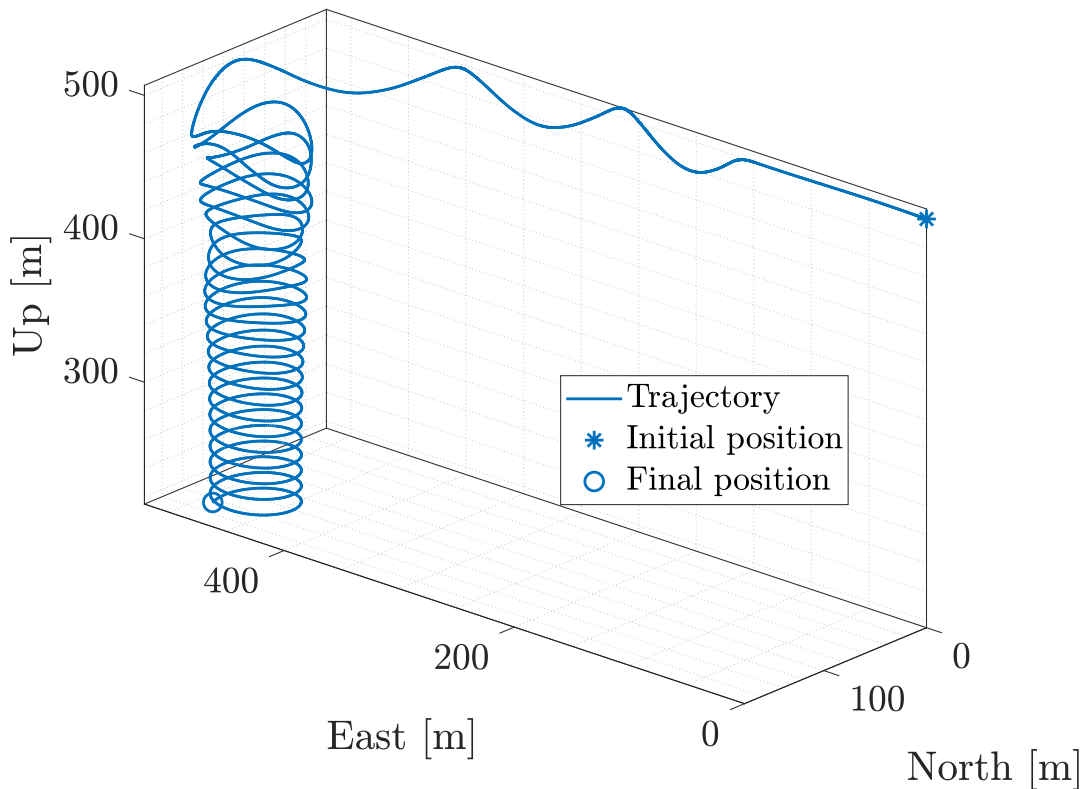


Figure 4.13: Trajectory of the aircraft after an elevator singlet excitation. The initial phugoid and successive descending spiral motions are clearly visible.

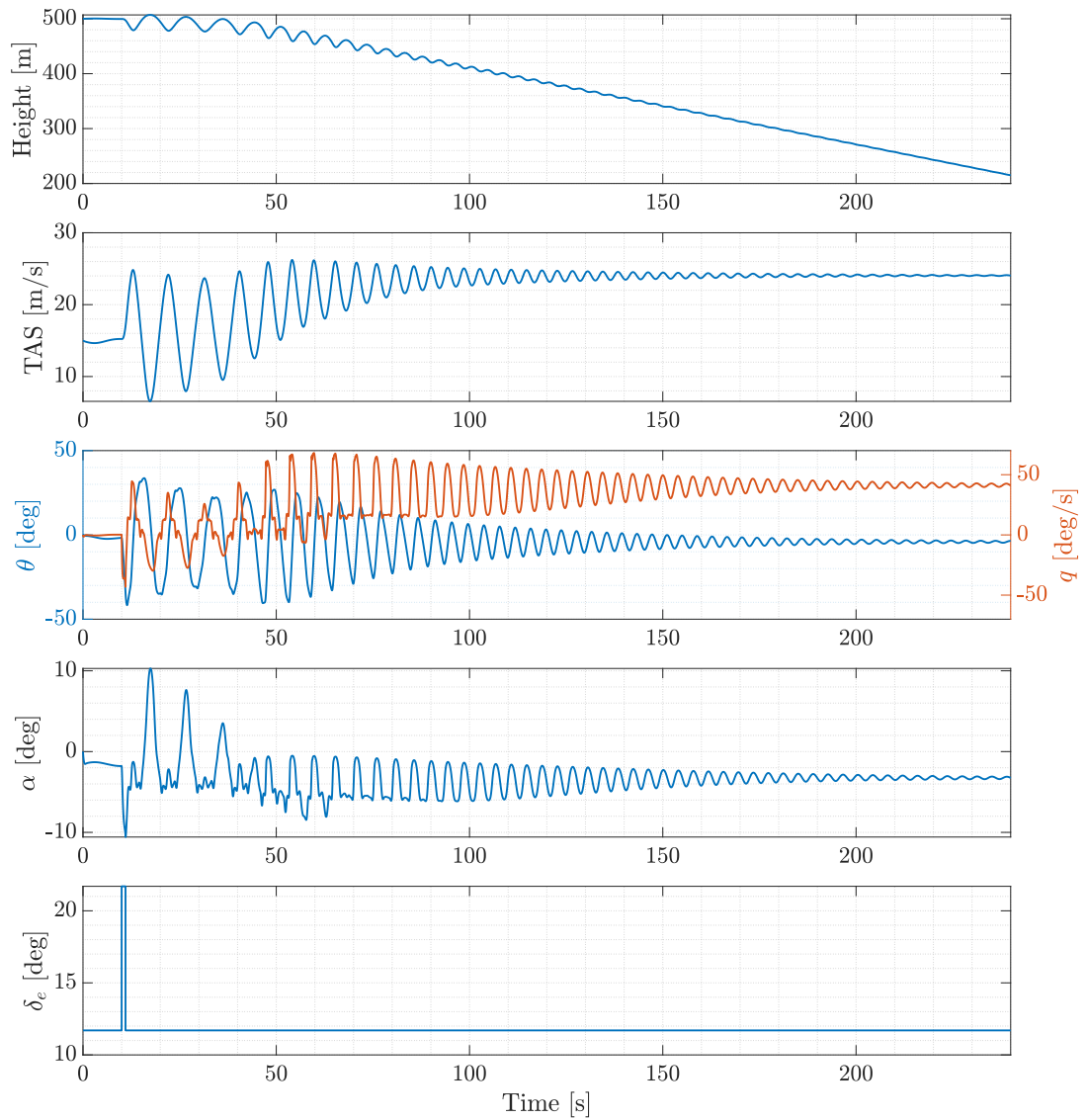


Figure 4.14: Time histories of the aircraft's longitudinal response after elevator singlet.

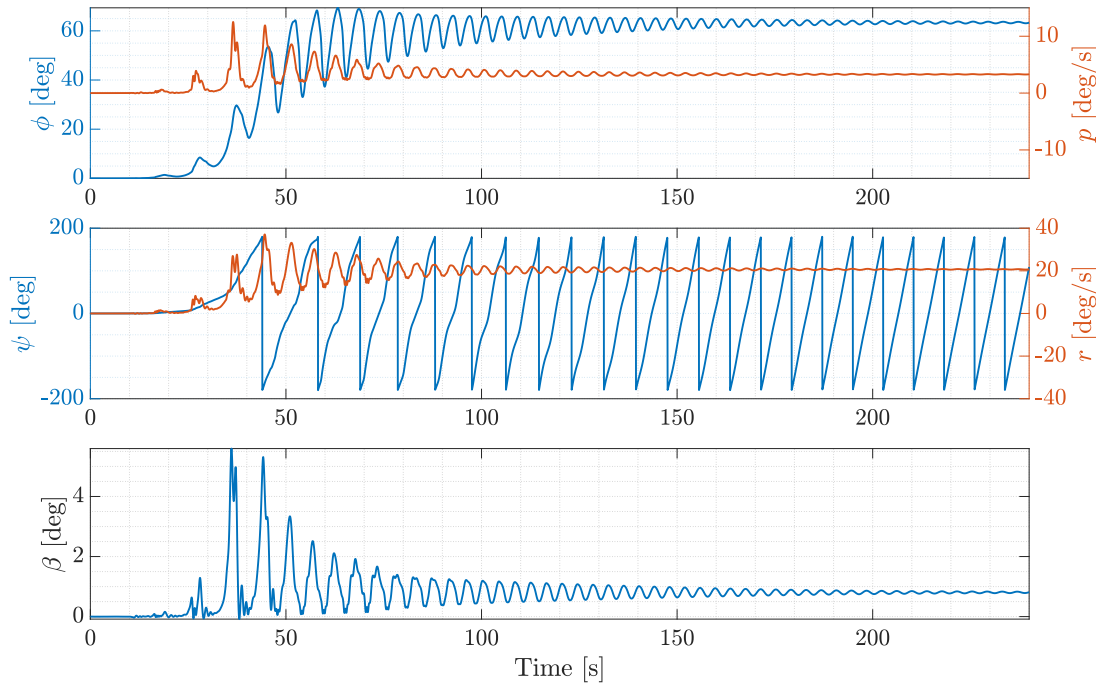


Figure 4.15: Time histories of the aircraft's lateral-directional response after elevator singlet.

Better estimates for the aircraft's endurance and range were also calculated. By using a simple proportional controller to stabilize the aircraft's unstable spiral mode, it is possible to trim the vehicle at a constant airspeed and altitude, and hence to evaluate its maximum endurance and range. In Table 4.3 such performance parameters are presented, together with the trim conditions for the three characteristic airspeeds of the UAV: the stall speed of 11 m/s, the cruise speed of 15 m/s and the maximum airspeed of 22 m/s. These performance figures were calculated with the aircraft, without any additional payload, starting from a stable and level attitude at the reference airspeed and at the cruising height of 120 m AGL, battery charged at 100% and the battery discharging to 15%.

Table 4.3: Endurance and range for different airspeeds.

Airspeed	Trim elevator deflection ¹ [deg]	Trim throttle percentage [%]	Endurance [min]	Range [km]
11 m/s	1	35.7	83	53.8
15 m/s	11.7	32.4	98	87.1
22 m/s	14.4	51.7	41	52.6

¹ Positive deflection for trailing edge down.

The original requirement on the aircraft's endurance from [1] was of 85 minutes, together with a 5 minutes hover. Given that at the cruising airspeed the endurance estimate is of 98 minutes, the requirement is most certainly met and probably exceeded.

Chapter 5

Integration

In this chapter the integration and assembly process of the UAV is described. For each main component of the vehicle (wings, tail, fuselage and motor booms), changes with respect to the original design of [1] are outlined, and then the assembly process is both thoroughly reported and documented with pictures. The first four sections are hence each one dedicated to one aircraft component. They are presented in the following order: starting from the wings, then tail, fuselage and finally motor booms. In the last section the assembly of those four main components is described.

5.1 Wing

5.1.1 Changes with respect to the original design

The original wing design, which was based on the UAVs built by the University of Southampton described in [33], had to be changed.

The preliminary wing structural layout, as described in [1], was made up of mixed materials:

- carbon fiber tubes for the spars;
- hot wire cut foam for the aerodynamic shape;
- Mylar cladding to protect the foam during ground handling;
- 3D printed ribs to transfer loads to the spar and to connect the foam blocks.

A drawing of the original layout is presented in Figure 5.1.

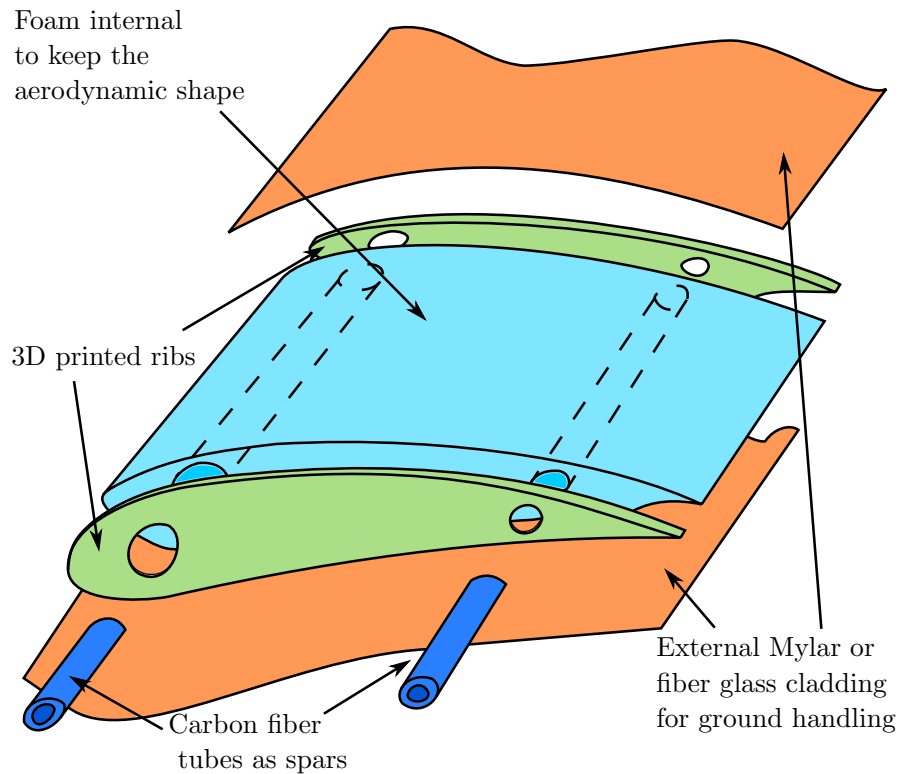


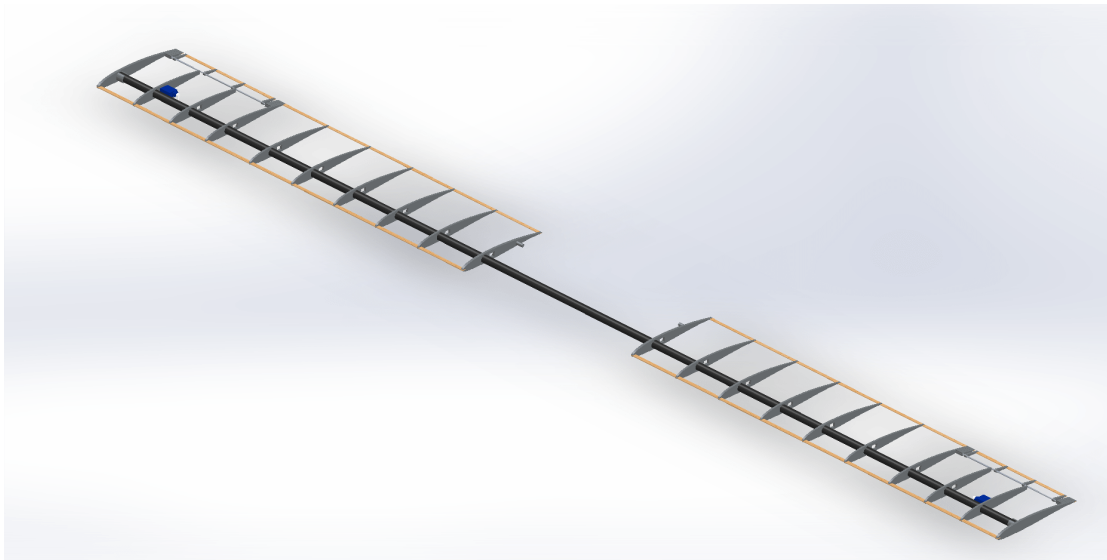
Figure 5.1: Original design of the wing, from [1].

When contacting manufacturers for the production of the wing, a problem arose with the fact that the wing profile was too thin to be accurately cut with the hot wire technique in Styrofoam. It could be cut by hand, but this would require a lot of careful hand work and hence it was decided to look for other feasible manufacturing options. Another possibility could be to 3D print the whole wing; however, this would be difficult from the manufacturing point of view, since the semi-span of each wing is quite big (approximately 0.9 metres). The last option identified was to increase the number of wing ribs and to cover them with the Mylar cladding, without having any filler material inside the wing. This was deemed to be the easiest and most straightforward way; following the design principle of low cost and simplicity which was the driver of the design in [1], this last alternative was chosen. Hence, the new design consists of the following components:

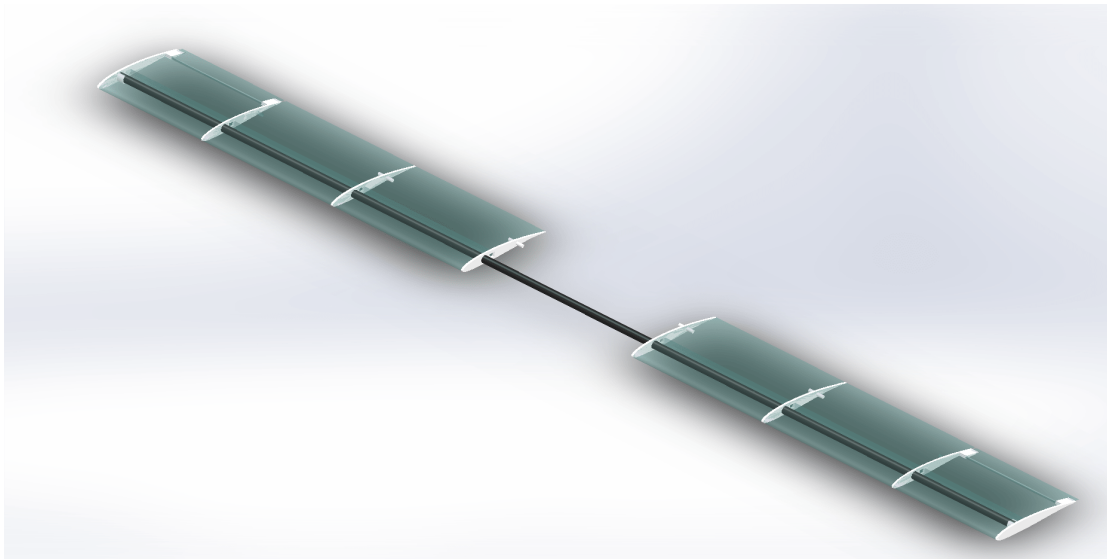
- carbon fiber tubes for the spars;
- Mylar cladding used as skin, to cover up the wing structure;
- 3D printed ribs to transfer loads to the spar;
- thin wood laths running across the wingspan on the trailing and leading edges, to support and assure a smooth cover.

Note that the ailerons will also be built using the same technique. Another addition with respect to the original work was the actual design of the supports and the mechanism of the servomotors which actuate the ailerons.

Figure 5.2 shows the new and original wing design. Note that the colors are chosen so to have a better image quality, and do not represent the final materials colours. In the new design, the increased number of ribs, the additions of the wood laths and the lack of the filler material can be appreciated.



(a) New wing design.



(b) Original wing design.

Figure 5.2: CAD models of the (a) new and (b) original wing designs. Colors are not representative of the real materials used.

5.1.2 Building and integration

The building phase began with arranging all the wings' structural components in place, without any glue, in order to evaluate if all the dimensions and production tolerances were correct. The ribs were inserted one by one inside the wing spar and subsequently the wood laths for the leading and trailing edge were positioned. Finally, on the left wing, the Pitot tube and its transducer chip were positioned. Once the design was validated, glue was applied on the contact area of the woods laths with the ribs, of the ribs with the wing spar and of the Pitot tube with its supporting rib. At this point the load carrying wing structure was completed. Figure 5.3 shows the final result for this first phase and Figure 5.4 shows the detail for the aileron part of the wing structure.



Figure 5.3: Wing load carrying structure, both wings.

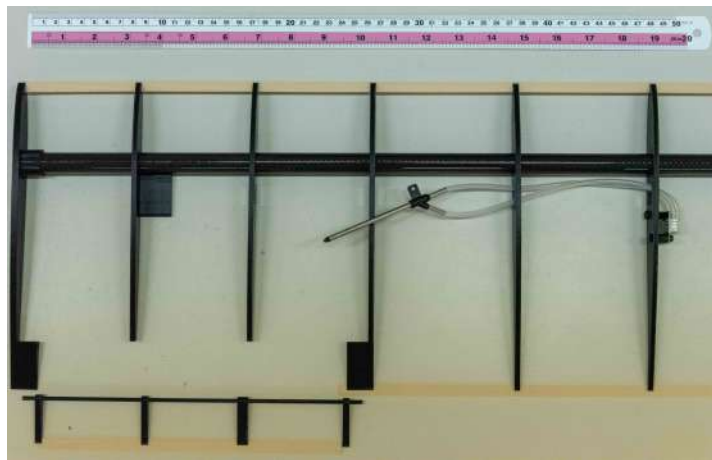


Figure 5.4: Detail of the aileron area of the load carrying structure, left wing.

The next step was to position the servomotors (which are the actuators for the ailerons) and their extension cable which runs outside the wing spar and connects to the Power Distribution Board inside the fuselage. It was decided to route the wire outside the spar, to avoid having to drill a hole inside the carbon fibre tube. The wire is kept in place by zip ties. Before proceeding with the wing coverage, the aileron servomotors were tested using an *Arduino* to ensure their correct functioning. A similar wire routing was done for the Pitot tube wire, for the left wing. Figure 5.5 shows the complete wiring for the right wing, whereas Figure 5.6 portrays the final position of the Pitot tube with its wiring in place.

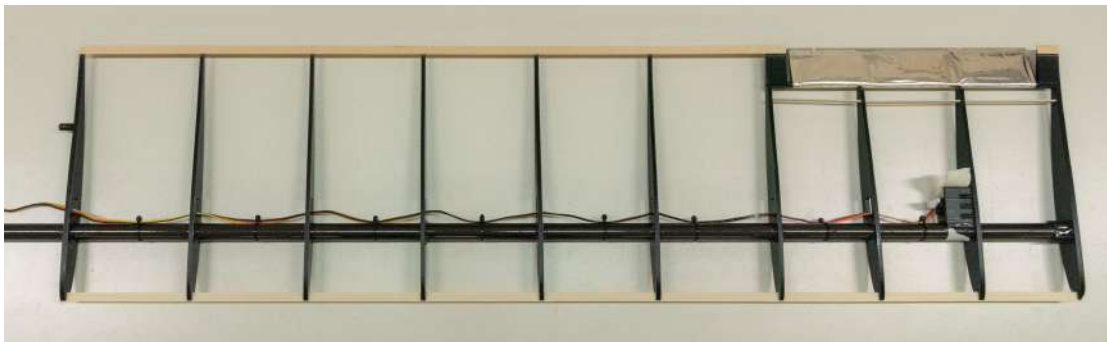


Figure 5.5: Wing structure with complete servomotor and wiring, right wing.

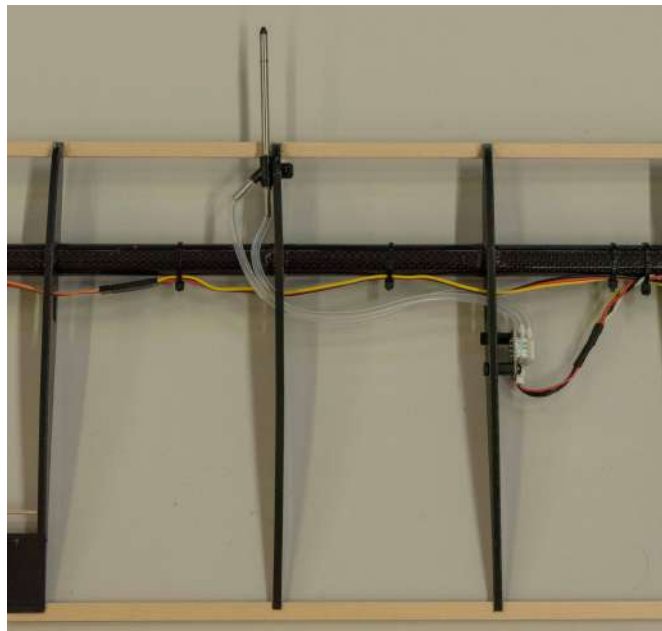


Figure 5.6: Detail of the Pitot area of the wing with complete wiring, left wing.

The following phase of the wing production was its coverage with the Mylar cladding. This material came in a spool and was sticky on one side (similar to tape). Some first trial and error coverage procedures were done on the ailerons.

However, a problem arose: the adhesion between the Mylar cover and some parts of the wing (namely the section near the trailing edge, leading edge and the wing spar) caused some deformations in the external shape of the wing, leading to a probable degradation of the aerodynamic performance. Hence, it was decided to cover the internal sticky sections of the skin which were in contact with such problematic parts with paper, to eliminate the stickiness of the material. This solution proved to be satisfactory and it was adopted for the whole wing. It was decided to cover each section between two adjacent ribs separately, because it was the only option, given the width of the available Mylar spool. Figure 5.7 shows the main steps of the coverage procedure. First, the piece of Mylar was cut to a correct shape and the parts that were in contact with the section near the trailing edge, the wing spar and the leading edge were covered with paper. Then, it would be positioned on one side of the wing and a good adhesion with the two adjacent ribs was achieved, by pulling and stretching the Mylar piece (Figure 5.7a). Then, the Mylar piece would be turned over to cover the remaining part of the wing, again taking care of stretching it so to achieve the least possible wrinkling (Figure 5.7b). This procedure was then repeated for all the wing sections, with cuts in the skin made in correspondence of the servomotors arms and of the Pitot tube. This procedure turned out to be somewhat difficult, with the need of at least two people.



(a) Step one: apply Mylar cladding on one side of the wing.



(b) Step two: wrap the Mylar cladding around the wing.

Figure 5.7: Wing coverage procedure.

Once both wings were covered, the final step was to attach the control horn on the aileron and the control rod that connects the aileron servomotor arm on the wing to the aileron horn. The actuation mechanism was hence complete and was successfully tested. The maximum deflection range was manually measured and was found to be $+32^\circ/-45^\circ$ for the left aileron and $+40^\circ/-35^\circ$ for the right aileron, with the control being both accurate and stiff. Figure 5.8 shows the completed control mechanism, Figure 5.9 shows a detail of the wing area around the Pitot tube and Figure 5.10 shows the completed wings, ready to be installed.

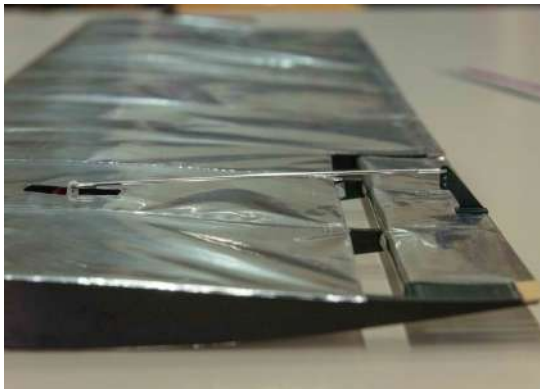


Figure 5.8: Detail of the aileron control mechanism, right wing.

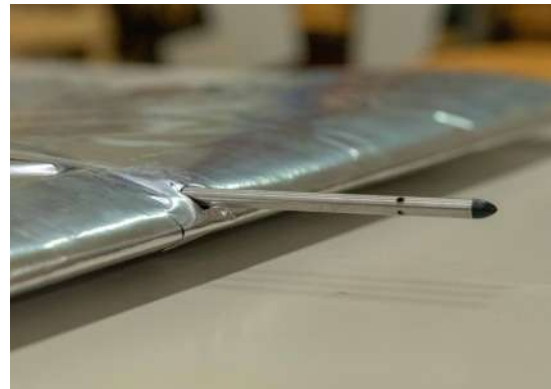


Figure 5.9: Detail of the wing's Pitot tube, left wing.



Figure 5.10: Completed wings, ready to be installed.

The main issue encountered with this design consists in the wrinkles present on the wing's skin, visible in Figure 5.10. This problem can be mitigated by stretching the Mylar cladding equally along all its directions as it is applied on each wing segment; however some corrugations are inevitable. The aerodynamic effect of this imperfection has to be evaluated either by wind tunnel or flight testing. Apart from this problem, this solution proved to be lightweight and stiff enough (the latter, however, has to be accurately evaluated again with actual testing) and the control surface design is rigid and precise. Minor skin issues (like small localized holes or tears) can be easily repaired with the application of a Mylar patch, whereas more serious damages to either ribs, the wing spar, Pitot or servomotor will most probably require more extensive maintenance work.

5.2 Tail

5.2.1 Changes with respect to the original design

The original tail design, see Figure 5.13c, also experienced some changes. Firstly, the production method, which originally was the same as the one for the wing, had to be changed as well, for the same reasons described previously in Section 5.1.1. However, due to the much smaller size of this component, it was decided to 3D print all the tail elements. Hence, the latter has been divided into 5 sub-sections: the two vertical stabilizers and the three parts which make the horizontal stabilizer (the two external ones and the central one). The spars for the two vertical surfaces and for the horizontal ones are kept in place as in the original design. Another major change is the size of the elevator, already described in Section 1.4 of Chapter 1. The position of the servo-motors was also studied in more detail and, as a first solution, the most straightforward installation was identified in being on the two booms, in vanes printed together with the pieces that allow the connection of the various horizontal and vertical parts of the tail. This original choice was driven by the fact that it was not possible to completely embed the servomotors inside the surfaces due to the limited thickness of the latter. After having printed and assembled the tail with the previous modifications, the servomotors were positioned, connected to the control surfaces utilizing thin metal rods used in RC aircraft and the actuation mechanism tested. This intermediate design is shown in Figure 5.13b and the control surfaces testing presented in Figure 5.11. Performance was not satisfactory: the control surfaces moved, but the movement was limited and not precise; moreover, the control mechanism was not stiff enough. This was probably caused by the deformation of the thin control rods. Hence, another design iteration was made, this time embedding the servomotors in the horizontal and vertical surfaces, so to have a direct push/pull on the rod configuration. Before proceeding with the production of the modified components, this new configuration was tested on the old tail, as shown in Figure 5.12,

and proved to work much better compared to the original design. Figure 5.13 shows the CAD model of the final (Figure 5.13a), intermediate (Figure 5.13b) and original (Figure 5.13c) version of the tail.



Figure 5.11: Test of the original control mechanism for rudders and elevator.

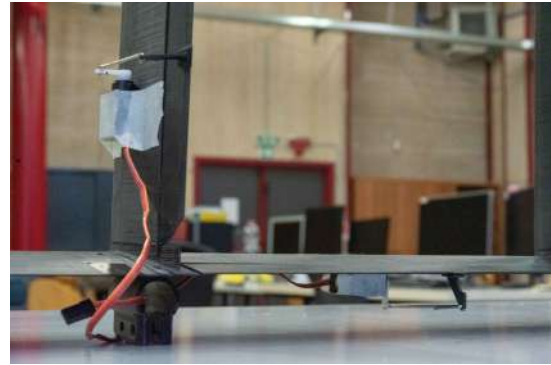
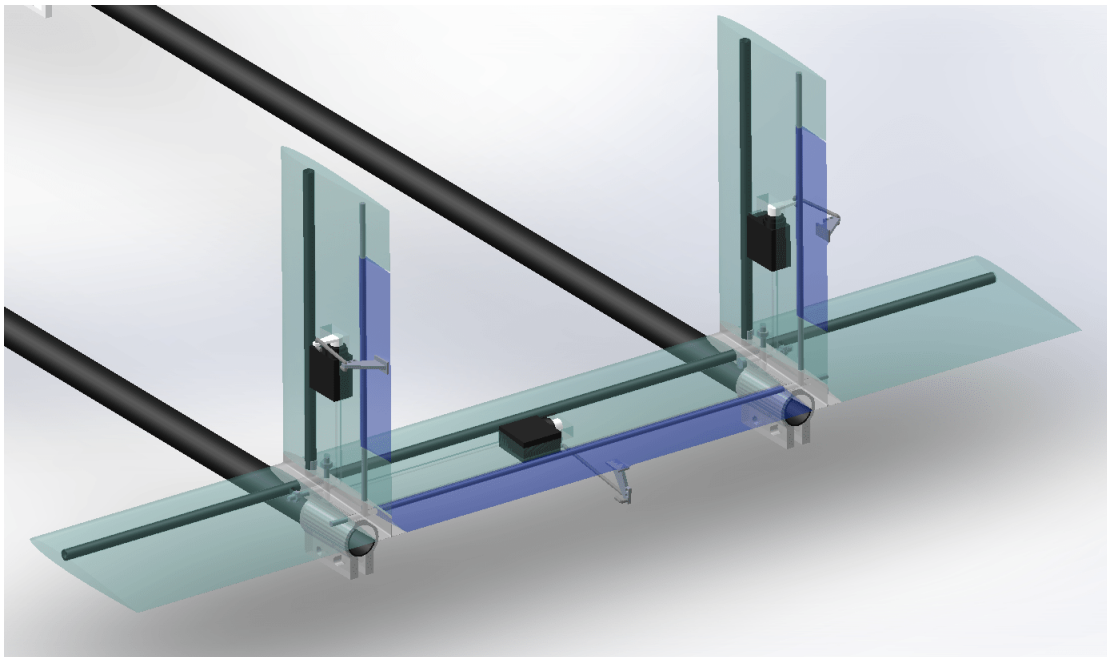
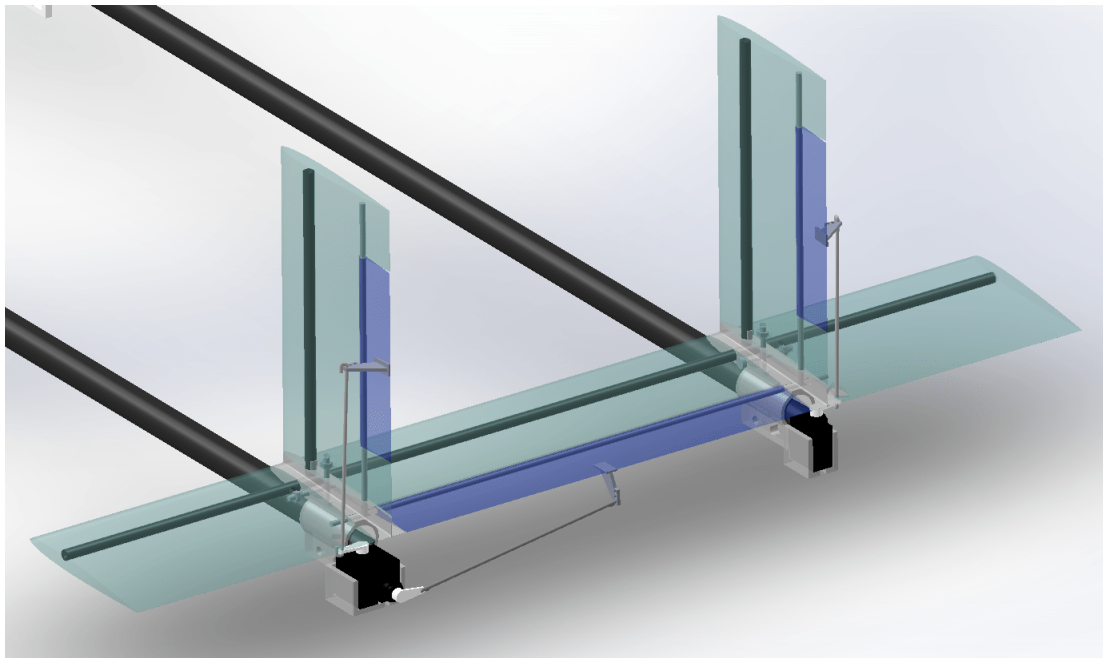


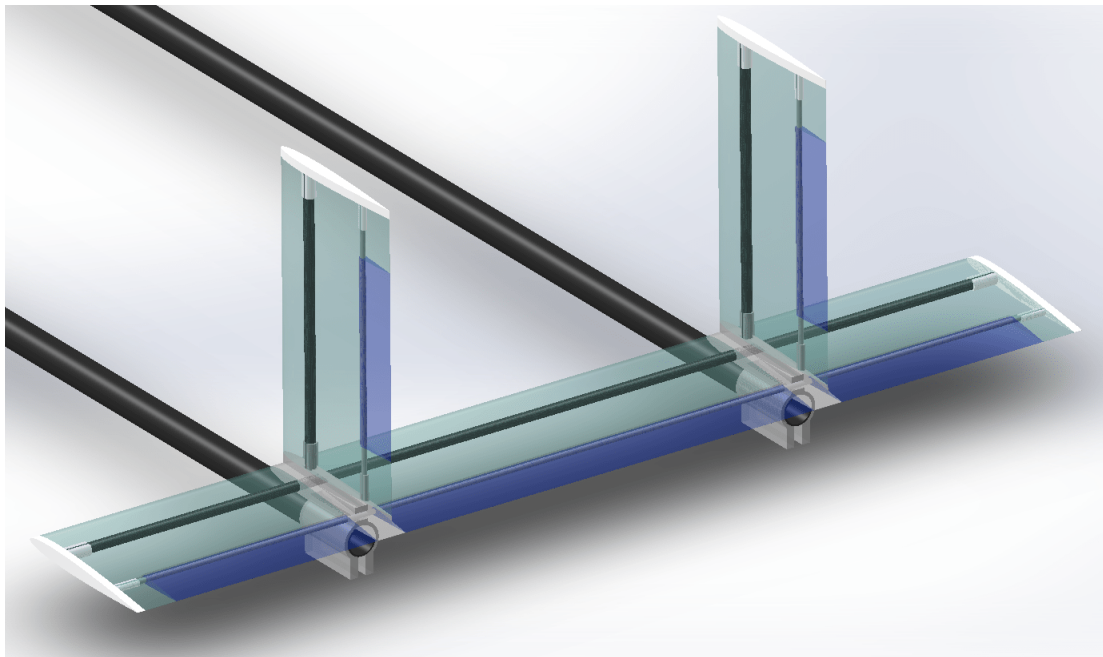
Figure 5.12: Test of the new control mechanism for rudders and elevator.



(a) CAD model of the new tail design.



(b) CAD model of the intermediate tail design.



(c) CAD model of the original tail design.

Figure 5.13: CAD drawing of the (a) new, (b) intermediate and (c) original tail design. The movable surfaces (elevator and rudders) are coloured in blue.

5.2.2 Building and integration

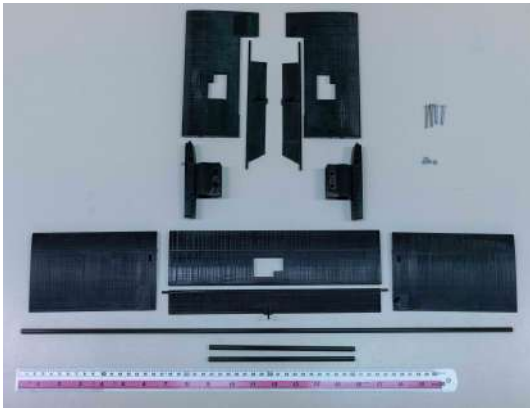
After cleaning the 3D printed components, the bottom part of both the left and right vertical stabilizers came loose. It was then decided to glue them to the rudders. However, due to their increased length, the rudders interfered with the elevator, when both of them were deflected at high angles towards the same direction. Hence, it was decided to cut away a triangular part in the lower part of both rudders, to solve this problem.

Figure 5.14 shows the mounting procedure for the whole tail. First, the vertical tail spar and the rudder is inserted in the vertical stabilizer (Figure 5.14b and Figure 5.14c). Then, the assembled vertical tail is screwed on the tail-boom support piece, which connects all the tail pieces together and attaches the whole tail to the rear motor booms (Figure 5.14d and Figure 5.14e). A nut, positioned in a dedicated hole on the vertical stabilizer, allows to firmly fix the screw, while maintaining a clean external aerodynamic profile. Afterwards, the horizontal stabilizer spar is inserted first in the assembled vertical tail, and then in the external horizontal stabilizer (Figure 5.14f and Figure 5.14g). The vertical tail and the external horizontal stabilizer are kept in place with a screw and nut mechanism similar to the one used for the vertical tail. Then, the central horizontal stabilizer is inserted in its spar and the elevator is positioned inserting its pivot inside a hole present in the tail-boom support piece (Figure 5.14h). Subsequently, the left vertical tail and external horizontal stabilizer are screwed together in the same way as it was done for the ones on the other side of the tail; then they are inserted in the horizontal stabilizer spar (Figure 5.14i and Figure 5.14j). Finally, the servomotors are glued and the connection rods attached. The protruding part of the servomotor is covered with tape, to improve the aerodynamic shape. Figure 5.15 shows the completed tail, ready to be assembled on the rear booms, whereas Figure 5.16a and Figure 5.16b show a detail of the nut and screw mounting mechanism for both the vertical and horizontal stabilizer.

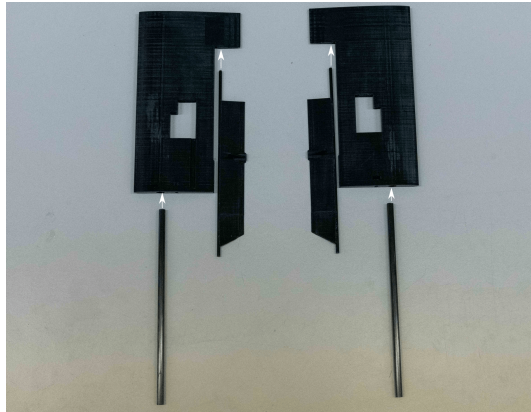
The maximum measured deflections for the rudders and elevator are summarized in Table 5.1; they are well beyond design requirements.

Table 5.1: Rudders and elevator measured maximum deflections.

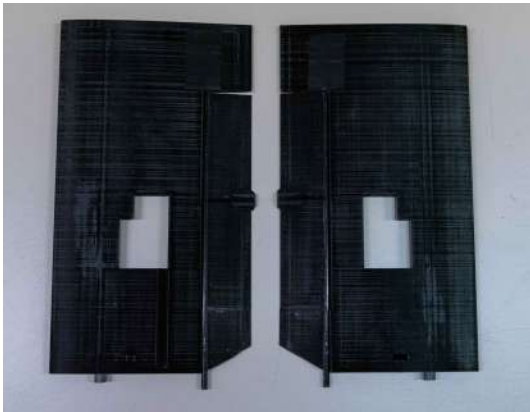
Left rudder	Right rudder	Elevator
40° Left	50° Left	+40° Down
50° Right	40° Right	−25° Up



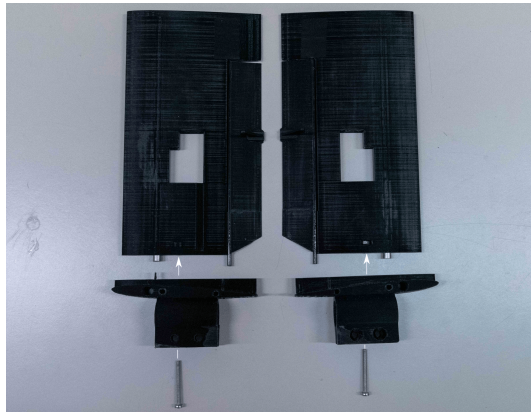
(a) Material needed.



(b) Step one: insert vertical tail spar and rudder on vertical stabilizer.



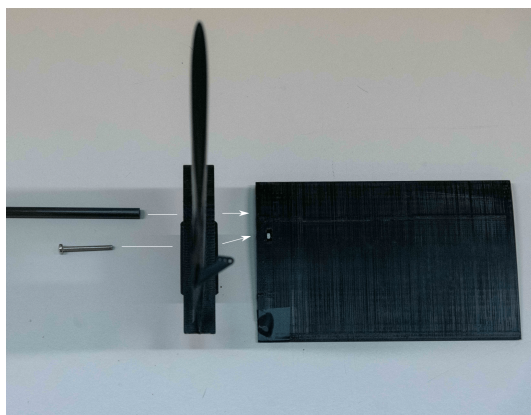
(c) Step one: insert vertical tail spar and rudder on vertical stabilizer (continued).



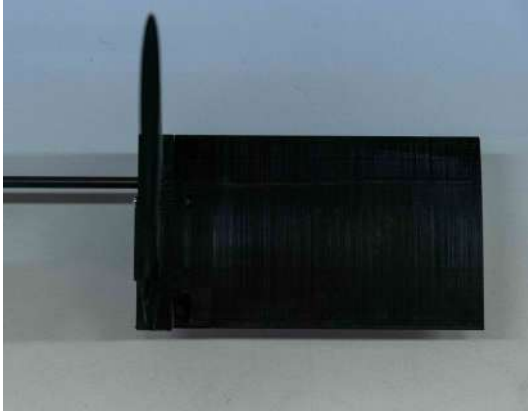
(d) Step two: screw the vertical tail to the tail-boom support piece.



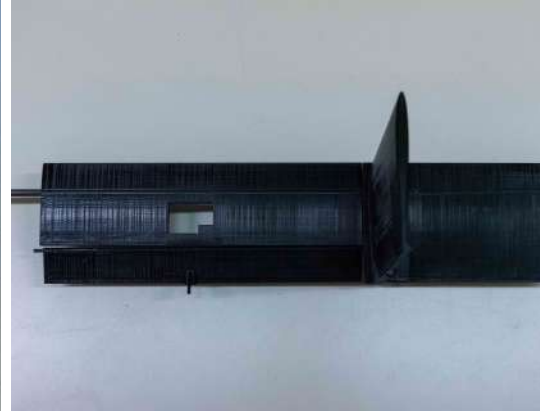
(e) Step two: screw the vertical tail to the tail-boom support piece (continued).



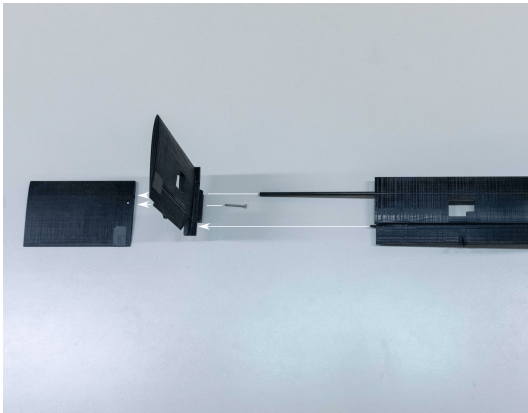
(f) Step three: insert and screw horizontal stabilizer spar in the vertical tail assembly and the external horizontal stabilizer.



(g) Step three: insert and screw horizontal stabilizer spar in the vertical tail assembly and the external horizontal stabilizer (continued).



(h) Step four: insert middle horizontal stabilizer and elevator on horizontal stabilizer spar.



(i) Step five: insert and screw vertical tail assembly and external horizontal stabilizer in the already assembled tail.



(j) Step five: insert and screw vertical tail assembly and external horizontal stabilizer in the already assembled tail (continued).

Figure 5.14: Tail mounting procedure.

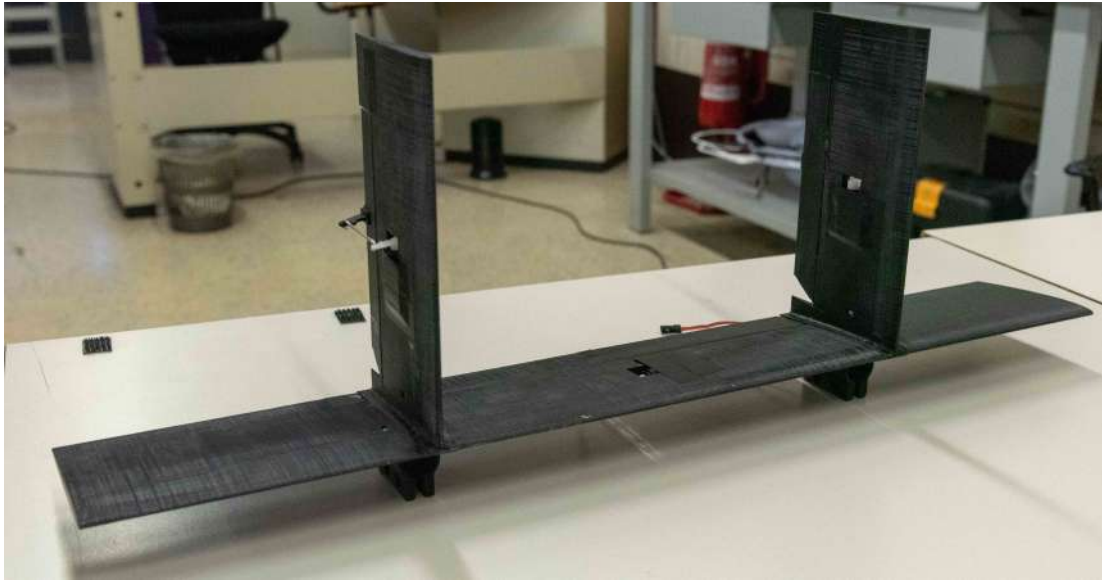
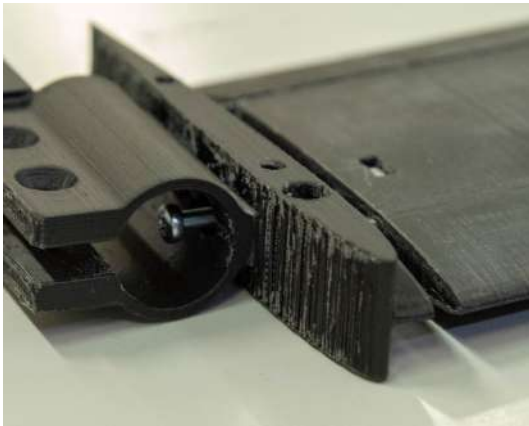
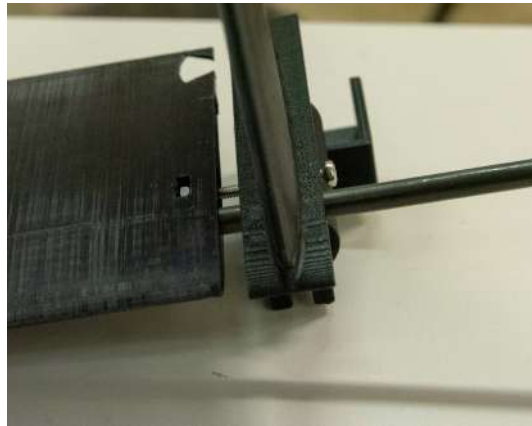


Figure 5.15: Complete tail, ready to be assembled on the rear booms.



(a) Vertical stabilizer.



(b) Horizontal stabilizer.

Figure 5.16: Details of the mounting screws and nuts for the (a) vertical and (b) horizontal stabilizer.

5.3 Fuselage

The original production method for the fuselage was the Selective Laser Sintering (SLS) 3D printing technology (see Appendix D for further information), which allowed to build the lower part of the fuselage as a single piece, without requiring extra temporary supports for the overhanging parts. However, problems arose with the external manufacturer and hence a different production method and supplier had to be found. The final choice came down to a new 3D printer present in the University's aerospace laboratory, which makes use of Fused Deposition Modeling (FDM) technology (see Appendix D for further information). With this printer, however, the fuselage lower part had to be divided in three sub-pieces, due to the limited printing volume available.

5.3.1 Mockup

Before starting the final print of the fuselage, a mockup was build with the 3D printer available in the *ASCL* laboratory, to verify the design and to check if all the components fit in place. Due to the limited size of the parts that could be printed with the Fused Deposition Modeling printer, the mockup fuselage has been printed in smaller pieces, which have then been glued together. Figure 5.17 shows the assembled mockup.

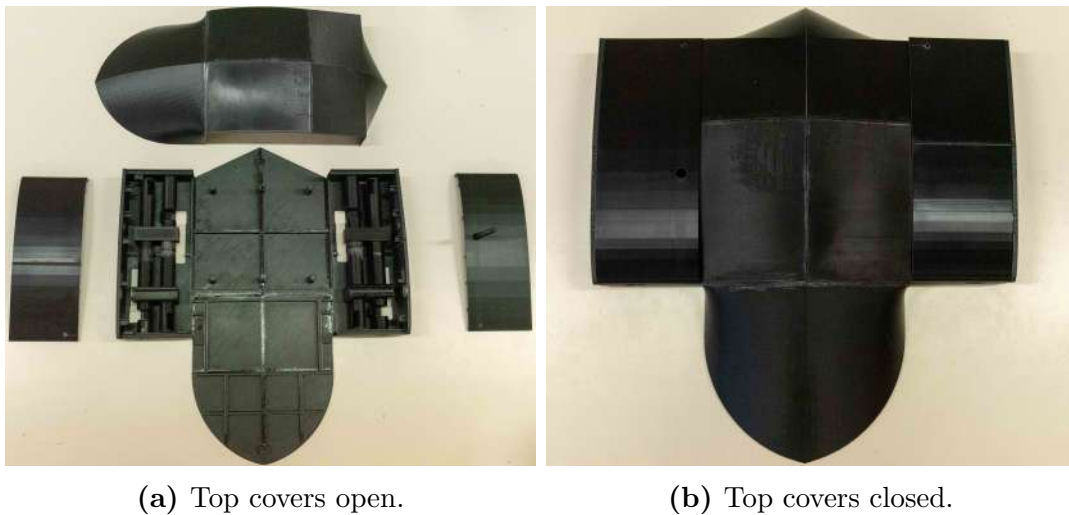


Figure 5.17: Fuselage mockup, with top covers (a) open and (b) closed.

Only small changes were needed, namely some thinning of the top cover, small changes in dimensions for better fit and some small adjustments to the positions of the various avionics components.

5.3.2 Avionics & Electronics

Numerous avionics and electronics components are present inside the fuselage, which serve a number of different purposes. Figure 5.18 shows all the avionics and electronics component present in the aircraft, with reference to the following descriptive list.

- (a) **Flight Control Unit (FCU).** The Flight Control Unit is responsible for controlling the aircraft and acquiring its states with a vast sensor suite. In the present design the *Pixhawk PX4*, by *Holybro* ([34]) has been used, which runs the open source autopilot *PX4*. It features its own processor and memory, two 3-axis accelerometers and gyroscopes, a magnetometer, a barometer and interfaces for a GPS module, Pitot tube and other external devices using different communications protocols, such as I²C, CAN, UART, SBUS, USB.
- (b) **Flight Companion Computer (FCC).** The Flight Companion Computer allows the aircraft to communicate with the ground control station when it is flown inside the *Fly-ART* drone cage facility. It also alleviates the computational load from the FCU, running heavy CPU control algorithms for the aircraft. A *NanoPi NEO Air-LTS* is used ([35]), which features a quad-core processor and 512 Mb of RAM on an extremely small PCB.
- (c) **Receiver (RX).** The receiver is used to communicate with the aircraft from a remote control station used by the UAV's operator, allowing to send inputs to the aircraft. The model used is the *FrSky X8R* ([36]).
- (d) **Telemetry module & Antenna.** The telemetry module allows to send and receive data to the aircraft's Flight Control Unit, using the *MAVLink* communication protocol, when the aircraft is used outside the *Fly-ART* drone cage facility. The model used is the 433MHz Transceiver Telemetry Radio by *Holybro* ([37]).
- (e) **GPS Module.** The GPS module is used for accurate aircraft location during outdoor flight. The module used in the UAV, made by *Holybro* ([38]), also features a compass and a safety switch, used to arm and disarm the UAV's electric motors, for safe operations during ground handling.
- (f) **Universal Battery Eliminator Circuit (UBEC).** The UBEC allows to power the on-board avionics and servomotors, which require a constant 5 Volts. It is connected between the battery and the Power Distribution Board (PDB). The model used is from *HobbyWing* ([39]) and can output up to 3 Amps.
- (g) **Power Distribution Board (PDB).** The Power Distribution Board is not strictly necessary, but highly recommended (especially in the current design)

in order to keep the wiring clean and organized. It connects to the battery, and outputs the power cables to the 10 motors, 5 servomotors, UBEC and the PWM control signals also to the 10 motors and 5 servomotors. The model used is from *Holybro* ([40]) and features enough outputs for the UAV needs.

- (h) **Battery.** The battery is the only power source for the aircraft. A 8500 mAh, 50C, 4S1P LiPo battery has been chosen ([41]).
- (i) **Airspeed sensor** (Not shown in picture). The airspeed sensor is composed of a Pitot tube and a transducer board, which then connects to the FCU. This is the only avionics component that is not installed inside the fuselage, but on the left wing, as shown by Figure 5.6 and Figure 5.9. The model used is from *Holybro* ([42]).

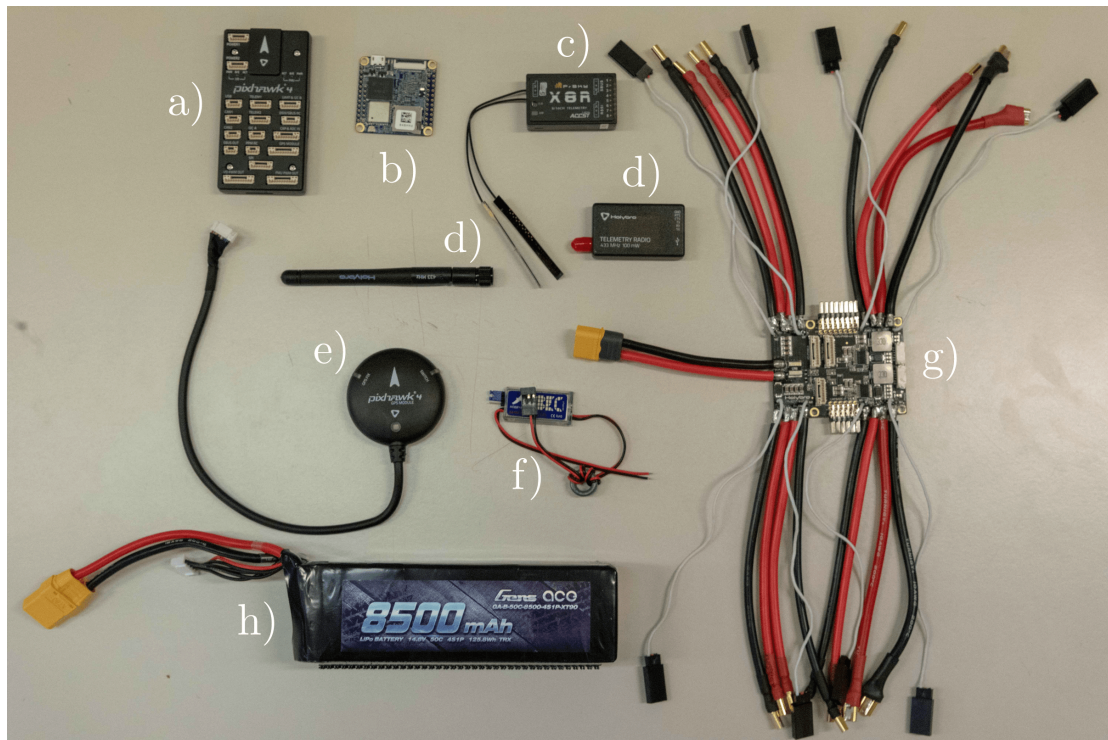


Figure 5.18: Avionics components. a) FCU; b) FCC; c) RX; d) Telemetry & Antenna; e) GPS; f) UBEC; g) PDB; h) Battery.

5.3.3 Integration

In Figure 5.19 the UAV's wiring diagram is shown. A single pair of power cables (ground and positive voltage) run from the Power Distribution Board to each individual motor, apart for VTOL motor 1a and the left forward motor, as well

as VTOL motor 2a and the right forward motor. Those two pairs of motors are each connected to only one power output of the PDB. This is because the board has a total of 8 power outputs, but the total number of motors on the UAV is 10. Similarly, on the PDB there are 8 signal only PWM outputs, next to the power contacts, as well as an additional 8 signal with 5V power (GND and +5V) connectors, located on the left of the board. All those connections are used: 10 PWM for the forward and VTOL motors, 5 for the servomotors and one as a 5V power input to the PDB from the UBEC, which on the other end is connected to the battery. The circuit diagram is completed by connecting the *Pixhawk 4* FCU to the Power Distribution Board and connecting the various avionics components to the *Pixhawk 4*. Note that these avionics connections are shown in green on the diagram, since they actually involve numerous power and signal cables, all grouped together in standardized connectors.

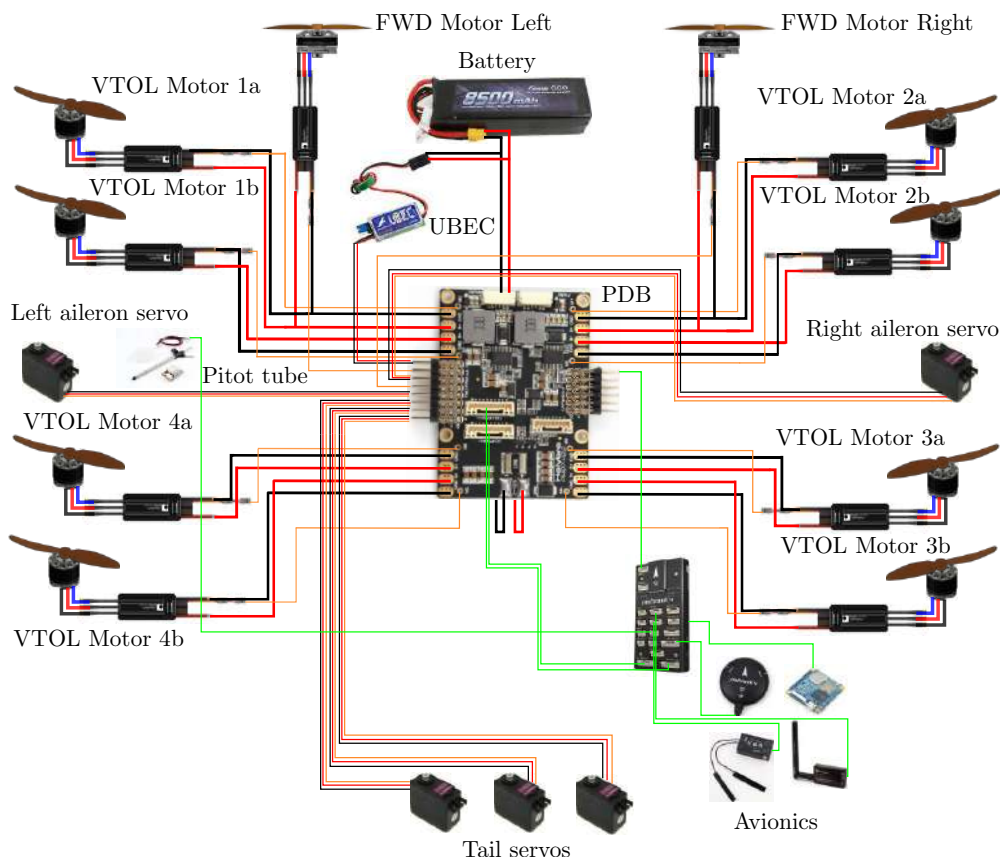


Figure 5.19: UAV's wiring scheme. Black represents ground wires, red represents positive voltage wires (either at battery voltage or 5 Volts), orange represents PWM signal wires and green represent more complex connections.

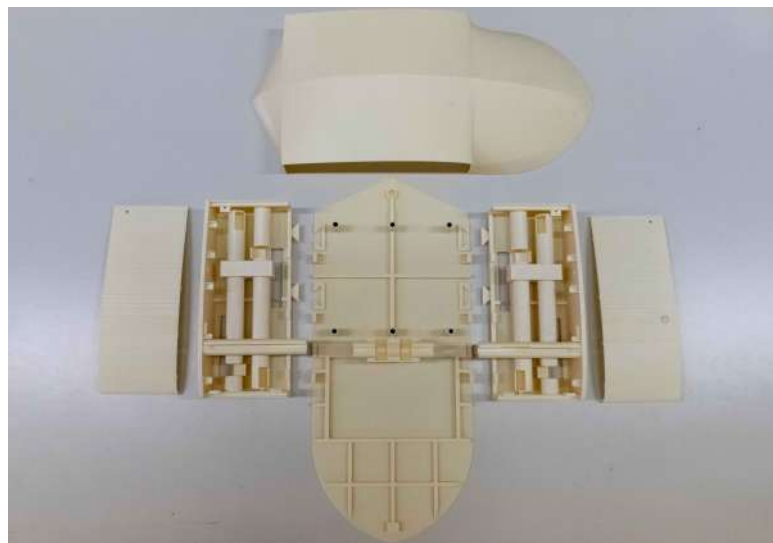
The power and signal cables to the motors and servomotors run inside the forward and rear engine booms, as well as alongside the wing spars. Small holes

are made in the carbon fibre booms to allow the wires to connect to their respective motor.

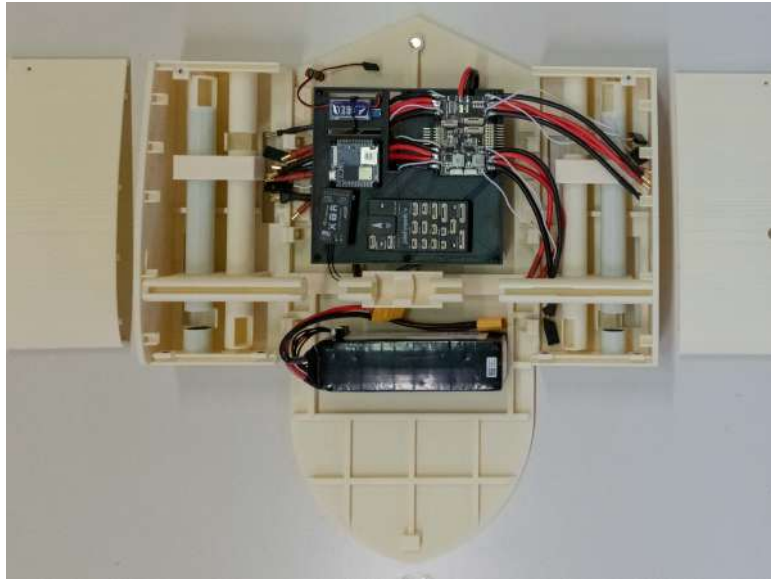
The power cables need to be chosen based on the maximum current which will flow through them. An international standard, called American Wire Gauge (AWG) standard, classifies cables based on their diameter and on the maximum current which can flow before damaging the wire. From the experimental tests on the motor (refer to Chapter 2), the maximum current at full throttle for the VTOL motors is around 28 Amps, whereas for the forward motors is around 20 Amps. AWG12 wires were chosen for all the wires connected to a single motor, whereas AWG10 wires were chosen for the two pairs of wires which each connect two motors (one VTOL and one FWD). AWG 12 wires have a diameter of 2 mm and can withstand up to 30 Amps, whereas AWG 10 wires have a diameter of 2.6 mm and can withstand up to 40 Amps. Both wire types are rated for temperatures up to 200 °C. A potential problem arises for the front wires which connect two motors through only a single wire pair: if both the VTOL and FWD motors are set at full throttle, the current through the wire would be around 48 Amps, exceeding the maximum limit of 40 Amps of the wire. However, this condition is rarely experienced: at the start of the transition as throttle is increased on the forward motor, it is reduced on the VTOL motors.

The servomotors were tested and they drew about 0.25 Amps each. Hence, in total, the current draw from the 5 servomotors is around 1.25 Amps. For the avionics suite, a total of about 0.3 Amps is drawn ([43]), excluding the NanoPi FCC. Hence, the 5 Volts 3 Amps UBEC is more than enough to power the servomotors and the avionics suite.

Figure 5.20 shows the final version of the fuselage, before (Figure 5.20a) and after (Figure 5.20b) joining permanently the three lower parts together, with both screws and glue, and inserting the avionics.



(a) Final fuselage before joining permanently the three lower parts.



(b) Final fuselage after joining permanently the three lower parts and inserting the avionics.

Figure 5.20: Final version of the fuselage, (a) before and (b) after the three lower parts were joined together permanently and avionics were inserted.

5.4 Motor booms

The propeller for forward flight was changed due to incompatibilities with the motor mounts. Instead of a 5x3 inch, two blades propeller, a 5x4.5 inch, 2 blades bullnose propeller is used (the same one of the VTOL motors). This new propeller was tested on the test bench (the results are presented in Chapter 2) and the performance is even slightly better compared to the original propeller.

No changes were made to the original design of the motor booms. A total of four carbon fibre tubes are used. The two front ones support four motors for vertical flight (two on each boom) and the two motors for horizontal flight (one on each boom). The rear tubes support the remaining four vertical flight motors (again two on each boom) and the tail. Figure 5.21 show the final CAD model for the four engine booms, together with the fuselage and the motors.

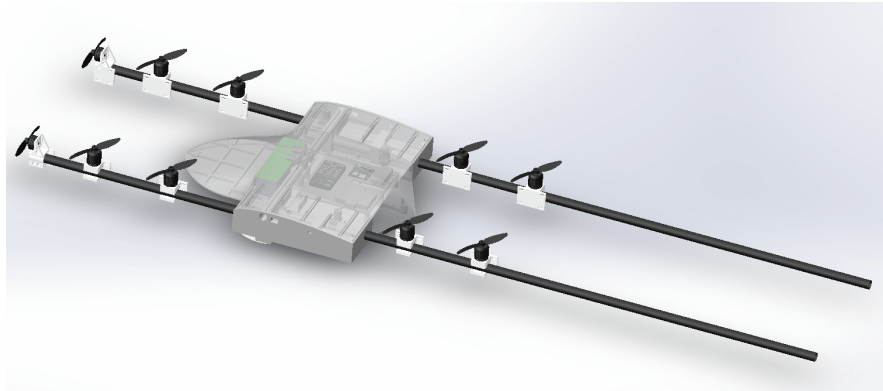
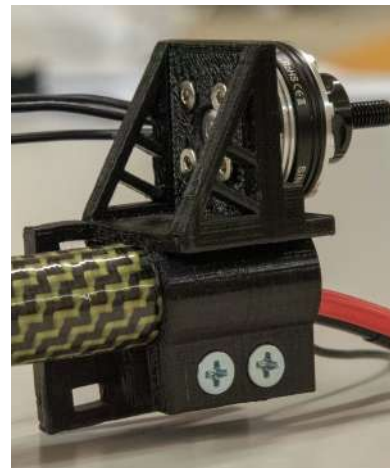


Figure 5.21: CAD model of the motor booms.

The supports which connect the motors to the booms were 3D printed. The forward motor supports work in a straightforward way. A single piece of plastic is printed. The motor is then screwed on one side of the support (Figure 5.22a); the support is then placed around the tube and then secured with two screws and nuts (Figure 5.22b).



(a) Step one: mount motor on support.



(b) Step two: mount support on the boom.

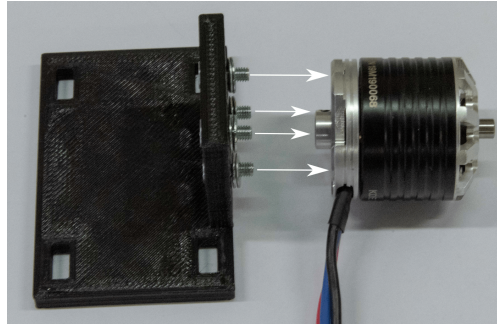
Figure 5.22: Forward motor and support mounting procedure.

The procedure for mounting the VTOL motors is more complicated. The whole support is made up of three different components (Figure 5.23a). First, the motor is mounted on the biggest support, which will also hold the ESC (Figure 5.23b and Figure 5.23c). Note that two washers per screw were inserted between the support and the motor because otherwise the screws were too long and touched the motor inner windings, causing a short circuit inside the motor and ultimately a motor total failure. Then, a second plate is attached, which will make a direct contact with the tube (Figure 5.23d and Figure 5.23e). This double plate mechanism is

needed because otherwise there would not be the possibility to mount the motor on the plate directly attached to the tube, without using excessively long and thin screws to mount the motor. Finally, the third plate is used to secure the whole motor and support assembly to the boom (Figure 5.23f). Figure 5.24 shows the mounted motor and support assembly on the boom.



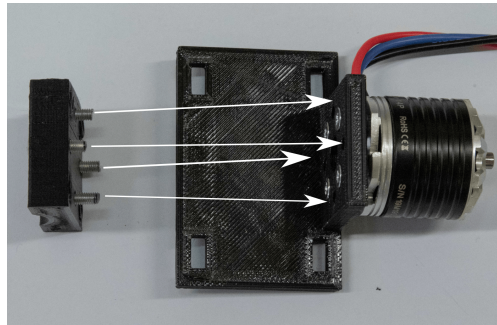
(a) Material needed.



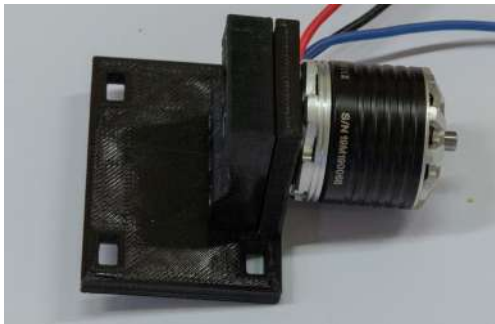
(b) Step one: mount motor on ESC plate.



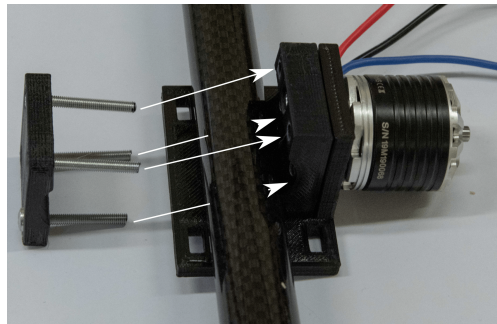
(c) Step one: mount motor on ESC plate (continued).



(d) Step two: mount boom plate.



(e) Step two: mount boom plate (continued).



(f) Step three: position support on boom and mount the second boom plate.

Figure 5.23: Vertical motor and support mounting procedure.

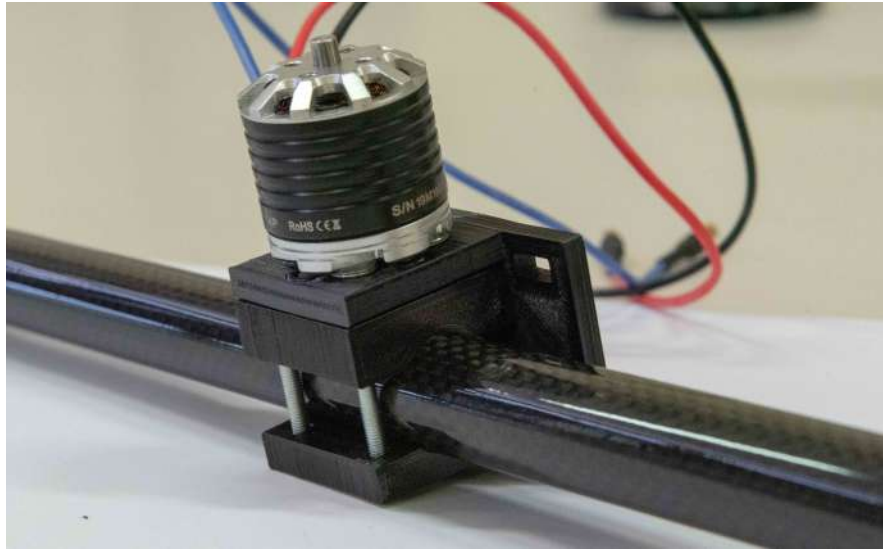


Figure 5.24: VTOL motor and support mounted on boom.

The ESCs are then secured on their support plate with zip ties and connected to the motor and to the power and signal cables running from the fuselage inside the booms. Figure 5.25 shows the complete assembly of the four motor booms, together with all the motors, ESCs and cables.



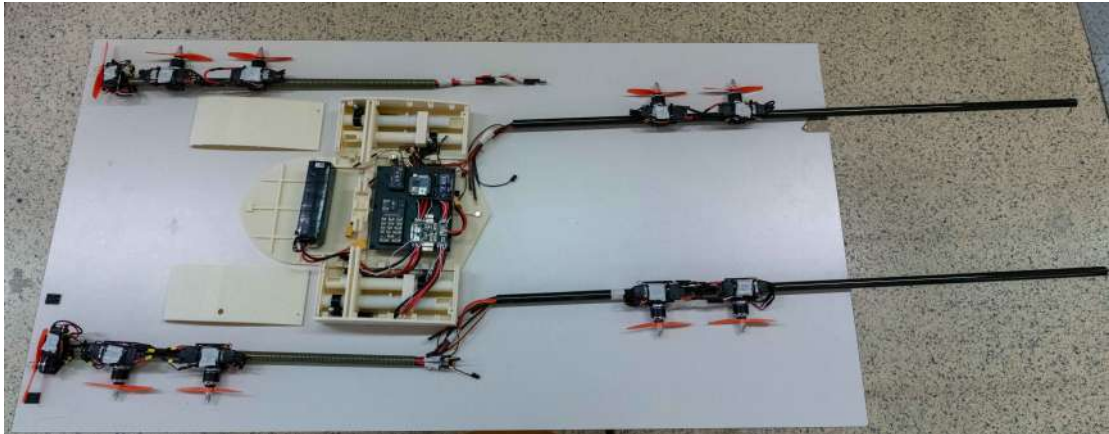
Figure 5.25: Completed motor booms, ready to be installed.

5.5 Final assembly of the UAV

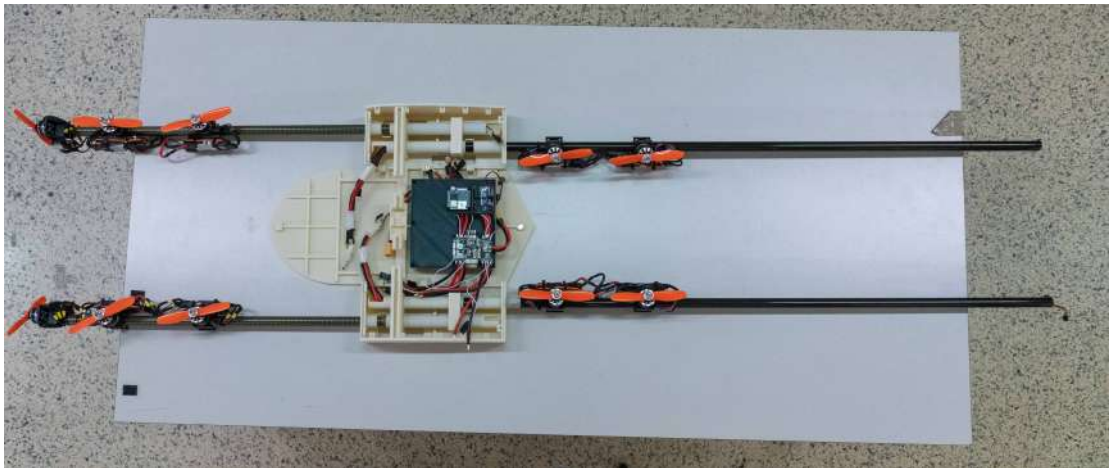
Once all the main components are built, the final assembly of the UAV is rather simple. First, the pieces which hold the forward and rear booms in place are positioned in their location inside the fuselage; one such component is shown in Figure 5.26. Those are annular pieces which tighten around the carbon fibre tubes using a screw; when they are tightened, the booms are fixed in place. Then, the booms are positioned (Figure 5.27a), inserted and secured (Figure 5.27b) inside their dedicated housings in the fuselage. The power and signal cables of the motors and servomotors coming from inside the forward and rear booms are connected to the Power Distribution Board. Subsequently the tail is screwed in place at the end of the rear booms and the servomotors connected (Figure 5.27c). Afterwards, the wings are slid in position (Figure 5.27d). The wing spar goes inside a dedicated housing in the fuselage and is kept in place by the same mechanism used for the motor booms. The wings' rotation about the pitch axis is blocked by a pin which is inserted in a corresponding hole on the side of the fuselage, one on each side for each wing. The ailerons servomotors and the Pitot tube are connected to the PDB and to Pixhawk FCU, respectively. Finally, the three fuselage covers are positioned and screwed in place. Figure 5.28 shows the final assembled eVTOL UAV.



Figure 5.26: Locking mechanism for the motor booms and wings. The carbon fibre tubes are placed inside this annular component and then are secured by tightening the screw.



(a) Step one: insert forward and rear motor booms inside the fuselage and secure them.



(b) Step one: insert forward and rear motor booms inside the fuselage and secure them (continued).



(c) Step two: insert and secure tail on rear booms.



(d) Step three: insert wings in fuselage.

Figure 5.27: Final integration of the UAV.



Figure 5.28: Complete assembled eVTOL UAV.

Conclusions

In this thesis, an eVTOL UAV, completely designed within the *ASCL* Laboratory of *Politecnico di Milano* ([1]), has been successfully built. This was the main objective of this work and, as easy as it sounds, it actually proved to be challenging and time consuming. This practical activity, with the many problems that arose and were solved along the way, helped gain extremely useful insights into the applied activity of integrating and assembling a fixed wing eVTOL UAV: something that cannot just be learned on paper-only projects. If the final aim is the actual construction of the vehicle, the aircraft design discipline has to take into account practical productions techniques, problems and limitation and eventually make compromises accordingly.

Hence it is seen as valuable, as reference for future works, to summarize the main lessons learned during this practical experience.

- For the wing production, the original Styrofoam filling material could not be used, since the wing was too thin. Hence, this is an important aspect to keep in mind for future works: either increase the thickness of the wing, or plan on other production techniques.
- The covering procedure of the wings with the Mylar adhesive cladding seems to be of satisfactory quality (although this has to be confirmed by flight tests), however wrinkles were produced as the cover was applied. A suggestion is to stretch the Mylar as hard as possible along all the directions and then apply it to the wing. Furthermore, due to the limited size of the ailerons, they can be easily produced with 3D printing for future works, avoiding the covering procedure of such small components.
- The original servomotors' positions in the tail was not effective in moving the control surfaces. Hence, it is strongly suggested for future works to always plan on a direct push/pull on the control rod configuration of the control surfaces, even at the cost of worsening the aerodynamic profile of the surfaces where the servomotors are mounted on.
- The particular ESCs used in this project are heavy and big. It is suggested to find some other products on the market that are more compact and light.

- By testing the motors on the motor test bench, performance proved to be quite different from the declared data by the manufacturer. This is most probably caused by small differences between the propellers that are used by the manufacturer and the ones that have been used in the project. Hence it is recommended, if possible, to test the combination of motor, ESC and propeller that will be used on the aircraft in order to evaluate its real performance, before actually flying the vehicle.
- When mounting the motors, extreme care must be taken with respect to the length of the screws that are used. Too long screws (even a couple of millimeters of extra length are enough) touch the inner motor windings, shorting the motor internally and causing it to overheat, smoke and ultimately fail (this has actually happened during the first tests on the motor test bench), without any external way of noticing the problem before it is too late. Hence, during the integration phase, after each motor was mounted it was tested with a multimeter for short circuits, between each power cable and each mounting screw. Washers were placed when the screws were too long.

Regarding the other objectives of the thesis, the UAV's aerodynamic model has been improved, by adding the fuselage, motor booms and motors, although for this work only a Vortex Lattice solver has been used. This has limitations in both the accuracy that can be reached and in the low angle of attacks that can be analyzed, which limits the simulation of complex transition maneuvers that involve the aircraft operating temporarily at high angles of attack. The motor test bench has been fully validated and is easy to use, providing all the necessary parameters to estimate the motor's both static and dynamic performance. A *Simulink* flight simulator has been developed and validated, using the aforementioned aerodynamic data and experimentally identified motors' parameters, together with mass properties coming from a precise *SolidWorks* CAD model. The simulator is able to simulate the three aircraft flight modes (VTOL, transition and hover) and accurately estimate the vehicle's endurance and range, since a precise LiPo battery model has been implemented. The simulator has been effectively validated with linearized models and tested only for the forward flight mode. This is because the aircraft in VTOL mode is extremely unstable, and there is no way to validate the simulator without applying any sort of control law.

From the aforementioned considerations, numerous future activities can be carried out on the project.

- Now that the aircraft has finally been built, an extensive flight test campaign can be extremely useful. By flying indoor as a VTOL and outside as a normal fixed wing aircraft, the UAV's stability and control derivatives can

be estimated in flight, not only improving the simulator's fidelity, but also providing an evaluation of the accuracy of the VLM software used in the preliminary design phase (*XFLR5*) and in the present work (*OpenVSP*). Furthermore, the real aircraft's performance and flying qualities can be obtained, which again can be compared to the results from the preliminary design and highlight the strengths and weaknesses of such design methodology.

- An improved aerodynamic model can be produced. This time it is strongly suggested to either use Computational Fluid Dynamics (CFD) software or to actually test the aircraft in wind tunnel. It would be extremely useful to model and/or test the UAV also for extremely high angles of attack (roughly from -90 degrees to +90 degrees), so that accurate simulator models for complex transition maneuvers can be produced.
- The transition phase can be deeply studied and analyzed. In this thesis considerations on some possible types of maneuvers have been made, but this just opens up the topic. Thanks to the redundant control inputs available during this phase (all the 10 motors and the 3 control surfaces) a lot of possibilities arise. It could be interesting to model such phase and look for transitions that minimize a certain objective function, depending on the mission: time for time-critical mission, energy used for long endurance missions and so on.
- The control laws for the aircraft have to be written and validated. Effectively three control laws are needed: for the VTOL, transition and forward flight phase.
- The modularity of the aircraft can be exploited to improve the design of single components. Different wings can be tested, a lighter fuselage can be designed and produced, longer or shorter motor booms can be evaluated, other tail designs can be analysed and different propulsive units can be installed.

Bibliography

- [1] N. Battaini. Design and dynamic modeling of a VTOL UAV. Master's thesis, Politecnico di Milano, 2020.
- [2] Overlander batteries. Supersport XL Series 8500 mAh, 14.8V, 20C battery datasheet (product code: 6265188). https://www.overlander.co.uk/pub/media/datasheets/LiPo/8500_20C_1.pdf.
- [3] R. Fail, J. A. Lawford, and R. C. W. Eyre. Low-speed experiments on the wake characteristics of flat plates normal to an air stream. *Aeronautical research council reports and memoranda*, 3120, June 1957.
- [4] DJI Mavic Air 2 picture Website. <https://www.mondoaeroporto.it/dji-mavic-air-2-prezzo-foto-e-data-di-uscita-del-nuovo-drone/>.
- [5] Trimble fixed wing drone picture Website. <https://www.payszpz.gq/products.aspx?cname=trimble+fixed+wing+drone&cid=7&xi=2&xc=25&pr=77.99>.
- [6] Baby Shark VTOL 250 Fixed wing picture Website. <https://www.droneassemble.com/product/baby-shark-vtol-250-fixed-wing-kit-air-frame-for-uav-building/>.
- [7] DHL. Unmanned aerial vehicle in logistics. https://www.dhl.com/content/dam/downloads/g0/about_us/logistics_insights/DHL_TrendReport_UAV.pdf.
- [8] C. Gómez and D. R. Green. Small unmanned airborne systems to support oil and gas pipeline monitoring and mapping. *Arabian Journal of Geosciences*, May 2017.
- [9] Hybrid UAV Enhances Drone Surveying Capabilities. <https://www.unmannedsystemstechnology.com/2019/09/hybrid-uav-enhances-drone-surveying-capabilities/>.
- [10] Helicopters versus drones: The cost of the war on rhinos. <http://edition.cnn.com/2013/10/16/world/africa/helicopters-versus-drones-rhino/index.html?iref=allsearch>.

- [11] UAVs perform autonomous search-and-rescue operations. <http://www.homelandsecuritynewswire.com/uavs-perform-autonomous-search-and-rescue-operations>.
- [12] Google spreads its wings, moving into drone deliveries. <https://www.cnet.com/news/google-announces-project-wing-for-drone-deliveries/>.
- [13] Soton UAV Website. <https://www.sotonuav.uk/>.
- [14] XFLR5 Website. <http://www.xflr5.tech/xflr5.htm>.
- [15] OpenVSP Website. <http://openvsp.org>.
- [16] OpenVSP Parasite Drag Tool Documentation. <http://openvsp.org/wiki/doku.php?id=parasitedrag>.
- [17] OpenVSP Forum, VLM vs panel methods discussion. <https://groups.google.com/g/openvsp/c/Mt22QS05udo>.
- [18] A. Deperrois. Analysis of foils and wings operating at low Reynolds numbers, 2013.
- [19] J. Gundlach. *Designing unmanned aircraft systems: a comprehensive approach*. Ed. by J. A. Schetz. AIAA education series. American Institute of Aeronautics and Astronautics, Inc., 2012.
- [20] J. Leishman. *Principles of Helicopter Aerodynamics*. Cambridge University Press, 2016.
- [21] M. Giurato. Design, integration and control of a multirotor UAV platform. Master's thesis, Politecnico di Milano, 2015.
- [22] D. Chevallard. Design, identification and control of a micro aerial vehicle. Master's thesis, Politecnico di Milano, 2017.
- [23] AVIA Semiconductor. HX711 chip datasheet. https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf.
- [24] RCbenchmark Series 1580/1585 Thrust Stand. <https://www.rcbenchmark.com/pages/series-1580-thrust-stand-dynamometer>.
- [25] S. O. H. Madgwick. An efficient orientation filter for inertial and inertial/magnetic sensor arrays. Technical report, University of Bristol, April 2010.
- [26] J. J. Craig. *Introduction to Robotics Mechanics and Control*. Pearson Education International, 2005.

- [27] G. Gozzini. UAV autonomous landing on moving aerial vehicle. Master's thesis, Politecnico di Milano, 2019.
- [28] R. W. Prouty. *Helicopter performance, stability and control*. John Wiley & Sons Ltd, 1986.
- [29] M. Lovera, F. Riccardi, and P. Panizza. Identification of the attitude dynamics for a variable-pitch quadrotor UAV. *European Rotorcraft Forum (ERF) 2014 proceedings*, September 2014.
- [30] M. Lovera, S. Panza, D. Del Gobbo, and L. Viganò. qLPV modelling of helicopter dynamics. *3rd IFAC Workshop on Linear Parameter Varying Systems LPVS 2019*, November 2019.
- [31] E. L. Tobias and M. B. Tischler. A model stitching architecture for continuous full flight-envelope simulation of fixed-wing aircraft and rotorcraft from discrete-point linear models. Technical report, U.S. Army Research, Development, and Engineering Command, April 2016.
- [32] United States Department of Defense. MIL F 8785 C. Military Specification: Flying Qualities of Piloted Airplanes. Technical report, U.S. D.O.D., November 1980.
- [33] A. J. Keane, A. Sóbester, and J. P. Scanlan. *Small Unmanned Fixed-wing Aircraft Design. A Practical Approach*. John Wiley & Sons Ltd, 2017.
- [34] Pixhawk Website. https://docs.px4.io/master/en/flight_controller/pixhawk4.html.
- [35] NanoPi NEO Air-LTS Website. https://www.friendlyarm.com/index.php?route=product/product&product_id=151.
- [36] FrSky X8R Website. <https://www.frsky-rc.com/product/x8r/>.
- [37] Transceiver Telemetry Radio V3 433MHz Website. <http://www.holybro.com/product/transceiver-telemetry-radio-v3/>.
- [38] Pixhawk 4 GPS Module Website. <http://www.holybro.com/product/pixhawk-4-gps-module/>.
- [39] HobbyWing 5V 3A Universal Battery Eliminator Circuit Website. <https://www.amazon.it/Switching-Hobbywing-UBEC-BEC-modelli/dp/B01CHW207U>.
- [40] Holybro Power Distribution Board Website. <http://www.holybro.com/product/pixhawk-4-power-module-pm07/>.

-
- [41] Gens Ace 14.8V 50C 4S1P 8500mAh Lipo Battery Pack Website. <https://www.genstattu.com/gens-ace-50c-8500mah-4s1p-xt60t-lipo-battery.html>.
- [42] Digital Air Speed Sensor Website. https://shop.holybro.com/digital-air-speed-sensor_p1029.html.
- [43] Pixhawk and avionics suite current draw. <https://diydrones.com/profiles/blogs/pixhawk-and-apm-power-consumption#:~:text=Using%20a%20standard%203%2Dcell,to%202.5W%20power%20consumption>.
- [44] KDE Motors Website. <https://www.kdedirect.com/pages/motors>.
- [45] MathWorks. Generate parameter data for datasheet battery block. <https://it.mathworks.com/help/autoblks/ug/generate-parameter-data-for-datasheet-battery-block.html>.
- [46] jiro. GRABIT, on MATLAB Central File Exchange. <https://www.mathworks.com/matlabcentral/fileexchange/7173-grabit>.
- [47] Ultimaker 3D printer Website. <https://ultimaker.com/3d-printers/ultimaker-s3>.
- [48] Stratasys Fortus 450 MC 3D printer Website. <https://www.stratasys.com/3d-printers/fortus-380mc-450mc>.
- [49] FDM schematics picture Website. <https://www.3dparts.co.uk/how-it-works/>.
- [50] SLS schematics picture Website. <https://www.spilasers.com/application-additive-manufacturing/selective-laser-sintering-and-melting/>.

Appendix A

Additional test cases results from Chapter 1

In this Appendix the results for cases 1 (airspeed of 11 m/s) and 3 (airspeed of 22 m/s) (see Table 1.1 of Chapter 1) of the aerodynamic comparison between the model developed in *XFLR5* in [1] and the improved model developed in *OpenVSP* during the present work are presented, as further material for Chapter 1.

Figure A.1 to Figure A.6 show the lift coefficient against angle of attack curve, the drag polar of the aircraft and the single contributions to the drag coefficient (viscous, induced, and total value) against angle of attack, whereas Table A.1 and Table A.2 summarize the differences between the aerodynamic coefficients and their derivatives, for the two *OpenVSP* and *XFLR5* models for test cases 1 and 3.

Similar conclusion to the ones presented in Chapter 1 apply also for these two test cases.

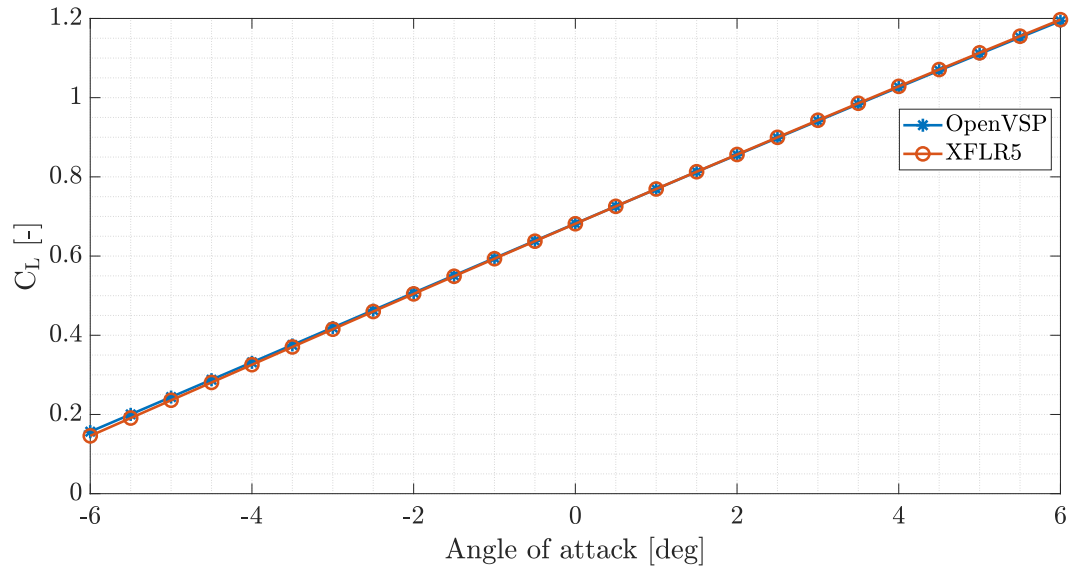


Figure A.1: *OpenVSP* and *XFLR5* lift coefficient against angle of attack curve comparison, case 1 (airspeed 11 m/s).

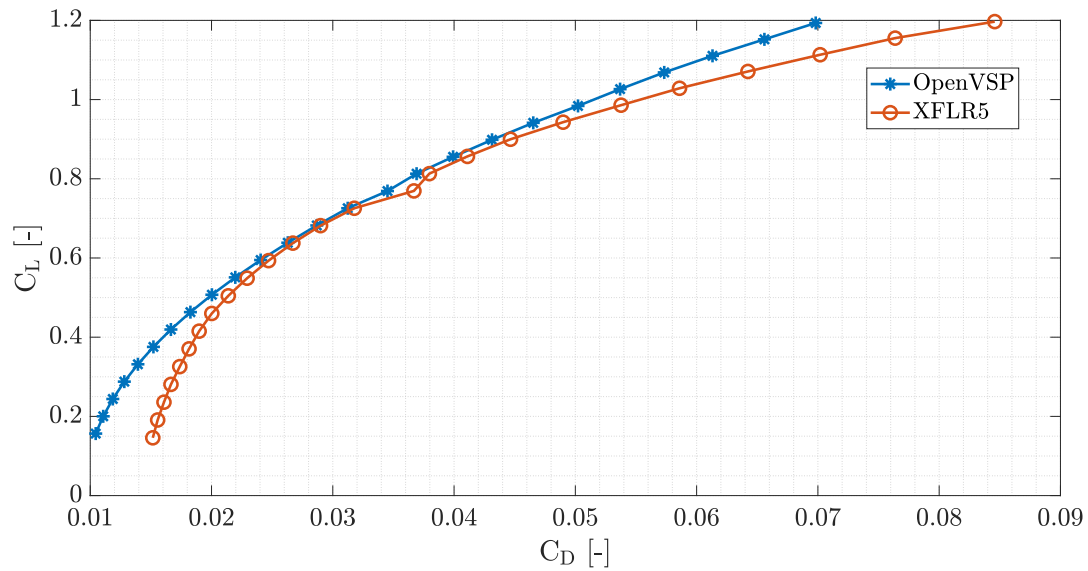


Figure A.2: *OpenVSP* and *XFLR5* drag polar curve comparison, case 1 (airspeed 11 m/s).

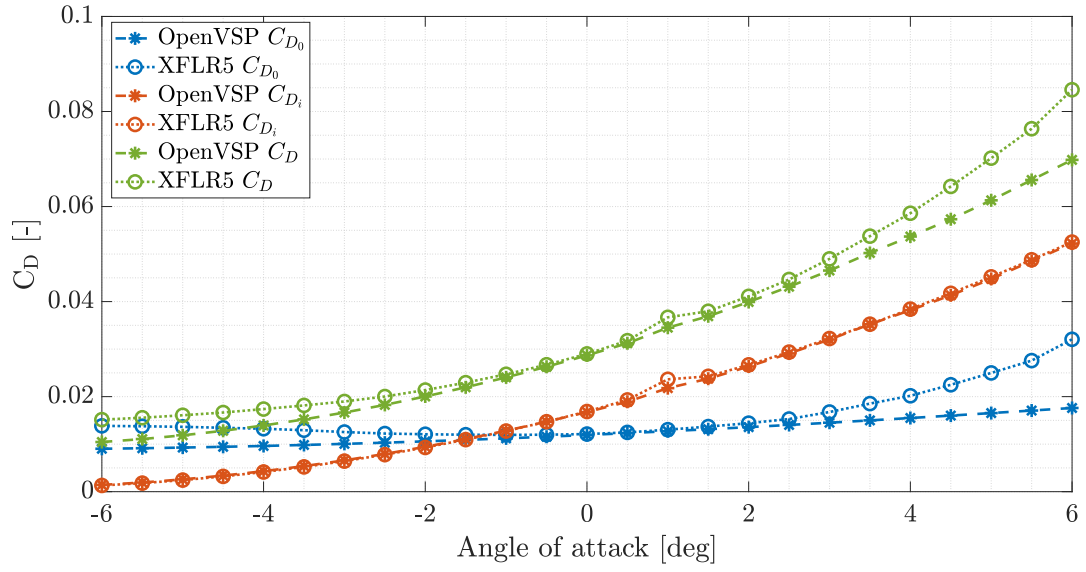


Figure A.3: *OpenVSP* and *XFLR5* single contributions to drag coefficient against angle of attack curve comparison, case 1 (airspeed 11 m/s).

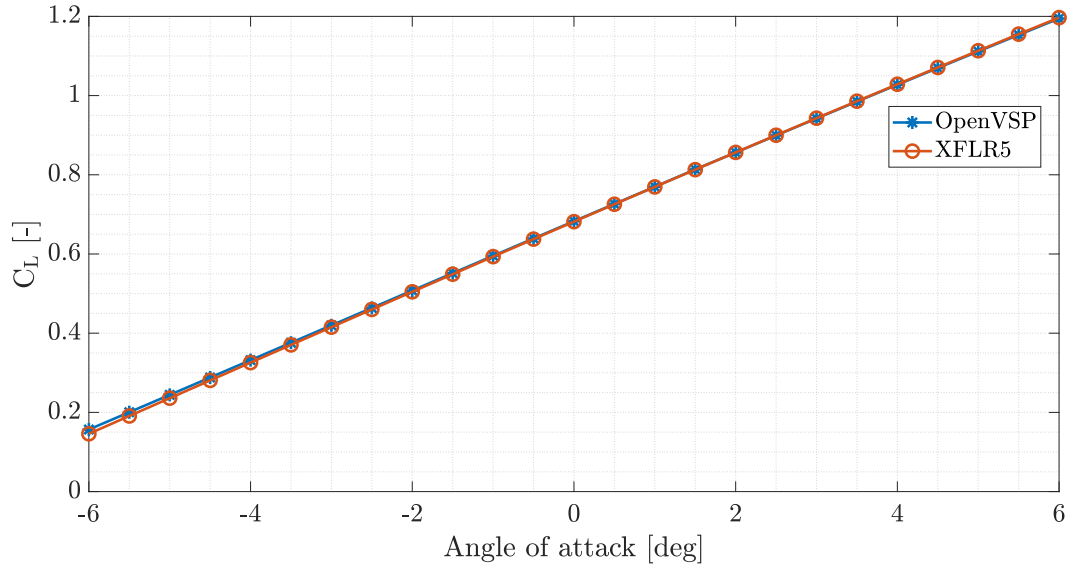


Figure A.4: *OpenVSP* and *XFLR5* lift coefficient against angle of attack curve comparison, case 3 (airspeed 22 m/s).

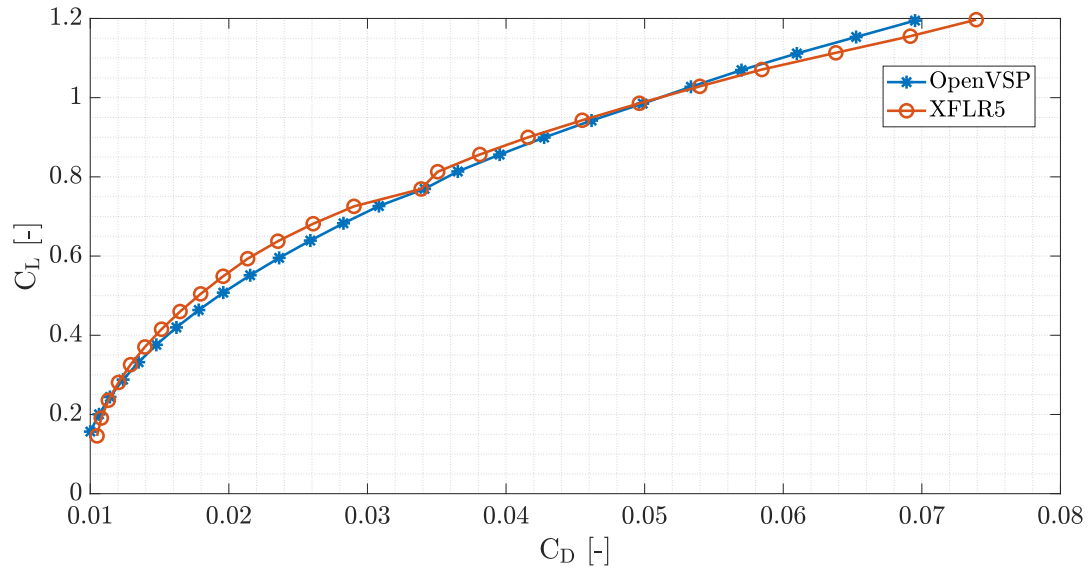


Figure A.5: *OpenVSP* and *XFLR5* drag polar curve comparison, case 3 (airspeed 22 m/s).

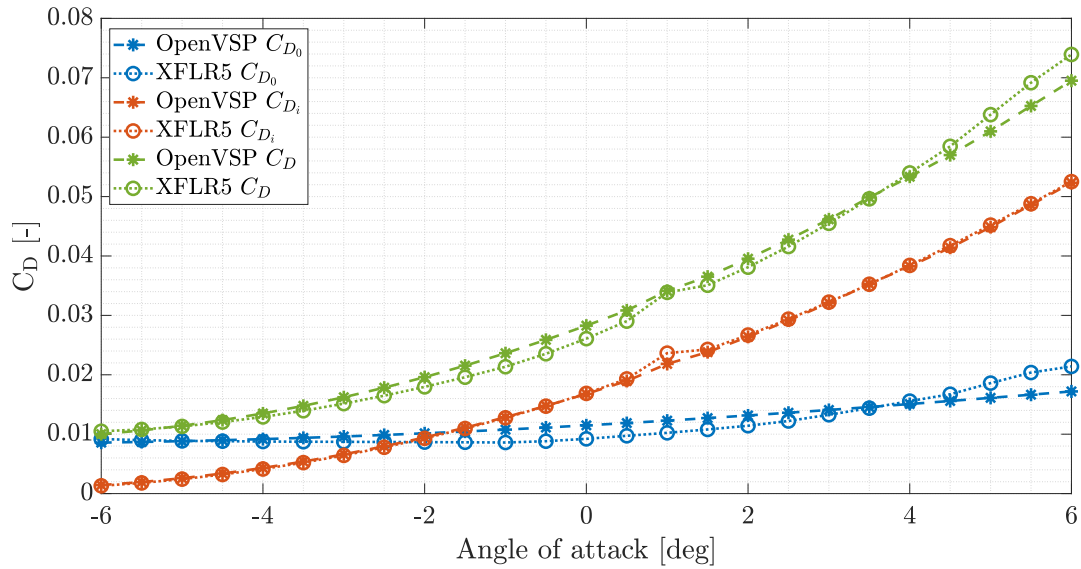


Figure A.6: *OpenVSP* and *XFLR5* single contributions to drag coefficient against angle of attack curve comparison, case 3 (airspeed 22 m/s).

Table A.1: *OpenVSP* and *XFLR5* aerodynamic coefficients and derivatives comparison, case 1 (airspeed 11 m/s).

	<i>Tot</i>	α	β	p	q	r	δ_a	δ_e	δ_r
C_X	<i>XFLR5</i>	0.0406	1.25	0 ¹	0.228	0 ¹	$9.69 \cdot 10^{-3}$	-0.0357	$-2.16 \cdot 10^{-5}$
	<i>OpenVSP</i>	0.0897	1.47	$-1.25 \cdot 10^{-3}$	0.0874	0.0159	$-4.05 \cdot 10^{-4}$	$-5.28 \cdot 10^{-3}$	$-1.17 \cdot 10^{-4}$
	% Diff.	-121%	-17.4%	Na ²	Na ³	Na ²	Na ³	114.8%	Na ³
C_Y	<i>XFLR5</i>	0.00	0 ¹	-0.176	0.0314	0 ¹	0.158	$4.47 \cdot 10^{-16}$	0.0522
	<i>OpenVSP</i>	$1.30 \cdot 10^{-6}$	$1.41 \cdot 10^{-4}$	-0.0385	0.0102	$-2.45 \cdot 10^{-4}$	0.167	$-2.20 \cdot 10^{-6}$	0.0781
	% Diff.	Na ³	Na ²	78.1%	67.5%	Na ²	-5.7%	Na ³	-49.8%
C_Z	<i>XFLR5</i>	-1.18	-4.95	0 ¹	-7.65	0 ¹	$1.49 \cdot 10^{-3}$	-0.315	$-5.61 \cdot 10^{-5}$
	<i>OpenVSP</i>	-1.18	-4.53	0.0191	-7.48	-0.0311	$1.02 \cdot 10^{-3}$	-0.291	$-4.60 \cdot 10^{-6}$
	% Diff.	0.5%	8.5%	Na ²	2.1%	Na ²	Na ³	7.5%	Na ³
C_L	<i>XFLR5</i>	0.00	0 ¹	$-1.20 \cdot 10^{-3}$	-0.560	0 ¹	0.248	$-1.97 \cdot 10^{-13}$	$-1.72 \cdot 10^{-3}$
	<i>OpenVSP</i>	$6.00 \cdot 10^{-7}$	$4.16 \cdot 10^{-5}$	-0.0121	-0.518	$-2.83 \cdot 10^{-4}$	0.240	$1.40 \cdot 10^{-6}$	$2.45 \cdot 10^{-3}$
	% Diff.	Na ³	Na ²	Na ³	7.5%	Na ²	3.2%	Na ³	Na ³
C_M	<i>XFLR5</i>	$-4.24 \cdot 10^{-3}$	-0.271	0 ¹	-15.1	0 ¹	$6.00 \cdot 10^{-4}$	-1.22	$-1.88 \cdot 10^{-4}$
	<i>OpenVSP</i>	-0.0871	-0.539	$-9.29 \cdot 10^{-3}$	0.0226	-0.0446	$2.22 \cdot 10^{-4}$	-1.19	$2.06 \cdot 10^{-5}$
	% Diff.	Na ³	-99.2%	Na ²	Na ²	Na ²	Na ³	2.7%	Na ³
C_N	<i>XFLR5</i>	0.00	0 ¹	0.0775	-0.267	0 ¹	-0.0610	$2.40 \cdot 10^{-14}$	-0.0239
	<i>OpenVSP</i>	$-6.00 \cdot 10^{-7}$	$-6.88 \cdot 10^{-5}$	0.0818	-0.184	-0.0757	$4.20 \cdot 10^{-3}$	$1.20 \cdot 10^{-6}$	-0.0371
	% Diff.	Na ³	Na ²	-5.6%	30.9%	24.0%	Na ³	Na ³	-55.4%

¹ Not calculated explicitly by *XFLR5*.² Not available because coefficient by *XFLR5* is not calculated explicitly.³ Not calculated because one of the values is 0 or really close to 0 (smaller than $1 \cdot 10^{-2}$).

Table A.2: *OpenVSP* and *XFLR5* aerodynamic coefficients and derivatives comparison, case 3 (airspeed 22 m/s).

	<i>Tot</i>	α	β	p	q	r	δ_a	δ_e	δ_r
C_X	<i>XFLR5</i>	-0.0349	0 ¹	0 ¹	-0.720	0 ¹	$2.30 \cdot 10^{-3}$	$-4.09 \cdot 10^{-3}$	$-8.07 \cdot 10^{-6}$
	<i>OpenVSP</i>	-0.0242	$2.84 \cdot 10^{-3}$	0.0487	-0.611	$5.60 \cdot 10^{-3}$	$-3.11 \cdot 10^{-4}$	-0.0241	$-1.22 \cdot 10^{-4}$
	% Diff.	30.6%	Na ²	Na ²	15.2%	Na ²	Na ³	Na ³	Na ³
C_Y	<i>XFLR5</i>	0.00	0 ¹	0.0309	0 ¹	0.155	$-2.13 \cdot 10^{-3}$	$5.43 \cdot 10^{-16}$	0.052
	<i>OpenVSP</i>	0.00	$-5.0 \cdot 10^{-7}$	0.0245	$-7.90 \cdot 10^{-6}$	0.153	$2.61 \cdot 10^{-3}$	$-3.00 \cdot 10^{-7}$	0.0749
	% Diff.	Na ³	Na ²	20.8%	Na ²	1.7%	Na ³	Na ³	-44.0%
C_Z	<i>XFLR5</i>	-0.292	-5.16	0 ¹	-7.14	0 ¹	$2.74 \cdot 10^{-3}$	-0.322	$-4.34 \cdot 10^{-5}$
	<i>OpenVSP</i>	-0.284	-5.00	$3.35 \cdot 10^{-3}$	$-9.60 \cdot 10^{-3}$	-0.0152	$4.40 \cdot 10^{-4}$	-0.300	$-4.36 \cdot 10^{-5}$
	% Diff.	2.6%	3.1%	Na ²	Na ²	Na ²	Na ³	7.0%	Na ³
C_L	<i>XFLR5</i>	0.00	0 ¹	$-1.36 \cdot 10^{-3}$	0 ¹	0.128	0.148	$-3.51 \cdot 10^{-13}$	$2.57 \cdot 10^{-3}$
	<i>OpenVSP</i>	0.00	$-1.0 \cdot 10^{-6}$	$-4.04 \cdot 10^{-3}$	$2.00 \cdot 10^{-7}$	0.129	0.128	0.00	$2.57 \cdot 10^{-3}$
	% Diff.	0%	Na ²	Na ³	Na ²	-0.7%	13.9%	Na ³	Na ³
C_M	<i>XFLR5</i>	$-5.90 \cdot 10^{-4}$	-0.542	0 ¹	-15.4	0 ¹	$6.90 \cdot 10^{-4}$	-1.24	$-1.47 \cdot 10^{-4}$
	<i>OpenVSP</i>	0.0766	-0.526	-0.0106	-16.4	-0.0257	$8.89 \cdot 10^{-5}$	-1.22	$-1.45 \cdot 10^{-4}$
	% Diff.	Na ³	2.9%	Na ²	-6.5%	Na ²	Na ³	1.3%	Na ³
C_N	<i>XFLR5</i>	0.00	0 ¹	0.0761	0 ¹	-0.0728	$9.31 \cdot 10^{-4}$	$2.52 \cdot 10^{-15}$	-0.0236
	<i>OpenVSP</i>	0.00	$3.00 \cdot 10^{-7}$	0.0755	$3.60 \cdot 10^{-6}$	-0.0786	-0.0118	$1.00 \cdot 10^{-7}$	-0.0356
	% Diff.	0%	Na ²	0.8%	Na ²	-8.0%	Na ³	Na ³	-50.5%

¹ Not calculated explicitly by *XFLR5*.² Not available because coefficient by *XFLR5* is not calculated explicitly.³ Not calculated because one of the values is 0 or really close to 0 (smaller than $1 \cdot 10^{-2}$).

Appendix B

Comparison of motor performance with manufacturer data

In this Appendix the motor test bed, described in Chapter 2 is used to compare the motor performance against the published data from the manufacturer website ([44]). The motor is powered by the laboratory DC power supply, with a voltage equal to the testing voltage used by the manufacturer. For the motor for horizontal flight *KDE2304XF-2350* this was 11.6 V, whereas for the motor for vertical flight *KDE2315XF-2050* this was 15.4 V. For this last motor, however, for throttle values higher than 25%, the power supply reached its maximum voltage, and hence a voltage drop for higher throttle settings is observed, which is shown in Figure B.7. As in Chapter 2, the plots of thrust against angular velocity, angular velocity against throttle percentage set, electrical and mechanical power against angular velocity will be presented in Figure B.1 to Figure B.6, whereas the identified static coefficients and their comparison to the ones obtained from manufacturer data are shown in Table B.1 and Table B.2.

Motor for forward flight

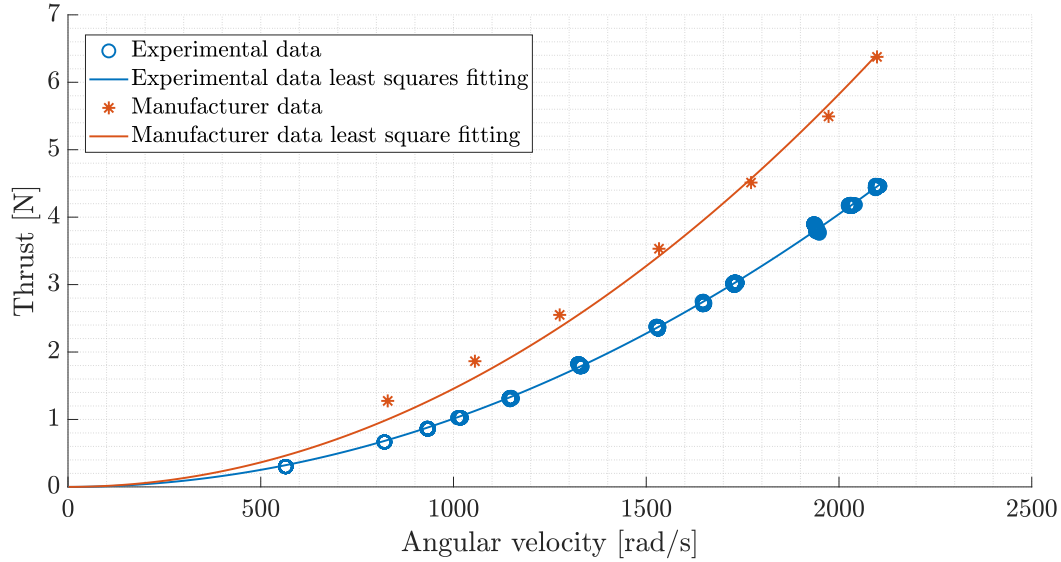


Figure B.1: Thrust against angular velocity, motor powered from DC power supply, motor for forward flight *KDE2304XF-2350*, 5x4.5 inch, 2 blades bullnose propeller.

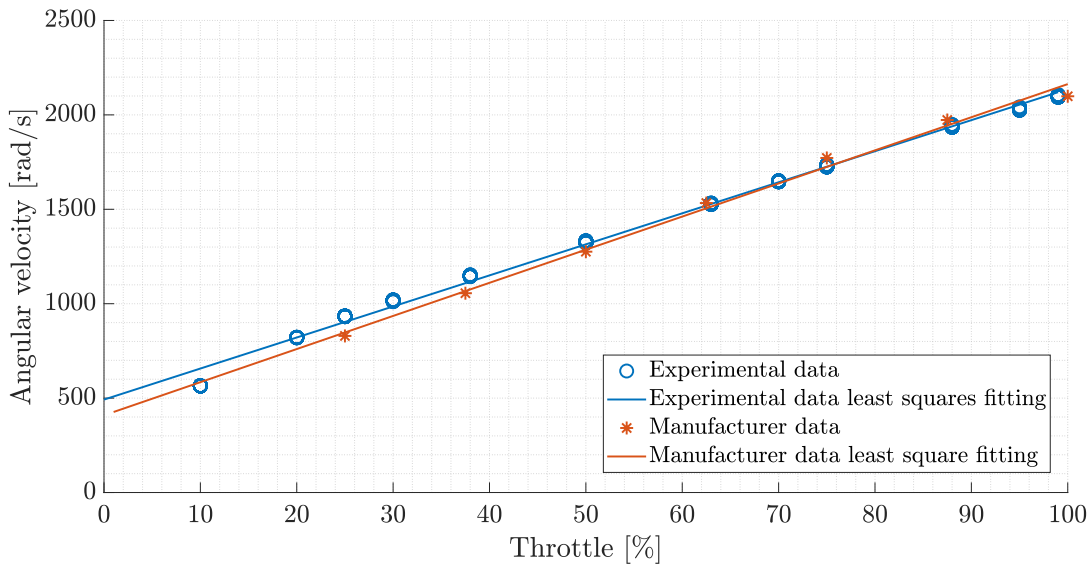


Figure B.2: Angular velocity against throttle, motor powered from DC power supply, motor for forward flight *KDE2304XF-2350*, 5x4.5 inch, 2 blades bullnose propeller.

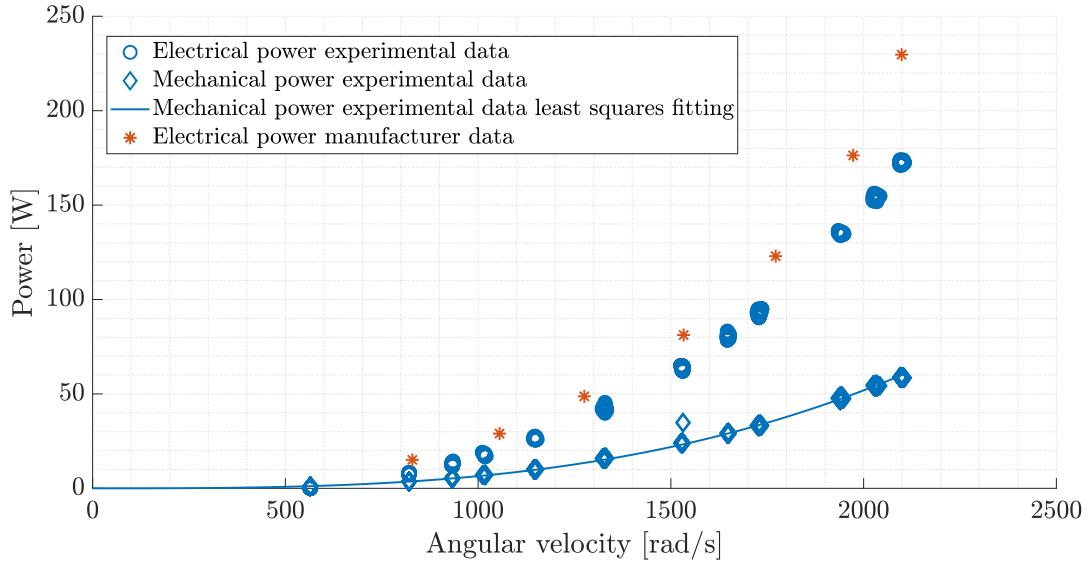


Figure B.3: Electrical and mechanical power against angular velocity, motor powered from DC power supply, motor for forward flight *KDE2304XF-2350*, 5x4.5 inch, 2 blades bullnose propeller.

Table B.1: Motor static coefficients comparison, motor for forward flight *KDE2304XF-2350*, 5x4.5 inch, 2 blades bullnose propeller.

	Experimental result	Manufacturer data
Thrust coefficient C_T	0.0164	0.0235
Torque coefficient C_Q	0.00166	0.00255
Power coefficient C_P	0.00166	0.00255

As it can be seen from the thrust against angular velocity curve (Figure B.1) the motor and propeller performance, in terms of thrust produced per angular speed, is lower compared to manufacturer's declarations, whilst the angular velocity against throttle percentage curve is in line with manufacturer data. This might be due to differences in the propeller's geometry: even though the tested propeller and the one used by the manufacturer have the same macroscopic characteristics (number of blades, diameter and nominal pitch), the actual realisation and manufacturing might be slightly different, hence explaining the different results.

Motor for vertical flight

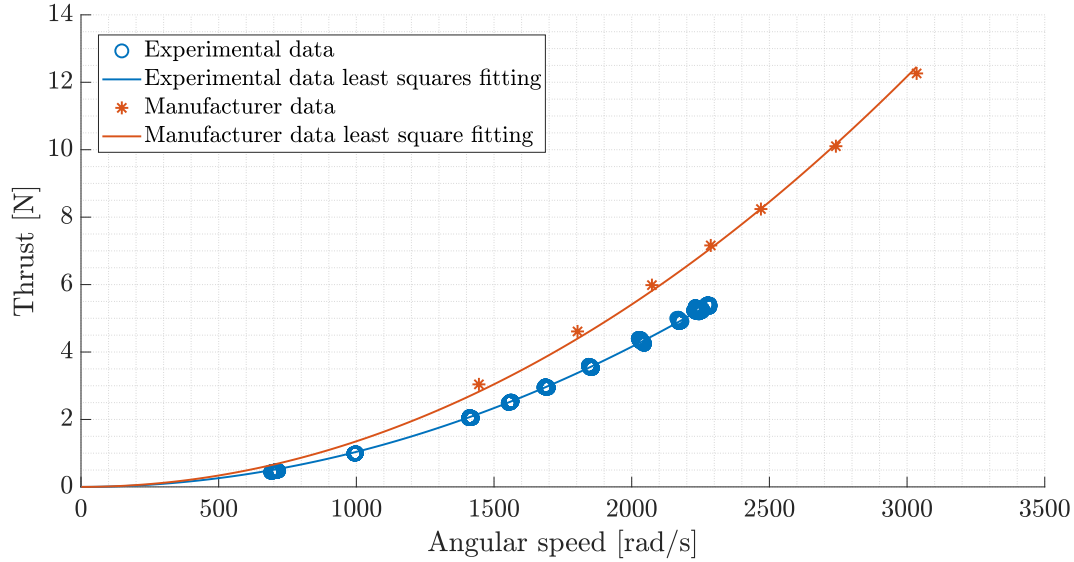


Figure B.4: Thrust against angular velocity, motor powered from DC power supply, motor for vertical flight *KDE2315XF-2050*, 5x4.5 inch, 2 blades bullnose propeller.

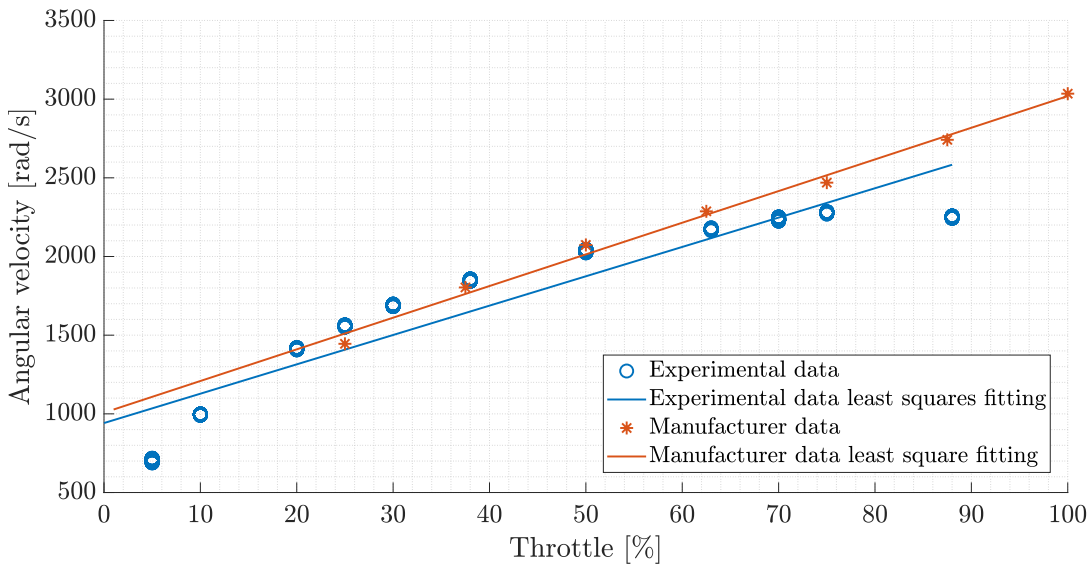


Figure B.5: Angular velocity against throttle, motor powered from DC power supply, motor for vertical flight *KDE2315XF-2050*, 5x4.5 inch, 2 blades bullnose propeller.

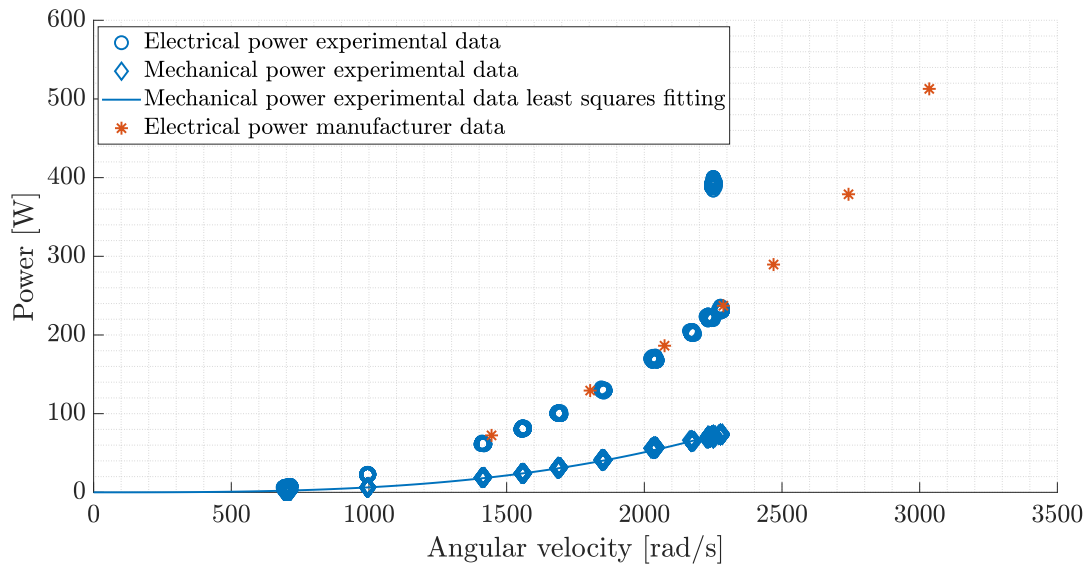


Figure B.6: Electrical and mechanical power against angular velocity, motor powered from DC power supply, motor for vertical flight *KDE2315XF-2050*, 5x4.5 inch, 2 blades bullnose propeller.

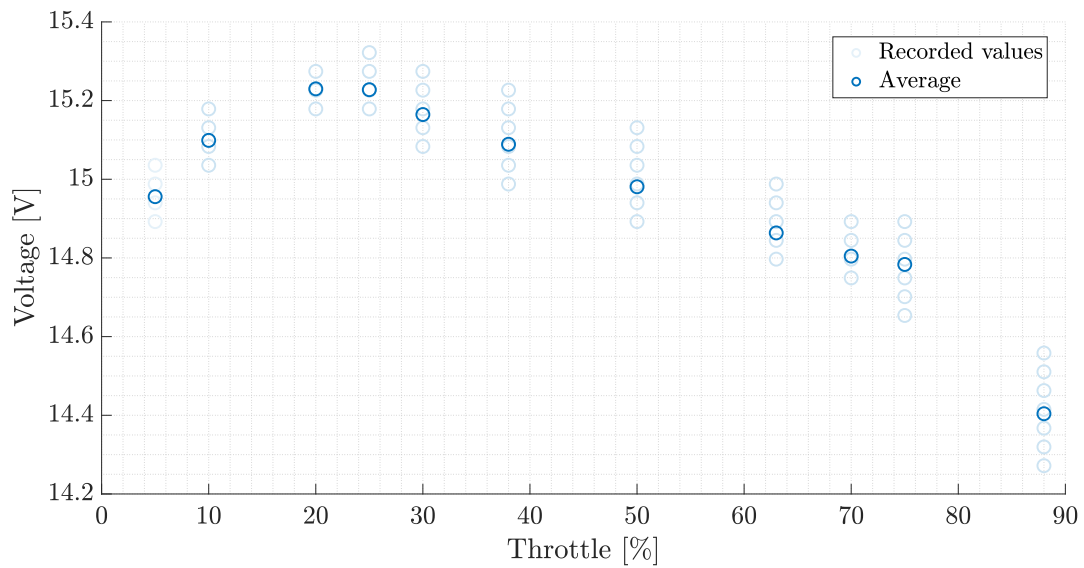


Figure B.7: Voltage against throttle, motor powered from DC power supply, motor for vertical flight *KDE2315XF-2050*, 5x4.5 inch, 2 blades bullnose propeller. The voltage drop at higher throttle setting is clearly visible.

Table B.2: Motor static coefficients comparison, motor for vertical flight *KDE2315XF-2050*, 5x4.5 inch, 2 blades bullnose propeller.

	Experimental result	Manufacturer data
Thrust coefficient C_T	0.0168	0.0219
Torque coefficient C_Q	0.00168	0.00229
Power coefficient C_P	0.00168	0.00229

Similar to the forward motor case, the vertical motor and propeller performance is slightly lower compared to the declared values. This propulsive system is capable of providing the declared angular velocity for throttle settings up to around 50%, then the performance drops, probably due to the voltage drop caused by the power supply. Again the mismatch between the declared and obtained performance might be caused by the propeller manufacturing and quality.

Appendix C

Determination of battery data from manufacturer datasheet

In this Appendix, the full procedure used to determine the battery data needed for the *Simulink* block that simulates battery performance in the flight simulator is presented. This routine has been implemented in a *MATLAB* script and is here reported because it is deemed to be useful for future similar works. The procedure is based on the suggestions from the *MathWorks* website (see [45]).

The data required for the *Simulink* block are the following:

- battery load current;
- battery temperature;
- initial battery capacity;
- rated capacity at nominal temperature;
- open circuit voltage lookup table data;
- open circuit voltage breakpoints;
- internal resistance lookup table data;
- battery temperature breakpoints;
- battery capacity breakpoints;
- number of cells in series and parallel.

The battery load current and temperature are calculated during the simulation, and the initial battery capacity is set by the user.

For the other parameters there is the need of the battery datasheet, which has to include the battery discharge charts. These diagrams plot the battery voltage

against the battery's state of charge (SOC) or remaining capacity, as a function of the discharge current (usually expressed in C-rate¹) and battery temperature. Figure C.1 reports, as an example, the datasheet used in the current work (from [2]), with highlighted the parameters needed for the *Simulink* block.

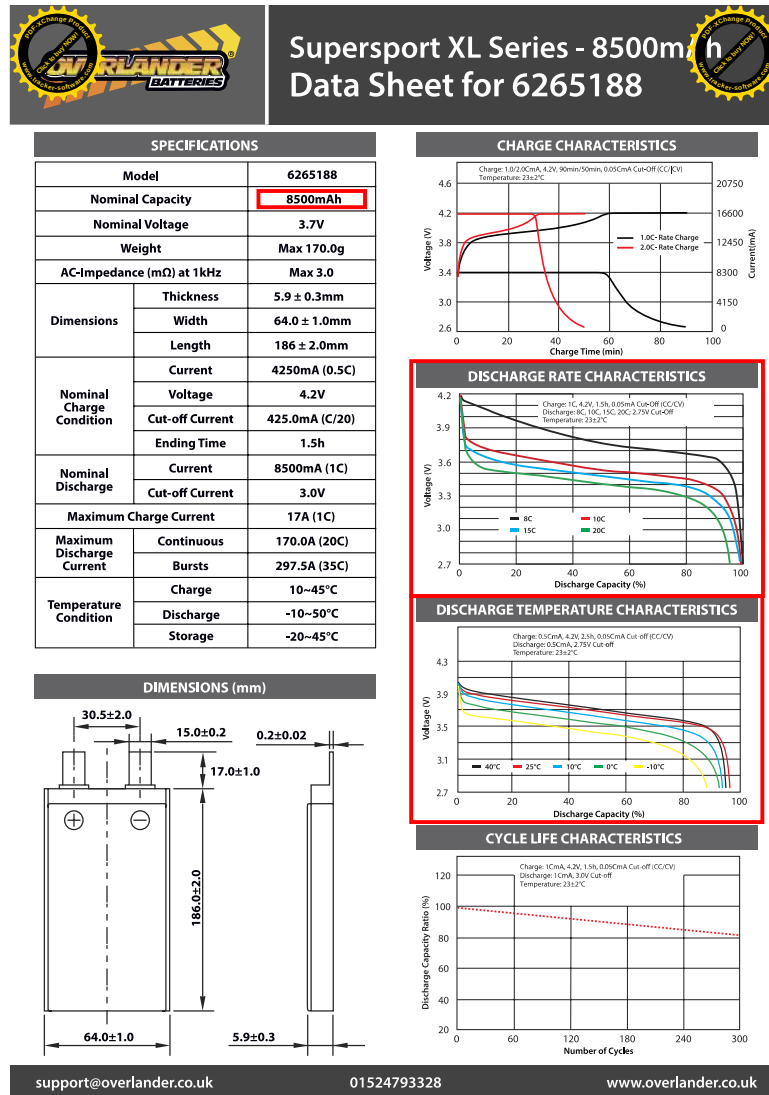


Figure C.1: Overlander 8500 mAh, 14.8 V, 20C datasheet (from [2]). Relevant data highlighted in red.

¹The C-rate is a measure of the rate at which a battery is being charged or discharged. It is defined as the current through the battery divided by the theoretical current draw under which the battery would deliver its nominal rated capacity in one hour. Hence, for example, for a battery rated at 8500 mAh, a 1C discharge delivers 8.5 Amps of current, a 10C discharge delivers 85 Amps of current, and a 20C discharge delivers 170 Amps of current.

Rated battery capacity is easily known (in this case 8500 mAh), as well as number of cells in series and parallel (in this case 4 cells in series and 1 in parallel, hence a 4S1P configuration).

The next step is to import in *MATLAB* the discharge rate and discharge temperature plots. This is done with the help of the free script called *Grabit*, available from [46]. Figure C.2 and Figure C.3 show the acquired data of voltage against Depth Of Discharge (DOD²) for constant temperature and varying current, and constant current and varying temperature.

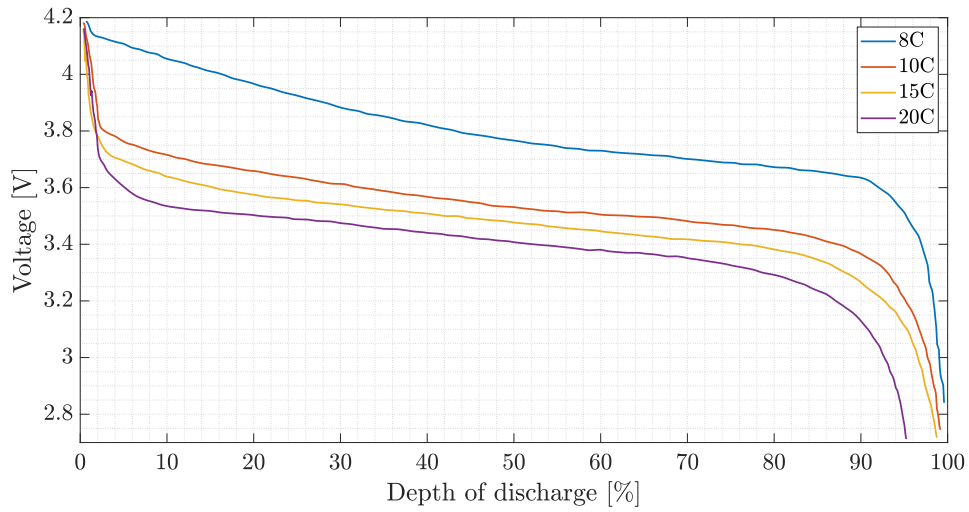


Figure C.2: Battery voltage against depth of discharge at constant temperature of 23 °C, varying discharge currents (C-rates).

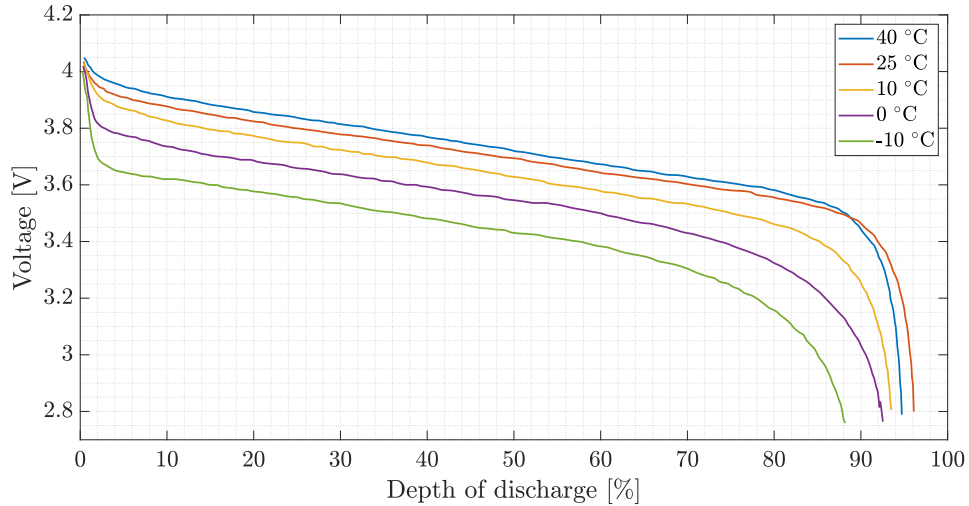


Figure C.3: Battery voltage against depth of discharge at constant current of 4.25 A (0.5C), varying temperatures.

²The Depth Of Discharge (DOD) indicates how much a battery has been discharged, and is the complement of the state of charge.

The following step is to normalize the depth of discharge data: hence, instead of having the depth of discharge going from 0% to 100%, it then goes from 0 to 1.

Then, a reduced set for the normalized depth of discharge data is chosen. For instance, going from 0.1 DOD to 0.9 DOD with a step of 0.1. With this reduced set, a new plot is created, which is just a rearrangement of the data already present in Figure C.2. For every value of the reduced set of the depth of discharge, a plot can be made of the voltage against the current drawn. Since the available data has 4 different discharge currents (8C, 10C, 15C, 20C, which in turn correspond to 68 Amps, 85 Amps, 127.5 Amps and 170 Amps), each of these plots will have four data points. This process is repeated for all the values in the reduced depth of discharge vector, leading to the plot shown in Figure C.4. Then, on this plot, for each value of the depth of discharge the open circuit voltage can be found by extrapolating the corresponding curve to the zero current point (represented by the circle on the plots; the extrapolation is shown by the dashed lines). With this process, the open circuit voltage lookup table data (the collection of the open circuit voltages for each value of the reduced depth of discharge set) and the open circuit voltage breakpoints (the reduced depth of discharge set) are found.

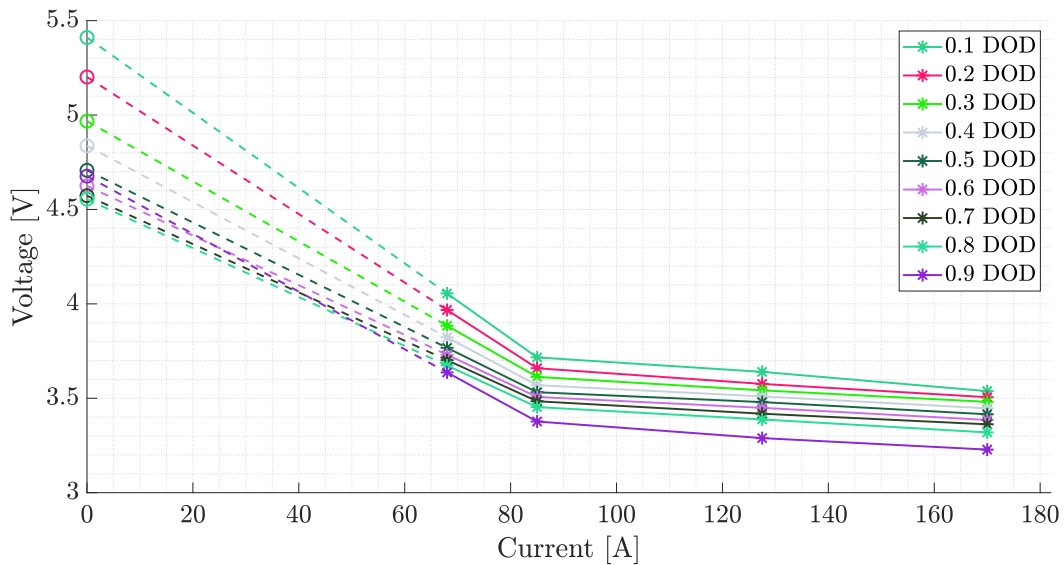


Figure C.4: Battery voltage against current for different depths of discharge. Extrapolation to zero current shown in dashed lines; open circuit voltages represented by circles.

The final step is to calculate the battery internal resistance, which in general is a function both of the depth of discharge (or the state of charge) and temperature.

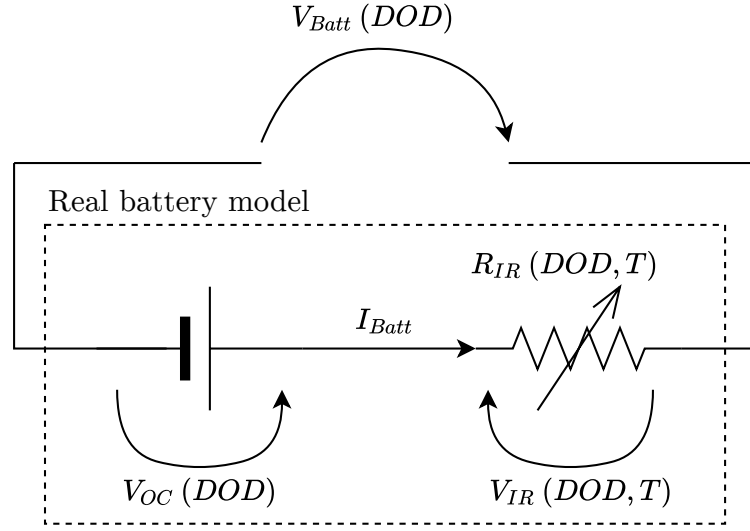


Figure C.5: Battery schematics electrical diagram.

With reference to Figure C.5 the real battery is modelled as a series connection of an ideal battery, with no internal resistance, which provides a voltage equal to the open circuit voltage $V_{OC}(DOD)$ previously calculated (function of the depth of discharge (DOD)) and a variable internal resistance, $R_{IR}(DOD, T)$, which depends on both the DOD and temperature and generates a voltage drop $V_{IR}(DOD, T)$ across it. The actual voltage delivered by the battery, $V_{Batt}(DOD)$, which also depends on the depth of discharge, is the one measured during the battery discharge characterization tests, shown in Figure C.2 and Figure C.3. According to Kirchoff's voltage law it is equal to

$$\begin{aligned} V_{Batt}(DOD) &= V_{OC}(DOD) - V_{IR}(DOD, T) \\ &= V_{OC}(DOD) - R_{IR}(DOD, T) \cdot I_{Batt} \end{aligned} \quad (C.1)$$

Rearranging equation (C.1), the value of the internal resistance is easily obtained, by knowing the open circuit voltage (already obtained) and the values of current and voltages as function of both the depth of discharge and temperature, from Figure C.3:

$$R_{IR}(DOD, T) = \frac{V_{OC}(DOD) - V_{Batt}(DOD)}{I_{Batt}} \quad (C.2)$$

Hence, the final required values for the battery characterization are found: the internal resistance lookup table data consists of the values of the internal resistance as function of both depth of discharge and temperature; the battery temperature breakpoints are the values of temperature used for the determination of the battery internal resistance; the battery capacity breakpoints are the values of the depth of discharge used for the determination of the battery internal resistance.

However, a problem arose with this calculation using the available data: resistance values in the order of 0.3Ω were found, way too high to be realistic (usual internal resistance values are in the order of milli-Ohms or centi-Ohms). This is probably caused by discrepancies in some datasheet reported values. In Figure C.3, the reported discharge current is $0.5C$. However, if looking at the plot of the discharge at 25°C and comparing it to the $8C$ discharge at 23°C of Figure C.2, it can be seen that for the former the voltage is constantly almost equal, or even lower, than for the latter. However this does not make sense, since for the same temperature, a much lower discharge rate (more than 10 times smaller in this case), should lead to a consistently higher voltage during the discharge. Hence, it is believed that some data is wrong in this datasheet.

The resistance data was then calculated using values only from the constant temperature data of Figure C.2, and hence assuming that the resistance is constant with temperature. This is a rough assumption, but since no other data was found for a similar battery and no actual battery temperature values during the simulation are calculated, it was seen as the only possible solution.

Figure C.6 shows the value of the battery internal resistance as a function of the normalized depth of discharge, for the temperature of 23°C .

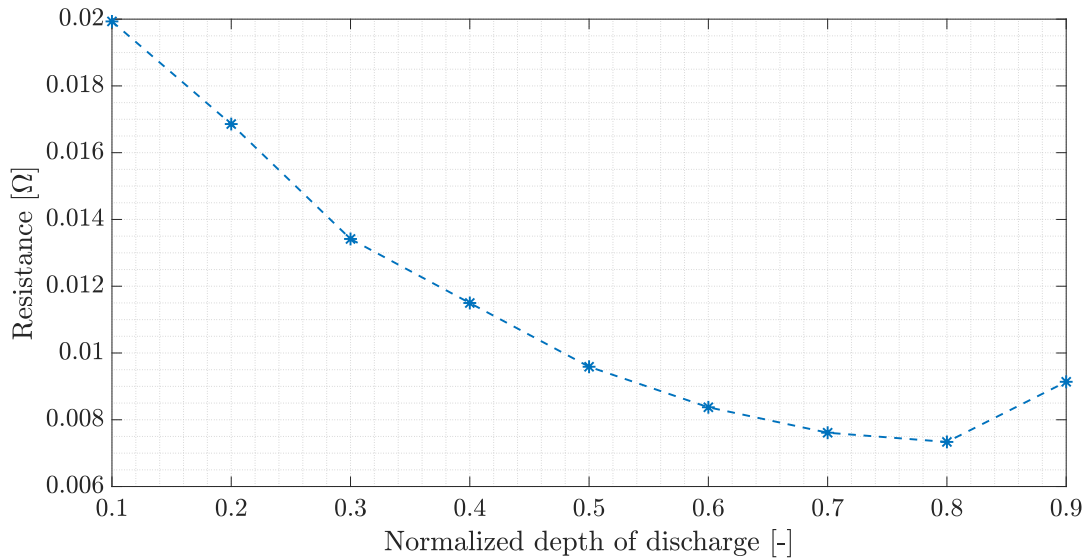


Figure C.6: Battery resistance as a function of the normalized depth of discharge, temperature of 23°C .

Appendix D

3D Printing technologies

In this Appendix, the two main 3D printing technologies which have been mentioned in Chapter 5 are briefly described. The usage of Additive Manufacturing (AM) allows for rapid prototyping and quick implementations of changes to the CAD model, all extremely important factors in this work.

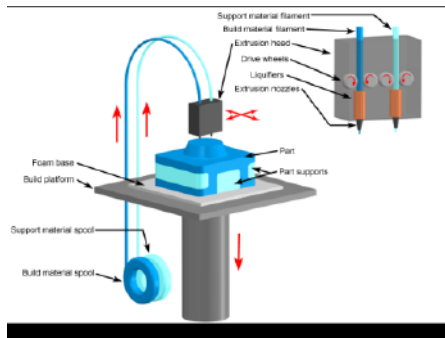
D.1 Fused Deposition Modeling (FDM)

Fused Deposition Modeling (FDM) (Figure D.1a) has been used for all the eVTOL UAV's components which were 3D printed. This additive manufacturing technology consists in the deposition of a molten plastic filament layer by layer. The CAD model is sliced and each layer deposited on top of each other by the printer's extruder. This technique has the advantage of being relatively simple, with small and cheap printers readily available. The main drawbacks are that delamination might happen along the direction of filaments' deposition and that for the cheaper printers thermal deformation of the plastic might cause distortions in the printed piece. Furthermore, in order to print overhanging parts, there is the need to print some sort of support, which will later be hand-removed when cleaning up the piece. All the 3D printed components but the fuselage (which are all the motor supports, wing ribs, avionics' support, fuselage feet, tail, tail supports) have been printed using a small *Ultimaker S3* ([47]) desktop printer (Figure D.2a), whereas the fuselage has been printed using the much bigger industrial *Stratasys Fortus 450 MC* ([48]) (Figure D.2b).

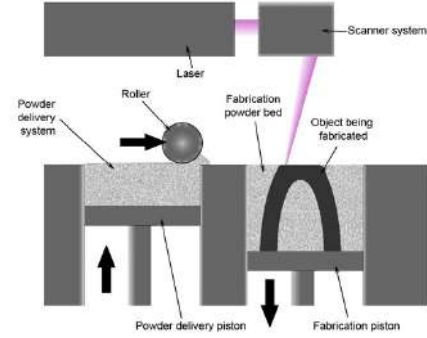
D.2 Selective Laser Sintering (SLS)

Selective Laser Sintering (SLS) (Figure D.1b) was the intended original production method for the fuselage, since it allowed to print its whole lower part in a single piece. The SLS 3D printing technology consists in the deposition of a thin layer of

plastic powder, which is then sintered¹ using a laser. This procedure is repeated layer by layer. It has the advantages of being more precise compared to FDM and most importantly there is no need to print additional supports for overhanging parts, since the unfused plastic powder acts itself as a support. No component of the eVTOL UAV has been printed with this technology.

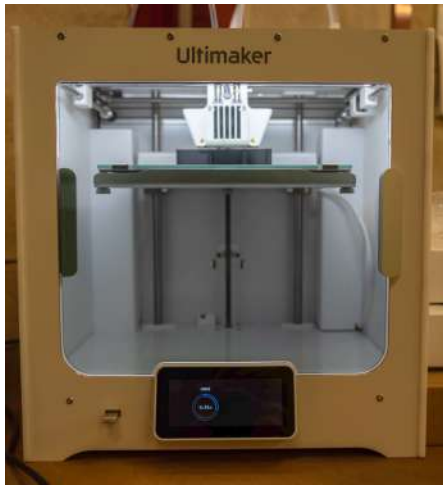


(a) Fused Deposition Modeling (FDM)¹.



(b) Selective Laser Sintering (SLS)².

Figure D.1: Schematics of 3D printing technologies: (a) Fused Deposition Modeling (FDM) and (b) Selective Laser Sintering (SLS).



(a) *Ultimaker S3* FDM printer.



(b) *Stratasys Fortus 450 MC* FDM printer.

Figure D.2: FDM printers used for the construction of the eVTOL UAV: (a) *Ultimaker S3* and (b) *Stratasys Fortus 450 MC*.

¹Sintering is the process of compacting and solidifying a powder material, without melting it to the point of liquefaction.

¹Picture from [49].

²Picture from [50].