Politecnico di Milano
School of Industrial and Information Engineering
Master of Science in Aeronautical Engineering

# Modelling, identification and control of a fixed-wing UAV

Advisor: Prof. Marco LOVERA
Co-Advisor: Ing. Mattia GIURATO

Thesis by:
Andrea GATTI Matr. 884196

Academic Year 2018–2019

*Giunto alla conclusione del mio percorso di formazione universitaria, mi trovo ora all'inizio di una lunga strada che, mi auguro, vedrà compiersi i sogni da lungo tempo racchiusi in angusti barattoli. Tuttavia, tali sogni non avrebbero neppure avuto la possibilità di formarsi se non mi fosse stata concessa l'opportunità di crescere all'interno di un'università supportato da amici, docenti e familiari. Un particolare ringraziamento va infatti alla mia famiglia, che con i loro sforzi mi ha permesso di avvicinarmi sempre più ai sogni che custodisco fin da bambino e che, nel tempo, hanno preso forma e acquisito consistenza.*

# Acknowledgments

Desidero ringraziare il laboratorio ASCL, diretto dal prof.Marco Lovera. Con la costante presenza ispiratrice dell'Ing.Mattia Giurato, del dott.Davide Invernizzi e del dott.Simone Panza ho avuto l'opportunità di confrontarmi e approfondire uno dei temi più attuali e ferventi dell'aeronautica moderna, ovvero gli aeromobili a pilotaggio remoto. Grazie alla loro passione e perseveranza ho avuto l'opportunità di confrontarmi con questo mondo e aggiungere ulteriori tasselli alla mia cultura aeronautica.

# Abstract

With the advent of low-cost electronic systems for the automation of certain functions related to the conduct of flight, many enthusiasts and scholars have had the opportunity to turn their model aircraft into real Unmanned Aerial Vehicle (UAV). These electronic systems are available on the market as "ready to fly": only the integration phase and performing a calibration procedure are enough to obtain an aircraft capable of performing automatic missions controlling position, airspeed and altitude, eventually without the intervention of the operator on the ground. During our experiments, a fixed-wing motor-glider was fitted with a "Pixhawk" Flight Contro Unit (FCU), one of the many hardware available on the market for such applications. On this hardware, the ArduPilot firmware was installed, being already known in the environment of the Polytechnic University Of Milan. The Pixhawk was then equipped with the telemetry module needed to receive and send data to the ground in order to give to the operator real-time data of the UAV. These solutions are having a major impact in the current economy: many activities can be carried out more effectively and economically than traditional solutions and, not least, new professionals have also appeared. Compared to multirotors, fixed-wing aircraft show the undeniable advantage of having greater range and endurance, as well as a greater payload capability. In this thesis, the main tasks necessary to tune the control laws inherent the FCU will be addressed to improve the performance of the attitude control, given the available hardware (servos, size and deflection of the aerodynamic surfaces, etc...) because the automatic tuning that such FCU's can perform online during appropriate flight missions are of course effective and reliable, but they certainly do not carry it out optimally nor even try to satisfy any user specification. The so-called "stock" tuning of the control laws obtained using the "Autotune" function available with ArduPilot, tunes only some gains of the attitude control loops during the dedicated tests, setting the remaining gains with pre-defined values based on the aggressiveness of the tuning desired by the user. The aim of the thesis is therefore to exclude the "Autotune" function from the UAV tuning procedure calibrating, through specific experiments and computations, the above gains of the control loops to achieve the performance desired by the user. These performance constraints can be various: control, mission, atmospheric disturbance rejection performance, and others. In this thesis, this procedure was carried out by exploring the potential of the Virtual Reference Feedback Tuning (VRFT) technique for calculating the gains of the control loops, combined with an iterative methodology to evaluate the obtained performance and to get a evaluation to the goodness of the closed-loop system with the user specifications. An iterative procedure based on the exploration of a finite domain of different combinations of the reference model's parameters used in the VRFT procedure, made it possible to explore a set of possible gain solutions. Those solutions have been inserted in the closed-loop system and the performance have been then evaluated through a cost function to find the best possible combination of parameters to control the system according to user's specifications. The computed performance was subsequently compared with the in-flight tests and the results, among with a detailed discussion, are presented in this thesis.

# Sommario

Con l'avvento dei sistemi elettronici a basso costo per lo svolgimento automatico di alcune funzioni legate alla conduzione del volo, numerosi appassionati e studiosi hanno avuto la possibilità di trasformare il proprio aeromodello in vero e proprio Aeromobile a Pilotaggio Remoto (APR). Tali sistemi elettronici si presentano come "pronti al volo", ovvero è sufficiente integrarli al proprio aeromodello ed eseguire alcune procedure di calibrazione per ottenere un velivolo in grado di eseguire missioni in maniera autonoma, gestendo posizione, velocità e quota senza l'intervento dell'operatore a terra. Durante la nostra sperimentazione, un motoaliante ad ala fissa è stato arricchito di un modulo "Pixhawk", uno dei tanti hardware disponibili sul mercato per tali applicazioni. Su tale hardware è stato installato ArduPilot come firmware, essendo già noto nell'ambiente del Politecnico di Milano. La Flight Control Unit (FCU) Pixhawk è stata quindi corredata del modulo telemetria necessario per la ricezione e l'invio dei dati a terra così da fornire all'operatore i dati in tempo reale delle prove in atto. Tali soluzioni stanno avendo un grande impatto nell'economia attuale: numerose attività infatti possono essere svolte in maniera più efficace ed economica rispetto alle soluzioni tradizionali e, non da ultimo, ha visto comparire anche nuove figure professionali. Rispetto ai multirotori, i velivoli ad ala fissa mostrano l'innegabile vantaggio di poter disporre di una maggiore autonomia (sia oraria che chilometrica) nonchè disporre di un maggior carico utile. In questa tesi verranno affrontati i principali temi necessari alla calibrazione delle leggi di controllo insite nel computer di bordo per migliorare le prestazioni sul controllo di assetto dato l'hardware a disposizione (servocomandi, dimensioni e movimento delle superfici mobili, ecc...) in quanto le calibrazioni che tali unità di controllo possono svolgere on-line durante apposite missioni di volo svolgono in maniera accettabile tale compito ma sicuramente non lo portano a compimento in maniera ottimale. La calibrazione cosiddetta "stock", ovvero la calibrazione delle leggi di controllo ottenuta usando la funzione "Autotune" disponibile con ArduPilot, regola solamente alcuni guadagni degli anelli di controllo dell'assetto durante le prove dedicate, impostando i rimanenti con valori predefiniti a seconda dell'aggressività della traratura scelta dall'utente. L'obiettivo della tesi è quindi escludere la funzione "Autotune" dalla messa in linea dell'APR e tarare, mediante specifici esperimenti ed elaborazioni, i suddetti guadagni degli anelli di controllo per ottenere le prestazioni desiderate dall'utente.

Tali vincoli sulle prestazioni possono essere di varia natura: prestazioni di controllo, di missione, di rigetto dei disturbi atmosferici e altri. In questa tesi si è esplorata tale procedura mediante l'uso della tecnica VRFT (Virtual Reference Feedback Tuning) per il calcolo dei guadagni degli anelli di controllo, successivamente inseriti nel sistema in anello chiuso e valutate le prestazioni. Una procedura iterativa basata sulla variazione dei parametri dei modelli di riferimento in uso nella procedura VRFT entro range prefissati ha reso possibile esplorare un insieme di possibili soluzioni, alla ricerca della soluzione ottima per il soddisfacimento delle specifiche minimizzando determinati indici di merito. Le prestazioni calcolate per via numerica sono state successivamente comparate con le prove in volo ed i risultati delle stesse presentati in questa tesi.

# Contents

# List of Figures

# List of Tables

# Introduction

## Overview

In the last few decades, a fervent interest has raised around the so-called Unmanned Aerial Vehicles.

Figure 1: Leonardo ”Falco-EVO”

The UAVs are, as often happens for new aerospace technologies, a reality that firstly came up in the military field and then found civil application. However, a huge evolution followed the miniaturization of the electronics and the falling of its costs, giving the possibility to group of people from companies, research institutes, or simply hobbyist, to find civil applications. A great example of civil companies using UAV's for their business is Zipline (see figure 2), an enterprise devoted to the delivery of medical equipment alongside remote rural areas, testing new technologies and strategies to succeed in their intent.

The UAVs definition includes a variety of different type of vehicles and sizes. Most common are fixed-wing vehicles and multi-rotors vehicles; this thesis is focused on the first category. This work puts focuses on the miniature UAVs or small UAVs (SUAVs), which are aerial vehicle small enough to be man portable. These platforms are gaining more and more interest in the academic and research fields

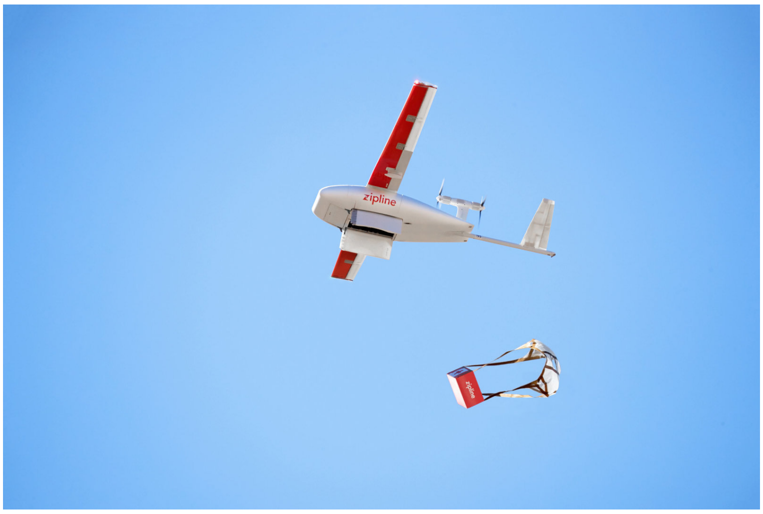

Figure 2: Zipline's UAV used to deliver blood products and medicines

because they allow the development and testing of new control algorithms, navigation systems and sensor-fusion algorithms. Civil and commercial applications are not developed as the military counterpart, although potential applications are extremely broad in scope, including:

- environmental monitoring;
- inspection or surveillance;
- homeland security;
- aerial mapping;
- search and rescue operations;
- cargo deliveries;
- first aid and remote medical assistance.

To extend the usefulness of fixed-wing UAVs beyond their current applications, the capability to plan paths and to follow them accurately is of great importance. Unlike piloted vehicles, which rely on the pilot to navigate over demanding terrain or to avoid obstructions, UAVs rely on automation to provide this functionality. As applications such as urban surveillance and rural search and rescue require UAVs to fly down city streets surrounded by buildings or near the surface of abruptly changing mountainous terrain, the ability to follow pre-planned paths with precision or maintaining a stable attitude, even in rough weather, is essential. For missions involving cooperation among a team of UAVs, precise path tracking (and of course following) is crucial to achieving the cooperation objective. For small aerial vehicles (sUAVs), such as those of primary interest in this study, wind disturbances, dynamic characteristics, and the quality of sensing and control all limit the achievable tracking precision. For UAVs, wind speeds are commonly 20%-50% of the desired airspeed. Effective control law tuning strategies could overcome the effect of this ever-present disturbance. For most fixed-wing UAVs, the minimum turn radius is in the range of 10-50 m. The more the UAV is maneuverable, the more will be its usefulness to follow pre-determined routes or, in general, to follow in a more accurate manner its flight path rejecting disturbances. To do this, the flight control unit must work in close cooperation with the aircraft, to extract all its potential.

# Thesis description

The focus of this thesis is to explore the capabilities of the tuning, using the VRFT approach, of the control loops that controls the attitude of a fixed-wing UAV. Since this approach requires a reference model that the user must find in a way that requires time and patience, a new algorithmic procedure for guessing the latter was built up to overcome this limitation. The algorithm accepts as input a finite domain of parameters along whom it will build the reference model to compute, using the VRFT procedure, the controller gains. Next, the gains are tested in the closed loop system and its performance evaluated according to a cost function that can be configured by the user accounting for his particular requirements, whether they are mission or performance specifications or both simultaneously. The algorithm ends the process when it finds a global minimum of the prescribed cost function, corresponding to the optimal gains of the control loops, among the input parameter domain. The performance evaluation has been carried out both in the simulation environment as well as experimentally, flying the aircraft with the computed gains both for longitudinal as well for the lateral-directional attitude control loops simultaneously. A detailed description and comment of the results obtained is then offered to the reader in the following thesis.

# Thesis structure

Chapter 1 provides a detailed introduction of the hardware used, as well with the safety procedures written to ensure safe operations;

Chapter 2 is focused on the procedures, methods and mathematical instruments used to perform the model identification of our aircraft. Starting from the input design, the reader is carried trough procedures, measurements, methods and assumptions employed in this work to identify the aircraft dynamics, both for the longitudinal as well for the lateral-directional channel. Finally, the identified model results, among with their limitations, are presented and discussed.

Chapter 3 provides a description of the methods employed to compute a new set of gains to achieve the best-possible performance of our identified model using a VRFT approach combined with an iterative evaluation of the closed-loop systems. In this chapter, both longitudinal and lateral-directional control loops are tuned, focusing particular attention to the decoupling of the lateral from the directional motion given a roll input. Finally, a comparison between simulated and experimental data is presented and discussed.

# Chapter 1

# The Multiplex Cularis UAV

In this chapter the aircraft used in the thesis work will be presented, together with a description of the systems carried on-board.

Furthermore, an overview about the safety procedures adopted to ensure safe operations during the flight test campaigns is presented.

## 1.1 Description of the platform

The Cularis (Figure 1.1) is an electric powered model glider produced by the German brand Multiplex, famous for the introduction in the r/c world of the Elapor, a material very similar to propylene foam, very light but durable at the same time. Mainly made up of the above-mentioned material, this model is a medium/high class glider with a 2.61 meters wing reinforced by carbon fiber spars and a 1.26 meters fuselage reinforced by PVC. The propeller is mounted in the nose in pulling configuration and the blades automatically folds when the motor is off in order to reduce the drag during glides and ensure safe landings due to the lack of landing gears

Mentioning the high aspect ratio and the slender body typical of a glider, this design is optimized to have a high efficiency necessary to ensure ascending flights in dynamic currents or thermals just as for the case of a real size piloted glider. Preliminary flight testing show a tendency of the electronics to overheat. This was due the reduced ventilation in the unique vane of the fuselage where the FCU was placed. To overcome this issue, as one can see in Figure 1.2, a custom nose ventilation air duct was added.

## 1.2 Hardware/software integration

The propulsive system installed on the Cularis makes the aircraft a hybrid between a powered airplane and a pure glider with the advantages of both configurations. In fact, the available power is way enough to take off without the need of another

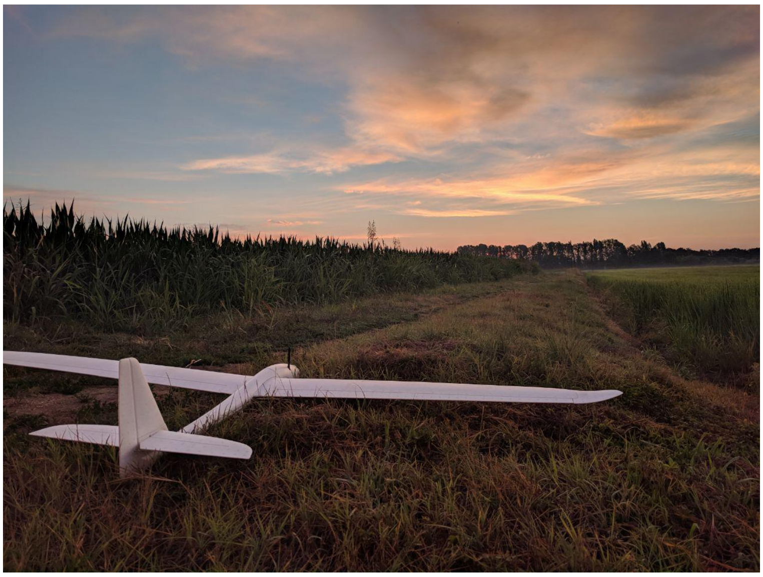

Figure 1.1: The Multiplex Cularis UAV

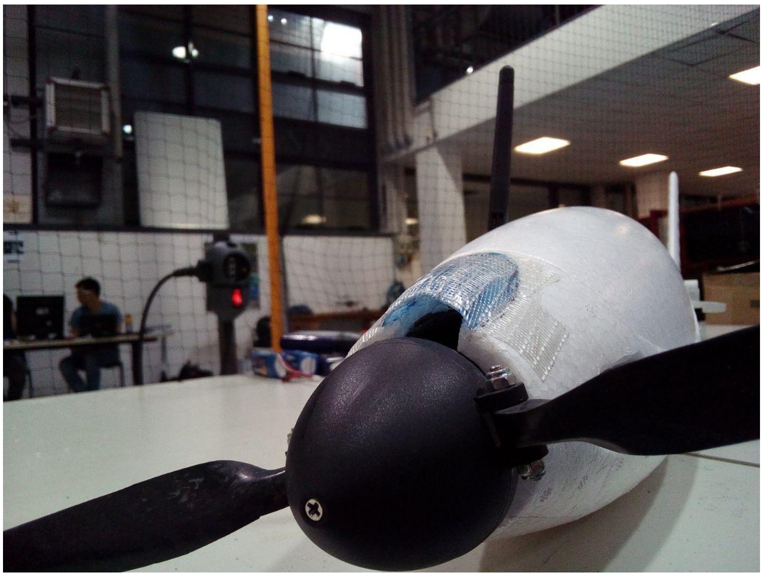

Figure 1.2: Custom nose ventilation duct

motorized airplane or a winch catapult system to get to the desired altitude and on the other hand, the folding propeller doesn't add too much drag during unpowered glides. The propulsive system includes four components connected to each other:

- Propeller
- Electric motor
- Electronic Speed Controller (ESC)
- Battery.

The propeller connected to the motor shaft is a twin blades fixed pitch, folding, of dimensions 12" x 6" with low warpage to ensure high performance especially at low speeds maneuvers as takeoff. We already mentioned the advantage of having a folding propeller in order to reduce drag during glides, but this feature also becomes fundamental during landings because the model doesn't have landing gear and it stops itself by friction simply contacting with the low grass of the landing field. If the propeller couldn't fold, an impact with the ground would be inevitable. The brushless motor installed on the Cularis can generate 400 W power and it is recommended for aerobatic models up to 1 kg and for glider models up to 4.5 kg. Brushless motors have many advantages compared to the old brushed motors as the high ratio torque/weight and their high efficiency. Moreover, these kinds of electric motors are highly reliable and have a longer life cycle. For all these reasons they became very popular in the world of airplane models.

The entire aircraft was built from two different uncomplete boxes of Cularis spare parts, which combined could form a new platform, but the motor mount was missing. To overcome this issue, a new motor mount was designed and 3D printed. A cut view of the 3D model is reported in Figure 1.3.

The Electronic Speed Control (ESC) is an electronic circuit with the function of varying the angular speed of the motor, its sense of rotation, and in some cases can also function as a dynamic brake. This component is an independent unit, connected at one end to the Power Module with two wires and on the other end to the motor. A third cable comes out from the ESC which is the BEC end (Battery Eliminator Circuit) connected to the receiver that controls the throttle signal and supply power to the receiver and servos without the need of another battery. A picture of the ESC used in our setup is shown in Figure 1.4.

As power source we use rechargeable Li-Poly batteries. The Li-Poly batteries are a technologic advancement of the Li-Ion batteries from which they differ for the electrolyte restrained in a solid polymer instead of in an organic solvent. The Li-Po batteries are the most used in r/c applications due to their lower weight and enhanced capacity, the most requested characteristics in this field that make them preferable over the other types of batteries despite a higher cost. The batteries in use on the Cularis are the “FullPower silver edition v2”, 3S, 2200mAh. The advantage of having many batteries at the field is that while one is being used

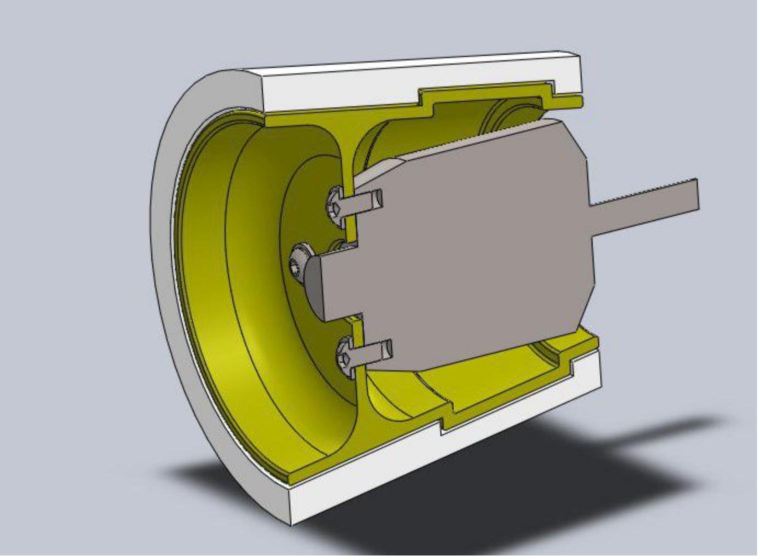

Figure 1.3: 3D printed motor mount cut view

Figure 1.4: AeroStar 30A ESC

Figure 1.5: Li-Po battery FullPower 2200mAh 3S

on the airplane, the others can be recharged at the same time ensuring an almost non-stop cycle of flights.

On the Cularis there are installed six analogic servomotor to move the control surfaces: one for each aileron (2x), one for each flap (2x) and two inside the canopy; one for the elevator and one for the rudder.

Each servo, through a small integrated electronic circuit, recognizes the input set-point coming from the receiver and realizes a closed-loop control on the angular position output of its shaft properly powering the small electric motor to which it is connected to a control rod through a reducer. In our particular setting, flaps will not be uset to avoid increasing the number of variables affecting our tests.

The chosen servomotor are the Hextronik HXT900, a reliable low cost off-the-shelf servomotor with an extremely good power to weight ratio. A complete list of the servo characteristics can be found in Table 1.1.

The radio control system in use (Figure 1.8) is composed by the digital transmitter FrSky Taranis X9D and by the receiver X4R (Figure 1.9) of the same brand. It is the state of the art for the RC system, with the transmitter capable of managing up to 16 radio channels in the 2.4 GHz band using the ACCST frequency hopping technology, which ensure high resistance to external interferences and permits a safe use of many of these systems. An optional telemetry channel was available.

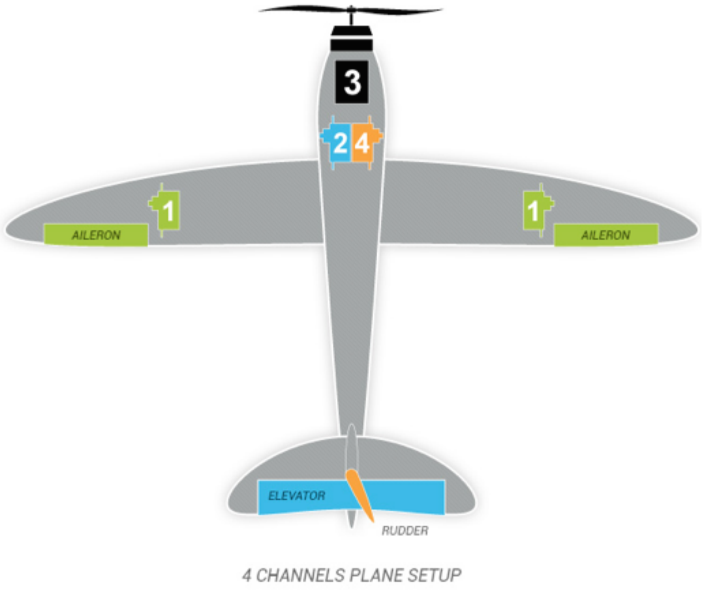

Figure 1.6: Servo collocation diagram

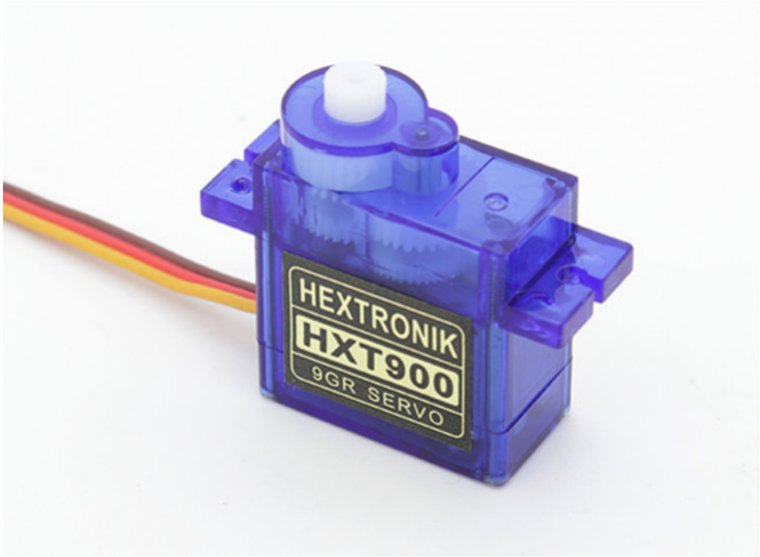

Figure 1.7: Hextronik HXT900 servo

| | |
|---|---|
| Dimensions | 21x12x22mm |
| Voltage | 3V to 6V |
| Weight | 9g |
| Angular speed | 0.12sec/60° |
| Torque | 1.6kg/cm |

Table 1.1: Hextronik HXT900 servo characteristics

Figure 1.8: FrSky Taranis X9D

The receiver can manage up to 16 channels (e.g. ailerons, elevator, motor, rudder, flaps. . . ) and normally handles the input signal from the transmitter and generates the correct output to the servos through PWM (Pulse Width Modulation) signals. The duration of the impulse is proportional to the position of the control sticks of the transmitter maneuvered by the pilot. As one can notice, our receiver dose not have many outputs as the old conventional receivers. Instead, each channel is multiplexed to the flight control system via SBUS, allowing the use of a single cable.

The Flight Control Unit, or FCU, is the core of each autonomous UAV. Choosing a right control unit among the existing wide range of options is not trivial. Before choosing the micro-controller hardware, the autopilot software must be defined. Recalling that the final goal is to tune the low-level controllers, an open source firmware is desirable because of the numerous sources of information about its architecture and functionalities would be freely available on the dedicated online website. The Pixhawk system (Figure 1.10) is an open-hardware fight controller specifically meant for UAV applications. It has two redundant IMUs, which are necessary for the estimation of the attitude of the aircraft. An IMU (inertial measurement unit) is an electronic device which integrates an accelerometer, a gyroscope, and usually a magnetometer. One IMU contains a 3-axis accelerometer, a 3-axis gyro, and one magnetometer, while the second one only contains

Figure 1.9: FrSky X4R receiver

an accelerometer and gyro, both different from the first IMU. Any accelerometer can output three acceleration measurements, one per axes, while gyroscopes measures the body angular rates on the three orthogonal axes. There is also a barometer for an indirect altitude measurement. Finally, the system can save data logs on an on-board MicroSD card. In the overall, this is an open-source autopilot of professional quality and the most advanced open-source IMU-based autopilot. In the intentions of its creators, it has been conceived to transform a standard R/C model into a real UAV. It can be configured interactively through a ground control station (GCS) and it's capable of flying by itself in accordance with some automatic modes. On the firmware side, ArduPilot can run on many different micro-controllers and platforms. We preferred this firmware instead of PX4 because some previously done work on the low-level controllers was already performed using this solution. However, it was discovered a huge confusion in the way the low-level controllers are labeled. We strongly encourage passing on PX4 for future works. Among the many flight modes available, for the purpose of this thesis we used the following:

- Manual: the FCU acts as a passthrough between the command inputs and the signal sent to the servos.

- Fly-by-wire-A (FBWA): the FCU interprets as input from the radio an at-

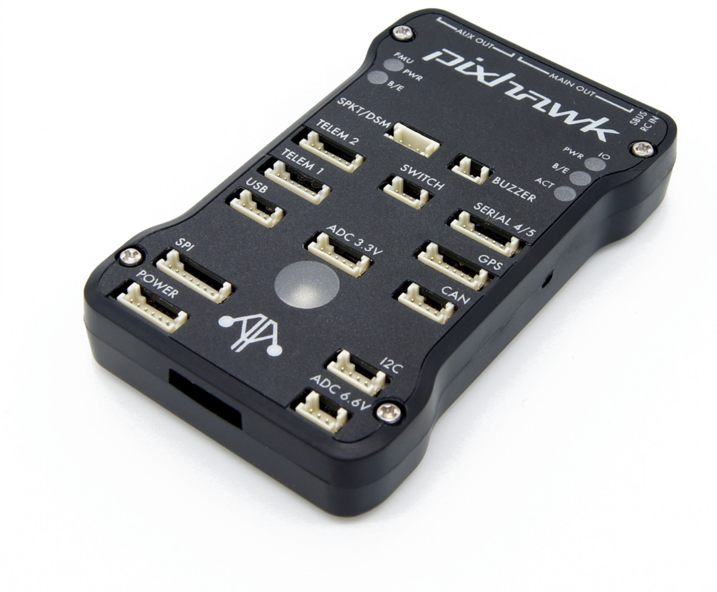

Figure 1.10: Pixhawk FCU

titude request and controls the aircraft accordingly to keep that attitude.

- Autotune: it is the same as the FBWA mode but, meanwhile uses the aircraft response to tune the pitch and roll controllers and adjusts the control parameters online. The aggressiveness of the tuning is settable by the user through a value which can range from 1 to 10. For our tests, we adopted an intermediate value of 5.

The Pixhawk has a dedicated connector for attaching the 3DR Power Module (PM, see Figure 1.11) which is useful because it provides a stable 5.3 V and 2.25 A power supply to the board. It allows monitoring of the battery's voltage and current making it possible to trigger a return-to-launch when the voltage becomes low or the total energy consumed during the flight approaches a certain threshold.

The Pixhawk (running ArduPilot) supports the use of an analog airspeed sensor, which is of paramount importance while trimming the aircraft, setting-up each test at the same indicated airspeed. It is a more accurate way of determining the speed of the airplane in the moving air, of course, than using the information coming from the GPS and can be also very useful in preventing stalls. It is made of two main parts: the pitot tube with both total and static pressure ports and the piezoelectric pressure sensor element that in this case is the Freescale Semiconductor® MPXV7002, as shown in Figure 1.12. The pitot was installed on the left wing, outside of the propeller slipstream, in a position that is outside to many aerodynamic interferences during normal flight.

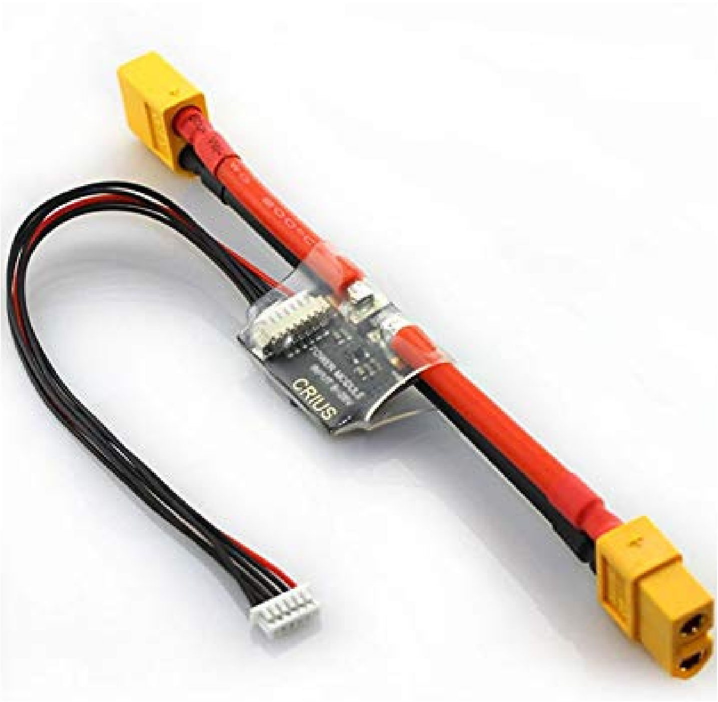

Figure 1.11: Power module

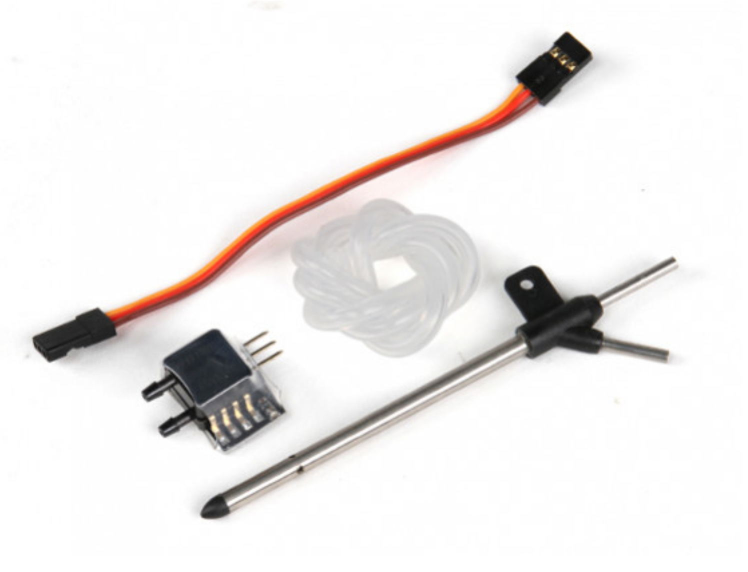

Figure 1.12: The Pitot installed on the Cularis

Figure 1.13: Telemetry modules

In order to obtain an inflight control and real-time monitoring of the parameters affecting the flight and all the information about it, a two-way telemetry is needed. 3DR radios (Figure 1.13) provide an air-to-ground data link between the autopilot and the ground station laptop or tablet. The telemetry works with the 433MHz transmission frequency (which is legal in Europe) and its maximum power output is adjustable up to 100mW.

The installation of the flight-testing equipment altered the longitudinal position of the center of gravity. To recover its correct position, an iron ballast was installed in the tail section. A schematic of the entire equipment carried on-board and its mutual connections is shown in Figure 1.14.

The GPS and a compass are essential for the UAV navigation and course estimation. The GPS is a Ublox NEO M8N, packed together with an extra 3-axis magnetometer, as reported in Figure 1.15. This magnetometer will be used instead of the Pixhawk integrated magnetometer because, being external, it can be placed far from the electronic devices and power cables. In fact, magnetometers are sensors known to be particularly affected by perturbations of the magnetic field in their proximity. In every mission performed, the GPS was used to ensure the automatic return to home in case of a failure of the radio control system.

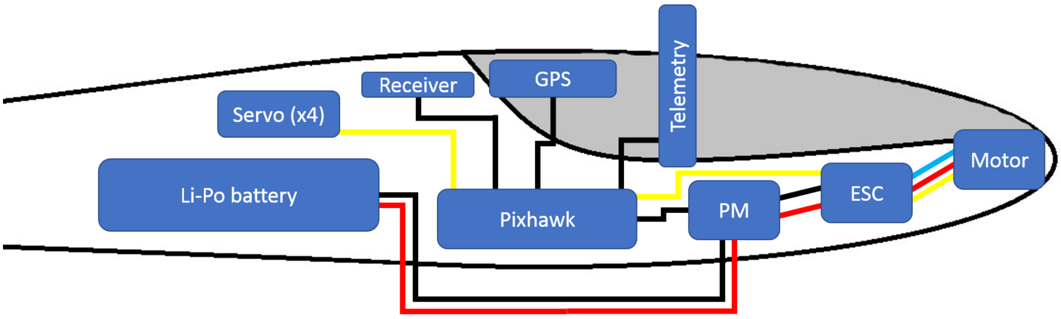

Figure 1.14: Pixhawk wiring diagram

Figure 1.15: The GPS module

## 1.3 Flight-testing safety procedures

The following pre-flight checks are performed before each flight, to ensure safe operations.

- Inspect for damages to the aircraft structure as well as the powerplant status;
- Inspect control links for damages;
- Power up and check the initial calibration and system warmup from the flight planner software;
- Check aerodynamic surfaces movement in manual mode: free and in the right direction;
- Check aerodynamic surfaces movement in FBWA mode: tilt the aircraft and check proper reaction of the aerodynamic surfaces to counteract the attitude variations;
- Close the canopy and reboot the Pixhawk. Since the the magnetometer is linked to the canopy and calibrated with the latter closed, the FCU expects consistent values to help the attitude and heading estimation process;
- Wait for GPS fix;
- Turn off the radio and check for the automatic engagement of the RTL failsafe function, then (if engaged), turn back on the transmitter and check the normal operations of the aircraft. Having the system disarmed at this stage doesn't affect the test;
- Check the transmitter settings for neutral trim on all axis;
- Arm the system and check for the propeller spinning in the right direction.

The takeoff procedure is studied to minimize the risk of an after-takeoff stall and crash. Due to the particular spanwise wing loading, a stall may easily end up in an uncommanded, abrupt roll motion typical of a spin departure. For this reason, during normal operations, each take-off is performed with the following check-list:

- Set the FBWA mode. In this mode, the UAV automatically reacts to wind gusts, particularly dangerous at low airspeed.
- Direct the nose of the aircraft headwind;
- Set the throttle stick to full-power position and maintain the pitch stick neutral, to avoid after-take-off stall. Leaving the pitch stick to neutral the UAV will try to maintain a leveled attitude, a desired condition for building-up airspeed after the hand-launch.

- Once the aircraft is airborne and at an airspeed beyond 12 m/s, gently pull the pitch stick full-backwards to set a constant pitch-up climb with 20° of pitch angle, and adjust the throttle stick as desired.

When the aircraft is airborne, we always let the aircraft accelerate at full throttle until the best-rate-of-climb airspeed of 12 m/s is reached. Then the pitch angle is increased to 20° and the throttle adjusted to keep the best rate of climb airspeed. It's important to notice that the 20° pitch-limit was choosen compatibly with the performance of our UAV. An UAV with a smaller motor may not have the same rate of climb of our model and therefore a different pitch-up limit will be necessary. After having performed the planned mission or when the battery is under the safe level of 40% capacity, the aircraft is prepared for landing. The limit of 40% of the battery is established to have a safe margin for two go-arounds in case of a missed approach. The flight mode is switched back to FBWA and the throttle is closed, gliding back to home and landing safely the aircraft. When stopped, the FCU is disarmed, blocking any throttle input to be sent to the motor, potentially harming the operators.

# Chapter 2

# Model identification

The model identification process is carried out to check the performance of the tuned system in the closed loop configuration and it takes into account only the movement of the three aerodynamic surfaces: ailerons, stabilator and rudder.

In the following sections, a list of the tasks performed to achieve a consistent identified model is shown.

## 2.1 PWM to aerodynamic surfaces deflection mapping

The construction of an identified model consistent with the units of measurement in use in the control loops of the flight control system requires to map the time histories of the control inputs saved on the SD memory (in PWM) with the actual deflection of the control surfaces (in degrees). Since the control loops in ArduPilot handle units such as degrees for angles and degrees per second for angular velocities, the identified model to be simulated in closed-loop must also be consistent with these units of measurement and with the sign convention for angles and rotations shown in Figure 2.5. To perform this transformation, we could use the chord theorem, shown in Equation 2.1 with the nomenclature reported in Figure (2.1).

$$AB = 2Rsin(\alpha) \qquad \alpha = \beta/2 \tag{2.1}$$

In each measurement, being consistent with Figure 2.1, $AB$ stands for the maximum distance between the trailing edge of a control surface while deflected alternatively in both directions, and $R$ is the distance from the trailing edge of the control surface hinge. The procedure for obtaining $AB$ and $R$ is:

- Place a piece of paper firmly stick to a rigid structure of the aircraft, not affected by the control surface movement;

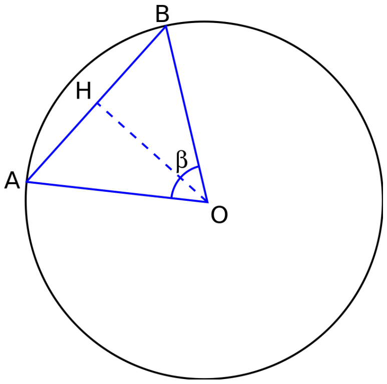

Figure 2.1: Chord theorem nomenclature

- Move the control surface full way in one direction and draw a mark from the hinge to the trailing edge with a pencil;
- Move the control surface full way in the other direction and draw a mark from the hinge to the trailing edge with a pencil as before
- Remove the piece of paper and perform any necessary geometrical computation necessary to determine the hinge position and other relevant data.
- Once an estimate of $AB$ and $R$ is obtained, one can compute the surface deflection beta by using equation (2.1).

For the ailerons deflection, as reported in Figure 2.2 we obtained the data collected in Table 2.1.

| | |
|---|---|
| AB | 28 mm |
| R | 45 mm |
| $\alpha$ | 18.1° |
| $\beta$ | 36.2° |

Table 2.1: Aileron deflection data

For the stabilator we obtained, as reported in Figure 2.3, the data collected in Table 2.2.

| | |
|---|---|
| AB | 21 mm |
| R | 70 mm |
| $\alpha$ | 8.6° |
| $\beta$ | 17.2° |

Table 2.2: Stabilator deflection data

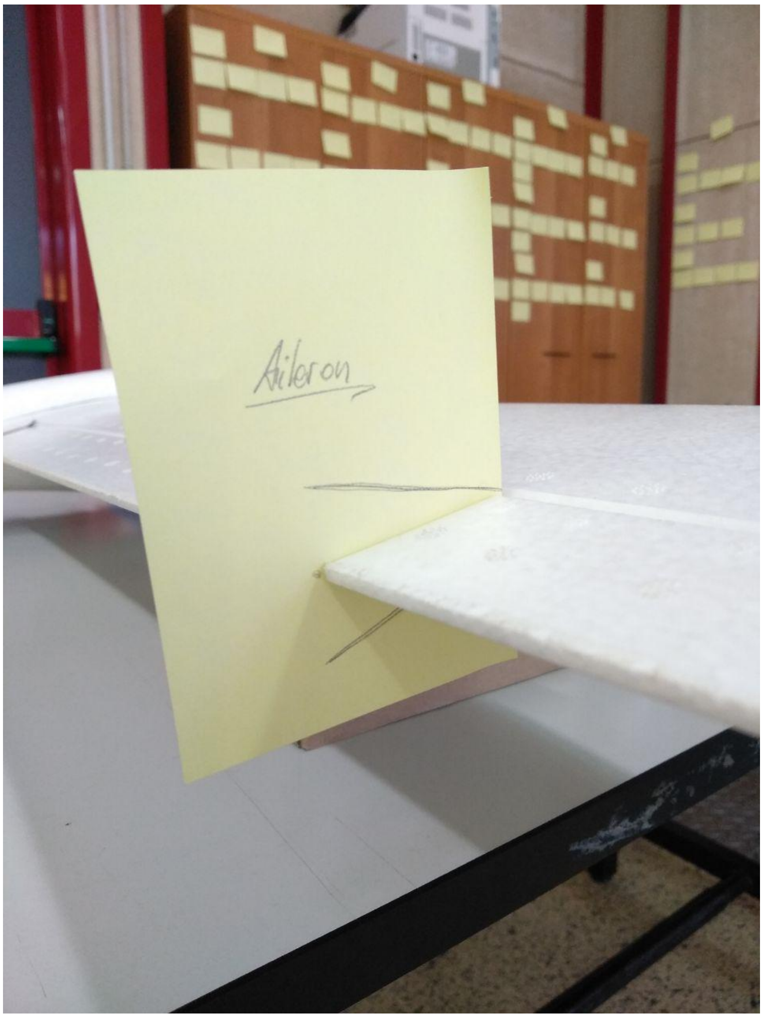

Figure 2.2: Aileron maximum deflection test

| | |
|---|---|
| AB | 81 mm |
| R | 78 mm |
| $\alpha$ | 31.3° |
| $\beta$ | 62.6° |

Table 2.3: Rudder deflection data

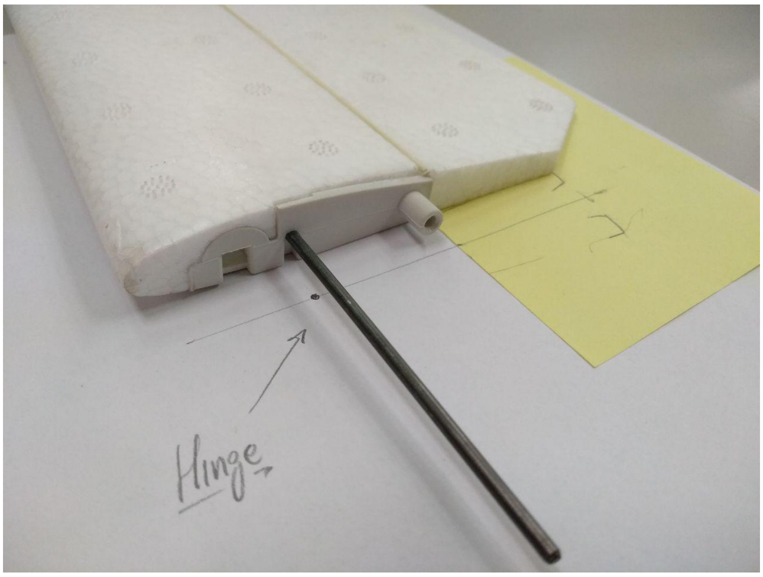

Figure 2.3: Stabilator maximum deflection test

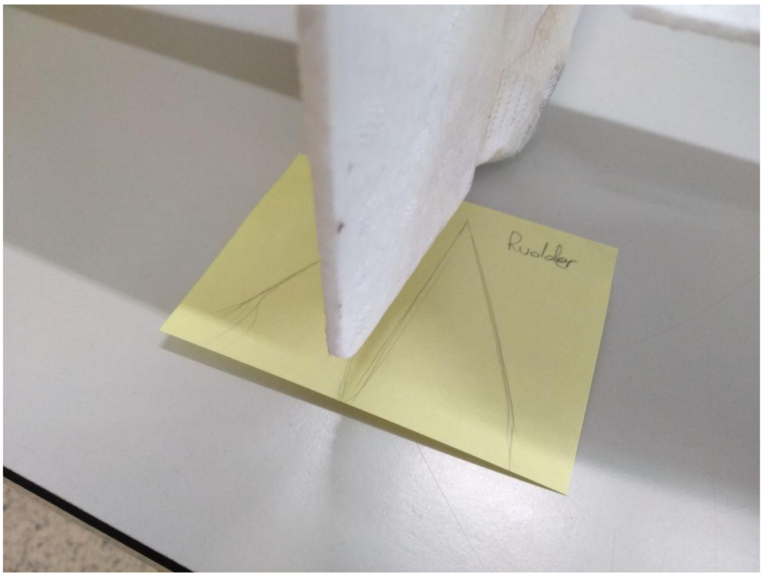

Figure 2.4: Rudder maximum deflection test

For the rudder, as reported in Figure 2.4, we obtained the data collected in Table 2.3.

The PWM values used to control the servomotors have a range from a minimum of 1000 units to a maximum of 2000 units. Normally a non-deflected control surface has a value in PWM units equal to 1500 units. Having the following measurements available, it is possible to build a map to correlate the deflection in degrees of the control surface with the PWM value sent by the flight control system to the servos.

## 2.2 Model identification approach

Identification tests are carried out at an indicated airspeed of 14 m/s, achieved with the motor running at 50% of the maximum power available. It is believed that this test airspeed is a good trade-off between testing the aircraft far from stall (which occurs at about 7 m/s IAS) and not too high to require excessive power which would excessively drain the battery. This airspeed is also very close to the minimum power required airspeed for level flight, but it is easier to keep for operational reasons. Having taken confidence with the machine, on-field testing had shown that an excitation frequency ranging from 0.5 Hz to 1.5 Hz is optimal to obtain a significant response. Below this range of frequencies, a significant and undesired variation in airspeed is obtained (during the longitudinal tests). For higher frequencies, on the other hand, it is difficult to manually perform the excitation.

This thesis uses the sign reference reported in Figure 2.5.

A total of four models representing the dynamics of the aircraft were identified. Both for the longitudinal channel and for the lateral-directional channel, continuous time was adopted. We're trying to estimate the coefficients of the matrices $A, B, C, D$ of the dynamic model reported in equation (2.2).

$$
\begin{aligned}
\dot{x} &= Ax + Bu \\
y &= Cx + Du
\end{aligned}
\tag{2.2}
$$

The choice to start the identification process from a grey box model has been carried out to verify the possibility of inserting the many known parameters (such as kinematic links) and characteristics of the aircraft, thus reducing the quantity of parameters to be identified. The defined state matrices coefficients of both the longitudinal and the lateral-directional channel are reported in equation 2.3 and equation 2.4 respectively, with the $free$ mark standing out where the coefficients are freely manageable by the identification algorithm.

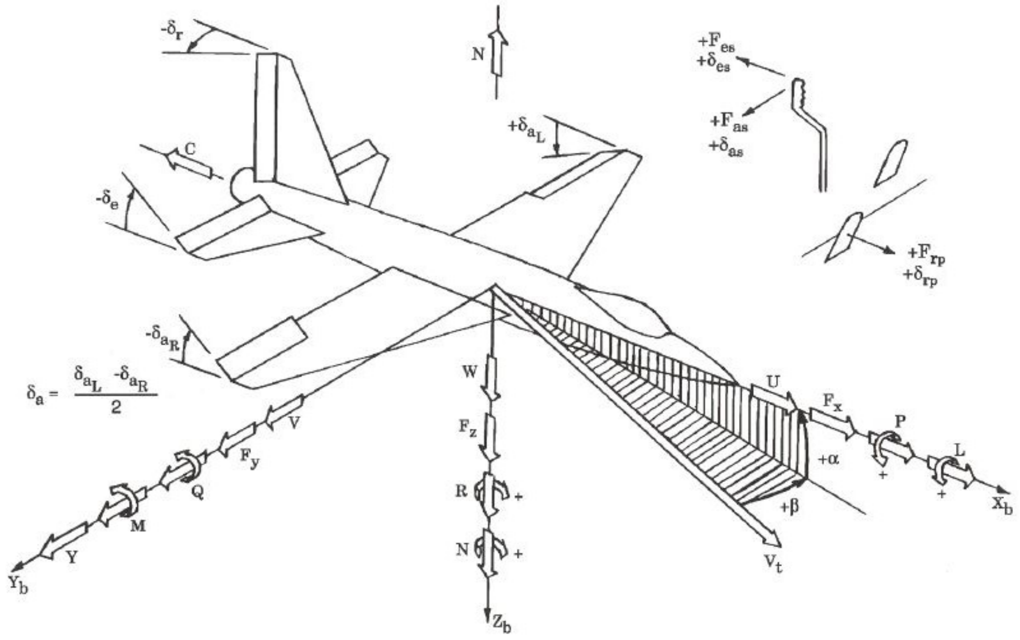

Figure 2.5: Sign convention

$$A_{longitudinal} = \begin{bmatrix} free & free & 0 & free \\ free & free & 0 & 0 \\ free & free & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.3}$$

$$A_{lateral-directional} = \begin{bmatrix} free & 0 & -1 & free & 0 \\ free & free & 0 & 0 & 0 \\ free & free & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{2.4}$$

$$x_{lon} = [u, \alpha, q, \theta]^T \tag{2.5}$$

$$x_{latdir} = [\beta, p, r, \phi, \psi]^T \tag{2.6}$$

Since an initial guess is necessary to start the grey box identification process, the estimation of the initial parameters of the state matrix was initially tackled by means of theoretical calculation, estimating the aerodynamic derivatives and inertia to be included in the flight dynamics equations. The aerodynamic derivatives in fact, for conventional aircraft, generally have orders of magnitude and sign comparable even between different aircraft. These parameters have been implemented relying on the robustness and convexity of the problem, aware of the fact that the derivatives we had were coming from general aviation aircraft or even large transport ones, literally different classes of aircraft. Found the absence of convergence, it was necessary to choose a different route that considered different aircraft dimensions.

$$A_{747_{Longitudinal}} = \begin{bmatrix} -0.0188 & 0.042 & 0 & -0.115 \\ -0.180 & -0.519 & 0.947 & 0 \\ -0.017 & -0.495 & -0.489 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.7}$$

$$A_{Navion_{Longitudinal}} = \begin{bmatrix} -0.045 & 0.036 & 0 & -0.182 \\ -0.370 & -2.026 & 0.972 & 0 \\ 0.337 & -6.985 & -2.966 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.8}$$

$$A_{Cularis_{Longitudinal}} = \begin{bmatrix} -0.5 & 0.01 & 0 & -0.25 \\ -1 & -10 & 1 & 0 \\ 5 & -100 & -10 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.9}$$

$$A_{747_{Latdir}} = \begin{bmatrix} -0.088 & 0 & -1 & 0.115 & 0 \\ -1.195 & -0.869 & 0.183 & 0 & 0 \\ 0.277 & -0.102 & -0.207 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{2.10}$$

$$A_{Navion_{Latdir}} = \begin{bmatrix} -0.250 & 0 & -1 & 0.182 & 0 \\ -16.05 & -8.417 & 2.197 & 0 & 0 \\ 4.571 & -0.350 & -0.761 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{2.11}$$

$$A_{Cularis_{Latdir}} = \begin{bmatrix} -0.5 & 0 & -1 & 0.1 & 0 \\ -15 & -10 & 1 & 0 & 0 \\ 15 & -1 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{2.12}$$

Considering, as a first approximation, that the Navion is an aircraft of an order of magnitude smaller than the B747 (just compare the wingspan) and the Cularis is in turn an order of magnitude smaller than the Navion, the idea of a quasi-linear extrapolation of the coefficients of both longitudinal state matrix 2.9 and lateral-directional state matrix 2.12 was built.

This procedure of re-scaling the state matrix coefficients according to the aircraft size in a quasi-linear fashion, led the 2.9 and 2.12 to be the initial state matrices to start the grey box estimation process. Moreover, both for the longitudinal as for the lateral-directional channel, the initial-guess input matrices were assumed to be full of free, unitary coefficients.

Once the coefficients of matrix A have been estimated through the Grey box algorithm, a check was performed on the order of magnitude of the obtained aerodynamic derivatives, finding however values not comparable with the theory. This anomaly is mainly due to the disturbances collected in the data acquisition process. The aerodynamic derivatives found may not necessarily have a physical meaning since in the identification process not only the dynamics of the aircraft are included but also disturbances of various nature, such as gusts and measurement errors of the on-board electronics. It follows that the quality of the identification process is to be judged only by the correspondence of the identified model with a data set spared by the identification process and used to validate the model. The initial conditions of the tests were not always equal to zero: due to the difficulties in evaluating the dynamic behavior of the aircraft from the ground before starting the injection of the input, the initial conditions typically have values different than zero. The identification process takes into account these initial conditions.

The identification of the black box was then carried out to assess the increase in accuracy with respect to the grey box, highlighting slight improvements. Significant improvements were further obtained by excluding the estimate of attitude angles and the aerodynamic angles from the identification process. The attitude angles obtained with the black box estimate in fact were not representative of the actual kinematic link with the related angular velocities, making the identification process uniquely more complex and worsening the quality of the identified model. The estimate of the aerodynamic angles instead, were of good quality but they would not be used in the following control law design. Due to this reason, they were discarded from the estimation process too. Moreover, looking at the free parameters in equation (2.4) and equation (2.3), eliminating the rows and the columns corresponding to the quantities excluded from the identification process, the Grey box model reduces to a form that is equivalent to the Black box model. The hypothesis of rigid rotations is then introduced, and the attitude angles will consequently be calculated by integrating the angular velocities in the identified model.

## 2.3 Identification of the longitudinal dynamics

To identify the open-loop aircraft dynamics required to build the aircraft model, several tests were performed to collect data time series. Based on the flight dynamics knowledge, two different excitation inputs have been used to evaluate the aircraft's longitudinal behavior: a relatively fast doublet input to evaluate the short period mode and a relatively slow doublet input to evaluate the phugoid mode. Of course, one could consider many types of inputs, but we choose the doublet because of its simplicity and ease of reproduction among different test campaigns. A MATLAB script ringing a pre-defined tone indicates, by acoustical means, the rhythm at which the doublet should be performed. All the tests were performed at 14 m/s of indicated airspeed, with doublet inputs having a frequency of 1 Hz ± 0.5 Hz. This excitation was the only we were able to perform manually since lower frequencies would mean too large height and airspeed variations. The following results are for the black box case only. In Figure 2.6, an example of open-loop longitudinal channel excitation test is presented. As one can see, a doublet input is applied to the stabilator. Since the elevator movement is positive for a downward motion of the trailing edge, a consequent pitch-down rotation is commanded and the indicated airspeed increases. Throttle was held constant during each test.

The whole amount of input and output time histories, like the one in Figure 2.6, were used to feed the Black box estimation process. As a result, the values reported in equation 2.15, with 2.13 as state vector and 2.14 as input vector, were found.

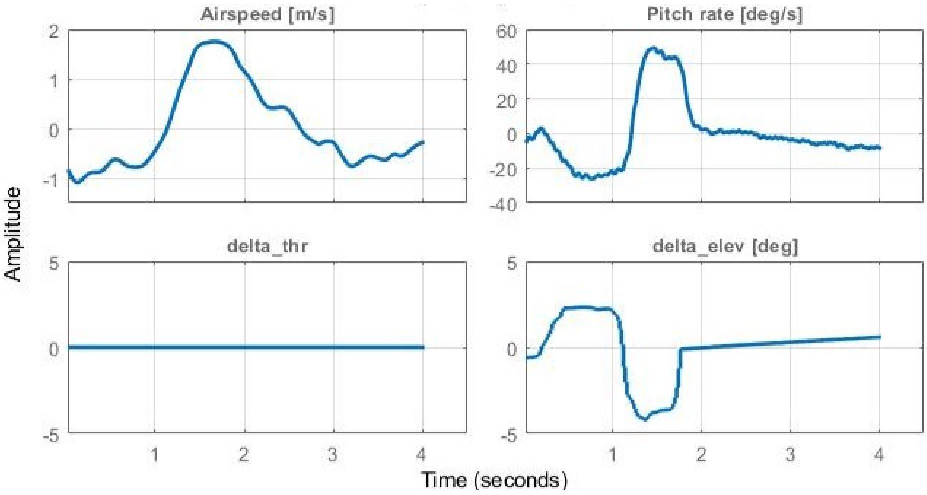

Figure 2.6: Longitudinal open-loop excitation experiment

$$x_{lon} = [u, q]^T \tag{2.13}$$

$$u_{lon} = [\delta_{throttle}, \delta_{stabilator}]^T \tag{2.14}$$

$$\begin{aligned}\dot{x}_{lon} &= \begin{bmatrix} -37.10 & 1.35 \\ -645.8 & 19.43 \end{bmatrix} x_{lon} + \begin{vmatrix} -69.02 & 0.856 \\ -1602 & -33.65 \end{vmatrix} u_{lon} \\ y_{lon} &= \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix} x_{lon} + \begin{vmatrix} 0 & 0 \\ 0 & 0 \end{vmatrix} u_{lon} \end{aligned} \tag{2.15}$$

Since the identification maneuvers performed on the longitudinal channel were mainly exciting the short period instead of the phugoid, results coming from a simulated response using a data-set like the one in Figure 2.6, are of the kind of Figure 2.7. In fact, it is known that short period excitation on the longitudinal channel has inherently low airspeed variations, leading to a poor estimate of the airspeed in our model. In contrast, the pitch rate estimate was acceptable. Since no tests were performed testing various inputs on the throttle stick, the contribution of this input channel is unacceptable, and the throttle step responses depicted in Figure 2.8 must be discarded.

| Pole | Damping | Frequency $[rad/s]$ | Time constant $[s]$ |
|---|---|---|---|
| -8.33+8.66i | 0.714 | 12.4 | 0.113 |
| -8.33-8.66i | 0.714 | 12.4 | 0.113 |

Table 2.4: Longitudinal dynamics identified poles

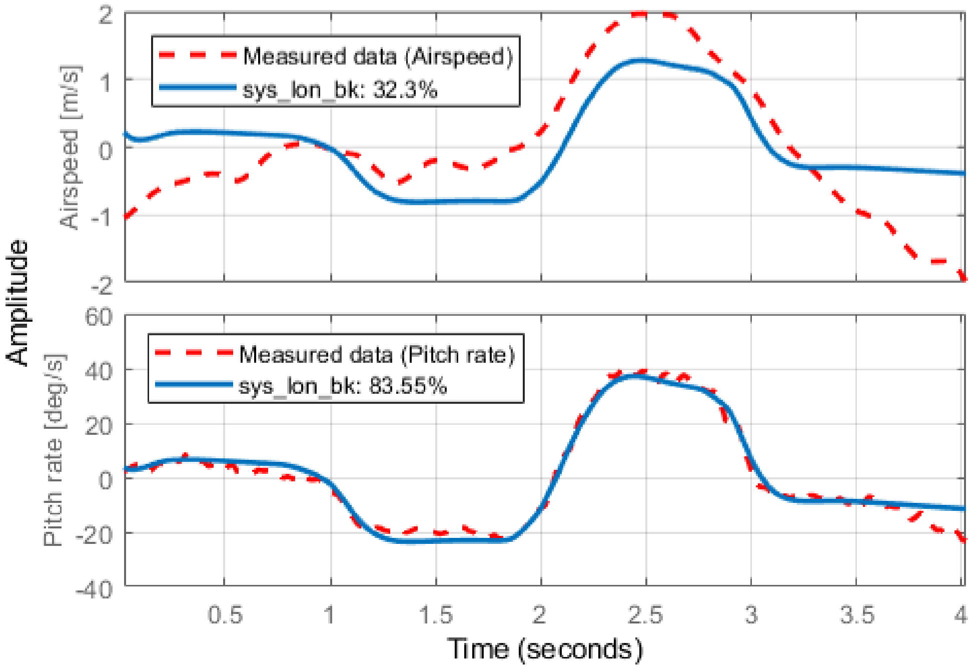

Figure 2.7: Longitudinal identified model simulated response

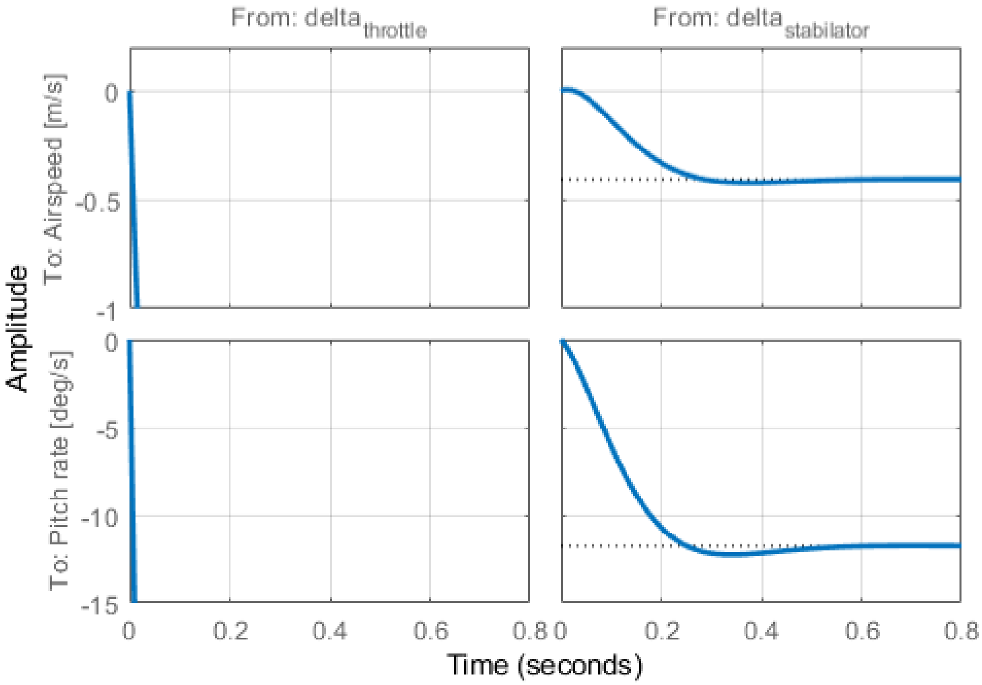

Figure 2.8: Longitudinal model step response

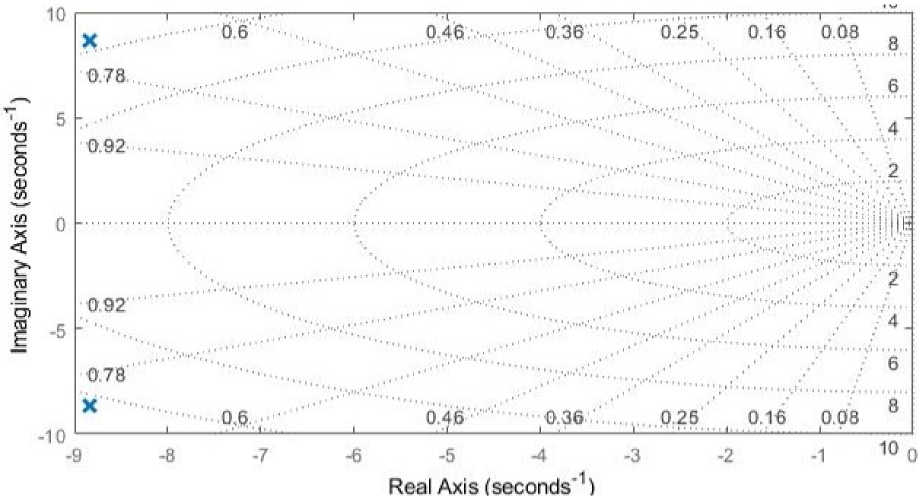

Figure 2.9: Longitudinal dynamics pole-zero map

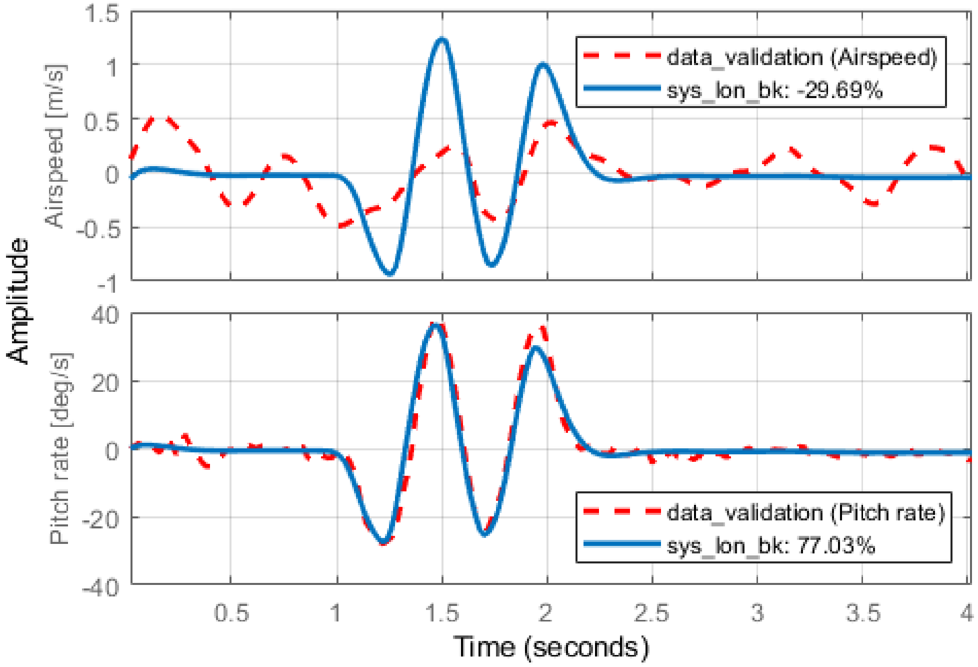

Figure 2.10: Longitudinal model validation test

To validate the identified model, a data-set was kept outside the estimation process to test the obtained dynamical system and check its representativeness to new data. As in Figure 2.10, the adherence of the estimated airspeed to the measured one is poor for the reasons previously described, but the pitch rate estimate is satisfactory for proceeding to further steps.

## 2.4 Identification of the lateral-directional dynamics

To identify the open-loop aircraft dynamics required to build the aircraft model, several tests were performed, collecting a number of data time series. Among the excitation inputs, also in this case we have chosen the doublet because of its simplicity and ease of reproduction. A MATLAB script ringing a pre-defined tone indicates, by acoustical means, the rhythm at which the doublet should be performed. All the tests were performed at 14 m/s of indicated airspeed, with doublet inputs having a frequency of 1 Hz ± 0.5 Hz. This excitation was chosen because it would be useless to have a lateral-directional model able to manage lower-frequency maneuvers than whose managed by the longitudinal model while both are flying on the same aircraft. Moreover, as shown in the last paragraph, performing manual excitation at different frequencies with the hardware we had was too skill-demanding to have useful data. The following results are for the black box case only. In Figures 2.11 and 2.12, an example of open-loop lateral-directional channel excitation test is presented. As one can see, in Figure 2.11 a doublet input is applied to the ailerons. The rudder was not moved during this test. In conjunction with a series of aileron doublet input, an equal number of rudder doublet inputs were performed.

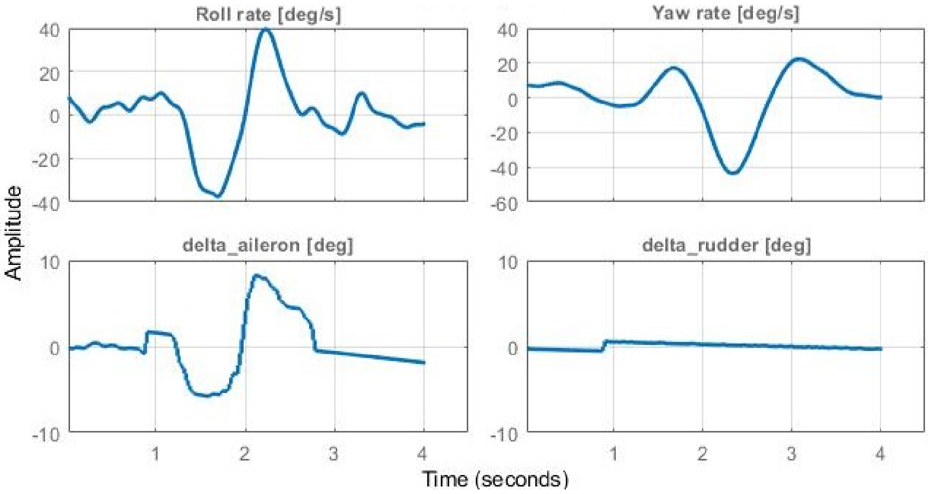

Figure 2.11: Lateral-directional open-loop excitation (roll experiment)

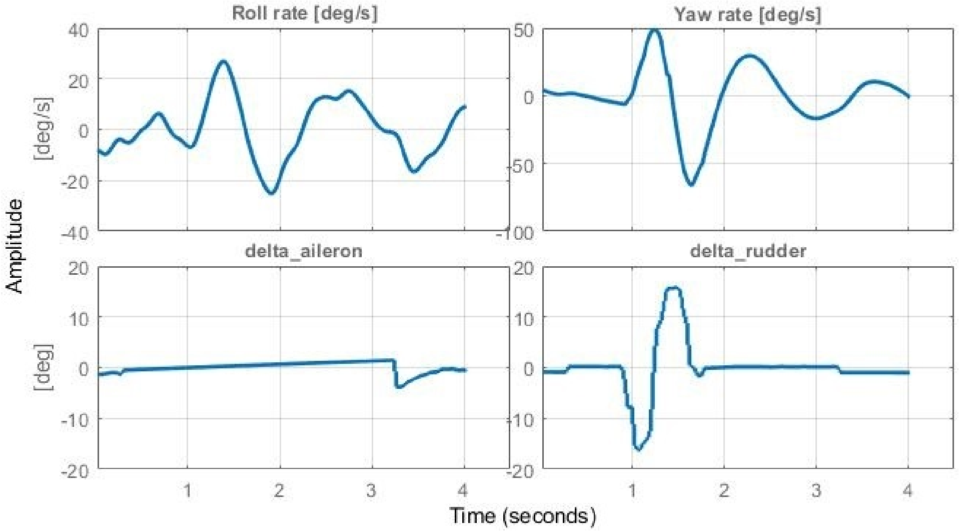

Figure 2.12: Lateral-directional open-loop excitation (yaw experiment)

The input and output time histories, related to the lateral-directional channel, were used to feed the black box estimation process. As a result, the values reported in equation 2.18 were found.

$$x_{latdir} = [p, r]^T \tag{2.16}$$

$$u_{latdir} = [\delta_{aileron}, \delta_{rudder}]^T \tag{2.17}$$

$$\begin{gathered}\dot{x}_{latdir} = \begin{bmatrix} -3.51 & 10.22 \\ -2.35 & -6.58 \end{bmatrix} x_{latdir} + \begin{vmatrix} 44.52 & 16.56 \\ -12.17 & -22.73 \end{vmatrix} u_{latdir} \\ y_{latdir} = \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix} x_{latdir} + \begin{vmatrix} 0 & 0 \\ 0 & 0 \end{vmatrix} u_{latdir}\end{gathered} \tag{2.18}$$

To validate the identified model, a data set was kept outside the estimation process to test the identified dynamical system and check its representativeness to new data. As shown in Figure 2.16, we obtained a model with a good representation of the roll behavior but in some way lacking on the yaw axis. A possible source of this poor fit could be addressed to the good rudder effectiveness: in fact, since the amplitude of the doublet inputs applied to the rudder were comparable with the amplitude of the doublet inputs applied to all the other surfaces, the aircraft's excessive oscillation may have reached a region of its dynamic behavior where the aerodynamic phenomena involved are no more linear. Due to this issue, the linear identified model lost fidelity while representing those excessive rudder inputs.

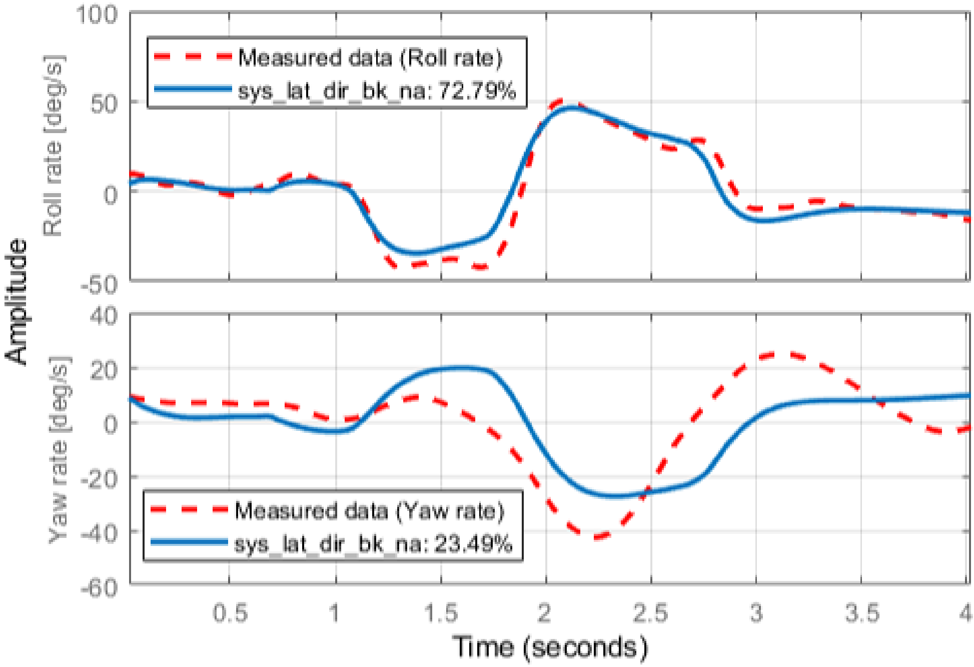

Figure 2.13: Lateral-directional identified model simulated response

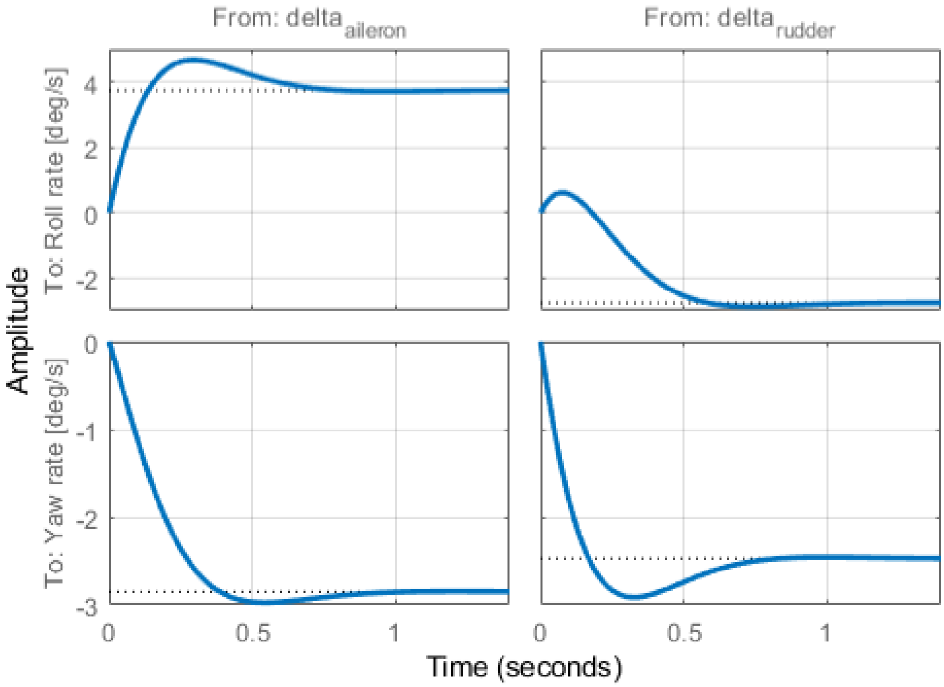

Figure 2.14: Lateral-directional model step response

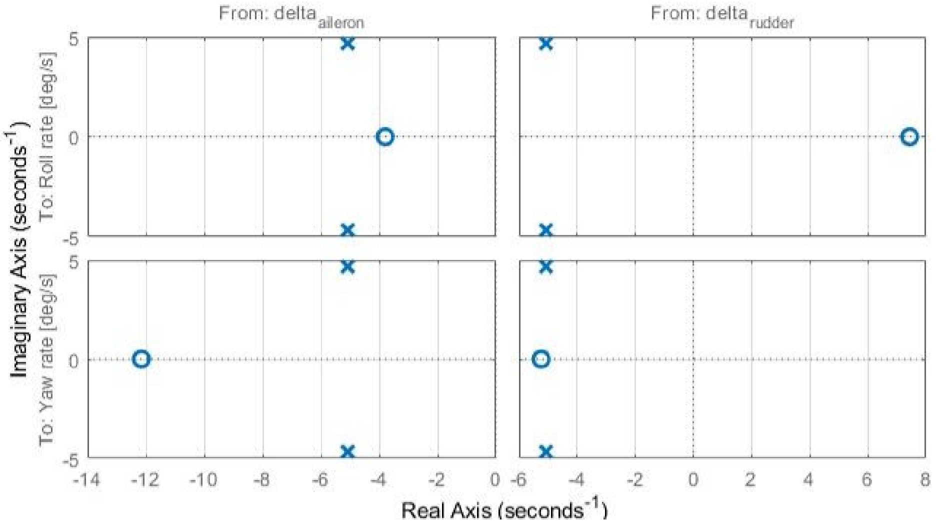

Figure 2.15: Lateral-directional dynamics pole-zero map

This lack should be further investigated by checking how a smaller-amplitude doublet input on the rudder could increase the fit of the estimated model. Particular attention should be hereby paid to the signal-to-noise ratio due to the sensitivity this aircraft has for any kind of atmospheric phenomena.

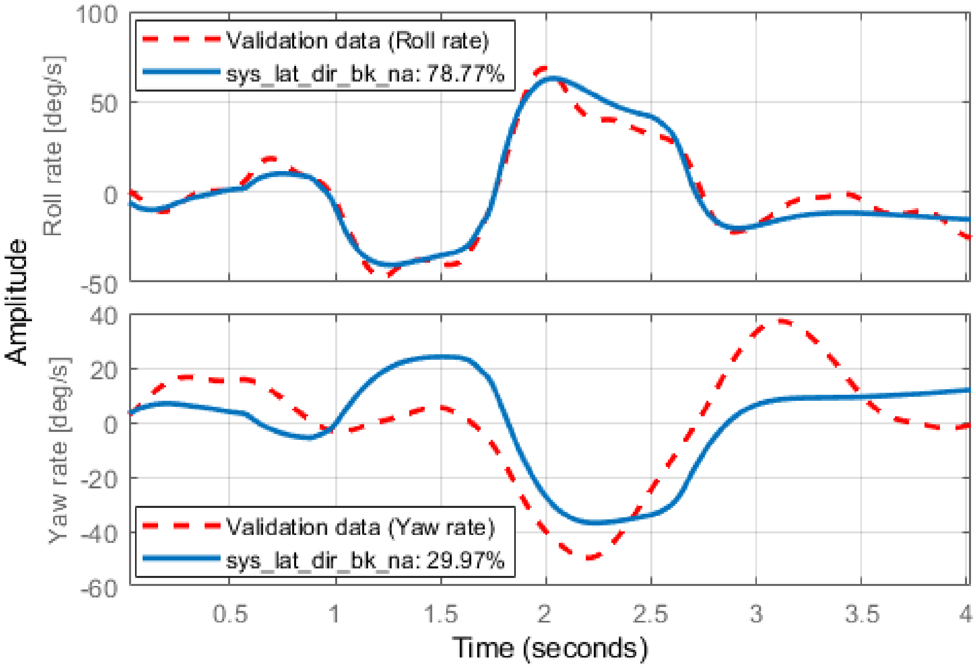

Figure 2.16: Lateral-directional model validation test

# Chapter 3

# Control law design

In this chapter, a detailed description of the techniques used to compute the controller gains, both for the longitudinal and for the lateral-directional channel, is presented.

Data-driven tuning methods have been formulated to overcome limitations of model-based controller synthesis, avoiding the definition of a system model. Other names by which these methods are known include model-free and data-based, highlighting the fact that only experimental data is used.

VRFT provides a one-shot solution. This feature makes VRFT extremely interesting for fast re-tuning of controllers, which could potentially be even carried out online.

## 3.1 Model-reference design

Model-following or model-reference control is one of the possible approaches to define the desired behavior of a closed loop system.

In model reference, the desired closed-loop behavior is expressed as a dynamical system, either in state-space form or as a matrix transfer function. It is generally easy to express the requirements on the output complementary sensitivity. The model-reference control problem can now be formalized giving the system plant $G(z)$, the reference model $M(z)$, a frequency weighting factor $W(z)$ and a controller $C(z)$. Solve for $\hat{C}(z)$, with:

$$\begin{gathered} J_{MR} = \left\| W(z)[M(z) - (I + G(z)C(z))^{-1}G(z)C(z)] \right\|_2^2 \\ \hat{C}(z) = \underset{C}{\operatorname{argmin}}\, J_{MR} \end{gathered} \tag{3.1}$$

A possible solution of this problem can be obtained directly if the controller is not constrained, resulting in:

$$\hat{C}(z) = [(I - M(z))G(z)]^{-1}M(z) \tag{3.2}$$

This solution might not be feasible to implement in real world applications, due to its potentially high order, presence of high frequency dynamics, and typically high control effort. Stability properties are also an issue, and finally, the relative degree of the transfer function, that is the difference between the number of poles and zeros, can be such that the system is not proper. Model reduction of the obtained controller or system might not be a solution either, due to the impossibility to guarantee the same performance and stability properties. Thus, in most circumstances, the structure of the controller is fixed, and only its parameters can be varied, leading to a weak formulation of the optimization problem. This formulation is also referred to as a structured, as opposed to the unstructured one presented in equation (3.1). The structured problem can be formalized as follows:

$$\begin{gathered} J_{MR}(\theta) = \left\| W(z)[M(z) - (I + G(z)C(z,\theta))^{-1}G(z)C(z,\theta)] \right\|_2^2 \qquad (3.3) \\ \hat{\theta} = \underset{\theta}{\operatorname{argmin}}\, J_{MR}(\theta) \end{gathered}$$

There are some remarks to be made regarding the solution of this problem. Firstly, in this criterion no stability requirement is imposed, only the minimization of the discrepancy between the reference model and the output complementary sensitivity. Secondly, this generally represent a non-convex optimization problem, which means that a global minimum might not be achieved as several local minim can be present. A convex approximation of the problem can be achieved by making the following assumptions:

1. The sensitivity function $S(z) = I - M(z)$ is close to the closed-loop sensitivity function for $\theta = \hat{\theta}$.

2. The controller family $C(z, \theta)$ can be linearly parameterized with the vectors of parameters $\theta$ such that $\mathscr{C}(\theta) = C(z, \theta) = \beta^T(z)\theta$

By replacing the above assumption in the equation (3.1), one obtains:

$$J_{MR}(\theta) = \left\| M(z) - (I - M(z))\beta(z)^T\theta G(z) \right\|_2^2 \qquad (3.4)$$

Note that depending on the specific application, the chosen controller structure might lead to an error as low as zero. However, it is usually greater than zero for structured problems. The assumptions that have been made should be checked a posteriori. The first assumption, for instance, states that the controlled system could be made arbitrarily close the desired model behavior, which might not be the case for the structured case.

## 3.2 The Virtual Reference Feedback Tuning approach

The VRFT method allows the tuning of structured controllers where limited information about the plant is available; not requiring any identification tests, it is a versatile method and, for its execution, even a single set of data (input and output) is sufficient.

Some prior knowledge is required for the definition of a reasonable reference model which must be compatible with the achievable system dynamics. VRFT is named after the signal which is exploited to define the model reference cost function to be minimized, the virtual reference. This signal is recreated from output data as the equivalent set point applied to a virtual closed-loop system that would be required to produce such output for the given reference model, as shown in Figure 3.1.

The idea can be summarized in the following steps:

- generate a set of I/O data by a single experiment on the plant.
- given the measured $y(t)$, generate in your computer a reference signal $r(t)$ such that $M(z)r(t) = y(t)$, where $M(z)$ is the desired reference model. $r(t)$ is called "virtual reference" because it was not used to generate $y(t)$ and it only exists as a computer file.
- compute the corresponding tracking error $e(t) = r(t) - y(t)$.
- even though plant $P(z)$ is not known, we know that when $P(z)$ is fed by $u(t)$ (the actually measured input signal), it generates $y(t)$ as an output. Therefore, a good controller is one that generates $u(t)$ when fed by $e(t)$. The idea is then to search for such a controller. Since both signals $u(t)$ and $e(t)$ are known, this task boils down to the problem of identifying the dynamical relationship between $e(t)$ and $u(t)$.

This procedure, however, has a drawback: the choice of the desired reference model. To overcome this limitation, a new procedure for the choice of a suitable reference model has been designed. The values of damping and natural frequency of the reference model have been searched using a global optimization algorithm.

To start this procedure, the following parameters must be given:

- a bounded domain of parameters such as natural frequency and damping for each control loop, like the ones in Table 3.1 and Table 3.4. The wideness of the range of the parameters should be carefully extended to avoid excessive computational cost if the mesh of the initial points is increased accordingly;
- the number of starting points to start the evaluation of the cost function, evenly distributed on the parameter domain, trying different combinations

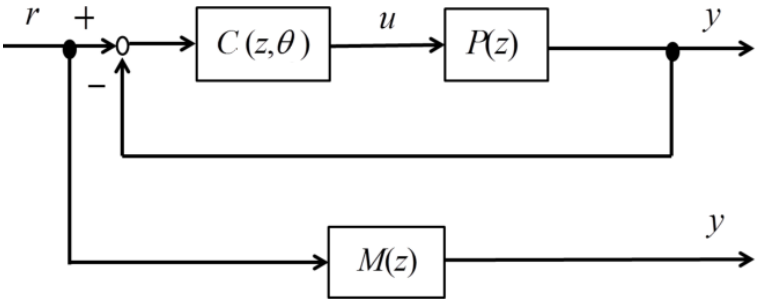

Figure 3.1: VRFT principle

> of parameters. The number of starting points should be large enough to ensure a fine mesh on the domain but at the same time this number should not be exaggerated to avoid excessive computational time and, eventually, non-physical cusp-shaped local minim.

Of course, it is not guaranteed that the found local minimum will be the global one but, since that result is coming from a reasonable mesh of starting points, that local minimum should be robust enough for real applications because the probability that a starting point was in the neighborhood of a cusp minimum of the cost function should be improbable.

Even in the case of the availability of unlimited computational power and time, very narrow local minimums should be avoided because they are typically faulty in robustness. Having chosen the range of the domain and the number of starting points, a global minimum should be found, inherently discarding non-physical solutions. In our case, the domains reported in Tables 3.1 and Table 3.4 have been explored using 1000 starting points. The cost function to be minimized is based on the mean square error between the tuned closed-loop system and the performance constraints that will be discussed in detail separately for the longitudinal and lateral-directional channel. In both longitudinal and lateral-directional cases, the Matlab “fmincon” algorithm was used. This procedure is no more only data-driven but uses the identified model to evaluate the quality of the closed-loop response tuned with different parameters.

The batch of tests used to run the VRFT method are, as shown in section 2.2 a series of doublets on the longitudinal and the lateral-directional channel as shown in Figure 2.6 and Figure 2.11. However, it must be stressed out that data used for the VRFT tuning respect from the data used for the model identification are coming from different test campaigns.

## 3.3 Longitudinal dynamics

The ArduPilot pitch control scheme consists of two cascade controllers. The inner controller is a PI controller with a feed-forward; the outer controller is a proportional gain which is in charge of transforming the pitch angle set-point error into an angular velocity set pointthat directly feeds the inner-loop feed-forward term. In Figure 3.2, nomenclature is not straightforward. The control structure we have today is the result of some recent modification carried out by the ArduPilot developers: the new update was meant to be rolled out inside a firmware upgrade.

To operate the conversion, new parameters reported in equation (3.5) need to be introduced.

$$\begin{aligned} \Omega &= \frac{1}{\tau} \\ \bar{K}_P &= (K_P - K_I\tau)\tau - K_D \\ \bar{K}_I &= K_I\tau, \end{aligned} \tag{3.5}$$

where $\tau$ is a new tuning parameter (which can be seen as the time constant in seconds from demanded to achieved bank angle). In its present configuration the scheme is shown in Figure 3.2. Thanks to this structure, the qualitative behavior of the control is unchanged with respect to the old control law and keeps intact the old users gains. Moreover the pure derivative actions of the old PID disappears so that there are no spikes in servo demands. There is also an optional feed-forward action with gain $K_{FF}$ that is not going to be used here and therefore is set to zero. Lastly a scaling is introduced to moderate the ailerons control action according to the airspeed in such a way that at high speed the surfaces are less controllable. Once the actual airspeed $V_a$ and the scaler airspeed $V_{scaling}$ are known, the scaling equation is:

$$scaler = \frac{V_{scaling}}{V}. \tag{3.6}$$

It is worth to notice that the associated term $K_D$ in the old controller structure is now a proportional term in the inner loop controller. The feed forward gain, parallel of the proportional term, is discarded to avoid confusion, although redundant in our setting. To explore this transition in a more detailed way, a map of this gain transformation was given in equation (3.5).

The pitch control scheme reported in Figure 3.2 has been simplified to Figure 3.3, being $C_\theta(z)$ the parallel between the $K_D$ gain and the integral term of the inner loop. In relation to the limitations of the identified model further simplifications are made:

- The turn coordination offset is not considered as the identified model does not support extended maneuvers over time (as a turn) but only the dynamics in the short period.

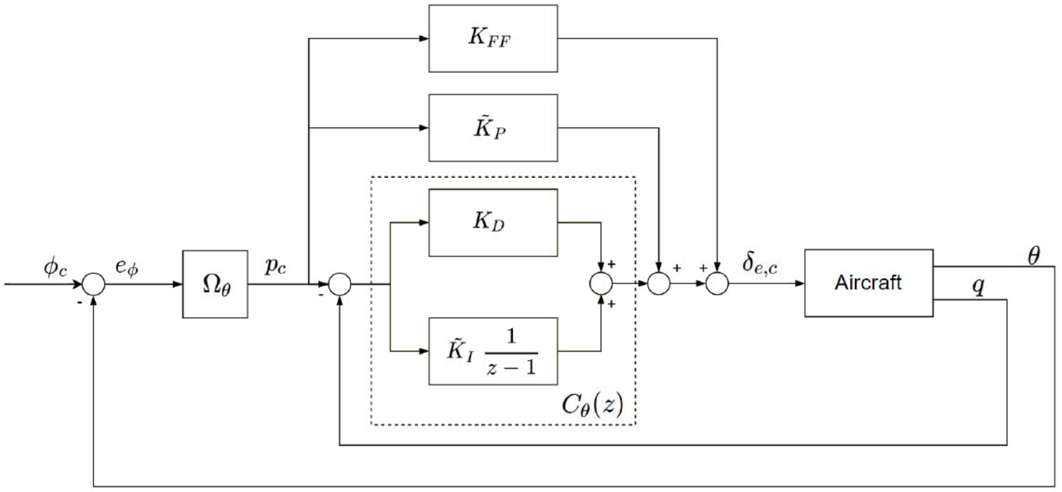

Figure 3.2: Inner and outer pitch loop control scheme nomenclature

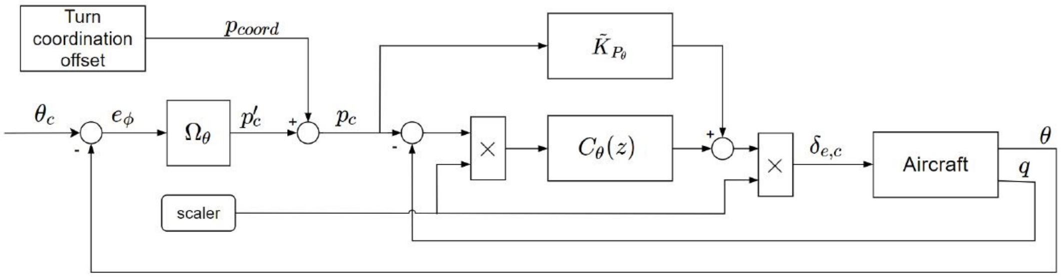

Figure 3.3: General ArduPilot pitch control scheme

- The scaler on the deflection of the stabilator is assumed to be equal to 1. To make this assumption valid, each test must be performed at an airspeed equal to the scaling airspeed, that is $14 m/s$.

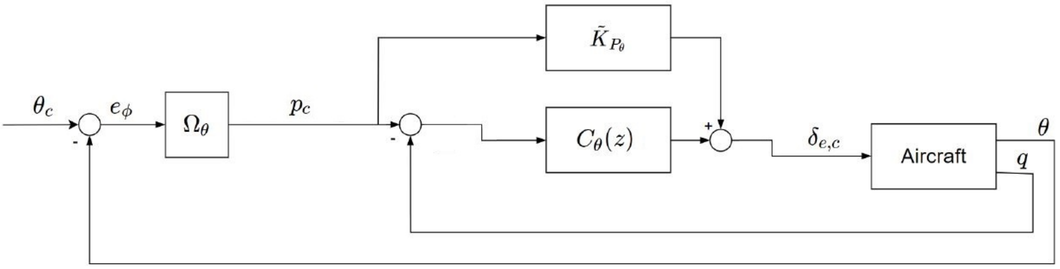

Figure 3.4: Simplified ArduPilot pitch control scheme

When the structure of the control loops and the ranges of the reference model's coefficients involved are defined, it is possible to start the search for the global minimum, guessing the reference model's parameters and checking, via a cost function, the compliance of the guess-tuned closed-loop system with the following reference criteria:

- The mean square error between the output of the guess-tuned, closed-loop system ($y_{lon}$) and a signal held constant to 1 ($y_{ref}$) both included in the partial cost equation (3.7). This cost term, if minimized alone, is devoted to search for a combination of reference model parameters that minimizes the time needed to achieve the set point, not dealing with overshoot and/or oscillations;

- A cost term devoted to the elimination of the oscillations of the closed-loop, guess-tuned, output of the external loop ($y_{lon}$), included in the partial cost equation ((3.8)). This term is proportional to the sum of the difference of two subsequent outer-loop output values, along the whole guess-tuned model output vector. This term would activate if the difference of two consecutive terms in the analyzed vector is negative. Being the result of this operation negative, it means that an oscillation phenomena is happening and it must be penalized in the cost function.

The cost function $C_{lon}$, in Matlab notation, is reported in equation (3.9) and the coefficient $w_1$ is a weight term that penalizes the oscillations, ruled according the (3.10). The diff() operator reported in equation (3.8) and equation (3.10) returns a vector containing the difference between two subsequent vector elements.

$$C_{1_{lon}} = \sqrt{mean((y_{lon} - y_{ref}).^2)} \tag{3.7}$$

| $\omega_i$ [rad/s] | | $\zeta_i$ | | $\omega_o$ [rad/s] | | $\zeta_o$ | |
|---|---|---|---|---|---|---|---|
| Min | Max | Min | Max | Min | Max | Min | Max |
| 0.1 | 300 | 0.4 | 2 | 0.1 | 300 | 0.4 | 2 |

Table 3.1: Longitudinal channel reference models parameters domain

| $\omega_i$ [rad/s] | $\zeta_i$ | $\omega_o$ [rad/s] | $\zeta_o$ |
|---|---|---|---|
| 226.4 | 1.48 | 139.1 | 0.98 |

Table 3.2: Longitudinal reference models tuned parameters

$$C_{2_{lon}} = w_1 |(min(\text{diff}(y_{lon}))| \tag{3.8}$$

$$C_{lon} = \sqrt{mean((y_{lon} - y_{ref}).^2)} + w_1 |(min(\text{diff}(y_{lon}))| \tag{3.9}$$

$$w_1 = \begin{cases} 1 & \text{if}\min(\text{diff}(y_{lon})) < 0 \\ 0 & \text{if}\min(\text{diff}(y_{lon})) \geq 0. \end{cases} \tag{3.10}$$

By suitably weighting the $w_1$ term, and according to the particular needs of the user and the characteristics of the system, a cost function to be minimized is obtained. The upper and lower bounds of the parameters usable by the minimization algorithm are reported in Table 3.1.

Through this tuning technique, the closed-loop performance of the longitudinal channel is presented in Figure 3.5, highlighting significant improvements in the achievement of the set point.

Before beginning our flight test campaign, a conservative value of Autotune aggressiveness of 5, over a range from 1 to 10, was chosen. For a detailed explanation of the flight modes, see section1.2. This value is indicated as a good starting point for many R/C model aircraft, on one side to avoid automatic abrupt control input and on the other side to have a maneuverable aircraft. This choice returned the "stock" values reported in Table 3.3.

Values reported in both Table 3.1 and Table 3.2 are the parameters of the inner and outer reference models, labeled respectively with the $i$ and the $o$ subscript.

| | $K_{P_\theta}$ | $K_{I_\theta}$ | $K_{D_\theta}$ | $K_{FF_\theta}$ | $\tau_\theta$ |
|---|---|---|---|---|---|
| Stock | 1.18 | 0.05 | 0.09 | 0 | 0.45 |
| VRFT | 0.82 | 0.15 | 0.10 | 0 | 0.24 |
| Variation | $-30,5\%$ | $200\%$ | $11,1\%$ | $0,0\%$ | $-46,6\%$ |

Table 3.3: Longitudinal channel tuned parameters comparison

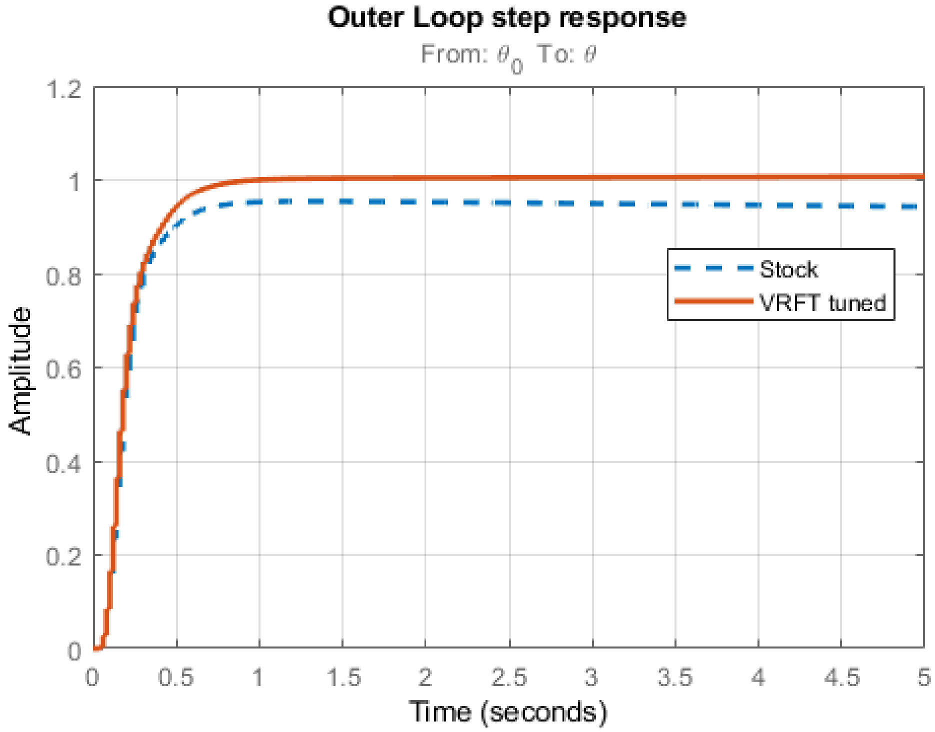

Figure 3.5: Longitudinal simulated outer loop step response

It is worth to notice that the stock controller was not able to bring the aircraft pitch angle to a nose-up (or nose-down) attitude desired by the pilot. This was mainly due to the low integral term that the "Autotune" flight control mode was not in charge of tuning it accordingly. In fact, the Autotune mode tunes the P gain directly, but sets the D and I gains conservatively based on the selected auto-tune aggressiveness.

Our tuning of the inner and outer control loops had shown, therefore, a significant improvement of the step response of the closed-loop system behavior, leading to a significant decrease of the time constant of the latter loop as shown in Table 3.3.

Moreover, we have discovered that the Autotune flight mode was able to tune the gains of the controllers to a settling time very close to the maximum reachable by the aircraft dynamics. In fact, if we assume that our tuning procedure is able to "squeeze" the best performance from the closed-loop system, the two time histories in Figure 3.5 are very close one to each other until the 80% of the amplitude step response. The values shown in Table 3.3 are already re-mapped, ready to be set in the user interface of the UAV through MissionPlanner. Although this procedure of mapping and inverse-mapping is quite annoying, the tuned values showed a slight variation from the stock settings, reassuring about the quality of the whole computational process before flight-testing the new gains.

## 3.4 Lateral-directional dynamics

The lateral and directional channels are, in most conventional aircraft, coupled. The aim of this study is therefore to obtain the gains of the controllers to ensure, known an input on one channel, the best possible response on that channel and a neutral behavior on the other. The tuning of these control loops should therefore provide for the possibility of processing the parameters in a single procedure that simultaneously considers the behavior of both channels, in order to minimize the unwanted couplings.

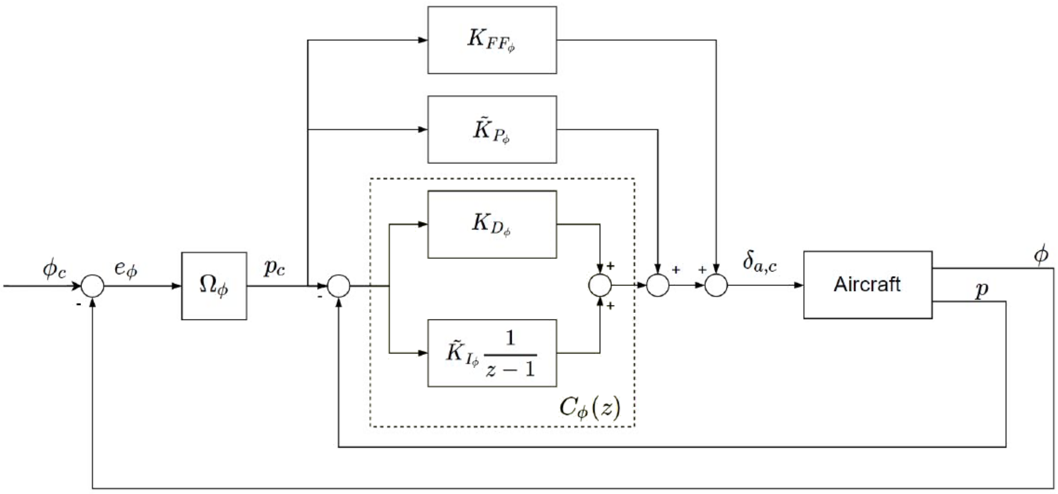

Figure 3.6: Inner and outer roll loop control scheme nomenclature

Control laws that handle the lateral-directional dynamics have, as inputs, roll angle and yaw angle set-points while, as output, they feature ailerons and rudder deflections. At a first glance, the controllers appears to be MIMO. However, if an attitude control is desired, only the roll angle input is of interest. In fact, when controlling the attitude of an aircraft it is unusual to require a set-point of yaw angle to be handled solely by controlling the rudder, that is, by performing a flat turn. We believe here that it is more appropriate to analyze the heading angle set-points in the navigation control level rather than here in terms of the attitude control.

The $K_D$ gain, as well as the integrator reported in the inner loop of the roll control scheme of Figure 3.6 have been condensed in the $C_\phi(z)$ controller in Figure 3.7 for clarity.

This study will then be limited to tuning the controller gains implemented in the ArduPilot control logic to achieve a desirable level of performance in terms of roll angle response while minimizing the adverse-yawing response. Since we are going to control the bank angle using both the aileron and the rudder deflection, the ArduPilot controller has been simplified from MIMO to SIMO.

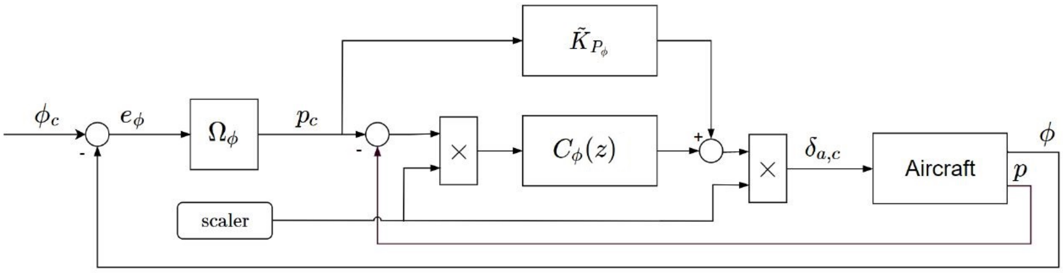

Figure 3.7: General ArduPilot roll control scheme

By analyzing the control scheme and the limits of validity in the frequency-domain of the identified model, it follows that any control block that operates on the typical dynamics of the turns, such as the "Turn coordination offset", and control on the sideslip with the $a_y$ lateral acceleration input, is to be excluded because they operate below the frequency range over which the model is valid. The latter, in our particular case, is not effective: with its glider-shaped fuselage, the Cularis leads to very small values of $a_y$, even for appreciable angles of side-slip.

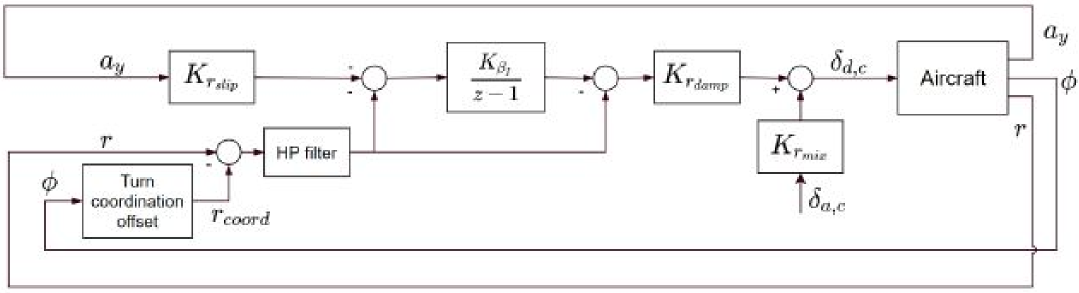

Figure 3.8: Complete ArduPilot scheme for the rudder surface deflection

The control loop with the rudder deflection on the output of Figure 3.8 is thus reduced to only a proportional gain fed by the deflection of the ailerons, as reported in Figure 3.9.

The minimization function, given in input a set-point of bank angle, works by minimizing:

- The mean square error between the output of the guess-tuned, roll-motion closed-loop system ($y_{roll}$)and a signal held constant to 1 ($y_{ref_{roll}}$) both contained in the partial cost equation ((3.11)). This cost term, if minimized alone, is devoted to search for a combination of reference model parameters that minimizes the time needed to achieve the set-point, not dealing with how much over-elongation and/or oscillations;

- The mean square error between the output time history of the yaw-motion,

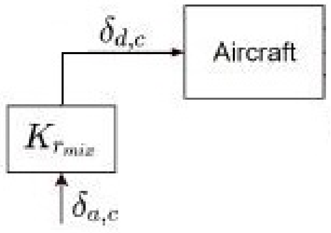

Figure 3.9: Simplified ArduPilot scheme for rudder surface deflection

closed-loop system tuned in the last attempt ($y_{roll}$) and a null signal ($y_{ref_{yaw}}$) both contained in the partial cost equation ((3.12)). The latter has a weight in the cost function at least 10 times higher than the previous parameter on the bank angle because the amount of error on the directional that the cost function must penalize is one order of magnitude smaller. For this reason, the value of $w_2$ is equal to 10.

- A cost term devoted to the elimination of the oscillations of the closed-loop, guess-tuned, output of the external roll loop ($y_{roll}$) active in the partial cost equation ((3.13)). This term, if active using the rule reported in equation (3.15), is proportional to the sum of the difference of two subsequent outer-loop output values, along the whole guess-tuned model output vector. This term would activate if the difference of two consecutive terms in the analyzed vector is negative. Being this operation negative, it means that an oscillation phenomena is happening and it must be penalized in the cost function.

The resulting cost function $C_{latdir}$, in Matlab notation, is reported in equation (3.14) and the coefficient $w_3$ is a weight term that penalizes the response oscillations, ruled according the equation (3.15). The diff() operator reported in equation (3.13) and equation (3.15) returns a vector containing the difference between two subsequent vector elements.

$$C_{1_{latdir}} = \sqrt{mean((y_{roll} - y_{ref_{roll}}).^2)} \tag{3.11}$$

$$C_{2_{latdir}} = w_2\sqrt{mean((y_{yaw} - y_{ref_{yaw}}).^2)} \tag{3.12}$$

$$C_{3_{latdir}} = w_3|(\min(\text{diff}(y_{roll}))| \tag{3.13}$$

$$C_{latdir} = C_{1_{latdir}} + C_{2_{latdir}} + C_{3_{latdir}} \tag{3.14}$$

| $\omega_i$ [rad/s] | | $\zeta_i$ | | $\omega_o$ [rad/s] | | $\zeta_o$ | | $K_R$ | |
|---|---|---|---|---|---|---|---|---|---|
| Min | Max | Min | Max | Min | Max | Min | Max | Min | Max |
| 0.1 | 300 | 0.4 | 2 | 0.1 | 300 | 0.4 | 2 | 0 | 2 |

Table 3.4: Lateral-directional channel reference models - parameters domain

| $\omega_i$ [rad/s] | $\zeta_i$ | $\omega_o$ [rad/s] | $\zeta_o$ | $K_R$ |
|---|---|---|---|---|
| 36,7 | 0,63 | 69,4 | 1,49 | 0.67 |

Table 3.5: Lateral-directional reference models tuned parameters

$$w_3 = \begin{cases} 1 & \text{if} \min(\text{diff}(y_{roll})) < 0 \\ 0 & \text{if} \min(\text{diff}(y_{roll})) \geq 0 \end{cases} \tag{3.15}$$

Any additional weight can be added to the various cost function additions to privilege settling time, channel decoupling and limit over-elongation. The upper and lower bounds of the tunable parameters are reported in Table 3.4.

This technique results in the closed-loop performance reported in Figure 3.10.

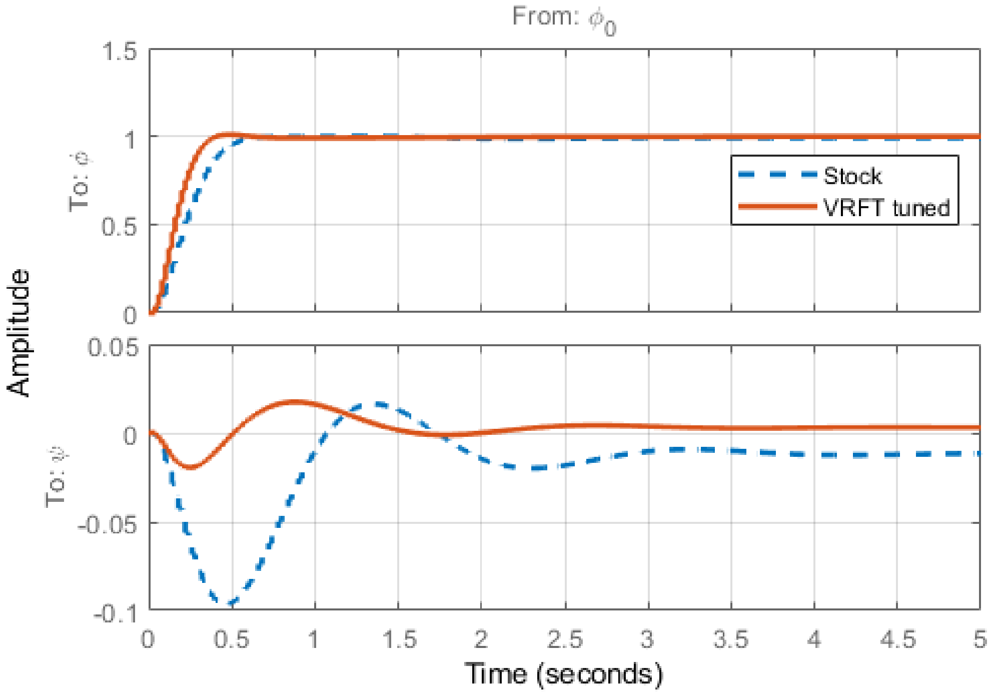

Figure 3.10: Lateral-directional simulated outer loop step response

Values reported in both Table 3.4 and Table 3.5 are the parameters of the inner and outer reference models, labeled rispectively with the $i$ and the $o$ subscript.

| | $K_{P_\phi}$ | $K_{I_\phi}$ | $K_{D_\phi}$ | $K_{FF_\phi}$ | $\tau_\phi$ | $K_{R_{mix}}$ |
|---|---|---|---|---|---|---|
| Stock | 1.03 | 0.05 | 0.08 | 0 | 0.45 | 0.5 |
| VRFT | 2.38 | 1.99 | 0.19 | 0 | 0.18 | 0.67 |
| Variation | +131% | +3880% | +137.5% | 0, 0% | −60% | +34% |

Table 3.6: Lateral-directional channel tuned parameters comparison

It is worth noticing that the stock controller was not tuned to ensure a proper decoupling of the lateral from the directional channel. ArduPilot features an automatic tuning flight mode called ”Autotune” mode, which is enough to make the aircraft safe to use but is certainly not able to bring it to its full potential with the stock settings.

It is worth to notice that the setting of many parameters related to the conduction of maneuvers of lower frequency, such as turns, where the control on lateral acceleration (which we excluded) and other similarly involved tuning parameters, were explicitly fixed to unit quantities because generally valid on most conventional model aircraft.

A detailed tuning, like the one carried out in this thesis and further developments would, for sure, enhance the performance derived by the incorporation of more parameters in the tuning process. However, as before for the longitudinal channel, the low integral term that the “Autotune” flight control mode has set, is now increased properly to reach the best performance that minimizes our functional in close coupling with the directional channel. As a result, we discovered that a faster response on the roll axis was possible, allowing a small over-shoot and, not least, it was possible to find a solution that at the same time significantly reduced the adverse yaw compared to the stock solution.

The explicit values of the control parameters are presented in Table 3.6. The values shown in Table 3.6, as for the longitudinal channel, are already re-mapped using the relations of equation 3.5 and ready to be set in the user interface of the UAV through MissionPlanner. In this case, the calculated gains have shown a significant change from the stock settings, since the gains on derivative and integral tuned controller exhibited changes of an order of magnitude or more. However, by first setting these values on the ground and testing them qualitatively by varying the aircraft’s attitude by hand before take-off, intuitive behavior of the control action was confirmed, and the flight test was subsequently carried out.

# Chapter 4

# In-flight validation

The main purpose of the flight tests is to check how close the simulated results are in comparison to the experimental data.

In-flight tests were performed by trying to manually reproduce the step inputs tested on MATLAB during simulations. To achieve this, the attitude angles reachable in FBWA mode have been limited by means of an appropriate choice of FCU tuning parameters. In this way, to test the closed loop step response on any channel, it was enough to set the FBWA flight mode, wait for the aircraft to automatically level the attitude and then quickly move the proper stick to the maximum reachable travel, waiting 5 seconds to accumulate a relevant attitude time story.

In FBWA flight mode, the aircraft responds to set-points of attitude controlled by the user on the ground, exactly as tested in the simulation phase by numerical means. Particular attention should be paid to the angular rate limit on both the pitch and the roll axis. The angular rate limiter is a control that acts as a saturation to the maximum pith and/or roll rate achievable by the aircraft in all automatic modes, included the FBWA mode. Without this fix, every step set point would be automatically managed as a ramp instead, with the slope indicating the maximum angular velocity that the FCU allows. To remove this control, the PTCH2SRV RMAX and RLL2SRV RMAX parameters in the ArduPilot parameters list should be set to zero.

In order to obtain an acceptable signal to noise ratio, while remaining reasonably within the identified model limits of linearity, the tests performed on the longitudinal channel, as well as on the lateral-directional channel, reach a maximum attitude angle of 20°.

For the longitudinal channel, every test has been performed preferably by injecting a negative pitch set-point. In fact, performing the same tests in a pull-up maneuver (while keeping the throttle constant), the speed decreases very quickly, coming out of the limits of the identified model, and the aircraft begins to show an oscillatory behavior on the lateral-directional channels, typical of the approaching of the stall condition. Increasing the throttle during the pull-up maneuver would inherently lead to a change in the yaw angle due to the propeller slipstream hitting the tail surfaces with greater strength and causing the aircraft to yaw. This yaw, coupled with the roll mode, would unnecessarily excite the lateral-directional closed loop channel, affecting the test.

Moreover, with regard to the decoupling of the lateral from the directional channel, the in-flight tests failed to clearly grasp the effects of the tuning obtained on the yawing decoupling. This problem is due to the signal to noise ratio.

Although this ratio proves to be excellent for measuring performance of the closed loop on the lateral channel, being the signal on the directional channel at least one order of magnitude smaller than the previous one, precisely for the purpose of decoupling the one from the other, the temporal trend of this signal does not stand out compared to the noise of various nature that affects the test.

Results obtained from the different flight tests considered to be most significant are presented in Figure 4.1 and Figure 4.2.

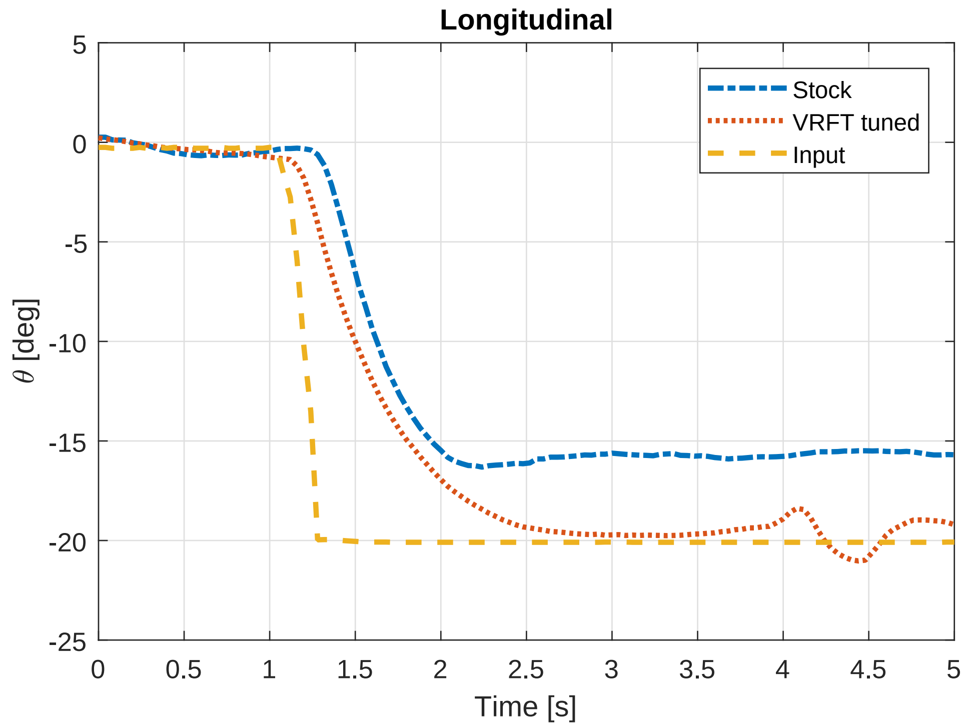

Figure 4.1: Longitudinal in-flight closed loop performance comparison

With regard to the longitudinal channel, there is a significant performance increase of the closed-loop, tuned system, especially in terms of settling time as well as the ability to maintain the requested attitude. The oscillations shown in Figure 4.1 of the tuned system step response at $t = 3.5s$ are to be attributed to atmospheric disturbances. On the lateral-directional channel the improvement is still present, roughly halving the time to reach the roll set point and, at the same

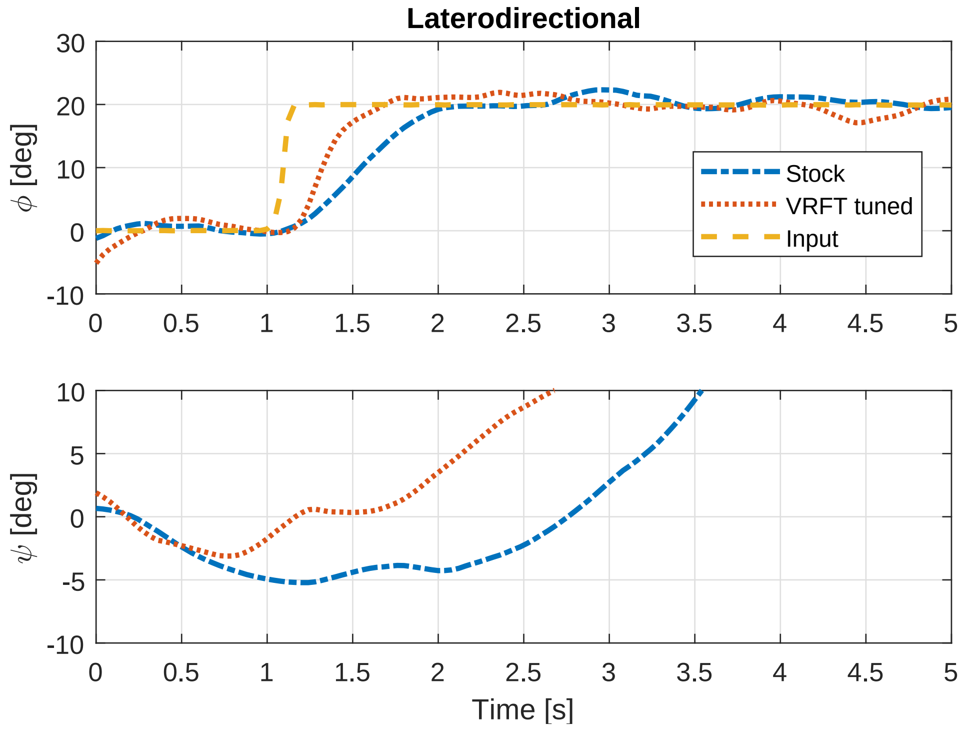

Figure 4.2: Lateral-directional in-flight closed loop performance comparison

time, guaranteeing more decoupling on the directional channel.

It can also be seen in Figure 4.4 that, despite the difficulty in stabilizing the yaw angle of the aircraft at the beginning of the test, at the injection of the bank angle set point the controlled closed-loop system shows a qualitatively lower perturbation of the yaw angle respect than the stock controller. See Figure 4.4 at $t = 1.2$ s where the aircraft, in both cases, actually begins performing the maneuver to increase the angle of bank. The controller tuned according to the VRFT approach keeps the yaw angle almost null until the turn dynamics come into play (that is, indicatively, when the bank angle reaches the set-point), while the controller tuned with the stock gains exhibits a variation of some degrees, reaching the set-point of bank angle for $t = 2$ s and there continuing the change of the yaw angle governed by the turn already taking place.

Although the VRFT-tuned system is qualitatively more aggressive than with the stock settings, while comparing the simulated response with the experimental data, discrepancies were found. The experimental test failed to proper grasp the exact response seen in the simulation environment.

In fact, by comparing the two time histories for both longitudinal and lateral-directional channel, the experimental results shown a slower response to the same input. This difference may be accounted to the scaler action described in section 3.3. Although tests were performed starting at 14 m/s ± 2 m/s, we were not

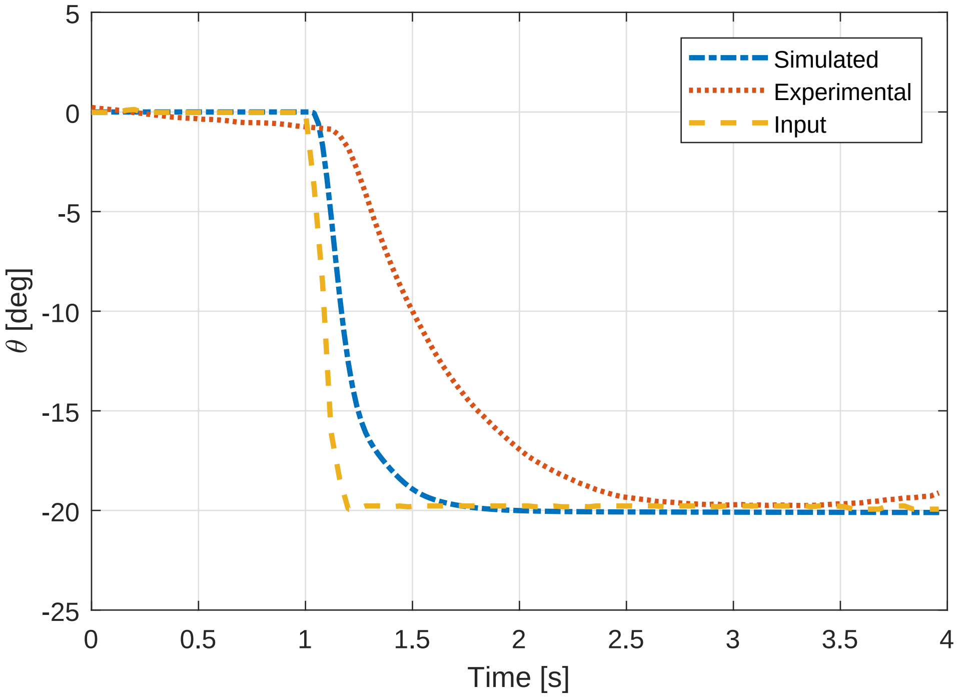

Figure 4.3: Longitudinal performance comparison (simulated - experimental)

able to maintain precisely that particular airspeed and, in general, fluctuations of several meters per second of indicated airspeed were inevitable. For this reason the scaler action could not be considered negligible and, therefore, to an increase of airspeed (as happened in most cases) a constrain on the aerodynamic surfaces deflection is applied. Figure 4.3 and Figure 4.4 clearly show this discrepancy.

A possible improvement could be the conduction of the tests at higher altitudes, where atmospheric convective activity is much lower than in the first hundred of meters above the ground. The latter has not been pursued for legal reasons about the occupation of the airspace and due to the range limitations of the available hardware.

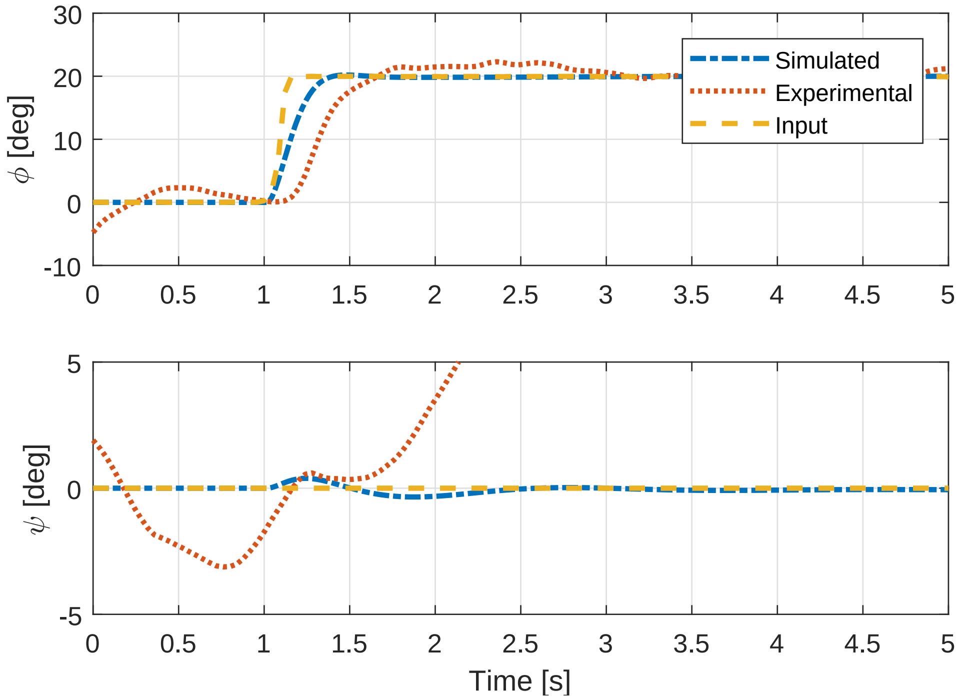

Figure 4.4: Lateral-directional performance comparison (simulated – experimental)

# Conclusions

This tuning procedure employing the VRFT method, has shown results that would otherwise be unattainable with stock settings.

Testing the new VRFT tuned gains has immediately shown the limitations of the “Autotune” mode: the aircraft was in fact more responsive and precise. However, while flight-testing the new controller gains, the noise of the servomotors was important, underlying an important control effort to correct the attitude of the aircraft, making adjustments not always appreciable from the ground. Those corrections, mainly due to atmospheric disturbances, inevitably lead to wear of the servomotors, faster than with the “stock” setting. For this reason, further studies could add more cost-terms to our minimization function, adding a penalty proportional to any servo movement.

The quality of the tests was greatly influenced by the limitations of the available systems. The initial conditions of the tests, in fact, were checked from the ground station, in contact with the aircraft via the telemetry channel. Checking the parameters on the GCS required to look away from the UAV to read the data of interest on the specially configured laptop and then look back at the UAV again.

The limited range of telemetry imposed a limit on distances, beyond which no more information was obtained on the status of the aircraft. It was therefore necessary to fly very close from the ground station, with the inevitable difficulties in estimating the attitude (especially for open-loop tests) and to carry out continuous re-positioning to begin the subsequent tests.

In this sense, a longer-range telemetry system, and a FPV (First Person View) system with an on-board mounted camera would certainly help in estimating the attitude and other relevant flight data during open-loop tests, monitoring more effectively initial test conditions as those information are shown through an head-up display.

It is suggested to continue the studies using PX4 firmware instead of Ardupilot: in fact, apart from the notation issues previously described, PX4 is built over the experience gained with years of ArduPilot development and therefore has a cleaner control structure, with a number of users and developers around the world constantly increasing.

Further studies are encouraged to progressively remove the lower-frequency limits of the identified model and allowing for the simulation (and flight-testing) of longer closed-loop attitude-controlled flights.

Moreover, in parallel, the study of an atmospheric disturbances model applicable to small scale UAV's flying at low altitude could be then added to our simulation input for the tuning, by minimizing a new cost function, of the gains of the control-loops. By doing so, aggressive solutions would be penalized, leading to an interesting weight-study on whether or not is better to accept attitude variations (due to disturbances) or keeping the desired attitude as best as possible scarifying the servo life and, of course, on-board power.

However, the calibration of the controllers carried out despite the limitations listed before, was satisfactory and highlighted a new technique for choosing the reference model useful for the application of the VRFT method. This technique allows the user to avoid the choice of the reference model (which in this procedure happens in a completely automatic way) focusing on the choice of desired performance, imposing constraints according to the specifications of interest. Such constraints can also transcend control theory and also lead to mission requirements, such as the search for the ideal gains to minimize the energy spent on control action.

# Bibliography

G.L. Teper. Aircraft stability and control data. 1969.

R.W. Beard and T.W. McLain. *Small unmanned aircraft, Theory and practice.* Princeton university press, Oxfordshire OX20 1TW, 2012.

R.V. Jateganokar. *Flight Vehicle System Identification: A Time Domain Methodology.* American Institute of Aeronautics and Astronautics, 1801 Alexander Bell Drive, Reston, VA 20191, 2006.

V. Klein and E.A. Morelli. *Aircraft System Identification Theory and Practice.* AIAA, Virginia Polytechnic institute and State University Blacksburg, Virginia, 2006.

B. Etkin. *Dynamics of atmospheric flight.* Dover Publications inc., Mineola, New York, 2000.

K.P. Valavanis and G.J. Vachtsevanos. *Handbook of Unmanned Aerial Vehicles.* Springer, 2015.

F.Lewis B. Stevens and E. Johnson. *Dynamics, Control design and Autonomous systems.* Wiley, 2016.

S. Farì. Guidance and control of a fixed-wing UAV. 2016.

P. Panizza. Data driven control system design for multirotor systems. 2017.

A. Zangarini. Data-driven multivariable attitude control design of multirotor uav platforms. 2018.

F. Bus. Fixed-wing uav performance flight testing. 2015.

Matlab grey box model estimation. `https://it.mathworks.com/help/ident/grey-box-model-estimation.html`, 2019.

Matlab black box model estimation. `https://it.mathworks.com/help/ident/ug/black-box-modeling.html`, 2019.

Matlab multistart. `https://it.mathworks.com/help/gads/multistart.html`, 2019.

Ardupilot reference guide. `http://ardupilot.org/plane/`, 2019.

# Appendix A

# ArduPilot parameter list

This appendix reports in detail all the ArduPilot configuration parameters of the control loops after the tuning.

For a complete explanation of the parameters presented, please consult [ard, 2019].

| Parameter | Symbol | Value |
|---|---|---|
| PTCH2SRV TCONST | $\tau_\theta$ | 0.23 |
| PTCH2SRV P | $K_{P_\theta}$ | 0.82 |
| PTCH2SRV I | $K_{I_\theta}$ | 0.15 |
| PTCH2SRV D | $K_{D_\theta}$ | 0.1 |
| PTCH2SRV IMAX | - | 4500 |
| PTCH2SRV RMAX UP | - | 0 |
| PTCH2SRV RMAX DN | - | 0 |
| PTCH2SRV RLL | - | 1 |
| LIM PITCH MAX | - | 20 |
| LIM PITCH MIN | - | 20 |

Table A.1: Pitch control loop parameters

| Parameter | Symbol | Value |
|---|---|---|
| RLL2SRV TCONST | $\tau_\phi$ | 0.18 |
| RLL2SRV P | $K_{P_\phi}$ | 2.38 |
| RLL2SRV I | $K_{I_\phi}$ | 1.99 |
| RLL2SRV D | $K_{D_\phi}$ | 0.2 |
| RLL2SRV FF | $K_{FF_\phi}$ | 0 |
| RLL2SRV RMAX | - | 0 |
| RLL2SRV IMAX | - | 4500 |
| LIM ROLL CD | - | 20 |

Table A.2: Roll control loop parameters

| Parameter | Symbol | Value |
|---|---|---|
| YAW2SRV SLIP | - | 0 |
| YAW2SRV INT | - | 0 |
| YAW2SRV DAMP | - | 0 |
| YAW2SRV RLL | $K_{r_{coord}}$ | 1 |
| KFF RDDRMIX | $K_{r_{mix}}$ | 0.67 |
| YAW2SRV IMAX | - | 4500 |

Table A.3: Yaw control loop parameters

| Parameter | Symbol | Value |
|---|---|---|
| SCALING SPEED | - | 14 |

Table A.4: Scaling airspeed