ADCS Design for a Sounding Rocket with Thrust Vectoring

Pedro Emanuel Moutinho dos Santos

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisor: Professor Paulo Jorge Coelho Ramalho Oliveira

Examination Committee

Chairperson: Professor Afzal Suleman
Supervisor: Professor Paulo Jorge Coelho Ramalho Oliveira
Member of the Committee: Dr. Alain Souza

December 2022

I declare that this document is an original work of my own authorship and that it fulfils all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

To my grandpa, that I know is always watching me.

Acknowledgments

First of all, I would like to thank Professor Paulo Oliveira for all the guidance and support he has given me throughout this journey and for showing me what is left to explore.

To RED, who provided me with the opportunity to develop my skills by working in a team with ambitious goals and talented people who I can now call true friends.

To my friends, Eduardo, Filipe e Gonçalo, who have shared with me all the ups and downs and helped me until the last minute.

To my girlfriend, who is always there for me and cheered me up in the hardest times.

To my mom, for always having the best advice, making me think clearer, and, somehow, always knowing the answer. To my dad, who inspires me everyday to reach higher in the hopes of one day being as wise as him. To my grandma, for always making sure that I have everything I need. Finally, I would like to thank all my family.

Resumo

A presente dissertação aborda o desenvolvimento de um sistema de determinação e controlo de atitude (ADCS) para um foguete-sonda capaz de controlar a direção do vetor de força propulsiva. A principal motivação foi a inclusão do sistema num foguete-sonda a desenvolver pelo RED (Rocket Experiment Division), um projeto do núcleo de estudantes de Engenharia Aeroespacial do Instituto Superior Técnico (AeroTéc).

Para desenhar o ADCS, um modelo não-linear da dinâmica e cinemática do foguete nos seis graus de liberdade deduz-se e implementa-se em ambiente de simulação, recorrendo às ferramentas *MATLAB* e *Simulink*®.

Através da representação em espaço de estados, variante no tempo, do foguete, é projetado um controlador ótimo, baseado no regulador quadrático linear (LQR), com uma ação integral adicional. Este controlador é testado no ambiente de simulação, demonstrando robustez a incertezas no modelo e um seguimento de referências satisfatório. O desempenho evidenciado é significativamente superior ao da solução clássica com um controlador proporcional-integral-derivativo (PID).

Recorre-se a um estimador de parâmetros linear para fornecer estimativas em tempo real das forças e momentos aerodinâmicos. Estas estimativas são utilizadas para desenvolver um controlador adaptativo que determina os ganhos em tempo real. O controlador adaptativo não demonstra uma melhoria significativa no desempenho devido à elevada robustez do controlador original.

Finalmente, um sistema de navegação, baseado nas medições disponíveis a bordo, é projetado para fornecer estimativas precisas das variáveis de estado. O ADCS é o resultado da integração dos sistemas de controlo de atitude e navegação, sendo testado em simulação e demonstrando um desempenho satisfatório.

Palavras-chave: ADCS, Foguete-sonda, TVC, Controlo de atitude, Estimação de atitude.

Abstract

This thesis concerns the development of an attitude determination and control system (ADCS) for a sounding rocket using Thrust Vector Control (TVC). The main motivation was the inclusion of the designed system in a future sounding rocket, to be developed by a student rocketry team from the Aerospace Engineering Student's Association of Instituto Superior Técnico (AeroTéc), named Rocket Experiment Division (RED).

To design the ADCS, a non-linear six-degrees-of-freedom model for the rocket dynamics and kinematics is deduced and implemented in simulation environment using *MATLAB* and *Simulink*®.

The attitude control system design is done using the derived linear, time-varying, state-space representation of the rocket. An optimal controller is designed using the linear quadratic regulator (LQR) with an additional integral action. This controller was tested in the simulation environment and demonstrated satisfactory attitude tracking performance, and robustness to model uncertainties, significantly outperforming the classical proportional-integral-derivative (PID) control solution.

A linear parameter estimator is implemented to provide real-time estimates on the aerodynamic forces and moments. These estimates are then used to develop an adaptive controller that computes the gains in real-time after correcting the state-space model. The adaptive controller did not show significant performance improvement due to the already high robustness of the original controller.

Finally, a navigation system is designed, based on measurements available onboard, to provide accurate estimates on the rocket's state. The ADCS is the result of the integration of the attitude control and navigation systems, with the complete system being implemented and tested in simulation and demonstrating satisfactory performance.

Keywords: ADCS, Sounding rocket, TVC, Attitude control, Attitude estimation.

Contents

List of Tables

List of Figures

Nomenclature

Greek symbols

α	Angle of attack.
β	Angle of side-slip.
$\boldsymbol{\omega}$	Angular velocity vector
μ_1	Pitch control input.
μ_2	Yaw control input.
ϕ	Roll angle.
ψ	Yaw angle.
ρ	Air density.
θ	Pitch angle.

Roman symbols

$\mathbf{F}$	Force vector.
$\mathbf{J}$	Inertia matrix.
$\mathbf{M}$	Moment vector.
$\overline{q}$	Dynamic pressure.
C_A	Coefficient of axial force.
C_m	Coefficient of pitching moment.
C_N	Coefficient of Normal force.
C_n	Coefficient of yawing moment.
C_Y	Coefficient of lateral force.
d	Fuselage diameter.
g	Gravitational acceleration.

l	Thrust vector moment arm.
m	Mass.
R	Rotation matrix from body frame to Earth frame.
S	Fuselage cross-sectional area.
SM	Static stability margin.
T	Thrust force.
x_{cm}	location of the centre of mass.
x_{cp}	location of the centre of pressure.
$\mathbf{v}$	Linear velocity vector.
$\mathbf{v_w}$	Wind velocity vector.
p, q, r	Angular velocity components in the body frame.
u, v, w	Linear velocity components in the body frame.

Subscripts

∞	Free-stream condition.
a	Aerodynamic.
g	Gravitational.
lat	Lateral mode.
lon	Longitudinal mode.
p	Propulsive.
rel	Relative to the wind.
x, y, z	Cartesian components.

Superscripts

B	Expressed in the body frame.
E	Expressed in the inertial frame.
T	Transpose.

Abbreviations and Acronyms

ACF	Attitude Complementary Filter
ADCS	Attitude Determination and Control System
AeroTéc	Núcleo de Estudantes de Engenharia Aeroespacial do Instituto Superior Técnico
CEA	Chemical Equilibrium Applications
CFD	Computational Fluid Dynamics
EuRoC	European Rocketry Challenge
GNSS	Global Navigation Satellite System
IMU	Inertial Measurement Unit
IST	Instituto Superior Técnico
LPE	Linear Parameter Estimator
LQI	Linear Quadratic Integral
LQR	Linear Quadratic Regulator
PCF	Position Complementary Filter
PID	Proportional Integral Derivative
RED	Rocket Experiment Division
TVC	Thrust Vector Control

Chapter 1

Introduction

1.1 Motivation

The main motivation behind this work is the development of an attitude determination and control system (ADCS) for a future sounding rocket from a student rocketry team from Instituto Superior Técnico (IST), named Rocket Experiment Division (RED). RED is a project created by students in 2017, under the Aerospace Engineering Student's Association (AeroTéc), with the aim of designing, building, testing, and launching sounding rockets for scientific purposes and to participate in university level competitions. So far, the largest vehicle built by RED was $3\,\mathrm{m}$ long and was launched to the approximate altitude of $3\,\mathrm{km}$. However, the ultimate goal of the team is to lead the space launchers sector in Portugal by designing rockets capable of reaching higher altitudes, passing the Kármán line and entering space, hence the growing need to actively control the stability and trajectory of the vehicle.

Although focusing on the practical implementation in a student rocket, this thesis aims to describe the attitude control and determination system design as generically as possible, so that it can be easily adapted to any small scale launcher. The designed attitude control system assumes that the rocket uses Thrust Vector Control (TVC) technology as the actuation method, and aims to control the rocket's pitch and yaw angles. The roll angle is assumed to be controlled by an additional roll control system whose design is out of the scope of this thesis. The objective for the integrated attitude control and navigation systems is to estimate and control the attitude of the rocket, to be able to respond to attitude commands from an external guidance system, and track a desired trajectory. A system with the high level architecture presented in Fig.1.1 should be designed, based on the original contributions of this work.

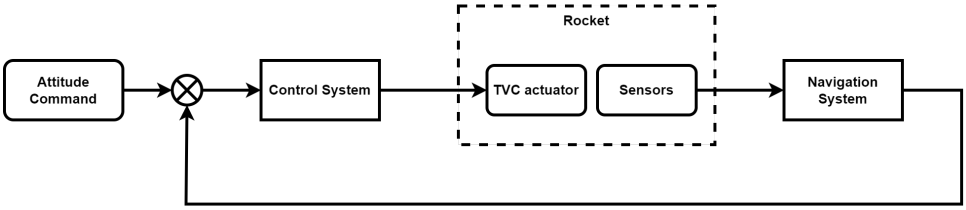


Figure 1.1: Proposed high level system architecture.

1.2 Topic Overview

All the relevant dynamics of a small scale rocket are similar to those of a space launcher, apart from specific difficulties that emerge with the increase in mass and size. Among these difficulties, it is important to highlight the decrease of the natural frequencies of an increasing number of flexible modes, which get closer to the range of the control bandwidth [1]. In this way, the sounding, or small scale, rocket attitude control problem can be seen as a simplified version of the generic launcher attitude control problem.

During the atmospheric flight phase of a rocket, stabilization can be achieved through the use of aerodynamic fins. With a correct design of the fins, the centre of pressure of the rocket is located further away from its nose than the centre of mass, which causes the vehicle to be naturally stable [2, 3]. However, the rocket is subjected to various external disturbances, such as wind gusts, which prevent the vehicle to follow a desirable, pre-calculated trajectory or, even more intense, completely destabilize it [4]. In addition to external disturbances, the rocket's mass, centre of mass, inertia and aerodynamic properties considerably change throughout the flight, adding even more uncertainty on the flight dynamics. It is then clear the necessity of having an active attitude control and stabilization system that not only ensures the stability of the rocket, but allows to actively correct its trajectory in order to achieve specific mission goals. Moreover, the atmospheric flight phase is undeniably the most challenging one. The aerodynamic load increases significantly right after lift-off, reaches its peak at the maximum dynamic pressure region and decreases greatly afterwards. Through the use of an active attitude control system, it is possible to reorient the vehicle according to the direction of significant wind gusts in order to minimize the aerodynamic load [5].

The fundamental task of a launch vehicle attitude determination and control system (ADCS) is to estimate its orientation with respect to an inertial frame and to drive it into a desired condition. Generally, this condition is provided by a guidance algorithm, which may either be a closed-loop steering algorithm or an open-loop attitude profile. In both cases, the desired attitude command of the vehicle can be expressed as a time-varying attitude quaternion or Euler angles, possibly with an associated attitude rate command. These attitude and rate commands are issued to the ADCS by a given guidance function, and compared with sensed outputs in order to generate an error signal. These error signals are then used to generate torque or acceleration commands through the use of a control algorithm, which are then distributed to actuators to effect the proper orientation of the vehicle, while maintaining the required performance and suppressing undesirable dynamics [1, 6].

As for the actuation technology, a first option is to rely on movable fins, capable of adjusting the aerodynamic moment and, consequently, the rocket's attitude. Although used in early stage vehicles, such as the German V-2 from the second world war, and in some sounding rockets, its applicability is limited to the denser regions of the atmosphere and it is dependant on the rocket's velocity to have sufficient authority. This means that this actuation method imposes limits on the altitude and velocity regimes. Besides that, fins create additional aerodynamic drag on the vehicle which limits its efficiency by increasing the amount of propellant to reach the same altitude [6, 7].

Technology exists, that allows for a wider range of operating conditions and provides better efficiency, denominated as Thrust Vector Control (TVC), or thrust vectoring for short. This actuation method is used by most launch vehicles and works by redirecting the thrust vector in order to create a control torque [5]. For a single gimballed engine or nozzle configuration, which is the common case for small scale launchers, the TVC actuation can only correct pitch and yaw attitude errors. Roll has to be controlled by a dedicated reaction control system, normally using cold gas thrusters. By combining these systems, the rocket's attitude can be controlled from launch up to orbital insertion (if that is the case), even if it means having multiple stages, each with its dedicated control system. For multiple engine configurations, roll attitude control can be achieved through differential thrust vectoring [8]. The TVC technology is also an integral part in the emerging sector of reusable launch vehicles, allowing for the controlled landing and recovery of the first stage of the vehicle with the help of additional systems, as seen in SpaceX's Falcon 9.

The control system design tends to be very conservative in the aerospace industry. Linear frequency-domain methods are standard because they allow fairly arbitrary representations of the vehicle dynamics with fewer theoretical limitations. Restricting the dynamic analysis to accommodate more sophisticated control design techniques risks the later realization that such restrictions would have to be lifted and would invalidate the control design. As a result, classical techniques are the established and accepted practice in the launch vehicle design industry. Among the classical techniques, the Proportional-Integral-Derivative (PID) control is on the core of most commonly used launch vehicle control systems [1, 5, 9]. Although widely used, PID control has its downsides when it comes to robustness and external disturbances rejection [10, 11]. The problem of controlling ascending launch vehicles is dominated by parameter uncertainty, which in face of the lack of robustness of the PID controller may be a concerning issue. Moreover, the rocket flight parameters considerably change throughout the flight. To overcome this, gain scheduling techniques have been proposed, that rely on the linearization of the dynamics at different operating conditions. Still in the linear domain, an advance with respect to the PID controller is the use of optimal controllers, such as the Linear Quadratic Regulator (LQR), that not only provide more robustness but ensure an optimal solution for a given cost function [6, 11].

As a way to improve the robustness of linear time-varying controllers, real time parameter estimators can be introduced in the control loop to form an adaptive control system. The online identification of system parameters allows the controller to act on a more accurate representation of the system dynamics, which in the case of rockets can considerably change throughout the flight. Traditionally, time-invariant estimation is used for modal parameter identification, however, due to the time-varying dynamics it is no longer applicable, and it is necessary to develop the time-varying structural modal parameter identification method [12].

Non-linear control techniques have also been proposed for launch vehicle attitude controllers [13], and come with the advantage of ensuring a global solution for the control problem, not dependant on the specific mission nor vehicle. In this way, the controller is able to function in a wider range of conditions with respect to the linear time-varying techniques. However, this type of controllers often have to simplify the complete non-linear dynamics in order to obtain a global solution. If relevant dynamics are discarded, the controller might fail in a real implementation scenario. Besides that, these methods all have particular design characteristics which make it harder to develop a standardized verification and validation procedure to meet the imposed system requirements [1, 5, 9].

1.3 Thesis Contributions

This thesis has several contributions, all connected with the design of an ADCS for a small scale rocket. The first contribution is the development of a six degrees-of-freedom (6 DoF) non-linear model for the dynamics and kinematics of a rocket, that serves as a tool for control and navigation system design. This model is linearized and results in a generic state-space representation for a rocket with TVC. Additionaly, a preliminary design for a future solid motor rocket with TVC, to be implemented by RED, is performed, and used as reference to test the developed controllers and estimators.

The second relevant contribution is the development of an optimal pitch and yaw controller using the linear version of the model and resorting to the Linear Quadratic Regulator (LQR) technique, with the addition of an integral action. This controller is assessed and compared with that of the classical PID controller solution.

Thirdly, a linear time-varying parameter estimator which relies on the developed rocket model is designed in order to estimate in real time the aerodynamic forces and moments acting on the vehicle. This estimator is then used to feed the state estimated on the LQR controller, making it an adaptive controller resorting to an online gain computation.

Finally, a navigation system is designed, based on measurements available onboard, to provide accurate estimates on the rocket's state, to be used by the attitude controller.

1.4 Thesis Outline

The structure of this thesis reflects the research and development time-line. Firstly, a theoretical background covering the dynamics and stability of rocket flight, the existent active attitude control systems and the basics of control theory, is presented. With the established knowledge on rockets and associated flight dynamics, the rocket dynamics and kinematics modelling process is explained. After detailing the modelling procedure, the 6 DoF non-linear model for a thrust-vector-controlled rocket is obtained and linearized. At this point, the preliminary design of the rocket that is used as reference to verify the proposed controllers and estimators is detailed, alongside with the intended mission and nominal trajectory. Using the linear version of the model, the optimal attitude controller is designed as well as the parameter estimator. Following the attitude controller design, the navigation system design is presented. Finally, with the design phase completed, the implementation in simulation of the non-linear model and the proposed controllers and estimators is explained, with all the simulation results presented and discussed in the following chapter, right before the conclusion and the final remarks.

Chapter 2

Theoretical Background

2.1 Sounding Rockets

Sounding rockets take their name from the nautical term ”to sound,” which means to take measurements. They are small rockets designed for sub-orbital flight and capable of carrying a scientific payload to a wide range of altitudes, from tenths to hundreds of kilometres. They can carry instruments to take measurements and/or perform experiments in the high atmosphere, in near space and/or in microgravity conditions. They are able to produce higher-quality microgravity conditions for longer periods than airplanes and drop towers or tubes [14, 15].

This class of rockets can have one or several stages and is normally divided into two sections (Fig. 2.1): a motor, most of the times a solid-fuel rocket motor but it can also be a liquid or hybrid one, and a payload. Many of the motors used in sounding rocket programs are surplus military motors, which keep down the cost of the rocket. The payload is the section that carries the instruments to conduct the experiment and can either send the data back to Earth, which implies real-time tracking by one or more ground telemetry stations, or store it on board, which implies a successful recovery of the rocket in order to extract the data [16].

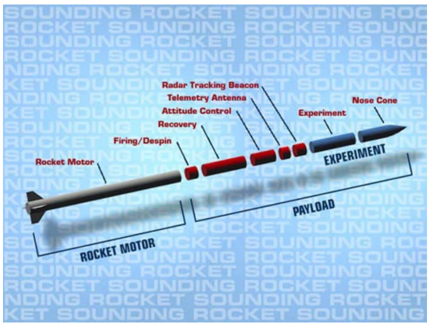


Figure 2.1: Typical sounding rocket sections [14].

As opposed to an orbital trajectory, the sub-orbital flight of sounding rockets is essentially a parabolic, or inverted ”U-shaped”, trajectory, providing nearly vertical paths along their ascent and descent. The flight path looks like a massive u-turn, from Earth to space and back, and can be divided into the following phases: the burning or powered phase, which starts from launch and continues until the rocket motor burns the entirety of its propellant; the coasting phase, where the payload continues into space after separation from the motor(s) (when it has separation capability) and begins conducting the experiments; and, finally, when the experiments are completed, the payload re-enters the atmosphere and a parachute is deployed, bringing the payload (or the entire vehicle when separation doesn't occur) gently back to Earth [14].

2.2 Flight Dynamics and Stability

2.2.1 Forces

From the dynamic point of view, sounding rockets experience four main forces during a flight: Weight, Thrust, Lift and Drag (Fig.2.2). Weight is the force produced by the interaction with Earth's gravitational field, Thrust is the propulsive force that causes the rocket to move, and Lift and Drag are the aerodynamic forces that result from the interaction between the rocket's body and the atmosphere.

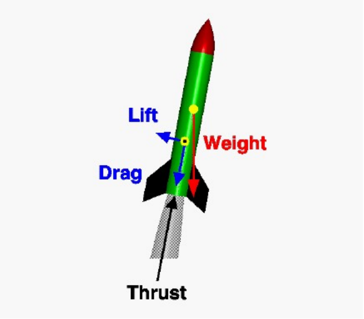


Figure 2.2: Forces acting on a rocket [17].

The magnitude of the Weight force depends on the mass of all of the parts of the rocket, is always directed towards the centre of the Earth and acts through the centre of gravity.

As for Thrust, it is a reaction force caused by the expelled exhaust gases at very high speeds in the direction opposite to the rocket's motion which, according to Newton's third law and momentum conservation, propels the rocket. These gases are generated by combustion in a chamber, reaching high pressure, and are expanded through a convergent-divergent nozzle, granting them a very high exhaust velocity. As the engine consumes its propellant, the rocket's mass will drop depending on the propellant mass flow rate, meaning that it is a variable mass system [8, 17].

Aerodynamic Forces

Aerodynamic forces appear when there is a relative motion between a solid and a fluid. In this case, the solid is the rocket body and the fluid is the air composing the atmosphere. Lift is the component of the aerodynamic force perpendicular to the relative velocity of the rocket with respect to the surrounding air, while Drag is in line with the velocity vector in the opposite direction, which means it is a force opposing the rocket's motion.

Lift

Lift is a mechanical force resultant from a pressure gradient in the fluid passing through the solid body, caused by different fluid velocity in different parts of the body. In basic terms, when the fluid encounters the body, it is forced to go around it and, if the fluid leaves the body in a different direction from the one it approached it, momentum conservation implies that a force must appear in the opposite direction - Lift. In the case of rockets, due to their axial symmetry, Lift only appears when the relative velocity is not aligned with the longitudinal axis or, in other words, when the angle of attack (the angle between the velocity vector and the longitudinal axis in the two dimensional case) is not zero. As opposed to aircrafts, rockets do not use Lift to counteract their weight but it can be used to grant stability and keep the rocket on its path through its fins [18, 19].

Drag

The Drag force can be divided into three main types that differ on their cause: Parasitic Drag, Wave Drag, and Lift-induced Drag.

- **Parasitic Drag** - Parasitic Drag corresponds to the sum of Pressure Drag and Skin Friction Drag. Similarly to Lift, Pressure Drag is caused by a pressure gradient in the fluid imposed by the shape of the body, with the difference that it corresponds to the component aligned with the relative velocity vector. However, contrary to Lift, Pressure Drag is present even when the angle of attack is zero, since there is a high pressure region in the front of the rocket due to the presence of a stagnation point and a low pressure region in the back due to the separation of the flow from the body. This pressure gradient generates a force - Pressure Drag - that provides resistance to the rocket's motion. On the other hand, Skin Friction Drag is the one caused by shearing stresses acting over the surface of the body due to the viscosity of the fluid [20].
- **Wave Drag** - Wave Drag appears during transonic and supersonic flight due to the formation of shock waves. As the speed of the rocket approaches the speed of sound, pressure waves produced by the motion in the fluid are compressed. When the speed of sound is reached (Mach equal to one), a sonic boom occurs and the shock waves cause a sudden increase in Drag [18].
- **Lift-induced Drag** - Lift-induced Drag only occurs on finite, lifting wings. It occurs because the distribution of lift is not uniform on a wing, but varies from root to tip. For a lifting wing, there is a pressure difference between the upper and lower surfaces of the wing. Vortexes are formed at the wing tips, which produce a swirling flow that is very strong near the wing tips and decreases toward the wing root. The local angle of attack of the wing is increased by the induced flow of the tip vortex, giving an additional, downstream-facing, component to the aerodynamic force acting on the wing. For rockets, fins are considered a lifting wing when the angle of attack is not zero (since they have a symmetric profile), and so, in this condition, they will produce Lift-induced Drag [21].

The total aerodynamic force can also be decomposed in the parallel and perpendicular directions to the longitudinal axis of the body. For the two-dimensional case, the Normal and Axial forces relate to Lift and Drag through the aerodynamic angle of attack in the following manner:

$$L = N\,\cos\alpha - A\,\sin\alpha\,, \tag{2.1a}$$

$$D = N\,\sin\alpha + A\,\cos\alpha\,. \tag{2.1b}$$

As previously mentioned, the aerodynamic forces acting on the rocket are due to the distributed loads imposed by the the pressure and shear stress distributions. If the aerodynamic force on a body is specified in terms of a single resultant, or its components such as Normal and Axial, it must be applied in a point such that it produces the same effect as the distributed loads. This location is called the centre of pressure. Since it is the point where the resultant of a distributed load effectively acts on the body, the aerodynamic moments taken at the centre of pressure are zero [18]. If the location of the centre of mass and centre of pressure do not coincide, there are aerodynamic moments on the rocket, classified as pitching or yawing moments depending if they are acting on the pitch or yaw axes.

The location of the centre of pressure is crucial for the stability of a passively controlled sounding rocket, as it will be shown ahead.

Aerodynamic Coefficients

The aerodynamic forces and moments are most commonly represented through dimensionless coefficients, the aerodynamic coefficients, which allow to better compare the aerodynamic characteristics of different bodies. In order to obtain the dimensionless coefficients, the freestream dynamic pressure is used, and it is defined as

$$\overline{q} = \frac{1}{2}\,\rho\,{V_\infty}^2\,, \tag{2.2}$$

where ρ is the density of the fluid and V_∞ is the unperturbed flow velocity. In addition, let S be a reference area and l be a reference length. The dimensionless force and moment coefficients are then given by:

$$C_F = \frac{F}{\overline{q}\,S}\,, \tag{2.3a}$$

$$C_M = \frac{M}{\overline{q}\,S\,l}\,. \tag{2.3b}$$

In the above coefficients, the reference area S and reference length l are chosen to pertain to the given geometric body shape; for different shapes, S and l may represent different quantities. In the case of rockets, the norm is to use S as the cross-sectional area of the fuselage and l as its diameter. Moreover, the force, F, can be replaced by any aerodynamic force, such as Lift, Drag, Normal or Axial, and the moment, M, can be replaced by any given aerodynamic moment, such as the aerodynamic pitching or yawing moments [18].

By looking at the aerodynamic coefficients' definition, one can infer that the freestream dynamic pressure is a very important quantity to determine the aerodynamic load acting on a body. For a given set of aerodynamic coefficients and reference geometric dimensions of the body, it will determine the aerodynamic forces and moments acting on it, generating higher values for a higher dynamic pressure. During a rocket launch, a maximum value for the dynamic pressure will be reached, known as *max q*, due to the combined time evolution of air density and the rocket's velocity. As the vehicle gains altitude, air density will drop and velocity will increase, which implies a maximum value for the dynamic pressure at a given instant. This value is very important when designing the structure of the rocket in order to ensure its structural integrity and, when the engine's thrust is controllable, it is reduced when passing through this point so that the overall load on the vehicle is minimised [2].

2.2.2 Stability

Stability is the property of a body that causes it, when disturbed from a condition of equilibrium or steady motion, to develop forces or moments that restore the original condition. It is essential for a rocket to be stable in order to keep a safe and nominal trajectory throughout its flight, and this can be achieved either by a passive or active way. A passively stable rocket is designed in such a way that its inherent aerodynamic properties will grant sufficient stability, also referred to as aerodynamic stability. On the other hand, an actively stable rocket uses active control mechanisms which stabilise the trajectory at the expense of energy consumption. Active control techniques will be covered later on and for now the focus will be on the passive aerodynamic stability of rockets.

In agreement with the definition of stability, an aerodynamic stable vehicle, will, after a small perturbation, return by itself to its original state by producing restoring aerodynamic forces and moments. If the dynamic effects are not considered, it is referred to as static stability, otherwise as dynamic stability.

Static Longitudinal Stability

If we consider a steady-flight of a rocket in the longitudinal plane, it is possible to characterise its static longitudinal stability. Given the typical symmetry of rockets, the lateral case is in every sense similar. A small increment in the angle of attack, $\Delta\,\alpha$, yields an incremental normal force,

$$\Delta\,C_N = C_{N_\alpha}\,\cdot\Delta\,\alpha\,, \tag{2.4}$$

where the linear evolution of the normal force coefficient with the angle of attack, valid for small angles, is used, and its slope, C_{N_α}, is called a stability derivative. The total normal force applies at the centre of mass, while the increment in normal aerodynamic force will act on the centre of pressure. This implies an incremental aerodynamic moment about the centre of mass,

$$\Delta\,C_m = C_{m_\alpha}\,\cdot\Delta\,\alpha = C_{N\,\alpha}\,\cdot\Delta\,\alpha\,\cdot\,(\frac{x_{cm}-x_{cp}}{d})\;\Leftrightarrow\;(\frac{x_{cp}-x_{cm}}{d}) = -\frac{C_{m_\alpha}}{C_{N_\alpha}}\,, \tag{2.5}$$

where x_{cm} and x_{cp} are, respectively, the positions of the centre of mass and centre of pressure on the longitudinal axis of the rocket from the tip of the nose, and d is the diameter of the fuselage. The dimensionless quantity $(\frac{x_{cp}-x_{cm}}{d})$ is called static stability margin (SM) and it determines the static stability of a rocket. It is seen that for $SM > 0$, i.e. $x_{cp} > x_{cm}$, or equivalently, $C_{m_\alpha} < 0$ (kowing that C_{N_α} is always positve), the vehicle is statically stable, while it is neutrally stable if x_{cp} and x_{cm} coincide. For $SM < 0$, i.e. $x_{cp} < x_{cm}$, or equivalently, $C_{m_\alpha} > 0$, the vehicle is statically unstable. The necessary negative value for the stability derivative C_{m_α} is quite intuitive. If a disturbance causes a positive deflection of α, the produced aerodynamic moment shall have a negative sign in order for it to be restoring. Essentially, the static longitudinal stability is determined by the relative position of the centre of mass and centre of pressure, knowing that a stable configuration implies that the centre of pressure is aft of the centre of mass, as illustrated by Fig. 2.3.

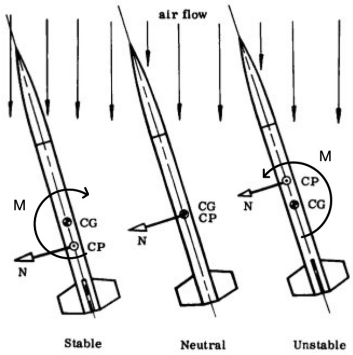


Figure 2.3: Rocket passive stability and static margin [3].

It is now possible to understand the role of fins in stabilising the rocket, as their contribution to the overall pressure distribution makes the cp move backwards. The position of the cp depends mainly on the Mach number and angle of attack, while the the cm usually shifts due to propellant consumption, which implies that the static margin varies during flight. The limits of the static margin can be controlled either by the fin area, where a larger fin area leads to a cp location closer to the fins themselves, or by adding a ballast, ideally in the nose cone, in order to shift the cm location.

A large positive static margin is very effective, but it also has some disadvantages. On one side, it can lead to large bending moments in the structure, especially for larger launch vehicles. On the other, in the presence of a cross-wind, the angle of attack will be increased and an aerodynamic stable rocket will counteract the increment, turning the nose into the wind direction. This phenomenon is known as *weathercocking* and the higher the static margin the more relevant it becomes, making the trajectory excessively deviate from the nominal one. Moreover, if one desires to actively guide the rocket, a large positive static margin will oppose to the turning moments provided by the steer mechanism making it harder to achieve the desired guidance [2, 3].

Dynamic Stability

Dynamic stability concerns the way a system evolves after being disturbed from its state of equilibrium. As already explained, with a positive static stability margin the rocket having an angular deviation is turned by aerodynamic forces to its initial position. When the angle of attack becomes zero, the restoring force will become zero. However, since the rocket maintains angular velocity, the motion will continue in an opposite direction. An angle of attack with an opposite sign will develop and angular oscillations of the rocket will arise. If the rocket is dynamically stable, due to the presence of damping forces the oscillations will decay. Oscillations of this type are called short period oscillations and their frequency depends on the stability margin and the moment of inertia of a rocket. The higher the margin of stability and the lower the moment of inertia, the higher the frequency of these oscillations, the greater the damping, the quicker the oscillations will decay, and the faster the rocket is restored to the steady state motion. The short period angular oscillations described above may occur, as is easily understood, about the three central axes.

The various kinds of rocket oscillations are not independent. During transverse angular oscillations, due to periodic development of lift force, transverse oscillation of the centre of gravity appears about the steady state trajectory. If the rocket moves with an angle of attack, then, with the appearance of the yaw angle, the rocket will roll. This happens due to a difference in velocity between both pitch plane fins. The one with additional velocity has an increase in lift, whereas the other has a decrease. A moment develops, rotating the rocket about its longitudinal axis and giving rise to rocket roll.

The problem of stabilisation consists of limiting and damping out as quickly as possible all oscillations of the types described [2, 4].

Roll Dynamics and Spin Stabilisation

Roll is the rotational movement of a vehicle around its longitudinal axis. For an ideal sounding rocket, with no manufacturing anomalies, roll is negligible. However, imperfections such as a cant angle in the fins can induce a significant rolling motion with spin rates in the range of several Hz [22].

The rolling motion acquired by the rocket contributes to its stability to some extent due to the gyroscopic effect. As the rocket spins around the longitudinal axis, it is less sensitive to perturbations in the other axes, pitch and yaw, which makes the vehicle follow a straighter path. This is the reason why some sounding rockets are designed to spin at high rates, either by introducing a cant angle on the fins or by actively deflecting a control surface [2, 4].

Nevertheless, spin stabilised rockets carry some downsides. One straightforward disadvantage is the intrinsic energy loss due to the rolling motion. In order to spin, the rocket has to generate Lift through its fins which causes an increment in Drag and, consequently, it will reach a lower altitude. In addition, a set of instabilities may occur due to non-linear rolling motion. For small angles of attack, the roll rate is proportional to the fin cant angle, and independent of the angle of attack. This is called linear rolling. However, at higher angles of attack, the roll rate is no more linear and resonance and coupling between modes of oscillation can occur. This may lead to several undesirable phenomena such as roll lock-in, where the spin rate goes to zero; roll speed-up, where the vehicle reaches extremely high spin rates; catastrophic yaw, where the pitch-yaw angular motion reaches high amplitudes that may cause vehicle disintegration; and Magnus instability, where lateral forces appear due to the combination of circulation due to spin and cross flow at an angle of attack [2].

2.3 Active Attitude Control Systems

With relatively simple static stabilization, the effects of short duration, random disturbing influences acting on the rocket during flight are satisfactorily excluded, but the influence of systematic, constantly acting factors, which are not considered beforehand, remains. To their number belong, first of all, technological errors. The rocket cannot be fabricated exactly. The rocket motor always has a certain eccentricity, and the line of thrust does not pass through the center of gravity of the rocket. This creates a moment, turning the rocket sideways. The external contours of the rocket cannot be fabricated ideally. Therefore, aerodynamic forces also have a certain eccentricity. Fins themselves, whose purpose it is to insure stability of motion, always have inaccuracies due to fabrication which lead to the deviation of the rocket from the required trajectory. The aerodynamic stabilization cannot be effective during flight beyond the limits of the atmosphere. Additionally, we have seen that statically stable rockets will have a tendency to turn in the wind direction which increases with the static margin and makes it deviate from the nominal trajectory and, for strong wind gusts, it can completely destabilise the vehicle.

As already covered, another possible way of assuring stability of motion of an unguided rocket consists of imparting to the rocket rapid rotation about its longitudinal axis, the so called spin stabilisation method. In this case, the rocket maintains the direction of its axis due to the gyroscopic effect, however, we have seen that it has negative aspects and, especially for larger rockets, it can be difficult to ensure the structural integrity at the spin rates necessary for stabilisation. Another downside is the difficulty in guiding a vehicle with such spin rates [2, 4].

To tackle these difficulties, active attitude control are implemented. An attitude control system is a specific type of control system whose function is to control the attitude of a vehicle, meaning its orientation relative to a frame of reference. By controlling the attitude of the vehicle, it is possible to control its trajectory, making sure it is stable and that the effects of external perturbations are minimised. To control the attitude of a flight vehicle, like a sounding rocket, there is the need for the actuators to produce torques that induce angular motion, hence changing its attitude.

For sounding rockets, and rockets in general, the two main actuation techniques consist on the deflection of aerodynamic surfaces and/or directing the engine's thrust, the so called thrust vector control (TVC). Reaction control systems are also a possibility, but are normally used in coordination with one or both the other methods [7, 23]. In the following sections, these actuation methods will be described, as well as their applicability.

2.3.1 Thrust Vector Control (TVC)

Thrust Vector Control, or thrust vectoring, consists in controlling the direction of the thrust force (or vector) in order to produce torques that act on the rocket's centre of mass, influencing its rotation in pitch, yaw, and roll. Most often, and particularly for sounding rockets, there is a single nozzle whose thrust vector is in the direction of the vehicle axis and goes through its centre of mass. Thus, it is possible to obtain pitch and yaw control moments by simple deflections of the single thrust vector, whereas roll cannot be controlled.

Figure 2.4 illustrates how a pitch or yaw moment is obtained from a hinged thrust chamber or nozzle. The side force varies with the sine of the effective angle of thrust vector deflection, where the pitch (or yaw) moment is given by the product between the side force and the distance from the hinge point to the rocket's centre of mass. As the propellant burns, the centre of mass will shift and with it the moment arm, influencing the magnitude of the control torque. It is evident that TVC only functions during the powered flight phase and other means of stabilisation must be used for the coasting phase.

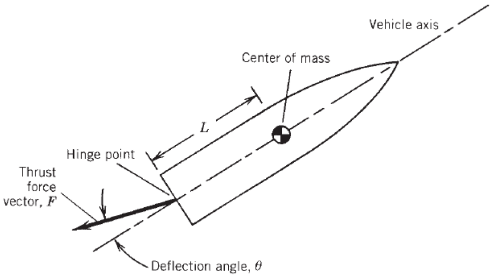


Figure 2.4: TVC representation [8].

TVC can be achieved through a set of different mechanisms. For the case of interest of a single nozzle they can be classified into four categories:

1. Mechanical deflection of the main nozzle or thrust chamber.
2. Insertion of heat-resistant movable bodies into the exhaust jet; these experience aerodynamic forces and cause deflection of part of the exhaust gas flow.
3. Injection of fluid into a side portion of the diverging nozzle section, causing an asymmetrical distortion of the supersonic exhaust flow.
4. Separate thrust-producing unit that is not part of the main propulsion system providing flow through one or more of its own nozzles.

Amongst the mechanical deflection mechanisms there are two main families: the gimbal or hinge scheme and the movable nozzle. In the gimbal or hinge scheme (a hinge permits rotation about one axis only, whereas a gimbal is essentially a universal joint), the whole engine (thrust chamber and nozzle) is pivoted on a bearing and thus the thrust vector rotates. For small angles this scheme has negligible losses in specific impulse and is used in many vehicles. This method is specific for liquid engines and it requires a flexible propellant-piping unit or bellows to send the propellant from the fixed vehicle tanks to the movable engine. Movable nozzles are ideal for solid propellant engines since the thrust chamber is fixed and the thrust vector is changed with the nozzle deflection. The two principal movable nozzle technologies consist either on a flexible laminated bearing or a flexible nozzle joint. Of all the mechanical deflection types, movable nozzles and gimballed engines are the most efficient since they do not significantly reduce thrust or specific impulse and are weight-competitive with other mechanical types.

Under the insertion of movable bodies into the exhaust jet, jet vanes, the jetavator and jet tabs are the main available technologies. By comparison with the mechanical deflection mechanisms, the disadvantages are the thrust loss due to drag and the erosion of the surfaces which limits their duration. As for the advantages, they provide an higher actuation speed and, for the case of thrust vanes, they can provide roll control even for a single nozzle configuration.

Regarding the fluid injection mechanism, secondary fluid injection through a nozzle wall into the main gas stream induces oblique shocks in the nozzle's divergent section, causing an unsymmetrical distribution of the main gas flow, which produces the side force. Secondary fluids can originate from a stored liquid or from a separate hot gas generator (the gas would need to be sufficiently cool to be piped), a direct bleed from the chamber, or the injection of a catalysed monopropellant. When the needed deflections are small, this is a low-loss scheme, but for large enough moments (large side forces) the amount of secondary fluid can become excessive.

The auxiliary thrust-producing unit is used only with liquid engines and they grant a low performance loss, however, the moments they can produce are relatively small [8, 23, 24].

2.3.2 Aerodynamic Control Surfaces

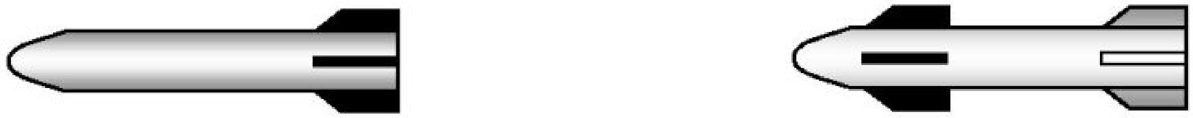
Another method for attitude control actuation is to implement movable aerodynamic surfaces capable of generating aerodynamic moments. This type of actuation is very common in missiles and, due to their similarity, can also be used in sounding rockets. Since it relies on the aerodynamic interaction with the atmospheric air, fin control, as it is commonly referred to, only functions during atmospheric flight and the actuation capacity will decrease with altitude, as air density also decreases. This means that sounding rockets aiming to pass the atmosphere while maintaining guidance and stabilisation must rely on thrust vector control, while the engine is still burning, and reaction control systems after burnout.

If we assume that a rocket is flying with no angle of attack and with all fins in neutral position (no local angle of attack), due to symmetry, no lift nor aerodynamic moments are present. By rotating a fin, a local of angle of attack appears and said fin starts to generate lift, which increases with the angle of attack. If the fin is not aligned with the centre of mass of the rocket, the local increase in lift leads to an aerodynamic moment whose arm depends on the distance between the fin and the centre of mass. Normally, fins are positioned by pairs defining the symmetry planes of the rocket - pitch and yaw. By commanding each pair of fins in a symmetric manner it is possible to control the pitch and yaw rotations, while an unsymmetrical deflection allows roll control.

There are different possible configurations depending on the position of the fins, movable and fixed, on the rocket, amongst which the two most common ones (Fig. 2.5) are the following:

- **Tail control** - In this configuration, the rear located fins, as the ones normally used for passive stabilisation, are movable. They are allowed to rotate around a spanwise axis and provide pitch,yaw and roll control.

- **Canard control** - This configurations consists in the addition of movable canards, which are fins located closer to the nose of the vehicle, while still having fixed fins at the tail. The rear fins provide passive stability and the canards allow pitch, yaw and roll control, also through their rotation around a spanwise axis.



(a) Tail control

(b) Canard control

Figure 2.5: Aerodynamic surfaces control configurations (movable in black) [25].

With respect to thrust vectoring control, movable aerodynamic surfaces have the advantage of working both in the powered and coasting flight phase whenever the desired apogee is located inside the atmosphere. On the other hand, fins will bring additional drag contributing to performance losses. Additionally, when it comes to controller design for actuated fins, it is essential to characterise the aerodynamic properties of the fins for numerous flight regimes in order to obtain a successful design. Throughout its flight, a rocket will experience a wide range of velocities and, consequently, the control torque variation with the fins' deflection angle will change. One can argue that for TVC with solid motors the thrust is not controllable and will change with time creating an analogous problem, however, it is also admissible to reply that accurately characterising the thrust variation with time is a much easier task than to determine the response of the fins at high Mach numbers under turbulent flow [4, 7, 26].

2.3.3 Reaction Control Systems

Reaction Control Systems (RCS) are another type of attitude control mechanism that relies on thrusters and/or reaction wheels to produce reaction control torques. RCS are widely used in spacecrafts and can also be used in sounding rockets as an additional attitude control system. When it comes to thrusters, they are propulsive devices, generally monopropellant or cold gas based, that produce a propulsive force in the desired directions, and are strategically positioned on the body of the rocket so that the moment arms with respect to the centre of mass are such that pitch, yaw and roll control is possible. As for reaction wheels, they individually induce angular motion around one body axis by spinning in an axis parallel to it. By increasing the rotation speed they are able to increase the reaction torque, limited by the maximum rotation speed. If a constant disturbance needs to be countered, the reaction wheel will accumulate angular momentum until the limit rotation speed is reached, meaning the reaction wheel is saturated. After that point it can no longer be used to counteract the disturbance and needs to be desaturated by an external device like the reaction thrusters [27, 28].

RCS have limited actuation capacity and most of the times cannot be used independently during the atmospheric flight phase of a sounding rocket. In this way, they are the most useful in the following situations:

- In case of separation, provide three-axis control to the payload during the space flight phase;
- In combination with TVC, provide roll control during the powered flight phase and three-axis control after burnout;

- In combination with movable fins, provide three-axis control when atmospheric density is too low for the fins to control the vehicle.

2.4 Control Theory

Control is the name given to the general task of achieving a desired result by appropriate adjustments. The object to be controlled, in this case a rocket, is referred to as the plant, while the process that exercises the control is called the controller. A system has variables applied to it externally, called the input vector, and produces certain variables internally, called the output vector, which can be measured. In modelling a system, one must account for the relationship between the input and output vectors. This relationship generally takes the form of a set of differential and algebraic equations, if the system is governed by known physical laws. A system having known physical laws is said to be deterministic, whereas a system with unknown (or partially known) physical laws is called non-deterministic or stochastic. Every system has certain unwanted external input variables – called disturbance inputs – that cannot be modelled physically and are thus treated as stochastic disturbances.

The state of a system at a given time is specified by a set of scalar variables, called state variables, or, in vector form, the state vector. The vector space spanned by the state vector is called a state space. The number of state variables required to represent a system is called order of the system, because it is equal to the net order of differential equations governing the system. While the size of the state space (i.e., the order of the system) is unique, any given system can be described by infinitely many alternative state-space representations. For instance, a flight vehicle's state can be described by the position, velocity, angular velocity and orientation relative to a frame of reference. However, the state can be transformed into any number of different state vectors depending upon the choice of the reference frame.

A system consisting of the plant and the controller is called a control system. The controller manipulates the plant through a control input vector, which is actually an input vector to the plant but an output of the controller. In physical terms, this output can take the form of either a force or a torque (or both) applied to a flight vehicle. Often, only electrical (or mechanical) signals are generated by the controller through wires (cables, hydraulic lines), which must be converted into physical inputs for the plant by a separate subsystem called an actuator. Also, controllers generally require measurement of the output variables of the plant. Whenever a measurement of a variable is involved, it is necessary to model the dynamics of the measurement process as a separate subsystem called a sensor. Generally, there are as many sensors and actuators as there are measured scalar variables and scalar control inputs, respectively. The sensors and actuators can be modelled as part of either the plant or the controller. The design of a control system requires an accurate mathematical model for the plant, hence one of the milestones of this work of developing one for sounding rockets. [6, 29]

Chapter 3

Rocket Dynamics and Kinematics Modelling

As mentioned in Section 2.4, in order to design a controller it is essential to have an accurate mathematical model that represents the physics of the system to control. In particular, to design an attitude controller for a rocket, the model shall represent its translational and rotational dynamics and kinematics. Next, the modelling process will be described.

In order to model the dynamics and kinematics of a rocket, the main references were [1, 30–33], since they not only explain thoroughly the reasons behind some of the necessary assumptions that were taken into account, but also describe in detail all the equations that were necessary to model the rocket dynamics and kinematics.

The dynamics will be described for the generic case, meaning that they can be adapted for either a passively or actively stabilized rocket, and include both movable fins and TVC actuation.

3.1 Assumptions

- **The rocket is considered to be a rigid body**, meaning no elastic behaviours are modelled. This assumption allows to only consider the rigid body equations of motion and, given the smaller size of sounding rockets with respect to orbital launchers, it is well suited for a preliminary controller design.

- **The rocket is assumed to be axially symmetric, as well as the mass allocation**, which means that the principal inertia axes coincide with the body axes, the centre of mass is on the longitudinal axis, and the aerodynamic behaviour is identical in both the pitch and yaw planes.

- **The flat Earth model is used**, meaning that neither the curvature nor rotation of the Earth are taken into account. Especially for the current sounding rockets of RED, for which the apogee and ground range are relatively small, this assumption is also valid for a preliminary attitude controller design since the neglected effects have a low impact on the overall dynamics.

3.2 Reference Frames

In order to describe the dynamics and kinematics of the rocket, it is crucial to define the reference frames to be used. Two reference frames are used: a body-fixed one, where the equations of motion are written; and an inertial space-fixed one, where the trajectory will be defined by translating into it the motion variables from the body-fixed reference frame.

The body-fixed reference frame has its origin located in the center of mass of the vehicle, as shown in Fig. 3.1. The x-axis (X_b) is along the rocket's longitudinal axis; the z-axis (Z_b) is perpendicular to the x-axis and points downward, in-plane with a set of fins (in case they exist); and the y-axis (Y_b) defines an orthogonal reference frame.

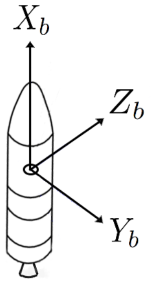


Figure 3.1: Body-fixed reference frame.

As for the inertial space-fixed reference frame, given that neither the curvature nor the rotational motion of the Earth are taken into account, a simple orthogonal frame centered in the launch location is used. The x-axis (X_e) is pointing upwards, so that for a zero inclination launch the x-axes of both reference frames are aligned; and the other two axes (Y_e and Z_e) are preferably aligned with a pair of cardinal directions.

3.2.1 Euler Angles

The equations of motion will be described in the body-fixed reference frame, while some of the dynamics, such as the gravity force, are more easily defined in the Earth frame. Moreover, the resulting kinematics in the body frame have to be translated to the Earth frame to obtain the trajectory of the vehicle. In this way, it is necessary to define the coordinate transformation between both reference frames.

This is done by using a sequential rotation of the body frame relative to the Earth frame defined by the three Euler angles:

$$\mathbf{R}(\phi, \theta, \psi) = \mathbf{R}_z(\psi) \cdot \mathbf{R}_y(\theta) \cdot \mathbf{R}_x(\phi) \tag{3.1}$$

- Roll (ϕ): the Euler angle of rotation of the body relative to the x-axis of the Earth frame;
- Pitch (θ): the Euler angle of rotation the body relative to the y-axis of the Earth frame;
- Yaw (ψ): the Euler angle of rotation of the body relative to the z-axis of the Earth frame.

The Euler angles, $\boldsymbol{\lambda} = [\phi \ \theta \ \psi]^T$, describe the attitude of the rocket, representing the variables to be controlled by the attitude control system. The coordinate transformation from the body frame to the Earth frame is then defined by the following transformation matrix:

$$ {}^{E}_{B}\mathbf{R} := \mathbf{R} = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix} , \tag{3.2} $$

where c and s stand as abbreviations for the trignometric functions. The inverse transform, from the Earth frame to the body frame, is defined by the transpose:

$$ {}^{B}_{E}\mathbf{R} = \mathbf{R}^T = \begin{bmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ s_\phi s_\theta c_\psi - c_\phi s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & s_\phi c_\theta \\ c_\phi s_\theta c_\psi + s_\phi s_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi & c_\phi c_\theta \end{bmatrix} . \tag{3.3} $$

3.3 Gravity Model

The vehicle will be subjected to gravity during the entirety of its flight. Using the flat Earth model assumption, the gravitational acceleration is considered to vary only with altitude. This variation is given by:

$$ g = g_0 \frac{R_E^2}{(R_E + h)^2} , \tag{3.4} $$

where g_0 is the gravitational acceleration constant at surface level, R_E is the mean Earth radius and h is the altitude. The gravitational force in the Earth-fixed reference frame is equal to

$$ {}^{E}\mathbf{F_g} = \begin{bmatrix} {}^{E}F_{g_x} \\ {}^{E}F_{g_y} \\ {}^{E}F_{g_z} \end{bmatrix} = \begin{bmatrix} -mg \\ 0 \\ 0 \end{bmatrix} , \tag{3.5} $$

and the transformation matrix is used to translate it into the body frame:

$$ {}^{B}\mathbf{F_g} = \mathbf{R}^T \cdot {}^{E}\mathbf{F_g} = \begin{bmatrix} {}^{B}F_{g_x} \\ {}^{B}F_{g_y} \\ {}^{B}F_{g_z} \end{bmatrix} = \begin{bmatrix} -mg \cdot c_\theta c_\psi \\ -mg \cdot (s_\phi s_\theta c_\psi - c_\phi s_\psi) \\ -mg \cdot (c_\phi s_\theta c_\psi + s_\phi s_\psi) \end{bmatrix} . \tag{3.6} $$

3.4 Propulsion and TVC Actuation

The propulsion model was derived using equations mainly obtained from [8], considering ideal propulsion and all its underlying assumptions. The thrust produced by the rocket motor is simply:

$$T = \underbrace{|\dot{m}| \cdot v_e}_{\text{Dynamic}} + \underbrace{(p_e - p_a) \cdot A_e}_{\text{Static}} , \tag{3.7}$$

where $\dot{m}$ is the mass flow rate, v_e is the effective exhaust velocity, p_e is the nozzle exit pressure, p_a is the atmospheric pressure, and A_e is the nozzle exit area. Note that two separate contributions can be identified: the dynamic one, caused by the exhaust of the expanded combustion gases; and the static, caused by the pressure gradient between the nozzle exit and the atmosphere. For an ideal expansion, only the dynamic part remains and so it is also referred to as ideal thrust.

Thrust will be determined by the pressure generated in the combustion chamber and by the acceleration of the combustion gases through the nozzle. Considering an ideal isentropic expansion, the pressure ratio between the nozzle exit and the combustion chamber $\left(\frac{p_e}{p_c}\right)$ can be obtained through the following equation:

$$\frac{1}{\epsilon} = {p_{\text{ratio}}}^{1/k} \left(\frac{k+1}{2}\right)^{\frac{1}{k-1}} \sqrt{\frac{\left(1 - {p_{\text{ratio}}}^{\frac{k-1}{k}}\right)\,(k+1)}{k-1}} , \tag{3.8}$$

where the nozzle expansion ratio is $\epsilon = \dfrac{A_e}{A_t}$, with A_t being the throat area, and k is the specific heat of the combustion mixture. The effective exhaust velocity is then,

$$v_e = \sqrt{\frac{2\,R T_c\, k\, \left(1 - {p_{\text{ratio}}}^{\frac{k-1}{k}}\right)}{M_{\text{mol}}\,(k-1)}} , \tag{3.9}$$

where R is the universal gas constant, T_c is the combustion chamber temperature and M_{mol} is the molar mass of the combustion mixture.

By knowing the thermodynamic properties of the combustion, thrust can be calculated combining (3.7), (3.8) and (3.9) with the combustion chamber pressure (p_c) and the mass flow rate ($\dot{m}$). For a liquid engine, these quantities are set by the propellant feeding system, while for a solid motor, the internal ballistics have to be computed in order to derive them.

The internal ballistics depend on the burning rate, burning surface and grain geometry. The burning rate (r_b) is the regression rate of the propellant surface, essentially in the direction perpendicular to it. It is possible to approximate the burning rate as a function of chamber pressure, particularly using the empirical Vieille's law:

$$r_b = a\, {p_c}^n , \tag{3.10}$$

where a is the temperature coefficient, as it depends on the ambient grain temperature, and n is the pressure exponent, since it doesn't depend on temperature but influences the chamber operating pressure and the burning rate. Both these numbers depend on the propellant type and are empirically determined inside different ranges of pressure. On the other hand, the quasi-steady combustion pressure is given by the following equation:

$$p_c = \rho_{prop}\,\frac{A_{burn}}{A_t}\,c^*\,r_b = \left(\rho_{prop}\,\frac{A_{burn}}{A_t}\,a\,c^*\right)^{\dfrac{1}{1-n}}\,, \tag{3.11}$$

allowing the computation of the mass flow rate,

$$\dot{m} = \frac{p_c\,A_t}{c^*}\,, \tag{3.12}$$

where c^* is the characteristic velocity (taken as the ideal one), ρ_{prop} is the propellant density and A_{burn} is the burning surface area. The burning surface area might change throughout the burn, depending on the grain geometry. In this way, a time varying chamber pressure and, consequently, mass flow rate, might be obtained.

At this point, all the equations necessary to compute the thrust produced by the motor were detailed and the only missing factor is the decomposition of the propulsive force in the body axes for a vehicle with thrust vector control capability. Said decomposition can be performed as shown in Fig. 3.2.

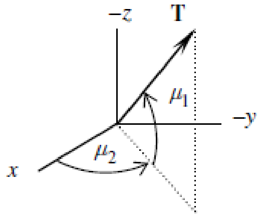


Figure 3.2: Thrust vector decomposition in the body axes [6].

According to it, the thrust vector is decomposed using the angles μ_1 and μ_2, where μ_1 is the gimbal angle that, on its own, produces a pitching moment, and μ_2 is the one that produces a yawing moment. Using these angles, the propulsive force in the body frame follows [6],

$${}^{B}\mathbf{F_p} = \begin{bmatrix} {}^{B}F_{p_x} \\ {}^{B}F_{p_y} \\ {}^{B}F_{p_z} \end{bmatrix} = \begin{bmatrix} T\,\cos\mu_1\,\cos\mu_2 \\ -T\,\cos\mu_1\,\sin\mu_2 \\ -T\,\sin\mu_1 \end{bmatrix}\,, \tag{3.13}$$

while the control moment in the body frame, produced by thrust vectoring, is

$${}^{B}\mathbf{M_p} = \begin{bmatrix} {}^{B}M_{p_x} \\ {}^{B}M_{p_y} \\ {}^{B}M_{p_z} \end{bmatrix} = \begin{bmatrix} 0 \\ -T\,\sin\mu_1\,l \\ T\,\cos\mu_1\,\sin\mu_2\,l \end{bmatrix}\,, \tag{3.14}$$

where l is the moment arm, which corresponds to the distance between the nozzle gimbal point and the centre of mass of the rocket.

3.5 Aerodynamic Forces and Moments

As described in Section 2.2, the rocket will be subjected to aerodynamic forces and moments resulting from its interaction with the fluid medium composing the atmosphere. Starting by the force, it is expressed in the body axes according to,

$$ {}^{B}\mathbf{F_a} = \begin{bmatrix} {}^{B}F_{a_x} \\ {}^{B}F_{a_y} \\ {}^{B}F_{a_z} \end{bmatrix} = \begin{bmatrix} -\overline{q}\, C_A\, S \\ \overline{q}\, C_Y\, S \\ -\overline{q}\, C_N\, S \end{bmatrix} , \tag{3.15} $$

where C_A is the axial aerodynamic force coefficient, C_Y is the lateral aerodynamic force coefficient, C_N is the normal aerodynamic force coefficient, $\overline{q}$ is the dynamic pressure and S is a reference area, usually corresponding to the cross sectional area of the fuselage.

The velocity of interest for the aerodynamic computations is the relative velocity between the rocket and the atmosphere, which is given by

$$ {}^{B}\mathbf{v_{rel}} = {}^{B}\mathbf{v} - {}^{B}\mathbf{v_w} \tag{3.16} $$

where ${}^{B}\mathbf{v_w}$ is the local wind velocity vector in the body frame, and ${}^{B}\mathbf{v} = [u,\, v,\, w]^T$ is the linear velocity vector in the body frame. The magnitude of the relative velocity is also represented through a dimensionless quantity, the Mach number, which relates it to the speed of sound (a_s),

$$ M = \frac{||\,{}^{B}\mathbf{v_{rel}}||}{a_s} \, . \tag{3.17} $$

The axial and normal aerodynamic forces correspond to the body axes components of Lift and Drag and are related through the aerodynamic angles - the angle of attack and the sideslip angle,

$$ \alpha = \arctan\left(\frac{w_{rel}}{u_{rel}}\right) , \tag{3.18a} $$

$$ \beta = \arcsin\left(\frac{v_{rel}}{||\,{}^{B}\mathbf{v_{rel}}||}\right) . \tag{3.18b} $$

The force coefficients can be determined using a linear relation with the aerodynamic angles, whose derivative depends mainly on the angle itself and Mach number,

$$ C_Y = C_{Y\beta}\, \beta \, , \tag{3.19a} $$

$$ C_N = C_{N\alpha}\, \alpha \, . \tag{3.19b} $$

As for the aerodynamic moment, in the body axes it is given by

$$ {}^{B}\mathbf{M_a} = \begin{bmatrix} {}^{B}M_{a_x} \\ {}^{B}M_{a_y} \\ {}^{B}M_{a_z} \end{bmatrix} = \begin{bmatrix} \overline{q}\, C_l\, S\, d \\ \overline{q}\, C_m\, S\, d \\ \overline{q}\, C_n\, S\, d \end{bmatrix} \,, \tag{3.20} $$

where d is a reference length, usually corresponding to the diameter of the body tube. For the case of having actively controlled fins, the aerodynamic moment coefficients about the centre of mass are the following:

$$ C_l = C_{l_\delta} \delta_p + \frac{d}{2V_{rel}}\, C_{l_p}\, p \,, \tag{3.21a} $$

$$ C_m = C_{m_{ref}} + C_N\, \frac{x_{cm} - x_{ref}}{d} + \frac{d}{2V_{rel}}\, (C_{m_q} + C_{m_{\dot{\alpha}}})\, q \,, \tag{3.21b} $$

$$ C_n = C_{n_{ref}} + C_Y\, \frac{x_{cm} - x_{ref}}{d} + \frac{d}{2V_{rel}}\, (C_{n_r} + C_{n_{\dot{\beta}}})\, r \,, \tag{3.21c} $$

with,

$$ C_{m_{ref}} = C_{m_\alpha}\, \alpha + C_{m_\delta}\, \delta_m \,, \tag{3.22a} $$

$$ C_{n_{ref}} = C_{n_\beta}\, \beta + C_{n_\delta}\, \delta_n \,. \tag{3.22b} $$

In the equations above, the terms depending on the angular velocities (p, q and r) are dynamic terms originated by the angular motion of the rocket that influence its dynamic stability by providing damping, while the others are static terms. The reference aerodynamic moment coefficients are obtained at a certain point, x_{ref}, which is the reference moment station and its location must be provided alongside all aerodynamic data. The angles δ_p, δ_m and δ_n are the angular deflections of the fins that cause, respectively, a rolling, pitching or yawing moment, where the angle δ_p can also include the fin cant angle in case it exists. If the reference moment station is defined as the centre of pressure, its location can be determined, and there are no control surfaces, the reference moments are zero and the moment coefficients take the following form:

$$ C_l = C_{l_\delta} \delta_p + \frac{d}{2V_{rel}}\, C_{l_p}\, p \,, \tag{3.23a} $$

$$ C_m = -C_N\, \frac{x_{cp} - x_{cm}}{d} + \frac{d}{2V_{rel}}\, (C_{m_q} + C_{m_{\dot{\alpha}}})\, q \,, \tag{3.23b} $$

$$ C_n = -C_Y\, \frac{x_{cp} - x_{cm}}{d} + \frac{d}{2V_{rel}}\, (C_{n_r} + C_{n_{\dot{\beta}}})\, r \,, \tag{3.23c} $$

where the stability margin ($\frac{x_{cp} - x_{cm}}{d}$) intuitively appears, and δ_p accounts only for the fin cant angle.

All the aerodynamic coefficients and stability derivatives are characteristic of the rocket and will depend on the flight conditions, meaning they will change throughout the flight. For this model, they are assumed to vary with Mach number and aerodynamic angles. This implies that the simulation model shall have built-in information regarding these quantities for the vehicle in case and must have a way to vary them during the simulation depending on the instant Mach number and aerodynamic angles.

3.6 6 DoF Equations of Motion

Now that all the external forces and moments that the rocket is subjected to were detailed, it is possible to define the non-linear differential equations that describe its motion within the six degrees of freedom: 3 translational and 3 rotational.

3.6.1 Translational Motion

The translational dynamics in the inertial frame are described by Newton's second law,

$$^{E}\mathbf{F} = {}^{E}\dot{\mathbf{P}}\,, \tag{3.24}$$

where $^{E}\mathbf{F}$ is the total external force and $^{E}\dot{\mathbf{P}}$ is the linear momentum, both expressed in the inertial frame. Since the body frame is a rotating one with angular velocity $^{B}\boldsymbol{\omega} = [p,\, q,\, r]^{T}$ (which coincides with the rocket's angular velocity), the translation dynamics in the body frame are given by:

$$^{B}\dot{\mathbf{v}} = \frac{1}{m}\left({}^{B}\mathbf{F_g} + {}^{B}\mathbf{F_p} + {}^{B}\mathbf{F_a}\right) - \mathbf{S}(\boldsymbol{\omega})\ {}^{B}\mathbf{v}\,, \tag{3.25}$$

where $\mathbf{S}(\boldsymbol{\omega})$ is the skew-symmetric matrix applied to the angular velocity vector. Note that the term of the linear momentum derivative accounting for the mass variation is inside the thrust force, here defined as an external force. By substituting for the previously derived expressions for the external forces, the translational equations of motion, expressed in the body frame, are obtained in terms of body acceleration components,

$$\dot{u} = m^{-1} \cdot ({}^{B}F_{g_x} + {}^{B}F_{p_x} + {}^{B}F_{a_x}) - q\,w + r\,v\,, \tag{3.26a}$$

$$\dot{v} = m^{-1} \cdot ({}^{B}F_{g_y} + {}^{B}F_{p_y} + {}^{B}F_{a_y}) - r\,u + p\,w\,, \tag{3.26b}$$

$$\dot{w} = m^{-1} \cdot ({}^{B}F_{g_z} + {}^{B}F_{p_z} + {}^{B}F_{a_z}) - p\,v + q\,u\,, \tag{3.26c}$$

where it is necessary to take into account that the mass varies with time due to the consumption of the propellant, given by $\dot{m}$. In order to obtain the translational motion in the Earth-fixed inertial frame, the velocity vector in the body frame is transformed using the correspondent transformation matrix:

$$^{E}\mathbf{v} = \mathbf{R} \cdot {}^{B}\mathbf{v}. \tag{3.27}$$

3.6.2 Rotational Motion

The rotational dynamics in the body frame are described by Euler's equation for rigid bodies,

$$^{B}\mathbf{M} = \mathbf{J}\,\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{J}\,\boldsymbol{\omega}\,, \tag{3.28}$$

where $^{B}\mathbf{M}$ is the total external moment in the body frame and $\mathbf{J}$ is the inertia matrix. Following the axial symmetry assumption, the cross-products of inertia can be taken as zero, resulting in

$$\mathbf{J} = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix}\,. \tag{3.29}$$

By substituting the inertia matrix $\mathbf{J}$ and the external moment in the body frame $^{B}\mathbf{M}$ in (3.28), the rotational equations of motion, expressed in the body frame, are obtained in terms of the body angular acceleration components,

$$\dot{p} = {J_x}^{-1} \cdot \left({}^{B}M_{a_x} - q\,r\,(J_z - J_y) \right), \tag{3.30a}$$

$$\dot{q} = {J_y}^{-1} \cdot \left({}^{B}M_{a_y} + {}^{B}M_{p_y} - r\,p\,(J_x - J_z) \right), \tag{3.30b}$$

$$\dot{r} = {J_z}^{-1} \cdot \left({}^{B}M_{a_z} + {}^{B}M_{p_z} - p\,q\,(J_y - J_x) \right), \tag{3.30c}$$

in which the moments of inertia will change with the mass decrease throughout the burning phase.

As mentioned, the attitude of the rocket will be given by the Euler angles. Their derivatives behave according to the following equations [31]:

$$\dot{\phi} = p + (q\sin\phi + r\cos\phi)\tan\theta\,, \tag{3.31a}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi\,, \tag{3.31b}$$

$$\dot{\psi} = \frac{q\sin\phi + r\cos\phi}{\cos\theta}\,. \tag{3.31c}$$

By using the Euler angles a singularity may arise for $\theta = \pm\,\frac{\pi}{2}$, however, the way the reference frames were defined prevents the rocket to achieve this attitude when inside the admissible range of operation.

Chapter 4

The Rocket Model

4.1 Thrust-Vector-Controlled Rocket Model

Referring to the modelling procedure detailed in Chapter 3, a six degrees of freedom, three in position and three in rotation, non-linear model is derived for the case under study: a finless, thrust-vector-controlled rocket. The dynamics and kinematics were re-written using more compact notation, resulting in the following model description:

$$\begin{cases} \dot{\mathbf{P}} = \mathbf{R}\ {}^{B}\mathbf{v} \\ \dot{\mathbf{R}} = \mathbf{R}\,\mathbf{S}({}^{B}\boldsymbol{\omega}) \\ m\ {}^{B}\dot{\mathbf{v}} = -\mathbf{S}({}^{B}\boldsymbol{\omega})\, m\ {}^{B}\mathbf{v} + {}^{B}\mathbf{F} \\ \mathbf{J}\ {}^{B}\dot{\boldsymbol{\omega}} = -\mathbf{S}({}^{B}\boldsymbol{\omega})\,\mathbf{J}\ {}^{B}\boldsymbol{\omega} + {}^{B}\mathbf{M} \end{cases} \tag{4.1}$$

where $\mathbf{P}$ is now the position in the inertial frame.

At this point we recall the assumptions presented in the modelling procedures (Section 3.1) and the particular characteristics of the case under study in order to derive the explicit dynamics. Firstly, the rocket is assumed to be axially symmetric and to have an axially symmetric mass distribution. This allows not only to consider that the off-diagonal elements of the inertia matrix J are null but also that the values for the y and z axes are equivalent. In this way the inertia matrix is composed by a longitudinal term (J_l), aligned with the x-axis, and a transverse term (J_t), equal for the y and z axes. Secondly, rolling moment will not be caused by fins misalignment since they don't exist. In this way, it will be described by a disturbance factor (τ_r) and by the rolling moment caused by the reaction roll control system (μ_r). Ideally, the roll control system is able to counteract the disturbance factor and the rocket won't have rolling motion.

Taking this into account, the external forces and moments are given by:

$$
{}^{B}\mathbf{F} = \begin{bmatrix} -mg\, c_\theta c_\psi \,+\, T\, c_{\mu_1} c_{\mu_2} \,-\, \overline{q}\, C_A\, S \\ -mg\, (s_\phi s_\theta c_\psi \,-\, c_\theta s_\psi) \,-\, T\, c_{\mu_1} s_{\mu_2} \,+\, \overline{q}\, C_Y\, S \\ -mg\, (c_\phi s_\theta c_\psi \,+\, s_\phi s_\psi) \,-\, T\, s_{\mu_1} \,-\, \overline{q}\, C_N\, S \end{bmatrix} , \quad {}^{B}\mathbf{M} = \begin{bmatrix} \tau_r + \mu_r \\ -T\, s_{\mu_1}\, l \,+\, \overline{q}\, S\, d\, C_m \\ T\, c_{\mu_1} s_{\mu_2}\, l \,+\, \overline{q}\, S\, d\, C_n \end{bmatrix} \tag{4.2}
$$

where the aerodynamic coefficients are given by the previously presented expressions (Section 3.5). By substituting the external forces and moments (4.2) in the rocket model (4.1), its is possible to obtain the explicit dynamics and kinematics:

$$
\begin{cases}
\dot{u} = -g\, c_\theta\, c_\psi - \frac{\overline{q}}{m}\, S\, C_A + \frac{T}{m} c_{\mu_1}\, c_{\mu_2} - q\, w + r\, v \\
\dot{v} = -g\, (s_\phi\, s_\theta\, c_\psi - c_\phi\, s_\psi) + \frac{\overline{q}}{m}\, S\, C_Y - \frac{T}{m}\, c_{\mu_1}\, s_{\mu_2} - r\, u \\
\dot{w} = -g\, (c_\phi\, s_\theta\, c_\psi + s_\phi\, s_\psi) - \frac{\overline{q}}{m}\, S\, C_N - \frac{T}{m}\, s_{\mu_1} + q\, u \\
\dot{p} = {J_l}^{-1}\, (\tau_r + \mu_r) \\
\dot{q} = {J_t}^{-1}\, \overline{q}\, S\, d\, C_m - J_t\, T\, s_{\mu_1}\, l \\
\dot{r} = {J_t}^{-1}\, \overline{q}\, S\, d\, C_n + {J_t}^{-1}\, T\, c_{\mu_1}\, s_{\mu_2}\, l \\
\dot{\phi} = p + (q\, s_\phi + r\, c_\phi)\, t_\theta \\
\dot{\theta} = q\, c_\phi - r\, s_\phi \\
\dot{\psi} = \dfrac{q\, s_\phi + r\, c_\phi}{c_\theta}
\end{cases} \tag{4.3}
$$

As the goal of this work is to design an attitude controller, the translation kinematics in the inertial frame will not be needed for the control action. Later, the attitude controller can be implemented in the inner-loop of a guidance system, in which an outer-loop position controller determines the required attitude over time to satisfy a specific guidance law.

4.2 Preliminary Rocket Design

The goal of this work is to define the attitude controller design process as generically as possible, so that it can be easily adapted to any small scale rocket, in particular the one built by RED, with any given mission goal, specially in terms of nominal trajectory. Nevertheless, it is necessary to use a reference model in order to test and validate the proposed ADCS in simulation environment. In this section, the preliminary design for a future RED's rocket with Thrust Vector Control is presented.

We have seen in Section 2.1 that sounding rockets, namely student-built ones, tend to have a very brief burning phase with respect to the overall ascent trajectory. For student-built sounding rockets, due to the normally reduced apogee, this means a burning phase of a few seconds with a large thrust-to-weight ratio that rapidly accelerates the rocket. This implies that the period in which thrust vectoring is possible is also very limited and that stability must be ensured differently during the coast phase.

To tackle this limitation, one possibility is to design a rocket that relies on fins to stabilise the rocket during the coast phase, either in a passive or active way. Although it comes with limited use of thrust vectoring, it can still be a very effective alternative due to the extreme importance of stability during the burning phase, in which the rocket is still gaining velocity for the fins to have sufficient stabilizing capacity and disturbances, for instance wind gusts, can have a greater impact. The other alternative is to design a rocket in such a way that the burning phase ending almost coincides with apogee and, consequently, with the beginning of the parachute recovery phase. In order to achieve this, it is necessary to have a close to unity thrust-to-weight ratio with a gradual thrust decrease, so that the velocity at burnout is low, minimizing the distance travelled between burnout and apogee and allowing a safe parachute deployment. In this way, thrust vectoring can be used to stabilize and guide the rocket throughout the entire ascent trajectory, making it unnecessary to use aerodynamic fins. Consequently, the rocket would be naturally unstable.

Taking all this into consideration, the second alternative was chosen, not only because it allows a larger period for the controller to be used but also because it represents a closer scenario to a micro-launcher with orbit insertion capacity, a long term goal of RED in which TVC will be essential. In addition, the natural instability of the rocket creates a more demanding scenario for the control system, while giving it more authority at the same time. As for the TVC technology, the movable nozzle was used as reference since it grants the best performance in solid motors (as seen in Section Y) and the actuation is more easily modelled as opposed to jet vanes. This implies that only pitch and yaw can be controlled by thrust vectoring and that an additional reaction control system, likely based on gas thrusters, has to be used for roll control. The task of the reaction control system is to prevent the appearance of rolling motion and its design is outside the scope of this thesis.

The preliminary design was an iterative process done with a realistic mass allocation, based on typical sounding rockets, always having in mind its feasibility. It was intended that the dimensions wouldn't considerably deviate from the current rockets built by RED, so that it can be used as a preliminary design for a vehicle to be built in the near future, with the goal of testing TVC technology. Figure 4.1 shows a sketch of the preliminary rocket design in the open source simulator -*OpenRocket*- where the different sections of the rocket are identified, while Tab. 4.1 displays the main characteristics of the rocket and Tab. 4.2 shows the total mass distribution in percentage.

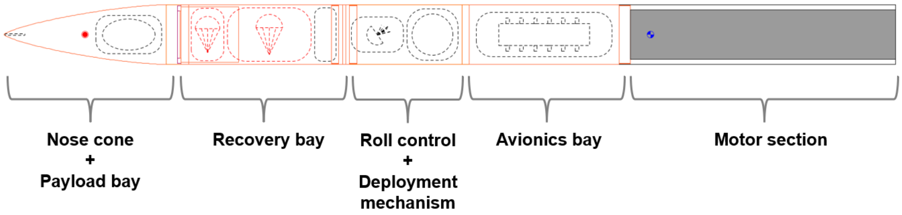


Figure 4.1: Preliminary rocket design.

Table 4.1: Main rocket characteristics.

Total mass	82.9 kg
Dry mass	40.0 kg
Length	3.57 m
Max diameter	24 cm

Table 4.2: Mass distribution in percentage.

Motor section	71.7 %
Avionics bay	9.0 %
R.C. + Deployment mech.	5.6 %
Recovery bay	8.3 %
Nose cone + Payload bay	5.4 %

As for the motor, solid propulsion was selected since it is the technology that RED currently relies on, meaning that thrust cannot be controlled. This is a limitation for the controller design and one can argue that future implementations with hybrid or liquid engines could rely on the same type of controller with an additional degree of freedom that simplifies the overall control task. Knowing that thrust cannot be controlled, a tailored thrust curve capable of meeting the aforementioned mission goals had to be obtained in the solid motor design process by selecting an appropriate grain geometry, nozzle expansion ratio and propellant mass. This being said, the end-burn configuration was selected for the grain geometry, with a decreasing burning surface area to allow a gradual combustion chamber pressure decrease and, consequently, a gradual thrust decrease. Figure 4.2 is a schematics of the grain geometry and burning direction.

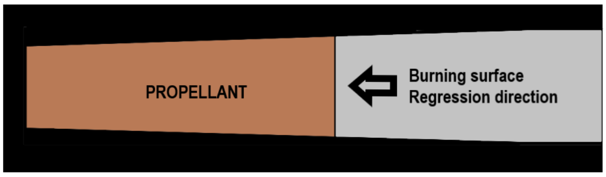


Figure 4.2: Motor schematics.

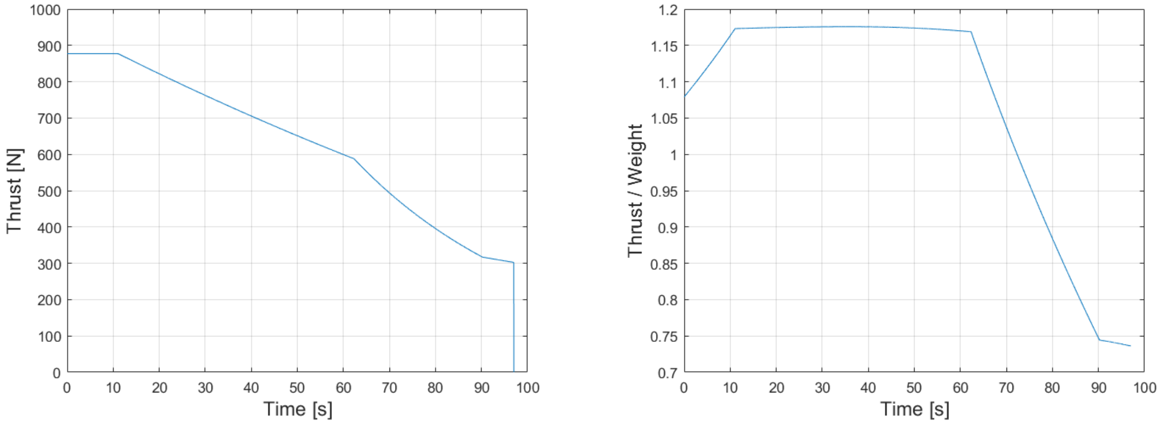
This configuration was achieved by iteratively testing different alternatives using the described propulsion model (Section 3.4). Figure 4.3 presents the obtained thrust curve and the resultant thrust-to-weight ratio, both as a function of time throughout the burning phase.

Table 4.3 details the main motor characteristics, both in design parameters and performance.

Table 4.3: Main motor characteristics.

Propellant	KNO_3 / Sorbitol (KNSB)
Propellant mass	42.9 kg
Nozzle expansion ratio	14.154
Hardware mass	12 kg
Total impulse	61319 Ns
Max thrust	877 N
Max pressure	105 Bar
Burn time	97 s

Even though the rocket is naturally unstable, it is still possible to simulate a disturbance-free, fully



(a) Thrust

(b) Thrust-to-Weight ratio

Figure 4.3: Thrust and Thrust-to-weight ratio.

vertical flight without the need of control action. The simulation results, obtained with the developed simulation model, are present in Tab. 4.4.

Table 4.4: Simulation results for vertical trajectory.

Apogee	4945 m
Max velocity	82 m/s
Max acceleration	1.7 m/s^2
Time to apogee	100 s
Velocity at burnout	27 m/s
Vertical distance after burnout	38 m

These results confirm that the mission requirements are achieved with the preliminary rocket design. The period of time and distance travelled between burnout and apogee are very reduced, and the velocity at burnout is inside the the safety limits for nose cone ejection and parachute deployment, according to the current recovery systems designed by RED. Additionally, the reduced maximum velocity and acceleration simplify the structural design requirements.

Overall, the preliminary design allows a simplified implementation which enables RED to have, in the short-term, a testing platform for thrust vector control systems. This vehicle can also be seen as a scaled down first-stage of a rocket capable of reaching space, in which atmospheric flight represents a very demanding scenario for the controller.

4.2.1 Nominal Trajectory

The nominal trajectory is the one that the vehicle will follow under nominal conditions. However, in reality, there will always be perturbations and model uncertainties which cause the vehicle to deviate from the nominal condition. The task of the attitude controller is to keep the attitude of the vehicle as close to the desired nominal evolution as possible, by controlling the system inputs.

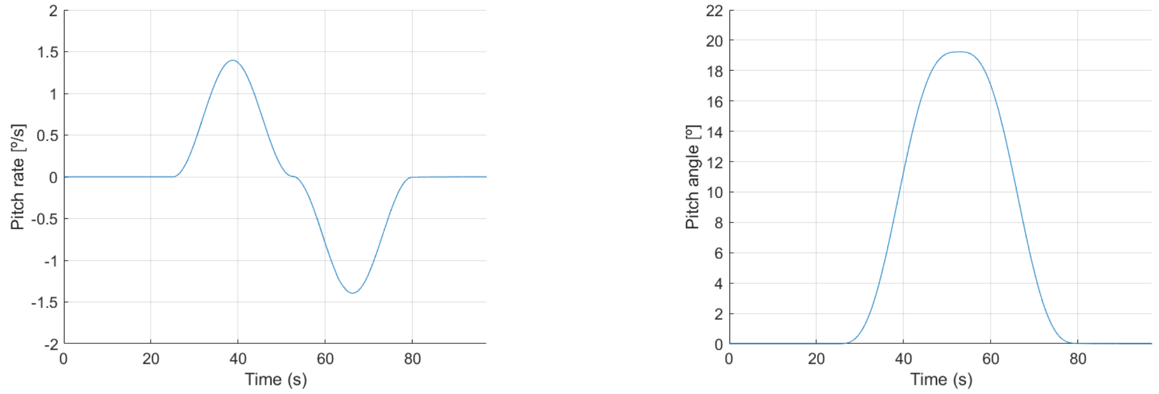
There are numerous possibilities for the nominal attitude evolution that depend on the mission goals and are limited by the actuation nature of the system and overall dynamics. As thrust approximately follows a predefined curve and the TVC system can only influence the pitch and yaw axes, not all trajectories can be performed.

Since the system is naturally unstable, it is necessary to find the time evolution of the nominal control inputs that allows the rocket to nominally follow the trajectory, defined by an attitude reference over time. Ideally, the nominal trajectory should be obtained through a trim condition, which consists in finding constant system input values that would make the first-order derivatives equal to zero. As the system would be in an equilibrium (or trim) condition, the task of the controller in keeping the nominal trajectory would be much easier. However, the dependency on varying parameters makes it impossible to achieve said condition. Even for a vertical trajectory, where the control inputs would be constant and equal to zero, the longitudinal acceleration cannot be set to zero due to the fact that thrust is not controllable.

This being said, the nominal control inputs evolution could be found by solving the non-linear equations of the system over time such that the attitude reference is correctly followed. However, this is a very hard task. One way of tackling this problem is to design a stable controller that allows the rocket to nominally follow the attitude reference and save the actuation to then use in real-time as the nominal control input. As long as the model is valid and the varying parameters are approximately known, the stable controller can be designed in simulation environment to generate the nominal actuation, and then, the real-time closed-loop controller will be able to correct deviations from the nominal trajectory by calculating the necessary adjustments to the nominal control input value. In this way, the selected controller was a proportional, integral and derivative controller (PID) that in simulation environment can stabilize the model and make the rocket nominally follow the attitude reference.

Regarding the attitude reference selection, there are different possibilities that depend on the mission objective. The goal of this work is to define the design process in a generic way so that the controller can be re-adapted to different nominal trajectories. As the objective of the reference rocket is to have low final velocity and to keep a vertical trajectory close to apogee (Section 4.2), two alternatives are identified: a strictly vertical trajectory in which the nominal control inputs are simply zero, and a varying pitch trajectory, in which the controller restricts the motion to the pitch plane (yaw equal to zero) and makes the rocket deviate from the vertical and later recover it, to ensure that the apogee is reached further away from the launch site and increase safety. Regarding the second alternative, in case of recovery failure, the rocket wouldn't fall close to the launch site and the trajectory can be selected taking into account the wind profile so that the rocket touches down close to the launch site in case of a successful recovery.

Given that the second alternative is more demanding for the controller and also contains sections where the trajectory is vertical, it was selected as reference for the attitude controller design. In this way, the pitch reference was defined in terms of pitch rate, with the aim of ensuring a smooth transition between the section where the pitch angle is increasing and the one where it is decreasing to restore the vertical orientation. Taking this into account, the pitch rate was defined by two symmetric half periods of a 27.5 Hz frequency sinusoidal wave with an amplitude of $0.7\,^\circ$ /s, starting 25 seconds after launch. Figure 4.4 shows the pitch rate over time for the nominal trajectory and the resulting pitch angle. By using this pitch rate reference, the maximum rate of change in pitch angle is equal to 1.4 ° /s, and the maximum pitch angle is approximately 19.2 degrees.



(a) Pitch rate (b) Pitch angle

Figure 4.4: Nominal Pitch rate and angle over time.

To achieve this nominal pitch evolution, the PID controller used in simulation calculated the necessary nominal evolution of the pitch control input (μ_{1_0}), represented in Figure 4.5. Since the nominal trajectory has no motion in the yaw plane, the nominal value of the yaw control input (μ_{2_0}) is zero.

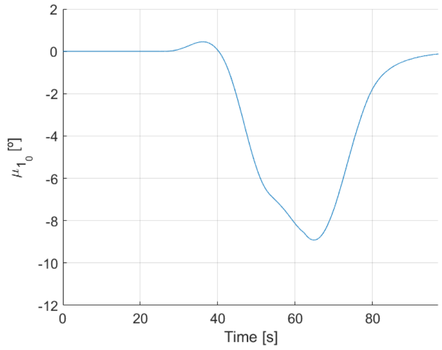
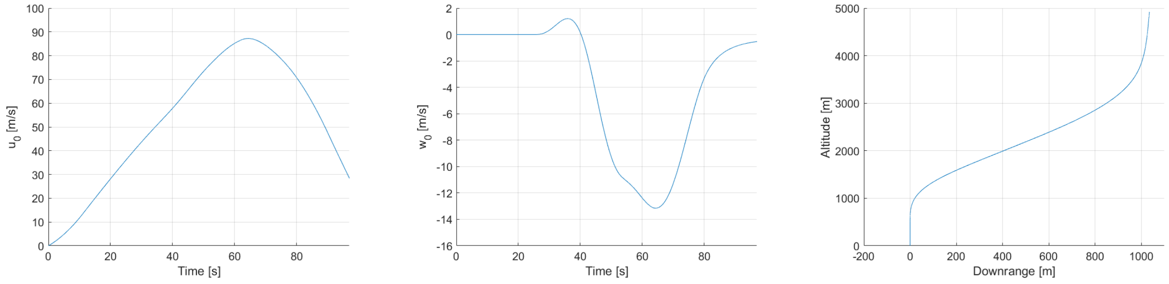


Figure 4.5: Nominal pitch control input over time.

Figure 4.6 presents the nominal evolutions of the remaining variables of interest for the pitch plane motion, including the resulting trajectory (Figure 4.6 (c)).



(a) Longitudinal velocity (u_0) (b) Normal velocity (w_0) (c) Trajectory

Figure 4.6: Nominal pitch plane motion variables.

4.3 Model Linearization

In order to design an optimal controller and use all the tools from linear control analysis, it is necessary to obtain a linear version of the model and respective state-space representation. The non-linear dynamics can be linearized at equilibrium points of the system by using a taylor series expansion, considering small perturbations. For the case of a rocket, conditions change considerably throughout the flight, mainly due to the accelerated motion caused by the propulsive force, hence, it is not advisable, nor correct, to choose a single equilibrium point to linearize the system. Instead, a nominal trajectory is selected and the system is linearized at multiple operating points that represent the dynamics of the system at different regimes. The selected trajectory imposes the nominal values of system states and inputs, and the outcome is a linear time-variant system.

Firstly, the following variable transformations are defined: $\delta x = x - x_0$ and $\delta u = u - u_0$; where x is an arbitrary system state and u is a system input, x_0 and u_0 its nominal values, and $\delta\, x$ and δu are small perturbations around the nominal values. By using the variable transformation in the non-linear differential equations of the system (denoted by $\dot{x} = f(x, u)$), generically, we have that

$$\delta\dot{x} = f(x, u) - f(x_0, u_0) = f(x_0 + \delta x, u_0 + \delta u) - f(x_0, u_0)\,. \tag{4.4}$$

Using the Taylor series expansion of $f(x_0 + \delta x, u_0 + \delta u)$ around (x_0, u_0), and neglecting the higher-order terms, we obtain

$$\delta\dot{x} = f(x_0, u_0) + \left.\frac{\partial f}{\partial x}\right|_{x_0,u_0} \cdot \delta x + \left.\frac{\partial f}{\partial u}\right|_{x_0,u_0} \cdot \delta u - f(x_0, u_0) = \left.\frac{\partial f}{\partial x}\right|_{x_0,u_0} \cdot \delta x + \left.\frac{\partial f}{\partial u}\right|_{x_0,u_0} \cdot \delta u\,. \tag{4.5}$$

4.3.1 Model Simplification

When obtaining the linear version of the system, it is advantageous to consider some assumptions. First of all, we recall that the aim of this work is to develop an attitude controller that relies on a TVC system to control pitch and yaw. Roll cannot be controlled by the actuation system and so it is considered that an additional, and sufficiently rapid, roll rate control system is implemented. This means that the roll rate in the body frame (p) can be taken as zero, reducing the order of the system. Secondly, it is assumed a no wind condition, which means that the linear velocities in the body frame coincide with the velocities with respect to the atmosphere and can be used in the aerodynamic forces, moments and angles expressions. Finally, for a first approach, the actuators' dynamics are not included in the linear model, which means that the controller won't take them into account. However, the performance of the controller will be analysed with the inclusion of said dynamics and the design parameters can be tuned accordingly. If deemed necessary, the actuator dynamics will be taken into account when deriving the linear model for controller design.

These assumptions, to be used in the linearization, result in the following simplified version of the rocket model, in terms of explicit dynamics and kinematics:

$$\begin{cases} \dot{u} = -g\, c_\theta\, c_\psi - \frac{\overline{q}}{m}\, S\, C_A + \frac{T}{m} c_{\mu_1}\, c_{\mu_2} - q\, w + r\, v \\ \dot{v} = -g\,(s_\phi\, s_\theta\, c_\psi - c_\phi\, s_\psi) + \frac{\overline{q}}{m}\, S\, C_Y - \frac{T}{m}\, c_{\mu_1}\, s_{\mu_2} - r\, u \\ \dot{w} = -g\,(c_\phi\, s_\theta\, c_\psi + s_\phi\, s_\psi) - \frac{\overline{q}}{m}\, S\, C_N - \frac{T}{m}\, s_{\mu_1} + q\, u \\ \dot{q} = {J_t}^{-1}\, \overline{q}\, S\, d\, C_m - {J_t}^{-1}\, T\, s_{\mu_1}\, l \\ \dot{r} = {J_t}^{-1}\, \overline{q}\, S\, d\, C_n - {J_t}^{-1}\, T\, c_{\mu_1}\, s_{\mu_2}\, l \\ \dot{\phi} = (q\, s_\phi + r\, c_\phi)\, t_\theta \\ \dot{\theta} = q\, c_\phi - r\, s_\phi \\ \dot{\psi} = \dfrac{q\, s_\phi + r\, c_\phi}{c_\theta} \end{cases} \tag{4.6}$$

4.3.2 Generic linearization

By applying the described linearization method to the simplified non-linear system, a linear time-variant system in the perturbation domain is obtained, that can be represented in the state-space form:

$$\delta\mathbf{x} = [\delta u \;\; \delta v \;\; \delta w \;\; \delta q \;\; \delta r \;\; \delta\phi \;\; \delta\theta \;\; \delta\psi]^T \ , \tag{4.7a}$$

$$\delta\mathbf{u} = [\delta\mu_1 \;\; \delta\mu_2]^T \ , \tag{4.7b}$$

$$\delta\dot{\mathbf{x}}(t) = \mathbf{A}(t) \cdot \delta\mathbf{x}(t) + \mathbf{B}(t) \cdot \delta\mathbf{u}(t) \, , \tag{4.7c}$$

where $\mathbf{A}(t)$ and $\mathbf{B}(t)$ are the state-space matrices given by the first-order Taylor derivatives with respect to system states and inputs, respectively, calculated at the operating points.

Before defining the nominal trajectory, the $\mathbf{A}(t)$ and $\mathbf{B}(t)$ system matrices can be generically calculated by applying the described linearization technique to the simplified non-linear model (4.6), assuming generic nominal values. The result is the following:

$$\mathbf{A}(t) = \begin{bmatrix} 0 & a_{12} & a_{13} & a_{14} & a_{15} & 0 & a_{17} & a_{18} \\ a_{21} & a_{22} & 0 & 0 & a_{25} & a_{26} & a_{27} & a_{28} \\ a_{31} & 0 & a_{33} & a_{34} & 0 & a_{36} & a_{37} & a_{38} \\ a_{41} & 0 & a_{43} & a_{44} & 0 & 0 & 0 & 0 \\ 0 & a_{52} & 0 & 0 & a_{55} & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{64} & a_{65} & a_{66} & a_{67} & 0 \\ 0 & 0 & 0 & a_{74} & a_{75} & a_{76} & 0 & 0 \\ 0 & 0 & 0 & a_{84} & a_{85} & a_{86} & a_{87} & 0 \end{bmatrix} , \quad \mathbf{B}(t) = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & 0 \\ b_{41} & 0 \\ b_{51} & b_{52} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} , \tag{4.8}$$

with,

$$
\begin{aligned}
a_{12} &= r_0 \\
a_{13} &= -q_0 \\
a_{14} &= -w_0 \\
a_{15} &= v_0 \\
a_{17} &= g\, s_{\theta_0}\, c_{\psi_0} \\
a_{18} &= g\, c_{\theta_0}\, s_{\psi_0} \\
a_{21} &= -r_0 \\
a_{22} &= \frac{\overline{q}\, S\, C_{Y_\beta}}{m\,\sqrt{1-\frac{v_0^2}{{V_0}^2}}\, V_0} \\
a_{25} &= -u_0 \\
a_{26} &= -g\,(c_{\phi_0}\, s_{\theta_0}\, c_{\psi_0} + s_{\phi_0}\, s_{\psi_0}) \\
a_{27} &= -g\,(s_{\phi_0}\, c_{\theta_0}\, c_{\psi_0} - c_{\phi_0}\, s_{\psi_0}) \\
a_{28} &= g\,(s_{\phi_0}\, s_{\theta_0}\, s_{\psi_0} + c_{\phi_0}\, c_{\psi_0}) \\
a_{31} &= q_0 + \frac{\overline{q}\, S\, C_{N_\alpha}\, w_0}{m\,({u_0}^2 + {w_0}^2)} \\
a_{33} &= -\frac{\overline{q}\, S\, C_{N_\alpha}\, u_0}{m\,({u_0}^2 + {w_0}^2)} \\
a_{34} &= u_0 \\
a_{36} &= -g\,(-s_{\phi_0}\, s_{\theta_0}\, s_{\psi_0} + c_{\phi_0}\, s_{\psi_0}) \\
a_{37} &= -g\, c_{\phi_0}\, c_{\theta_0}\, c_{\psi_0} \\
a_{38} &= -g\,(-c_{\phi_0}\, s_{\theta_0}\, s_{\psi} + s_{\phi_0}\, c_{\psi_0}) \\
a_{41} &= \frac{\overline{q}\, S\, d\, SM\, C_{N_\alpha}\, w_0}{J_t\,({u_0}^2 + {w_0}^2)} \\
a_{43} &= -\frac{\overline{q}\, S\, d\, SM\, C_{N_\alpha}\, u_0}{J_t\,({u_0}^2 + {w_0}^2)} \\
a_{44} &= \frac{\overline{q}\, S\, d^2\,(C_{m_q} + C_{m_{\dot{\alpha}}})}{2\, J_t\, V_0} \\
a_{52} &= -\frac{\overline{q}\, S\, d\, SM\, C_{Y_\beta}}{J_t\, V_0\,\left(1-\frac{v_0^2}{{V_0}^2}\right)^{\frac{1}{2}}} \\
a_{55} &= \frac{\overline{q}\, S\, d^2\,(C_{n_r} + C_{n_{\dot{\beta}}})}{2\, J_t\, V_0} \\
a_{64} &= s_{\phi_0}\, t_{\theta_0} \\
a_{65} &= c_{\phi_0}\, t_{\theta_0} \\
a_{66} &= (q_0\, c_{\phi_0} - r_0\, s_{\phi_0})\, t_{\theta_0} \\
a_{67} &= \frac{q_0\, s_{\phi_0} + r_0\, c_{\phi_0}}{c^2{\theta_0}} \\
a_{74} &= c_{\phi_0} \\
a_{75} &= -s_{\phi_0} \\
a_{76} &= -q_0\, s_{\phi_0} - r_0\, c_{\phi_0} \\
a_{84} &= \frac{s_{\phi_0}}{c_{\theta_0}} \\
a_{85} &= \frac{c_{\phi_0}}{c_{\theta_0}} \\
a_{86} &= \frac{q_0\, c_{\phi_0} - r_0\, s_{\phi_0}}{c_{\theta_0}} \\
a_{87} &= \frac{(q_0\, s_{\phi_0} + r_0\, c_{\phi_0})\, s_{\theta_0}}{c^2{\theta_0}} \\
b_{11} &= -\frac{T}{m}\, s_{\mu_{1_0}}\, c_{\mu_{2_0}} \\
b_{12} &= -\frac{T}{m}\, c_{\mu_{1_0}}\, s_{\mu_{2_0}} \\
b_{21} &= \frac{T}{m}\, s_{\mu_{1_0}}\, s_{\mu_{2_0}} \\
b_{22} &= -\frac{T}{m}\, c_{\mu_{1_0}}\, c_{\mu_{2_0}} \\
b_{31} &= -\frac{T}{m}\, c_{\mu_{1_0}} \\
b_{41} &= -\frac{T\, l}{J_t}\, c_{\mu_{1_0}} \\
b_{51} &= -\frac{T\, l}{J_t}\, s_{\mu_{1_0}}\, s_{\mu_{2_0}} \\
b_{52} &= \frac{T\, l}{J_t}\, c_{\mu_{1_0}}\, c_{\mu_{2_0}}
\end{aligned}
\tag{4.9}
$$

where V_0 is the norm of the linear velocity vector in the body frame, and the time dependency of the variables was omitted to simplify the notation.

As mentioned, the obtained system is a time-varying one. This can be seen in the state-space matrices, where a dependency on time-varying parameters and nominal values is present (the time function symbol, (t), is omitted to avoid unnecessary repetition). When calculating the first-order Taylor derivatives, all these parameters were assumed to have no dependency on system states. In reality, some of the parameters depend on system states, such as dynamic pressure and aerodynamic coefficients. However, these dependencies are secondary and the parameters can be assumed to be constant at each operating point and only to vary independently between them.

The time evolution of the nominal values is imposed by the nominal trajectory. Deviations from the nominal condition are described by the linear system in the perturbation domain.

The time-varying parameters will also follow a nominal evolution that depends on the selected trajectory or is directly a function of time. However, the dependency on the trajectory is not direct, and the system itself has no knowledge on how these quantities change when a deviation from the nominal trajectory occurs. A list on the time-varying parameters and their dependencies follows:

- **Dynamic pressure ($\overline{q}$):** The dynamic pressures depends on the total relative velocity of the vehicle and on the air density. On the other hand, air density can be expressed as a function of altitude using the reference atmospheric model, and so, dynamic pressure is obtained by knowing the total velocity and altitude of the rocket.

- **Gravitational acceleration (g):**The gravitational acceleration can be expressed as a function of the altitude, using the assumed flat Earth model.
- **Aerodynamic coefficients and derivatives (C_A,C_{N_α}, C_{Y_β}, C_{m_q}, $C_{m_{\dot{\alpha}}}$, C_{n_r}, $C_{n_{\dot{\beta}}}$):** The aerodynamic coefficients and derivatives mainly depend on the aerodynamic angles and the rocket's velocity expressed in Mach number.
- **Thrust (T), mass (m) and inertia (J):** All these quantities can be considered to be directly a function of time. Thrust will approximately follow a known curve (Thrust curve), mass will also change according to a known mass flow rate, and inertia varies due to the known change in mass.
- **Static margin (Sm):** As previously explained, the static margin is a function of the center of pressure and the center of mass. The center of pressure depends on the rocket's velocity and aerodynamic angles, and the center of mass approximately follows a known time evolution related with the propellant consumption.

When designing a controller, it might be useful to take a linear parameter varying approach, where the selected state-space representation will depend on the value of the time-varying parameters, instead of referring to the expected nominal evolution. This alternative implies that there is the capacity to measure on-board the quantities that influence the varying parameters in order to estimate them.

4.3.3 Linearization Results for the Reference Rocket

The generic state-space representation can be applied to the reference rocket detailed in Section 4.2, using the defined nominal evolution of system states, parameters and inputs.

State-space Representation Simplification

Looking at the nominal trajectory, it is possible to separate two distinct sections: a first section up to $t = 25$ s in which motion is strictly vertical, and a second section up to burnout in which pitch is varying. Starting by the varying pitch section, it is defined by the following conditions for the nominal system states and inputs:

- **Yaw and Roll equal to zero** ($\phi_0 = 0$ and $\psi_0 = 0$);
- **Lateral velocity equal to zero** ($v_0 = 0$);
- **yaw rate equal to zero** ($r_0 = 0$);
- **yaw control input equal to zero** ($\mu_{2_0} = 0$).

By substituting these conditions in the generic state-space matrices (4.8 and 4.9), a simplified state-space representation is obtained in which the longitudinal and lateral modes are decoupled and can be represented by distinct state-spaced matrices (present in Appendix A), with the following state and input vectors

$$\delta \mathbf{x_{lon}} = [\delta u \;\; \delta w \;\; \delta q \;\; \delta\theta]^T \,, \qquad \delta u_{lon} = \delta\mu_1 \,, \tag{4.10a}$$

$$\delta \mathbf{x_{lat}} = [\delta v \;\; \delta r \;\; \delta\phi \;\; \delta\psi]^T \,, \qquad \delta u_{lat} = \delta\mu_2 \,. \tag{4.10b}$$

In the vertical section, we have the following conditions for the nominal system states and inputs:

- **Roll, pitch and yaw equal to zero** ($\phi_0 = \theta_0 = \psi_0 = 0$);
- **Normal velocities equal to zero** ($v_0 = w_0 = 0$);
- **Pitch and yaw rates equal to zero** ($q_0 = r_0 = 0$);
- **Control inputs equal to zero** ($\mu_{1_0}\mu_{2_0} = 0$).

This further simplifies the system, with the state-space matrices also being present in Appendix A. The state-space representation no longer includes the longitudinal velocity (δu) nor the roll angle ($\delta\phi$) as states, which reduces the order of the system by two, given by the following state and input vectors

$$\delta \mathbf{x_{lon}} = [\delta w \;\; \delta q \;\; \delta\theta]^T \,, \qquad \delta u_{lon} = \delta\mu_1 \,, \tag{4.11a}$$

$$\delta \mathbf{x_{lat}} = [\delta v \;\; \delta r \;\; \delta\psi]^T \,, \qquad \delta u_{lat} = \delta\mu_2 \,. \tag{4.11b}$$

Considering the assumption on axial symmetry of the vehicle, the decoupled modes for the vertical section are equivalent since the aerodynamic properties are homogeneous in both planes. In this way, when designing the controller it is possible to only refer to one of the modes for the vertical section of the trajectory, composed by three state variables (4.11).

Modal Analysis

Before designing a controller, it is essential to understand the location of the system poles and zeros, in order to derive the open-loop stability of the system. Since a linear time-variant system has been obtained, the location of poles and zeros will depend on the operating point that is being considered, and this analysis cannot give mathematical proof of stability. Nevertheless, a modal analysis for the different operating points of the nominal trajectory, depending on time since launch, is carried out to understand the behaviour of the systems throughout the flight. By using the previously presented nominal values for the system states and inputs taken at the operating points, and substituting in the state-space matrices, the poles and zeros of the system can be easily computed.

Firstly, the poles for the vertical section are computed using the correspondent state-space representation (A.2). The equivalence between the lateral and longitudinal modes is verified, with the poles being displayed in Fig. 4.7. The color of the poles goes from blue to green with the increase in time.

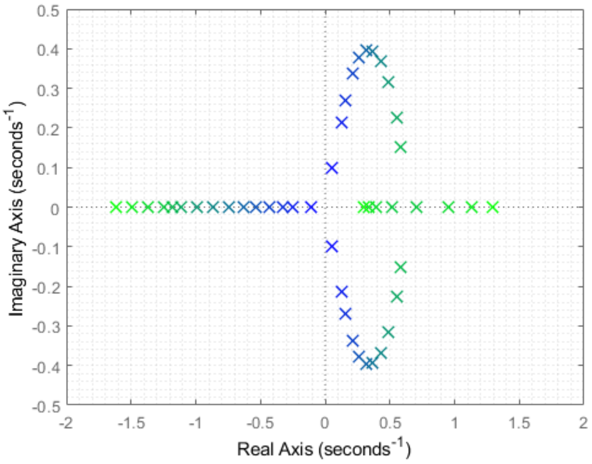
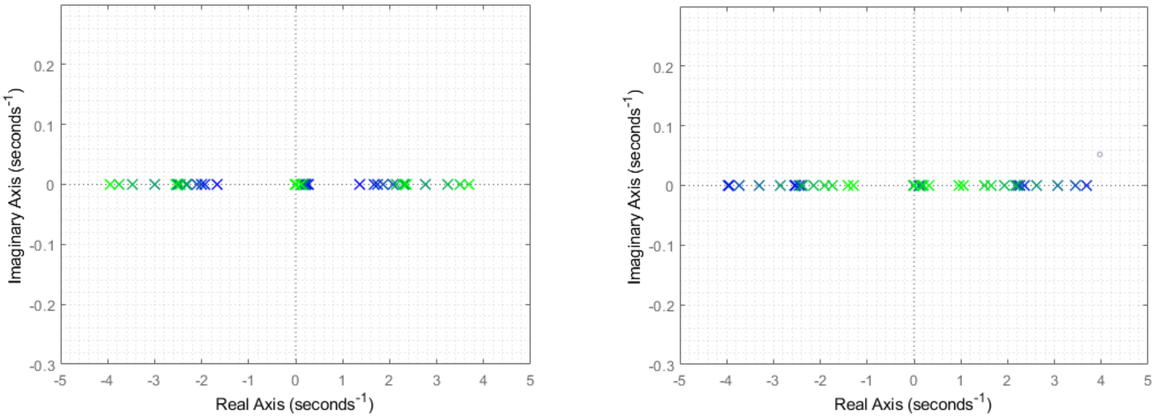


Figure 4.7: Poles for the vertical section (up to $t = 25$ s).

The equivalence between the lateral and longitudinal modes is determined by the existence of three double poles for each operating point, one for each equivalent state variable. During the vertical part of the nominal trajectory two distinct types of pole distribution are identified: One composed by a stable real pole and two unstable conjugate complex poles, and another composed by one stable real pole and two unstable real poles (one of them closer to the origin). The first type starts at launch and the transition occurs between 17 and 18 seconds after launch. This evolution is seen by the arc formed by the unstable complex poles which eventually degenerate into two unstable real poles. As time passes and the velocity of the vehicle increases, the magnitude of the poles also increases which means that the system becomes faster.

At $t = 25$ s the vehicle starts the pitch manoeuvre, which means that the decoupled modes are no longer equivalent and that each mode has an additional state variable (as seen in 4.10). However, after analysing the poles distribution, it is verified still similar behaviour between both modes, hence only the pole distribution for the longitudinal mode is displayed as it will be the most important mode to control for a pitch manoeuvre. Figure 4.8 presents the pole distribution for the longitudinal mode from the start of the pitch manoeuvre up to burnout, divided into two parts.



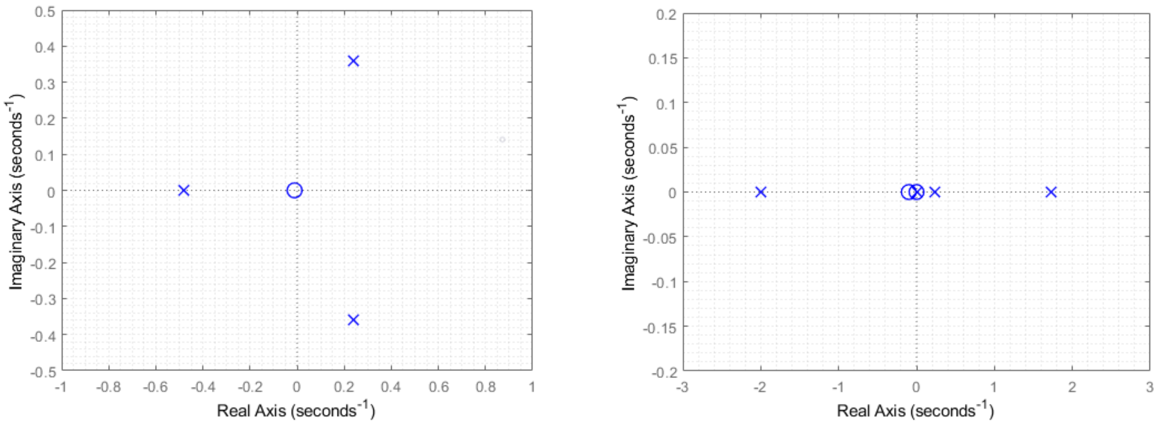
(a) From $t = 25$ s up to $t = 60$ s

(b) From $t = 60$ s up to $t = 95$ s

Figure 4.8: Poles for the pitch manoeuvre divided into two intervals.

For this section of the trajectory it is observed that the distribution of the poles is of the same type as the one reached at the end of the vertical section, one stable real pole and two unstable real poles, with an additional pole extremely close to the origin. As time passes, the magnitude of the two poles further away from the origin (one stable and one unstable) keeps increasing, while the pole at the origin almost doesn't move and the remaining unstable pole approaches the origin. This behaviour continues up to $t \approx 60$s, when velocity reaches a maximum value. After that, and as velocity decreases, the poles start moving in the opposite direction.

So far, only the location of the poles has been discussed. Regarding the zeros, by defining the yaw and pitch angles as the outputs of the system, it is possible to obtain its location. It was verified that the zeros of the system, for both modes and for the entire trajectory, are located in the real axis, ranging from approximately $-0.3\,\text{s}^{-1}$ to the origin. Figure 4.9 shows the poles and zeros for both types of distributions as exemplification for the longitudinal mode (the type of distribution is equivalent for the lateral mode with only the exact position changing).



(a) $t = 5\,\text{s}$

(b) $t = 30\,\text{s}$

Figure 4.9: Pole-zero map.

During the vertical part of the trajectory there is only one zero located in the positive real axis. During the pitch manoeuvre, it is noted that there is an additional zero cancelling the pole at the origin.

After performing the modal analysis some important conclusions can be made. Firstly, the system is naturally unstable, which was expected due to negative static stability margin caused by the absence of aerodynamic fins. Secondly, in the first seconds after launch the system has an unstable oscillatory behaviour indicated by the positive complex poles, after which all poles of the system are real. Finally, it is concluded that the velocity of the rocket is a driving factor for the poles' location, causing very distinct distributions throughout the flight. Moreover, at higher velocities the system is seen to have higher magnitude poles and hence faster dynamics.

Chapter 5

Attitude Control System Design

With the non-linear model for the dynamics and kinematics of the rocket, as well as its linear, time-varying, state-space representations, already derived, it is possible to proceed with the attitude control system design.

5.1 Controllability

Before designing an attitude controller based on the derived linear time-varying system, it is essential to verify its controllability. Analysing the controllability allows to determine whether the state of a dynamic system can be transferred from the initial state to an (arbitrary) final state in finite time by suitably adjusting the system input. In other words, a stabilizing state feedback law can only be obtained if the system is controllable.

As an initial approach, in order to determine the controllability of the obtained linear time-varying system, the theorem for the time-invariant case may be applied to the state-space realizations at each operating point. Although this method gives no controllability proof for the time-varying dynamics, it is assumed that if the system is controllable in all operating points, it is controllable as long as the selected operating points correctly represent the entire dynamics of the system. This assumption needs to be checked when implementing the controller in the complete non-linear model.

This being said, the controllability theorem for the time-invariant case states that the system $\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u}$ is controllable if and only if the rank $r(\mathcal{C})$ of the controllability test matrix

$$\mathcal{C} = \begin{bmatrix} \mathbf{B} & \mathbf{A}\,\mathbf{B} & ... & \mathbf{A}^{n-1}\,\mathbf{B} \end{bmatrix} \tag{5.1}$$

is equal to n, the order of the system [34].

5.1.1 Verification for the Nominal Trajectory

By applying the controllability theorem to the previously shown linearization points for the vertical section of the nominal trajectory, it was verified that for all points the system is controllable, meaning that the rank of the $\mathcal{C}$ matrix is equal to the order of the system (6). Regarding the varying pitch section, the rank of the $\mathcal{C}$ matrix was found to be equal to 7, one degree less than the order of the system (8), which means that one of the system states is not controllable. The uncontrollable state is roll ($\delta\phi$), which was expected since roll is meant to be controlled by an additional actuation system. Although uncontrollable, the pole associated with roll is the one extremely close to the origin and has an associated zero cancelling it. This mean that its dynamics might be discarded and the controller for the lateral mode during the pitch manoeuvre can be designed considering a reduced version of the system that doesn't include roll.

5.2 Linear Quadratic Regulator (LQR)

After obtaining the state-space representation of the system, and verifying its controllability, it is possible to design attitude controllers using linear and optimal control techniques. A widely used linear optimal control technique is the linear quadratic regulator or LQR.

The linear quadratic regulator is an optimal control technique that applies to linear systems and aims to find the optimal gain matrix $(\mathbf{k})$ for the linear control law

$$\mathbf{u} = -\mathbf{k}\,\mathbf{x} \tag{5.2}$$

that minimizes a specific cost function. The cost function J is expressed as the integral of a quadratic form in the state x plus a second quadratic form in the control u:

$$J = \int_t^T \left[\,\mathbf{x}'(\tau)\,\mathbf{Q}\,\mathbf{x}(\tau) + \mathbf{u}'(\tau)\,\mathbf{R}\,\mathbf{u}(\tau)\,\right] d\tau \tag{5.3}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are symmetric matrices, with $\mathbf{Q}$ being positive semi-definite and $\mathbf{R}$ positive definite. The $\mathbf{Q}$ and $\mathbf{R}$ matrices are respectively called state weighting matrix and control weighting matrix, and represent the design degree of freedom for the control problem.

In the cost function, the quadratic form, $\mathbf{x}'\mathbf{Q}\mathbf{x}$, represents a penalty on the deviation of the state, $\mathbf{x}$, from the origin, and the term, $\mathbf{u}'\mathbf{R}\mathbf{u}$, represents the cost of control. In this way, $\mathbf{Q}$ specifies the importance of the various components of the state vector relative to each other, while $\mathbf{R}$ imposes a limit to the magnitude of the control signal $\mathbf{u}$ in each of the system inputs. It is important to note that the objective of the regulator is to keep the states close to the origin, which applied to the case under study means to minimize the perturbations, hence keeping the total value of state variables close to the nominal condition.

Regarding the limits of the cost function integral, t is defined as the present time, while the upper limit T is the terminal time. The integral time interval between the present and terminal times is defined as the control interval. By fixing the value of T, the control interval is limited, which means that the control law shall minimize the cost function in finite time. However, if the terminal time is defined as infinite, it means that we are interested in the behaviour of the system starting from the present time up to the steady state. As it is not required by the mission that the control action has limited time, the steady state solution for the cost function minimization problem is used, also known as the infinite horizon problem.

It can be shown [35] that for the infinite horizon problem, the solution which minimizes the cost function and guarantees closed-loop asymptotic stability is the constant state feedback law

$$\mathbf{u} = -\mathbf{k}\,\mathbf{x}\,, \quad \mathbf{k} = \mathbf{R}^{-1}\,\mathbf{B}^T\,\mathbf{P}\,, \tag{5.4}$$

where $\mathbf{P}$ is the solution to the Algebraic Riccati Equation (ARE),

$$\mathbf{P}\,\mathbf{A} + \mathbf{A}^T\,\mathbf{P} - \mathbf{P}\,\mathbf{B}\,\mathbf{R}^{-1}\,\mathbf{B}^T\,\mathbf{P} + \mathbf{Q} = \mathbf{0}\,. \tag{5.5}$$

In order for the ARE to have a unique, positive definite solution $\mathbf{P}$, which minimizes the infinite time cost function when the defined control law is used, it is a sufficient condition that the system defined by the pair ($\mathbf{A}$, $\mathbf{B}$) is controllable.

5.2.1 Linear Quadratic Integral Control (LQI)

The LQR feedback control law ideally drives the states of the system in the perturbation domain to zero, ensuring that the nominal values throughout the trajectory are followed. However, it does not guarantee a zero tracking error for non-zero references in terms of attitude. In order to have a zero reference tracking error, and to increase the robustness of the controller, an integral action that acts on the attitude tracking error is added. This technique is commonly referred to as Linear Quadratic Integral control or LQI, and the resulting feedback control scheme is shown in Fig. 5.1. [36]

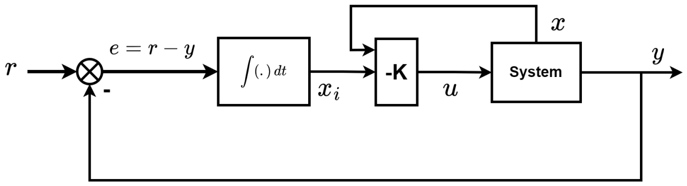


Figure 5.1: LQI control scheme.

Let the difference between the reference signal, r, and the output of the system, y, (the tracking error) be the time derivative of the state-space variable that results from adding the referred integrator, x_i. The state-space representation of the resulting regulator can be obtained by combining the open-loop state-space representation with the feedback law,

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u} \\ \dot{x}_i = r - y \\ y = \mathbf{C}\,\mathbf{x} \\ \mathbf{u} = -\mathbf{K}\,[\mathbf{x}\ x_i]^T \end{cases} \Leftrightarrow \begin{cases} \dot{\mathbf{x}} = (\mathbf{A} - \mathbf{B}\,\mathbf{k})\,\mathbf{x} + \mathbf{B}\,k_i\,x_i \\ \dot{x}_i = r - \mathbf{C}\,\mathbf{x} \end{cases} \tag{5.6}$$

where the gain vector $\mathbf{K} = [\mathbf{k} \;\; k_i]$ is composed by the the original state feedback gain vector, $\mathbf{k}$, and the integral action gain, k_i. Using state augmentation, one can define the new state vector as $\mathbf{z} = [\mathbf{x} \;\; x_i]^T$, composed by the original state vector, $\mathbf{x}$, and the integral action state, x_i. Combining the state augmentation with (5.6), the closed loop state-space representation is obtained,

$$\dot{\mathbf{z}} = \begin{bmatrix} \mathbf{A} - \mathbf{B}\,\mathbf{k} & -\mathbf{B}\,k_i \\ -\mathbf{C} & \mathbf{0} \end{bmatrix} \mathbf{z} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r \Leftrightarrow \dot{\mathbf{z}} = \left(\begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & \mathbf{0} \end{bmatrix} - \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \mathbf{K} \right) \mathbf{z} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r \tag{5.7}$$

Looking at (5.7), it is possible to identify the closed-loop dynamics matrix which is obtained by applying the LQR control law to the newly defined matrices

$$\overline{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & \mathbf{0} \end{bmatrix}, \quad \overline{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \tag{5.8}$$

with the augmented gain vector $\mathbf{K}$. We recall that in order to have a unique positive definite solution to the ARE, the pair composed by the system matrices has to be controllable, which is now given by $\overline{\mathbf{A}}$ and $\overline{\mathbf{B}}$.

5.2.2 Implementation in the linear domain

The described control law can be applied to the derived linear models for the nominal trajectory and the design parameters can be tuned according to the performance in the linear domain. Due to the time dependency of the system, the optimum gain matrices will have to be computed for each defined operating point of the nominal trajectory, possibly resulting in different gain matrices to be selected throughout the mission, also called as gain scheduling.

As already described, the nominal trajectory can be separated into the vertical and varying pitch sections, which translate into different state-space representations. Therefore, the controller design is done separately for the operating points coming from each section. Moreover, both representations were found to have decoupled longitudinal and lateral modes which allows to separate the design even further, with one controller for each mode. This means that there will be one controller for pitch, and one controller for yaw. As the state-space representation for the vertical section is a further simplification of the representation for the varying pitch motion, the design is focused on the varying pitch manoeuvre with the additional simplifications for the vertical part being later presented.

Starting from the state-space matrices for the LQI control (5.7), we recall that the original $\mathbf{A}$ matrix for the lateral mode has to be reduced in order to remove the uncontrollable state, that is roll ($\delta\phi$). In this way, the lateral and longitudinal state vectors, increased by the integrator state, are given by

$$\delta\mathbf{x_{lon}} = \begin{bmatrix} \delta u & \delta w & \delta q & \delta\theta & \delta\theta_i \end{bmatrix}^T \qquad \delta\mathbf{x_{lat}} = \begin{bmatrix} \delta v & \delta r & \delta\psi & \delta\psi_i \end{bmatrix}^T \tag{5.9}$$

Since the output of the system is the variable which shall have zero tracking error, the $\mathbf{C}$ matrix is intuitively the one that selects the pitch and yaw states (one for each mode), resulting in

$$\mathbf{C_{lon}} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} , \qquad \mathbf{C_{lat}} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \tag{5.10}$$

Given the selected operating point, the LQI is applied to the correspondent state-space model and the output gains are a function of the $\overline{\mathbf{A}}$ and $\overline{\mathbf{B}}$ system matrices and the weighting matrices $\mathbf{Q}$ and $\mathbf{R}$, for the longitudinal and lateral modes. The resultant gain vectors will be of the form

$$\overline{\mathbf{K}}_{\mathbf{lon}} = \begin{bmatrix} k_w & k_q & k_\theta & k_{\theta_i} \end{bmatrix} \quad \overline{\mathbf{K}}_{\mathbf{lat}} = \begin{bmatrix} k_v & k_r & k_\psi & k_{\psi_i} \end{bmatrix} . \tag{5.11}$$

The design degree of freedom is the selection of the weighting matrices. Regarding this procedure, some considerations about the control objective allow to reduce the admissible candidates. First of all, setting all non-diagonal entries to zero, and only focusing on the diagonal ones, allows for a more intuitive matrix selection given by the ”penalty” method [34]. According to this method, the diagonal entries of the $\mathbf{Q}$ matrix will determine the relative importance of the state variables in terms of origin tracking performance, while the diagonal entries of the $\mathbf{R}$ matrix allow to directly adjust the control effort for each input. Therefore, the weighting matrices have the following generic format, separated for each mode:

$$\mathbf{Q_{lon}} = diag\left(q_u,\, q_w,\, q_q,\, q_\theta,\, q_{\theta_i}\right), \quad \mathbf{Q_{lat}} = diag\left(q_v,\, q_r,\, q_\psi,\, q_{\psi_i}\right), \quad R_{lon} = r_{\mu_1}\,, \quad R_{lat} = r_{\mu_2}\,, \tag{5.12}$$

where the R matrix for each mode is in fact a scalar, given that there is only one input per mode.

Using the ”penalty” method , further restrictions can be imposed on the $\mathbf{Q}$ matrix. Taking into account the aim of controlling the rocket’s attitude, the state variables of higher importance are the pitch and yaw Euler angles and its associated tracking error integral state, hence, the entries q_θ, q_ψ, q_{θ_i} and q_{ψ_i} shall have the highest magnitude. Additionally, given the nature of the TVC actuation, trying to control the linear velocities will conflict with the attitude control, specially for non-zero attitude references. This means that q_u, q_v and q_w are set to zero. The angular velocities will have non-zero entries in the $\mathbf{Q}$ matrix with a lower magnitude than the previous terms in order to limit deviations from the nominal condition but allowing a certain freedom so that non-zero attitude references can be reached.

As for the r_{μ_1} and r_{μ_2} values, as mentioned, they will penalize the control action from each control input and have to be iteratively determined taking into account the dynamics of the actuation system. In particular, it is of high importance that the control input requests will not surpass the saturation of the actuators, in this case the gimbal angle, since the stability of the system will no longer be ensured. Moreover, given that during the pitch manoeuvre the nominal value of the pitch control input, μ_1, has a non-zero value, the control action of this input has a higher limitation in terms of magnitude.

During the vertical part of the trajectory, the longitudinal velocity δu is no longer a state of the system, and the equivalence between the longitudinal and lateral modes allows to use the same weighting matrices for both of them.

Given that the gains will have to be computed for all operating points, the $\mathbf{Q}$ and $\mathbf{R}$ values could be adapted to each operating point. However, in order to limit the degrees of freedom of the design problem, the values were initially kept constant for all operating points.

Taking all these considerations into account, the weighting values were tuned by observing the closed-loop stability, poles and zeros location, and response of the linear system in the different operating points, including the actuator dynamics in the loop. The aim was trying to achieve the fastest response time possible, while maintaining a smooth response and actuation behaviour. Table 5.1 shows the weighting values obtained during the design phase, for both sections of the trajectory and for both decoupled modes.

Table 5.1: Weighting matrices' values.

Matrix entry	q_u	q_v	q_w	q_q	q_r	q_θ	q_ψ	q_{θ_i}	q_{ψ_i}	r_{μ_1}	r_{μ_2}
Vertical section	N/D	0	0	2	2	500	500	50000	50000	5	5
Varying pitch section	0	0	0	2	2	500	500	25000	50000	10	5

As an initial approach, given the long duration of the mission and the knowledge obtained regarding the evolution of the system dynamics, the following time vector was selected for the operating points to be studied in the design phase:

$$t_{op} = [0.1,\ 5,\ 10,\ 15,\ 20,\ 25,\ 30,\ 35,\ 40,\ 45,\ 50,\ 55,\ 60,\ 65,\ 70,\ 75,\ 80,\ 85,\ 90,\ 95],\ [s] \tag{5.13}$$

By applying the defined weighting matrices in LQI design to each operating point, the resulting gains were obtained. Given that the weights relative to the linear velocities were set as zero, the corresponding gains were negligible. This allows to use a partial state vector feedback composed only by the angular velocities and Euler angles, meaning that the gain vectors for each mode are of the form

$$\overline{\mathbf{k}}_{\mathbf{lon}} = \begin{bmatrix} k_q & k_\theta & k_{\theta_i} \end{bmatrix} \ , \quad \overline{\mathbf{k}}_{\mathbf{lat}} = \begin{bmatrix} k_r & k_\psi & k_{\psi_i} \end{bmatrix} \ , \tag{5.14}$$

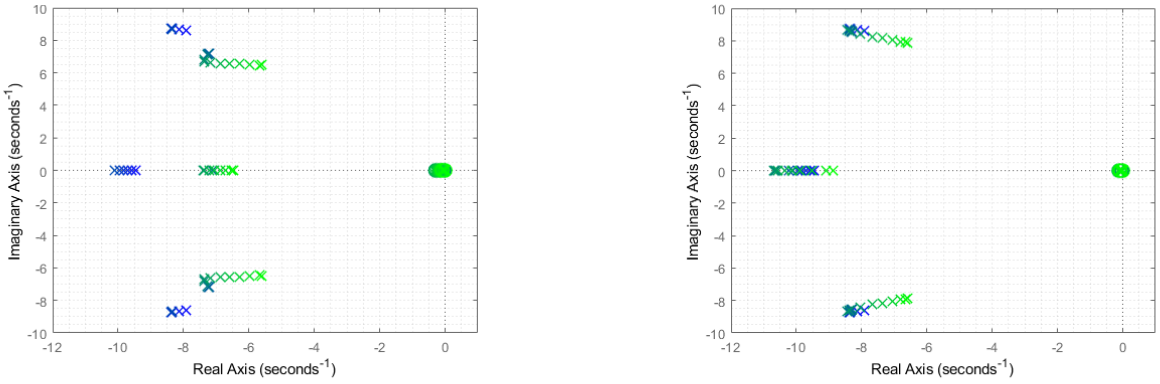
and will have the same dimension during the entire trajectory, as neither the longitudinal velocity nor the the roll angle, the differentiating states, are being considered.

As previously mentioned, the gains resulting from the controller design applied to each operating point will have to be scheduled during the flight according to a selection criteria. This will be approached in Chapter 7, in which the implementation of the designed controllers in the complete simulation model will be discussed. If necessary, the selection of the operating points in terms of overall number and location will then be revisited.

Closed-loop modal analysis

By closing the loop with the defined control law, the closed-loop poles and zeros can be determined for the different operating points of the nominal trajectory. The closed loop poles are simply given by the eigen values of the closed-loop dynamics matrix ($\overline{\mathbf{A}} - \overline{\mathbf{B}} \cdot \overline{\mathbf{K}}$), while the poles will also depend on the closed-loop $\mathbf{B}$ matrix ($[\mathbf{0} \ \ 1]^T$) and the output matrix $\mathbf{C}$, which is the one that selects the state variable to be tracked, the pitch or yaw Euler angles.

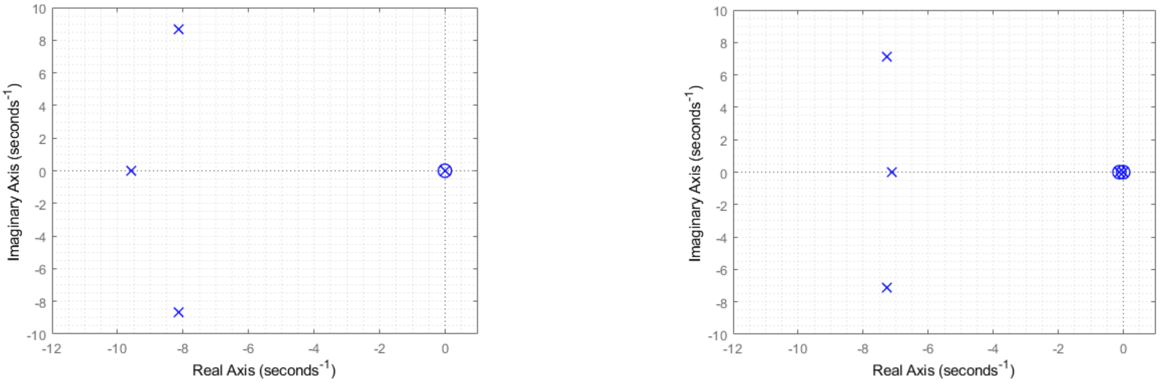
Figure 5.2 displays the obtained poles and zeros for all the selected operating points and for both decoupled modes. It was found that the poles and zeros distribution was identical for both modes throughout the entire nominal trajectory. In this way, Fig. 5.3 shows the poles and zeros for the longitudinal mode only at $t = 5\,$s and $t = 40\,$s, as exemplification.



(a) Longitudinal mode

(b) Lateral mode

Figure 5.2: Closed-loop pole-zero maps for all operating points.



(a) $t = 5\,$s

(b) $t = 40\,$s

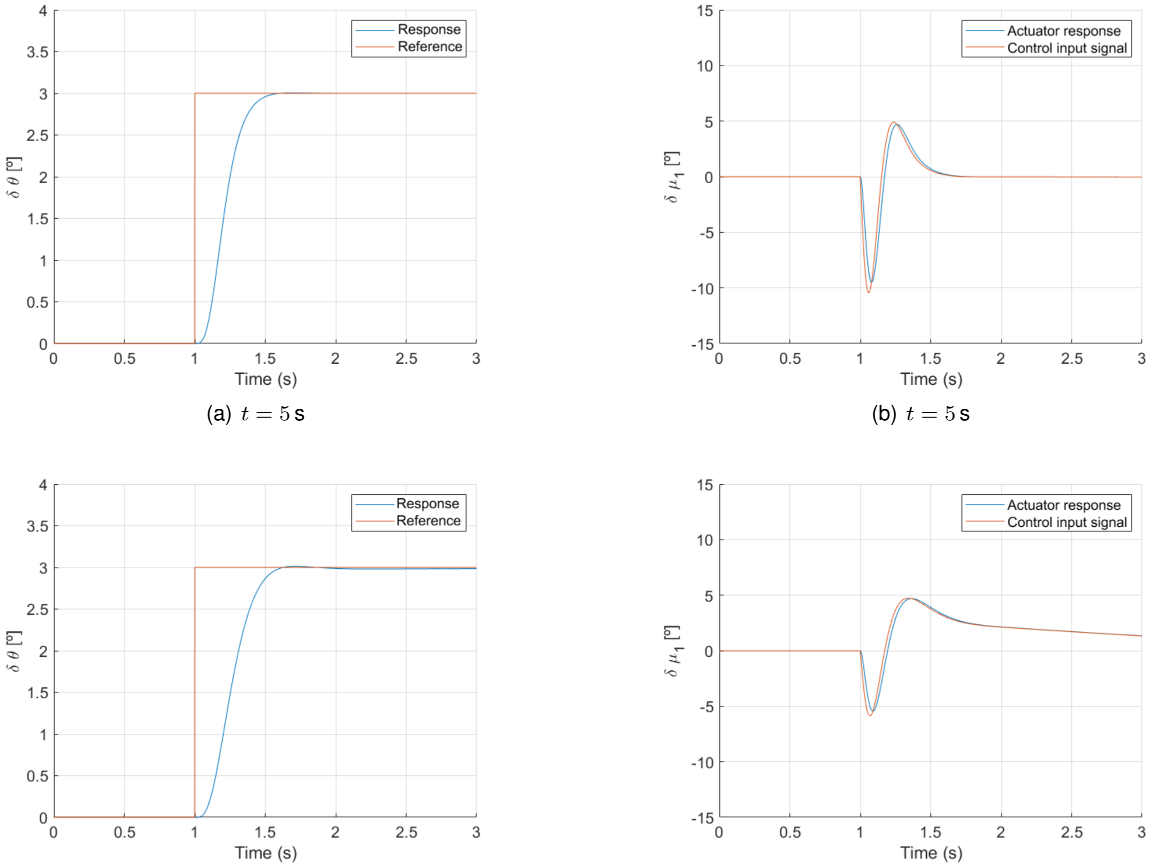
Figure 5.3: Closed-loop pole-zero map example.

Firstly, the control law allowed to stabilize all operating points, placing all closed-loop poles in the left-hand side of the complex plane. During the vertical section, there is a pole approximately at the origin, with a zero cancelling its dynamics. This also occurs during the pith manoeuvre, with an additional pole close to the origin, corresponding to the additional dimension of the state-space representation, also cancelled by a zero. The cancellation makes that during both phases the number of poles of each mode, relevant for the dynamics, is the same. The relevant poles correspond to a pair of conjugated complex poles and a real pole, all in the left-hand side of the complex plane. The complex poles are expected to cause oscillatory behaviour in the response of the system, however, it was the ideal compromise found between limiting oscillations while keeping a fast settling time.

As the Q and R matrices for the longitudinal mode during the pitch manoeuvre were selected so that the control action is limited, it is possible to identify that the relevant poles are closer to the imaginary axis. This was done in order to prevent the saturation of the actuator given the non-zero nominal input value, and will cause the a slower response of the system.

Closed-loop behaviour

The closed-loop linear system was simulated using the state-space representations of the different operating points and the respective LQI gain matrices, according to the block diagram of Figure 5.1. Through these simulations, it was possible to determine the closed-loop behaviour of the linear system, which allowed to tune the weighting matrices in the design phase. As the number of operating points is large, some key points throughout the nominal trajectory were selected to visually present the ideal performance of the designed controllers in a wide range of conditions. Taking into account the closed-loop poles location, two points were selected: the first one in the beginning of the trajectory ($t = 5\,$s), where the open-loop system presented unstable oscillatory behaviour, and the second one coinciding with the peak nominal pitch control input ($t = 60\,$s), which also approximately coincides with one of the two symmetric peaks in pitch rate. Additionally, the results here presented are for the longitudinal mode only, due to the existent mode equivalence during the vertical part and to the higher importance of this mode during the pitch manoeuvre. Figure 5.4 depicts the response of the system in terms of pitch angle ($\delta\theta$) and pitch control input ($\delta\mu_1$) for the selected operating points and for a $3\,°$ step, which was considered to be a reasonable value for an attitude correction during the trajectory. This selection allowed to appropriately tune the control effort, specially looking at the saturation of the actuator.



(a) $t = 5\,$s

(b) $t = 5\,$s

(c) $t = 60\,$s

(d) $t = 60\,$s

Figure 5.4: Closed-loop step response.

It is possible to infer that the step response is similar in both operating points. However, the limitation imposed on the control effort during the pitch manoeuvre created a slight decrease in performance, with a higher settling time and the appearance of a slight overshoot. Nevertheless, the objective of limiting the control input magnitude to avoid saturation due to the the non-zero nominal value was achieved without compromising the correct functioning of the control system. Additionally, the control input signal demonstrates a smooth behaviour, allowing the actuator to closely track the desired gimbal angle

Table 5.2 details some key parameters of the closed-loop system step response in the linear domain for a wide range of operating points.

Table 5.2: Closed-loop step response parameters.

Operating point	Rise time (s)	Settling time (s)	Overshoot (%)
$t = 5$ s	0.2686	0.4461	0.5710
$t = 15$ s	0.2658	0.4423	0.4986
$t = 25$ s	0.2641	0.4376	0.6312
$t = 35$ s	0.3401	0.5723	0.1239
$t = 45$ s	0.3356	0.5546	0.4587
$t = 55$ s	0.3281	0.5344	1.2921
$t = 65$ s	0.3278	0.5303	1.7586
$t = 75$ s	0.3392	0.5503	1.6825
$t = 85$ s	0.3546	0.5789	1.3589
$t = 95$ s	0.3667	0.6052	0.7995

Overall, the performance in the linear domain is satisfactory, with fast settling times and reduced overshoot for all operating points. As expected, the performance slightly worsens during the pitch manoeuvre as the control effort had to be limited to avoid actuator saturation. Later, in Chapters 7 and 8 , the designed controller will have to be implemented and tested in the full non-linear model, with the addition of noise and disturbances, to better detail the performance of the control system under conditions closer to a real scenario.

5.3 Linear Parameter Estimator

So far, the approach to the rocket attitude control problem was based on a linear version of the rocket model obtained through linearization around a nominal trajectory. The resulting linear time-varying model was then used to design an optimal controller recurring to the LQR technique.

Even though it was possible to obtain satisfying results in simulation, this approach presents two interconnected limitations that can cause the controller to fail under a real implementation scenario. One, is the dependency on model accuracy, and the other is the lack of generality of the controller. The LQR has a certain degree of robustness, meaning that it can still perform when model uncertainties are present. However, this robustness is limited since the controller is designed considering a nominal evolution of model parameters that might considerably differ from the real evolution during the mission. Also related to this, the controller can only work under a certain region close to the design condition (or the nominal trajectory), making it only suitable for the specific mission and, even more concerning, causing it to fail when considerable deviations from the nominal condition occur.

In order to tackle the first limitation, and potentially reduce the second, an online parameter estimator can be implemented so that the controller acts on an informed value of the model parameters. This can either be used to improve the LQR performance using an online gains computation with the estimated parameters or with another type of controller, for instance, a non-linear one, that allows a more global solution, tackling the second limitation.

5.3.1 Selected parameters for estimation

Amongst the identified parameters (Section 4.3.2), the ones related with the aerodynamic properties of the rocket are subjected to an higher level of uncertainty. This is due, on one side, to the difficulty in obtaining accurate aerodynamic coefficients and derivatives of the rocket for a broad range of velocities and aerodynamic angles, and, on the other, to the potential lack of sensors that allow to determine on-board the velocity with respect to the fluid and the aerodynamic angles of the vehicle, such as Pitot tubes and free movement vanes. Taking this into account, and to limit the number of parameters to be estimated, the parameter estimator is designed in order to get better online estimates on the aerodynamic parameters.

Looking at the external forces and moments (4.2), the aerodynamic parameters are hidden under the aerodynamic force and moment coefficients - C_A, C_Y, C_N, C_m, and C_n. Since a first estimate on these quantities is available using the stored aerodynamic data, the inertial velocity expressed in Mach number and the aerodynamic angles computed using the inertial velocities, a proportional error factor is multiplied in each aerodynamic force and moment and then estimated, instead of using the linear parameter estimator to directly estimate the coefficients. In this way, the error factors will account for the errors behind all aerodynamic parameters impacting the aerodynamic forces and moments, without estimating each of them separately. It is expected that if the aerodynamic forces and moments are being accurately computed with the initial guess, the error factors will be equal to one.

Once the parameters to be estimated are selected, it is possible to rewrite the external forces and moments, in the simplified form, in order to include them,

$$ {}^{B}\mathbf{F} = \begin{bmatrix} -mg\, c_\theta c_\psi + T\, c_{\mu_1} c_{\mu_2} - \bar{q}\, C_A\, S\, \delta_{a_x} \\ -mg\,(s_\phi s_\theta c_\psi - c_\theta s_\psi) - T\, c_{\mu_1} s_{\mu_2} + \bar{q}\, C_Y\, S\, \delta_{a_y} \\ -mg\,(c_\phi s_\theta c_\psi + s_\phi s_\psi) - T\, s_{\mu_1} - \bar{q}\, C_N\, S \delta_{a_z} \end{bmatrix}, \quad {}^{B}\mathbf{M} = \begin{bmatrix} 0 \\ -T\, s_{\mu_1}\, l + \bar{q}\, S\, d\, C_m \delta_m \\ T\, c_{\mu_1} s_{\mu_2}\, l + \bar{q}\, S\, d\, C_n \delta_n \end{bmatrix}, \tag{5.15} $$

where δ_{a_x}, δ_{a_y}, δ_{a_z}, δ_m and δ_n are the proportional error factors and the parameters to be estimated in the rearranged rocket model.

5.3.2 Estimator design

The estimator design follows along the methodology proposed in [37], where an hovercraft control system is designed based on dynamics parameter identification, which details a generic parameter estimator for time-varying systems, linear in the parameters.

Given the detailed rocket model, we are in the presence of a non-linear time-varying system that can be described by

$$\dot{\mathbf{x}} = \mathbf{f_0}(\mathbf{x}, \boldsymbol{\zeta}, t)\,, \tag{5.16}$$

where $\mathbf{x}$ is the system state and $\boldsymbol{\zeta}$ is the parameter vector. Taking into account the linearity in the system parameters (the aerodynamic correction factors) the system can be described by the rearranged dynamics

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t) + \mathbf{G}(\mathbf{x}, t)\,\boldsymbol{\zeta}\,. \tag{5.17}$$

Using state augmentation with the parameter vector, this system can be written in block form as

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\boldsymbol{\zeta}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{G}(\mathbf{x}, t) \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\zeta} \end{bmatrix} + \begin{bmatrix} \mathbf{f}(\mathbf{x}, t) \\ \mathbf{0} \end{bmatrix}, \tag{5.18}$$

and assuming full state measurements are available, we have

$$\mathbf{y} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\zeta} \end{bmatrix}\,. \tag{5.19}$$

It is important to note that this representation assumes constant, or slowly varying, parameters, which may not be the case for the correction factors. However, this will be considered as first approach to the estimation problem and the performance under varying errors will be studied.

The non-linear system, linear in the parameters, can thus be rewritten as a linear time-varying (LTV) system

$$\begin{cases} \dot{\boldsymbol{\xi}} = \mathbf{A}(t)\,\boldsymbol{\xi} + \mathbf{u}(t) \\ \mathbf{y} = \mathbf{C}(t)\,\boldsymbol{\xi} \end{cases} \tag{5.20}$$

with,

$$\mathbf{A}(t) = \begin{bmatrix} \mathbf{0} & \mathbf{G}(\mathbf{y}, t) \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{u}(t) = \begin{bmatrix} \mathbf{f}(\mathbf{y}, t) \\ \mathbf{0} \end{bmatrix}, \quad \mathbf{C}(t) = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}, \quad \boldsymbol{\xi} = \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\zeta} \end{bmatrix}\,. \tag{5.21}$$

where $\mathbf{x}$ was replaced by $\mathbf{y}$ in the system matrices so that it can be regarded as linear, as $\mathbf{y}$ is available for estimator design purposes.

By substituting the external forces and moments (5.15) in the simplified rocket dynamics (4.6), it is possible to obtain the rearranged explicit dynamics, where previously detailed assumptions are being used. With the rearranged explicit dynamics detailed, it is possible to obtain the derived LTV system for the case of the rocket:

$$\mathbf{x} = [\, u \;\; v \;\; w \;\; q \;\; r \;\; \phi \;\; \theta \;\; \psi \,]^T, \quad \boldsymbol{\zeta} = [\, \delta_{a_x} \;\; \delta_{a_y} \;\; \delta_{a_z} \;\; \delta_m \;\; \delta_n \,]^T \,, \tag{5.22a}$$

$$\mathbf{G}(\mathbf{y},t) = \begin{bmatrix} -m^{-1}\,\overline{q}\,S\,C_A & 0 & 0 & 0 & 0 \\ 0 & m^{-1}\,\overline{q}\,S\,C_Y & 0 & 0 & 0 \\ 0 & 0 & -m^{-1}\,\overline{q}\,S\,C_N & 0 & 0 \\ 0 & 0 & 0 & {J_t}^{-1}\,\overline{q}\,S\,d\,C_m & 0 \\ 0 & 0 & 0 & 0 & {J_t}^{-1}\,\overline{q}\,S\,d\,C_n \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{5.22b}$$

$$\mathbf{u}(t) = \begin{bmatrix} -g\,\cos\theta\cos\psi + m^{-1}\,T\cos\mu_1\cos\mu_2 - q\,w + r\,v \\ -g\,(\sin\phi\sin\theta\cos\psi - \cos\theta\sin\psi) - m^{-1}\,T\,\cos\mu_1\sin\mu_2 - r\,u \\ -g\,(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi) - m^{-1}\,T\,\sin\mu_1 + q\,u \\ -{J_t}^{-1}\,T\,\sin\mu_1\,l \\ {J_t}^{-1}\,T\,\cos\mu_1\sin\mu_2\,l \\ (q\,\sin\phi + r\,\cos\phi)\tan\theta \\ q\,\cos\phi - r\,\sin\phi \\ \frac{q\,\sin\phi + r\,\cos\phi}{\cos\theta} \end{bmatrix}. \tag{5.22c}$$

In order to design the estimator for this system, it is necessary for it to be observable. In the reference [ref], it is demonstrated that the system is observable if and only if there exists no unit vector d, with the dimension of the parameter vector, such that

$$\int_{t_0}^{t} \mathbf{G}(\mathbf{y},\sigma)\,d\sigma \cdot d = 0. \tag{5.23}$$

Taking the time derivative in both sides and substituting for the rocket dynamics, we get that the equivalent non-observability condition is

$$\begin{cases} -m^{-1}\,\overline{q}\,S\,C_A\,d_1 = 0 \\ m^{-1}\,\overline{q}\,S\,C_Y\,d_2 = 0 \\ -m^{-1}\,\overline{q}\,S\,C_N\,d_3 = 0 \\ J^{-1}\,\overline{q}\,S\,d\,C_m\,d_4 = 0 \\ J^{-1}\,\overline{q}\,S\,d\,C_n\,d_5 = 0 \end{cases} \Leftrightarrow \begin{cases} C_A\,d_1 = 0 \\ C_Y\,d_2 = 0 \\ C_N\,d_3 = 0 \\ C_m\,d_4 = 0 \\ C_n\,d_5 = 0 \end{cases} \tag{5.24}$$

where d_i, for $i = 1, 2, 3, 4, 5$, are the components of the unit vector, and the simplification is due to m, J, $\overline{q}$, d, and S being always different from zero.

Looking at condition (5.24), it is possible to infer that the system is observable only when the aerodynamic force and moment coefficients are all different from zero, since if one of them is not, the unit vector with $d_i = 1$, where i corresponds to component multiplying the null coefficient, satisfies the non-observability condition.

At this point, we recall how the coefficients are computed to identify under which conditions their initial estimate is equal to zero. Regarding the force coefficients, the lateral ones are cancelled when the respective aerodynamic angle is zero and the axial one is never zero under the admissible operation regime. On the other hand, the moment coefficients are cancelled for null aerodynamic angles and body angular rates, or for a non-zero instantaneous combination of both.

A possible solution to the observability issue is to make sure that the initial estimates on the aerodynamic coefficients are never zero. This can be done through a non-linearity introduced in the initial estimate computation that consists in forcing it to a small non-zero value whenever it is indeed null. The penalty is that if the real value is zero, an error is being consciously introduced and the estimator has to be continuously correcting it. However, it is expected that under a real scenario the true value of the aerodynamic coefficients can only be exactly equal to zero instantaneously, as both internal and external disturbances will break the aforementioned conditions.

Now that the have seen that the system can be made fully observable, a Kalman filter represents a simple and easily tunable solution for the estimation of the system state. The Kalman filter theory is detailed in Appendix B. Looking at (5.20) and (5.21), there is an immediate correspondence between the $A(t)$ and $C(t)$ estimator matrices and the process and output equations. The state to be estimated $x(t)$ is the defined state vector $\xi(t)$, while the $G(t)$ and $B(t)$ matrices are defined as the identity with appropriate dimensions.

5.3.3 Online LQR Control With Parameter Estimation

By using the designed linear parameter estimator, the LQR controller gains can be computed onboard recurring to the real time estimates of the aerodynamic correction factors. Instead of referring to the nominal evolution of the aerodynamic parameters and predetermine the gains to be scheduled during the flight, the gains are computed online. This is done by rewriting the state-space representation, including the estimated parameters , which results in a new dynamics matrix $A(t)$ for the generic linearization, given by

$$A(t) = \begin{bmatrix} 0 & a_{12} & a_{13} & a_{14} & a_{15} & 0 & a_{17} & a_{18} \\ a_{21} & a_{22}\,\delta_{a_y} & 0 & 0 & a_{25} & a_{26} & a_{27} & a_{28} \\ a_{31}\,\delta_{a_z} + (1-\delta_{a_z})\,q_0 & 0 & a_{33}\delta_{a_z} & a_{34} & 0 & a_{36} & a_{37} & a_{38} \\ a_{41}\,\delta_m & 0 & a_{43}\,\delta_m & a_{44}\,\delta_m & 0 & 0 & 0 & 0 \\ 0 & a_{52}\,\delta_n & 0 & 0 & a_{55}\,\delta_n & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{64} & a_{65} & a_{66} & a_{67} & 0 \\ 0 & 0 & 0 & a_{74} & a_{75} & a_{76} & 0 & 0 \\ 0 & 0 & 0 & a_{84} & a_{85} & a_{86} & a_{87} & 0 \end{bmatrix}, \tag{5.25}$$

where the matrix entries a_{ij} are the same as detailed in (4.9). It is noted that there is no dependency on the axial aerodynamic force in the obtained representation, which means that the correspondent correction factor δ_{a_x} is not used by the controller.

The new dynamics matrix can then be simplified for the nominal trajectory, as previously done, and be used to compute the LQI gains according to the defined control law. Once again, the design degree of freedom will be the selection of the Q and R matrices which will now be used on-board to compute the gains. In the original method, operating points had to be selected in order to predetermine the gains to be scheduled during the flight. For this method, the scheduling will be done by selecting the points where the gains computation will occur throughout the trajectory. The selection method will be further explained in the implementation Chapter 7.

It is expected that by using this method, the controller will be more robust to model uncertainties and will be able to operate in a wider range of conditions, further away from the nominal scenario.

Chapter 6

Navigation System Design

So far, it was assumed that the control system had access to an exact full-state measurement, meaning that the state vector was assumed to be exactly known at any given instant. In reality, this is not the case, and it is necessary to have a navigation system, composed by sensors and estimators, capable of providing an accurate estimate on the state vector.

Even with high grade sensors and estimators, the state vector estimates will always be corrupted by noise, and so, it is important to understand its impact on the control system performance once the navigation system has been designed.

6.1 On-board Sensors

In order to design a navigation system, it is necessary to select the sensor suite that will be on-board of the vehicle. The sensors might provide a direct measurement on the required state variables or on other variables that can then be used to estimate them. For the case of rockets, and taking into account the state variables to measured - linear velocities, angular velocities, and Euler angles - it is common to use an Inertial Measurement Unit (IMU) combined with a Global Navigation Satellite System (GNSS) receiver.

6.1.1 IMU

As indicated by its name, an IMU is composed by inertial sensors. Inertial sensors are used to measure the motion of an object with respect to an inertial reference frame and can be classified into accelerometers and gyroscopes. An IMU has 3-axis accelerometers and gyroscopes to provide measurements in all the vehicle axes. Often, the IMU includes an additional 3-axis magnetometer and a barometer.

Accelerometers

An accelerometer supplies a measure of the system's acceleration and can be used to determine the vehicle's velocity by integration. In order to do this, it is necessary to know the initial condition. Over time, the velocity measurement will drift from the ture value due to the inherent noise and bias properties of the accelerometer. If the accelerometer is assumed to be measuring gravity alone, it is possible to calculate the pitch and roll angles from the direction of the gravity vector. However, any biases or other errors in the accelerometer measurements cause errors in the calculation of the pitch and roll angles. In addition, since the accelerometer is assumed to be measuring gravity alone, any added dynamic motion also causes an error in the calculation of the system's pitch and roll.

Gyroscopes

A gyroscope provides a measurement of the system's angular rate. These angular rate measurements are can be integrated to determine an estimate of the system's attitude. However, in order to determine the current attitude, the initial attitude of the system must also be known. Over time, this calculated attitude drifts boundlessly from the true attitude of the system due to the inherent noise and bias properties of the gyroscope itself.

Magnetometer

Since the accelerometer can only measure pitch and roll, a magnetometer provides a measurement of yaw by comparing the measurement of the magnetic field surrounding the system to Earth's magnetic field, just like a traditional magnetic compass. In most units, the magnetometer measurements have no impact on the pitch and roll angle estimates.

6.1.2 GNSS receiver

A GNSS is a satellite configuration, or constellation, that provides coded satellite signals which are processed by a GNSS receiver inside the vehicle to calculate position, velocity, and time. There are many GNSS providers, with the american Global Positioning System (GPS) being the first to appear.

6.1.3 Sensor Fusion

The measurements from the gyroscope, accelerometer, magnetometer and GNSS receiver are combined to provide an estimate of the system's velocity, angular rates and orientation, using the estimator presented below. The estimation technique uses these raw measurements to derive an optimized estimate of the state variables, given the assumptions outlined for each individual sensor. The system estimates the accelerometer and gyro bias, or drift error, in addition to the state variables. The gyro and accelerometer bias are then used to compensate the raw measurements and aid in preventing the drift over time. By combining the data from each of these sensors into the estimator, a drift-free solution for the system can be obtained.

6.2 Estimator Architecture

The estimator architecture is based on [38], where the objective is to estimate the sideslip angle of a formula student car. The architecture is composed by three main filters, according to the scheme in Fig. 6.1.

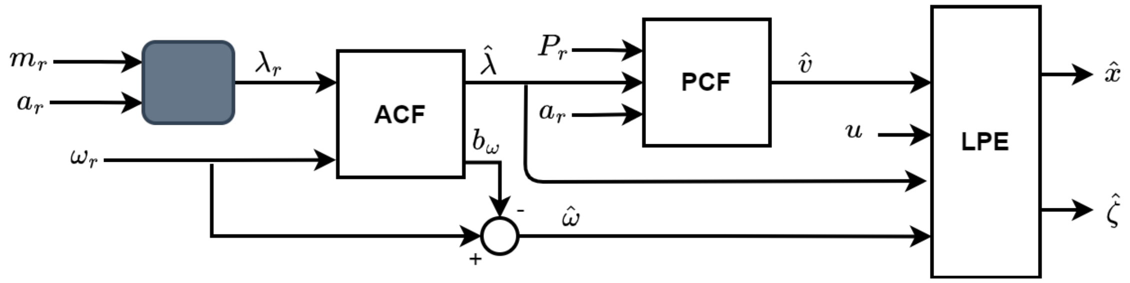


Figure 6.1: Estimator Architecture.

The first is an Attitude Complementary Filter (ACF), which uses the Euler angles readings ($\boldsymbol{\lambda}_r$), obtained through the combination of a magnetometer ($\mathbf{m}_r$) and accelerometer ($\mathbf{a}_r$), and the measured angular rates from the gyroscopes ($\boldsymbol{\omega}_r$), to provide a filtered attitude estimate ($\hat{\boldsymbol{\lambda}}$). Additionally, it provides an estimate on the angular rate bias ($\mathbf{b}_\omega$) to correct the signal from the gyroscope. As mentioned, accelerated motion will cause errors in the pitch and yaw measurements by the accelerometer. In this way, a pre-processing step occurs in which the GNSS measured velocity is used to account for the acceleration measured by the accelerometers and correct the attitude measurement.

The second one is a Position Complementary Filter (PCF), which merges the position reading from the GNSS receiver, translated into the inertial frame ($\mathbf{P_r}$), and the acceleration measurements from the accelerometer ($\mathbf{a_r}$) to provide an estimate on the velocity components ($\hat{\mathbf{v}}$). This filter is also self-calibrated, since it accounts for the bias in the accelerometer ($\mathbf{b_a}$).

Finally, the last filter is the linear parameter estimator (LPE) detailed in the previous Chapter, which uses the velocity, angular rate and attitude pre-filtered values to give a final estimate on the state vector ($\hat{\mathbf{x}}$) and parameters ($\hat{\boldsymbol{\zeta}}$) according to the developed dynamics and kinematics rocket model.

As opposed to the what is done in the reference, the estimator is designed considering continuous time, as in the control system design. In reality, the sensor readings will be available in discrete time, however, if the sensors update rate is high enough, the results do not substantially differ from the continuous time case. Nevertheless, the estimators can be easily adapted to discrete time.

6.2.1 Attitude Complementary Filter (ACF)

The ACF fuses the Euler angles readings with the angular rates readings from the gyroscopes to provide a more accurate estimate on the Euler angles of the vehicle and, consequently, its attitude. The ACF also provides a bias estimate for the angular rate measurements, in order to account for the static and dynamic offsets of the gyroscopes. This bias is then used to correct the angular rates for the following filters.

Firstly, it is assumed that the Euler angles measurement is corrupted by Gaussian white-noise ($\mathbf{w}_\lambda$) as well as the angular rates reading ($\mathbf{w}_\omega$). The gyroscope bias ($\mathbf{b}_\omega$) is described by a constant term ($\mathbf{b}_{\omega_c}$) with additional Gaussian white-noise ($\mathbf{w}_{b_\omega}$), yielding

$$\boldsymbol{\lambda}_r = \boldsymbol{\lambda} + \mathbf{w}_\lambda \tag{6.1a}$$

$$\boldsymbol{\omega}_r = \boldsymbol{\omega} + \mathbf{b}_\omega + \mathbf{w}_\omega \,, \quad \mathbf{b}_\omega = \mathbf{b}_{\omega_\mathbf{c}} + \mathbf{w}_{\mathbf{b}_\omega} \tag{6.1b}$$

This filter is based on the kinematic equations for the Euler angles (3.31), using directly the Euler angles readings in the process matrices so that the system can be regarded as linear. Its state-space representation follows according to the Kalman filter equation,

$$\dot{\hat{\mathbf{x}}} = \begin{bmatrix} 0 & 0 & 0 & -1 & -s_{\phi_r} t_{\theta_r} & 0 \\ 0 & 0 & 0 & 0 & -c_{\phi_r} & s_{\phi_r} \\ 0 & 0 & 0 & 0 & -\dfrac{s_{\phi_r}}{c_{\theta_r}} & -\dfrac{c_{\phi_r}}{c_{\theta_r}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \hat{\mathbf{x}} + \begin{bmatrix} 1 & s_{\phi_r} t_{\theta_r} & 0 \\ 0 & c_{\phi_r} & -s_{\phi_r} \\ 0 & \dfrac{s_{\phi_r}}{c_{\theta_r}} & \dfrac{c_{\phi_r}}{c_{\theta_r}} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_r \\ q_r \\ r_r \end{bmatrix} + \mathbf{K}\left(\mathbf{y} - \hat{\mathbf{y}}\right), \tag{6.2a}$$

$$\mathbf{x} = \left[\boldsymbol{\lambda} \;\; \mathbf{b}_\omega\right]^T \,, \quad \mathbf{y} = \boldsymbol{\lambda}_r + \mathbf{w}_\lambda \,, \quad \hat{\mathbf{y}} = \hat{\boldsymbol{\lambda}} \,. \tag{6.2b}$$

The gain matrix $\mathbf{K}$ will be of dimension 6×3, and can either be computed by recursively solving the matrix form differential Riccati equation, using the varying process and input matrices, or by selecting a time-invariant equivalent of the system. A possible time-invariant is choosing the vertical attitude, $\boldsymbol{\lambda} = [0 \;\; 0 \;\; 0]^T$, to define the process matrices and then compute the associated time-invariant Kalman gains. This solution is more efficient for a real implementation scenario as it requires less computational power.

6.2.2 Position Complementary Filter (PCF)

The PCF is used to estimate the velocity components in the body frame. It uses the position provided by the GPS, already translated into the inertial frame, and the acceleration readings from the on-board accelerometer. The Euler angles from the ACF are used to convert the acceleration readings to the Earth frame and the output velocity components to the body frame. This filter is also kinematic, as the previous, using the following equations of motion,

$$\dot{\mathbf{P}} = {}^E\mathbf{v} \,, \tag{6.3a}$$

$${}^E\dot{\mathbf{v}} = \mathbf{R}\,\mathbf{a} \,. \tag{6.3b}$$

where $\mathbf{R}$, as previously defined, is the rotation matrix that makes the conversion from the body frame to the Earth frame. It is assumed that both measurements, position ($\mathbf{P_r}$) and acceleration ($\mathbf{a_r}$), are corrupted with Gaussian random white-noise, $\mathbf{w_P}$ and $\mathbf{w_a}$ respectively. Besides that, it is also assumed that the acceleration readings have bias, in part due to the sensor offset, which is also described by a constant term $\mathbf{b_{a_c}}$ with additional Gaussian white-noise $\mathbf{w_{b_a}}$,

$$\mathbf{a}_r = \mathbf{a} + \mathbf{b}_a + \mathbf{w}_a \,, \tag{6.4a}$$

$$\mathbf{b}_a = \mathbf{b}_{a_c} + \mathbf{w}_{b_a} \,. \tag{6.4b}$$

By combining (6.3) and (6.4), it is possible to write the state-space system of the PCF, once again resorting to the Kalman filter,

$$\dot{\hat{\mathbf{x}}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{R} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \hat{\mathbf{x}} + \begin{bmatrix} \mathbf{0} \\ \mathbf{R} \\ \mathbf{0} \end{bmatrix} \mathbf{a_r} + \begin{bmatrix} \mathbf{K_1} \\ \mathbf{K_2} \\ \mathbf{R^T K_3} \end{bmatrix} (\mathbf{y} - \hat{\mathbf{y}}) \,, \tag{6.5a}$$

$$\mathbf{x} = \begin{bmatrix} \mathbf{P} & {}^{E}\mathbf{v} & \mathbf{b_a} \end{bmatrix}^{\mathbf{T}} , \quad \mathbf{y} = \mathbf{P_r} + \mathbf{w_P} \,, \quad \hat{\mathbf{y}} = \hat{\mathbf{P}} \,. \tag{6.5b}$$

The individual gain matrices $\mathbf{K_1}$, $\mathbf{K_2}$ and $\mathbf{K_3}$ are of dimension 3×3 and can once again be computed considering the vertical attitude time-invariant to define the rotation matrix $\mathbf{R}$. Note that the gain matrix $\mathbf{K_3}$ is associated with the bias vector estimate $\hat{\mathbf{b}}_a$ and so it has to be rotated from the inertial frame to the body one, due to the fact that the position measurements are given in the inertial frame.

Chapter 7

Implementation in Simulation

7.1 Simulation Model

Using MATLAB & Simulink® environment, the described dynamics and kinematics modelling of a generic rocket were portrayed in a generic computational model which served as a simulation environment to test the developed attitude determination and control system. The implementation details of this computational model will be described, as well the verification and validation procedure.

The Simulink model was segregated in subsystems according to the block diagram of Fig. 7.1. These subsystems are responsible for generating the environment, such as the atmospheric conditions (air pressure, temperature, density and wind), the aerodynamic and mass properties of the rocket, and the external forces and moments. By computing all the previous quantities, the resulting kinematics in terms of position, linear and angular velocities, and orientation, can be obtained.

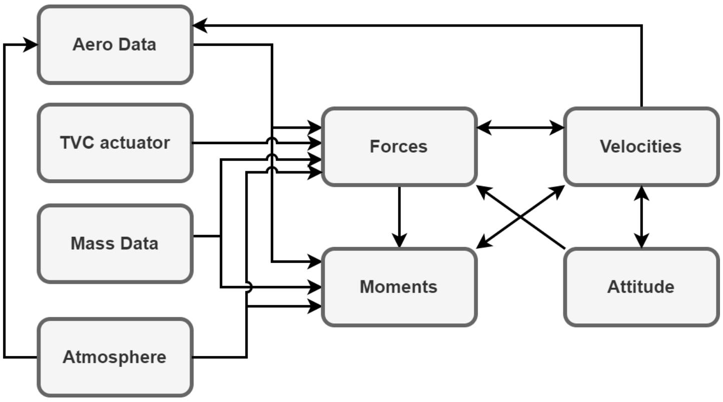


Figure 7.1: Simulation model block diagram.

7.1.1 Implementation Details

Support script

A support script was made to define all the necessary data, such as aerodynamic data, geometric and mass properties, initial conditions, physical constants and launch location. This script also performs thrust and mass flow rate calculations given the motor and propellant characteristics.

The aerodynamic data, such as stability derivatives, centre of pressure location and drag coefficient, is stored in a matrix format where lines and columns identify different values for the aerodynamic angle of interest and Mach number, which determine different values for the aerodynamic properties themselves. These matrices are then fed into the simulation model that selects the correct values depending on the instantaneous values of the aerodynamic angles and Mach number, recurring to interpolation. This data can be obtained in several ways, as for instance by using open source software like Open Rocket and RASaero II, where a model of the rocket can be made and the aerodynamic properties retrieved, by performing CFD (Computational Fluid Dynamics) analysis of the rocket or even by conducting wind tunnel testing.

Regarding the thrust and mass flow rate computation, the script follows the equations detailed in Section 3.4 to determine the pressure evolution inside the combustion chamber and, consequently, the dynamic component of thrust and mass flow rate. In order to do so, the script must have the necessary solid propellant characteristics, such as number of grains, grain geometry and density, nozzle dimensions and thermodynamic properties of the combustion that can be obtained using available software, as for instance, NASA's CEA (Chemical Equilibrium Applications).

Mass properties subsystem

The mass properties subsystem is responsible for computing the evolution of the mass, moments of inertia and centre of mass throughout the burning phase.

The mass is obtained by integrating the previously computed mass flow rate, outputting the mass variation. Regarding the centre of mass, the previously mentioned assumption that its location is fixed in the longitudinal axis of the body and can only move in said axis, due to an axial symmetry of the rocket geometry and axially symmetrically positioned components, is used. This is valid for the majority of cases and small perturbations can be later introduced when designing the attitude controller. Taking into account that the varying mass element is the propellant, the centre of mass (measured from the tip of the nose) is then obtained through the following equation (incorporated in the Simulink model):

$$x_{cm} = \frac{x_{cmdry}\, m_{dry} + x_{cmprop}\,(m - m_{dry})}{m} \tag{7.1}$$

which corresponds to the ponderation between the centre of mass of the rocket without propellant and the centre of mass of the propellant, that may or may not shift depending on the grain geometry evolution. Figure 7.2 illustrates the reasoning behind this computation.

As for the moments of inertia, they are obtained externally, for instance, through CAD models of the

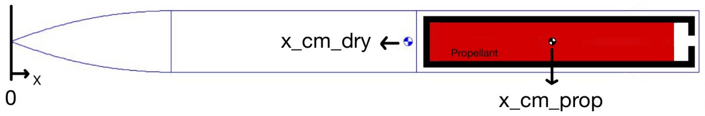


Figure 7.2: Centre of mass computation schematics

rocket, for different stages of propellant consumption, and are inserted into the script to be selected in this subsystem depending on the instantaneous mass of the vehicle.

Aerodynamic data subsystem

The Aerodynamic data subsystem outputs all the necessary aerodynamic coefficients to compute the aerodynamic forces and moments. To do so, it selects the correct values for the stability derivatives, Drag coefficient and position of the centre of pressure, according to the instantaneous values of the aerodynamic angles and Mach number. This selection is done using the look-up table Simulink block, that contains the stored data in matrix format from the support script.

Forces subsystems

As indicated by its name, the forces subsystem is responsible for computing all the forces detailed in Chapter 2: Gravity, Thrust and Aerodynamic. Regarding the propulsive force, it has two selectable modes, it can either use the dynamic thrust and mass flow rate computed in the support script and add the air pressure dependant static term, or to receive listed values for thrust and mass from an external file. This allows the user to either use the simulation model to obtain theoretical thrust data or to directly insert the already known values for the motor in case, for instance, from static fire tests. Additionally, the thrust force is decomposed in the three body axes according to the thrust vector control inputs (Equation 3.13).

This subsystem also has a built-in function that simulates the ground, so that the rocket doesn't ”fall” while the thrust-to weight-ratio is less than one during the ignition transient. This is obtained by simply setting gravity and thrust to zero until said ratio is equal to one.

Moments subsystem

Similarly to the forces subsystem, it refers to the equations from Chapter 2 to calculate the external moments applied to the rocket - aerodynamic and thrust vector caused. For the aerodynamic moments it takes into account the changing distance between the centre of pressure and centre of mass, which represents the moment arm. As for the thrust vector torque, the arm is given by the distance between the nozzle pivot point and the instantaneous position of the centre of mass.

TVC actuator modelling

The model for the actuator considers that the nozzle is actuated by two servo motors, one for the pitch control input μ_1 and the other for the yaw control μ_2. The response of a servo actuated system can be modelled using a continuous time first-order transfer function given by

$$\mu_r = \frac{1}{\tau\, s + 1}\, \mu \tag{7.2}$$

where μ_r is the actuator angular response and τ is the time constant. The time constant value can be adjusted according to the servo motor specifications. Additionally, servos normally have a maximum angular velocity value, also present in its specifications, which can be modelled by a adding a rate limiter block in Simulink.

As the actual system has not been designed, a ratio of 1:1 between the servos and nozzle angular rotations is assumed. The time constant and angular velocity limit values were retrieved from the specifications of the high class servos currently used by RED, and are equal to $0.02\,$s and 1 full rotation per second, respectively.

The actuation dynamics are very fast with respect to the system dynamics and so it is not expected for it to be a problem for the control system, even though they are not considered by the controller.

Velocities subsystem

The velocities subsystem is responsible for taking the forces and moments from the respective blocks and calculate both linear and angular velocities in the body axes, using the equations of motion from Section 3.6. To do so, the linear and angular accelerations have to be integrated in time, for a given set of initial conditions. It also computes the relative air velocity according to the wind speed and direction, and computes the norm of the linear velocity vector and the Mach number.

Attitude subsystem

In this subsystem, the Euler angles derivatives are computed using the set of equations 3.31, followed by an integrator block to obtain the angles.

Atmosphere subsystem

Atmospheric pressure and density will have an impact on the overall dynamics of the flight. On one side, as seen in Section 3.4, air pressure will impact the thrust force by originating a static term, and on the other, air density will impact the overall aerodynamic load on the vehicle, which is higher for larger air density. In this way, it is important to use a model of the atmosphere that characterises the variation of pressure and density with altitude. For reference, it was selected a standard atmosphere model used by NASA [39], which has three zones with separate curve fits for the troposphere, the lower stratosphere, and the upper stratosphere. It assumes that pressure and temperature change only with altitude, and derives density from the state equation.

Wind subsystem

Wind deeply affects the dynamics of the flight as it will determine, along with the inertial velocity, the relative velocity between the rocket and the atmosphere. As previously mentioned, it is the relative velocity that influences the aerodynamic forces and moments acting on the vehicle, specially since it determines the aerodynamic angles. In this way, it is necessary to have a model that specifies the wind velocity, in terms of magnitude and direction in the Earth frame. Moreover, it is important to model wind gusts as it will be one of the most relevant disturbances for the attitude controller to reject.

Two different options have been included in the model for the average wind speed and direction. The first is to simply define a constant wind velocity vector, which is a rough approximation but allows to observe the response of the rocket to a constant wind. The second option is to use the built-in Simulink wind model - *Horizontal Wind Model 07* [40]. This block implements the U.S. Naval Research Laboratory horizontal wind model routine to calculate the meridional and zonal components of the wind for a set of geographic coordinates: latitude, longitude, and altitude. The meridional and zonal components are interpreted as the East and North components, respectively, under the flat Earth assumption.

As for the wind gusts, the Dryden continuous turbulence model is used. As described in [33, 41], this model defines continuous gusts in terms of power spectral densities for the linear and angular velocity components, parameterised by turbulence length scales and intensities. The continuous gust is assumed to be stationary, homogeneous, isotropic and with a length scale larger than the vehicle. In simulink, there is a built-in block that implements the Dryden continuous turbulence model [42]. By feeding the transformation matrix from Earth to body to the block, the rocket's velocity in the Earth frame, and altitude, it outputs the wind turbulence expressed in the body-axes. These components are then added to the mean wind velocity components coming from the horizontal wind block after being transformed to the body frame.

7.1.2 Verification and Validation

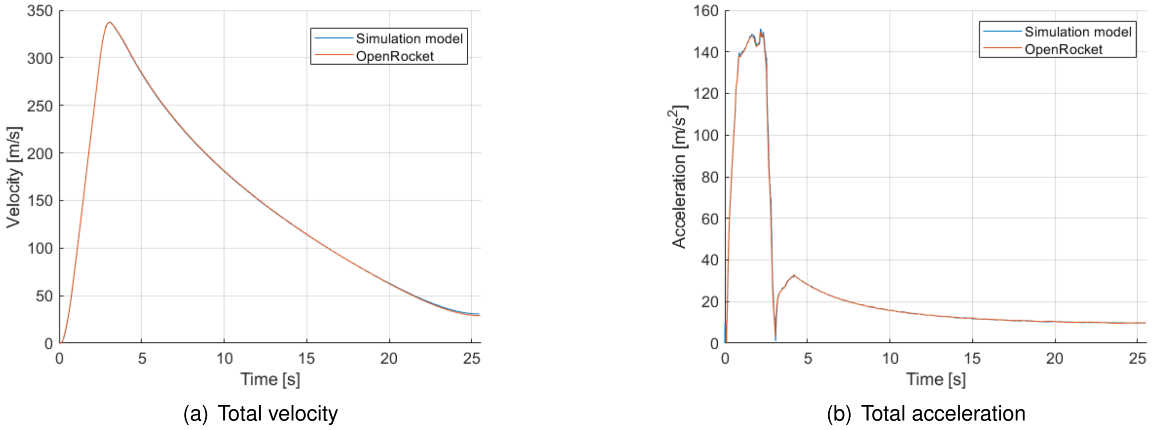
Before relying on the developed simulation model as a tool for the attitude controller design, it is essential to perform a verification and validation procedure. In order to verify the model, the simulation results obtained for the rocket built by RED for EuRoC 2022 - ”Baltasar” - were compared to the ones obtained through the open-source simulator *OpenRocket*. This software is the go-to rocket trajectory simulator in the field of rocketry and has been validated by several rocket launches. It is also the software used by international rocketry competitions, such as EuRoC, to determine launch safety for all rockets coming from the different contestants. ”Baltasar” is a 3 m long rocket, with a total mass of about 30 kg, aiming for a 3 km apogee. It relies on an active altitude control system, based on aerodynamic braking, to ensure that the 3 km mark is precisely achieved.

The tests were performed for a 5 m/s average windspeed, upwind launch scenario (directly against the wind), with an initial launch angle of 6 ° with the vertical. Additionally, the results were obtained for the case in which the braking system is not active. Table 7.1 displays some key flight parameters obtained in the simulation model and *OpenRocket*, with the respective relative error.

Table 7.1: Key flight parameters comparison.

Flight parameter	Simulation model	*OpenRocket*	Relative error (%)
Apogee	3457	3480 m	0.66 %
Max velocity	337.3	337.6 m/s	0.09 %
Max Mach number	1.000	1.002	0.20 %
Max acceleration	151.0 m/s^2	150.3 m/s^2	0.47 %
Time to apogee	25.5 s	25.6 s	0.39 %
Covered ground distance	940 m	900 m	4.44 %

Besides key flight parameters, it is also important to compare the time evolution of important flight variables to verify the simulation model. Figure 7.3 shows the total velocity and acceleration in the inertial frame obtained through both methods.



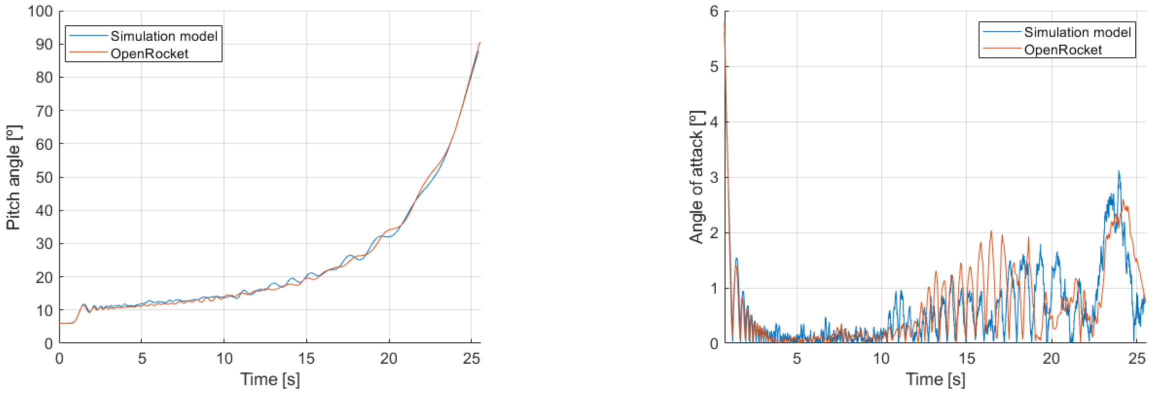
(a) Total velocity

(b) Total acceleration

Figure 7.3: Total velocity and acceleration comparisons.

The velocity and acceleration curves obtained though both methods are found to be extremely identical. Velocity has a peak value that coincides with the end of the burning phase, which can also be identified in the acceleration curve with a sudden drop after motor burnout. After the burnout, the rocket solely experiences the gravitational force and the aerodynamic forces (mostly drag). This leads to a velocity decrease up to apogee, close to which the acceleration is approximately equal to the gravitational acceleration.

As the main purpose of the simulation model is to serve the attitude controller design, the attitude values over time are of great importance and must also be checked against the open-source software. Since the simulated flight was for an upwind condition, only pitch is expected to significantly vary. In this way, Fig. 7.4 displays the pitch angle over time. Additionally, the aerodynamic angle of attack is also presented as it directly shows the wind influence. The pitch angle evolution is also very identical between both simulation methods. The initial pitch angle is equal to the launch rail inclination and, after exiting the rail, wind causes the rocket to turn, creating a stable oscillation in attitude. As the rocket climbs, pitch continues to increase and, close to apogee, the rocket has an approximately horizontal attitude. This is in agreement with the typical parabolic trajectory of sounding rockets. The reason behind the difference between both curves is the implemented wind gust model. As the altitude increases, wind gusts create an oscillatory response that depends on the gust intensity. Since each simulation will have random gust values, it is expected that the pitch angle response doesn't perfectly match. This reasoning is confirmed by the angle of attack time evolution, in which a deviation between both curves caused by different wind gust scenarios starts to appear approximately ten seconds after launch, when the rocket has reached an altitude of higher intensity gusts.

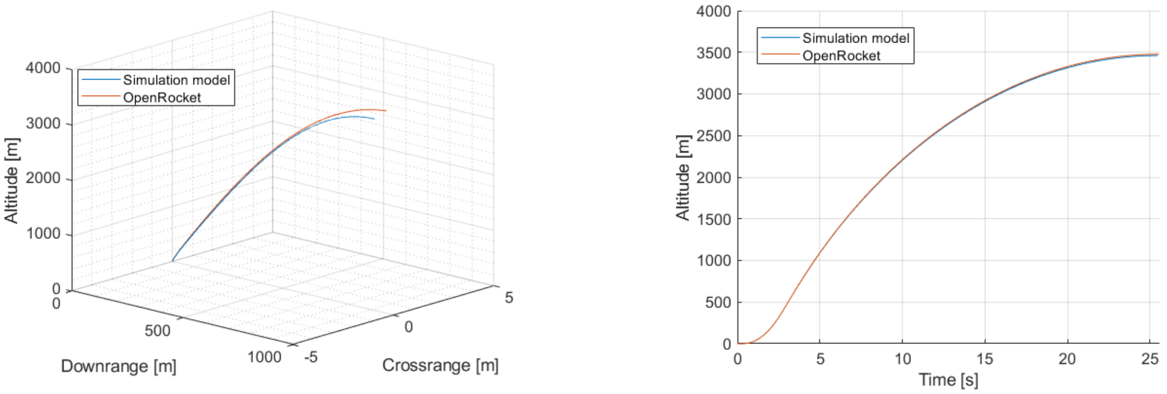


(a) Pitch angle

(b) Angle of attack

Figure 7.4: Pitch angle and angle of attack comparisons.

Finally, the three-dimensional trajectory plots coming from the simulation model and *OpenRocket* are shown in Figure 7.5, as well as the altitude profile.



(a) 3D trajectory

(b) Altitude profile

Figure 7.5: 3D trajectory and altitude profile comparisons.

Once again, the results coming from the developed simulation model are in agreement with the open-source software. The altitude profiles over time coincide, although with a small divergence as the apogee is approached. Looking at the 3D trajectory, it is expected that motion occurs only in the downrange plane since the rocket is launched against the wind. However, the trajectory obtained in *OpenRocket* shows a small crossrange displacement that justifies the difference in apogee and downrange displacement present in Tab. 7.1. This side displacement is due to Coriollis acceleration, a term that is neglected in the simulation model but not in *OpenRocket*. The fact that the maximum crossrange displacement, reached at apogee, is very reduced, allows to conclude that neglecting the Coriollis acceleration is an admissible assumption which will not compromise the validity of the simulation model as a tool for attitude controller design.

Looking at the results from Tab. 7.1, and Figs. 7.3, 7.4 and 7.5, it is possible to conclude that the simulation model was well implemented and correctly simulates the rocket flight, since the obtained simulation results are very close to the ones from *OpenRocket*. The fact that many aerodynamic parameters used in the simulation model are obtained through *OpenRocket*, further justifies the proximity in results.

7.2 LQI controller

After designing the controller, it is fundamental to test it in the complete non-linear simulation model of the system, in this case the, rocket. First of all, it is necessary to recall that the controller acts on the deviations from the nominal condition in the perturbation domain. In this way, the perturbations must be retrieved by subtracting the nominal value of the system variables to the total one, generically, $\delta x = x - x_0$. Similarly, the control input determined by the LQI has to be added to the nominal value, also referred to as the feedforward control.

Secondly, the gain matrices are obtained for different operating points along the nominal trajectory and must be selected, or scheduled, throughout the simulated flight. The operating points were selected as a function of time throughout the nominal trajectory. However, delays in the system may cause a wrong selection of the operating point. Instead, the gains are scheduled based on the measured altitude by storing the nominal altitude associated with each operating point. By using this method, not only a mismatch due to possible delays is prevented, but it is also ensured that altitude dependant terms, such as density and gravity, are consistent with the selected point. Nevertheless, significant deviations from the nominal trajectory can still cause a mismatch between the selected operating point and the true dynamics, for instance, in terms of velocity, Thrust force, and mass. The gain scheduling is done by interpolation, meaning that the pre-calculated gains for the defined operating points are stored in vectors and are interpolated throughout the simulation using the built-in Simulink look-up table block, with the linear interpolation option selected. The resultant control system high-level implementation is represented in the following diagram (Fig.7.6).

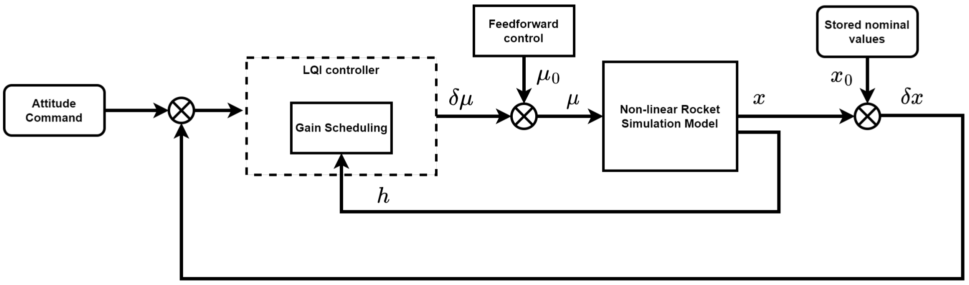
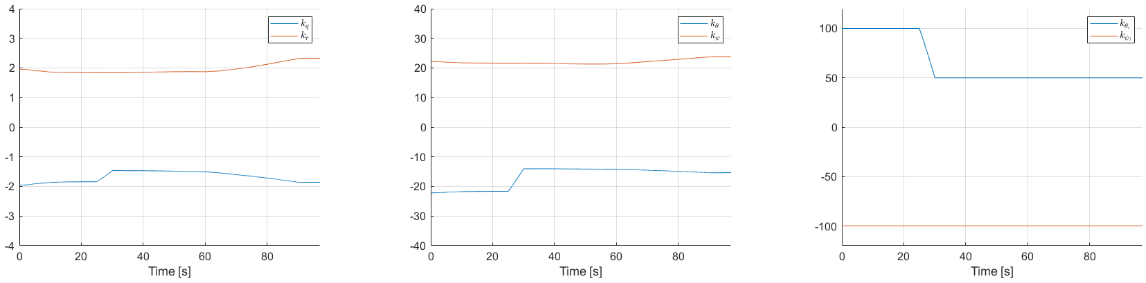


Figure 7.6: LQI implementation.

Regarding the final tuning of the controller, it is verified in simulation that the selected operating points and tuning matrices during the design phase provided good performance and so were kept the same. This results in the following time evolution for the different gains (Fig. 7.7).



(a) k_q and k_r (b) k_θ and k_ψ (c) k_{θ_i} and k_{ψ_i}

Figure 7.7: Controller gains over time.

The controller gains have a small variation during the trajectory. For the longitudinal mode, where the control effort was purposely reduced during the pitch manoeuvre to avoid saturation, a decrease in the magnitude is seen after the start of said manoeuvre. The relative magnitude of the gains for the angular rates, Euler angles, and integral action, is also in agreement with the tuning matrices. The integral action gain is seen to be very independent from the time-varying dynamics, only changing for the pitch angle, also due to the intended decrease in control effort.

7.3 Adaptive LQI controller with parameter estimation

As opposed to the regular LQI controller, the adaptive one uses the on-board estimates regarding the system parameters to calculate the state-space matrices and, consequently, to obtain the controller gains by solving the Riccati equation in real time. The parameters are estimated by resorting to the stored system data, which is a function of flight variables, with the aerodynamic parameters being corrected by the estimated correction factors from the LPE.

While in the off-line controller the gains have to be scheduled using interpolation on the stored values, for the adaptive controller it is necessary to define at which points the gains are updated. In simulation this was done by selecting a time interval between two consecutive computations of one second. In a real implementation this value has to be selected taking into account the available computation power and by looking at the rate of change in system dynamics.

7.4 Navigation System

The developed navigation system is implemented in simulation to test its performance with the non-linear rocket model. The first step is to introduce a model for the onboard sensors. According to Chapter 6, the sensors are modelled by adding Gaussian white noise to the measured state variables. This is done using the band-limited white noise block in Simulink, with the noise properties being retrieved by the specification of the sensors used by RED, namely, the VN-200 from Vectornav®. The VN-200 is a high performance GNSS-Aided Inertial Navigation System, containing 3-axis gyros, accelerometers and magnetometers, a high-sensitivity GNSS receiver. Although providing its own estimation and filtering, the unfiltered specifications for each sensor were used to test the designed navigation system.

The estimator composed by the combination of the ACF, the PCF, and the LPE, is implemented in simulation according to the scheme present in Fig. 6.1. The three individual estimators all rely on Kalman filtering and so, the tuning parameters were the Q and R matrices, respectively the process and output noise covariance matrices. The ACF and PCF use constant Kalman gains given by solution to the steady-state Riccati solution, considering the time-invariant equivalents of both process equations. On the other hand, the LPE uses the time-varying solution, as the process equation is constantly updated by the inputs from the other filters. In this way, the built-in continuous time Kalman filter block was used for the LPE.

7.5 Complete ADCS

Finally, two diagrams representing the implementation of the control system for each type of controller, alongside the navigation system in the simulation model is shown (Figs. 7.8 and 7.9). These diagrams represent the two versions of the implemented complete Attitude Determination and Control System for a rocket with TVC.

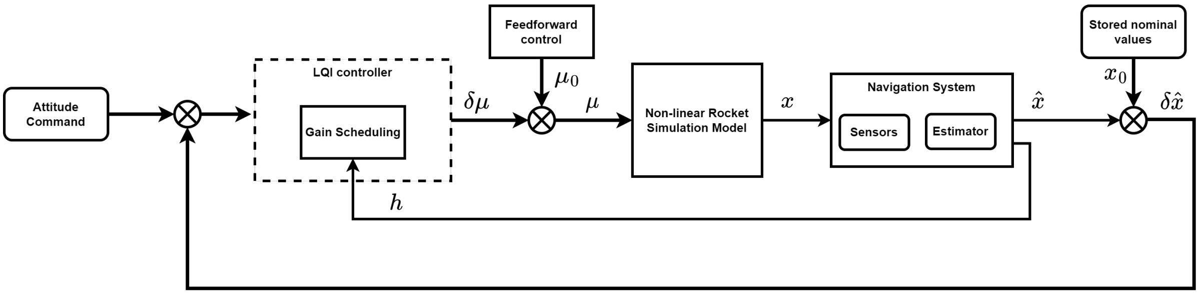


Figure 7.8: ADCS block diagram with regular controller.

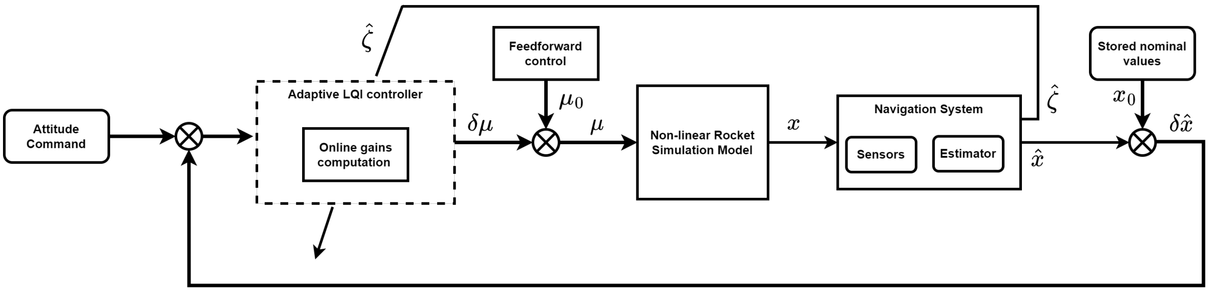


Figure 7.9: ADCS block diagram with adaptive controller.

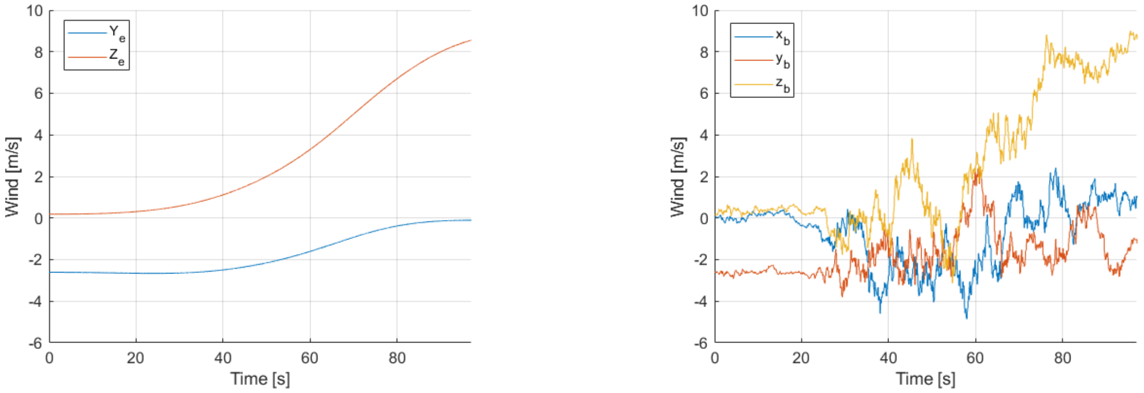
Chapter 8

Simulation Results

8.1 LQI control

Following the implementation in simulation described in Chapter 7, and using the complete non-linear model of the rocket, the performance of the developed LQI controller is analysed in the time domain. The described nominal trajectory for the reference rocket model is used and, initially, no sensor noise is added to the system. Since for nominal conditions only the feedforward control action is needed, in order to test the controller it is necessary to either add perturbations, or to provide non-null attitude references with respect to the nominal evolution.

In this way, the first tests are performed by adding wind as external perturbation, according to the already described wind model. Firstly, only the average horizontal wind model is used, and then, the wind gusts given by the Dryden model are introduced. Figure 8.1 shows the wind components, with and without gusts.

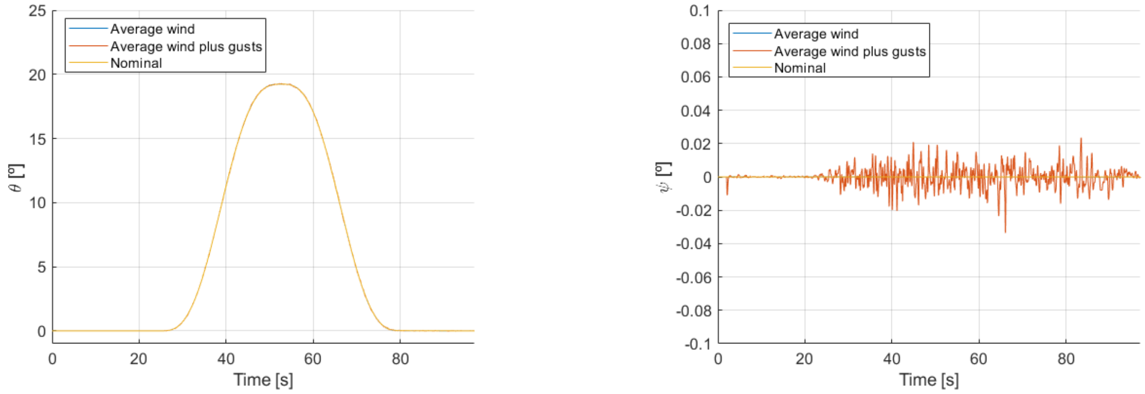


(a) Average wind components in the Earth frame

(b) Total wind components (with gusts) in the body frame.

Figure 8.1: Wind components used in simulation.

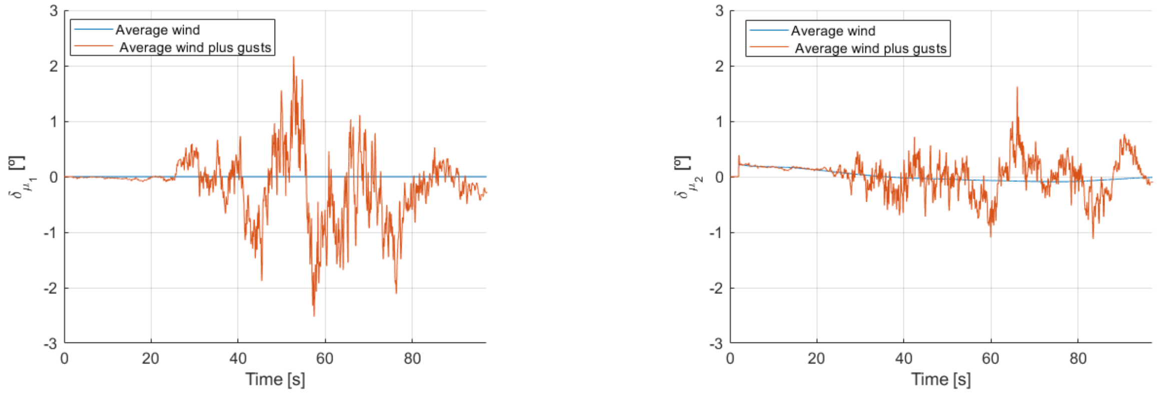
For both wind conditions, the LQI controller is able to to reject the perturbation and the pitch and yaw attitude references are correctly followed throughout the trajectory. Figure 8.2 details the evolution of said angles compared against the nominal desired one. The associated control input adjustments, determined by the controller, are shown in Fig.8.3.



(a) Pitch angle (θ)

(b) Yaw angle (ψ)

Figure 8.2: Attitude over time.



(a) Pitch control input adjustment (δ_{μ_1})

(b) Yaw control input adjustment (δ_{μ_2})

Figure 8.3: Control inputs over time.

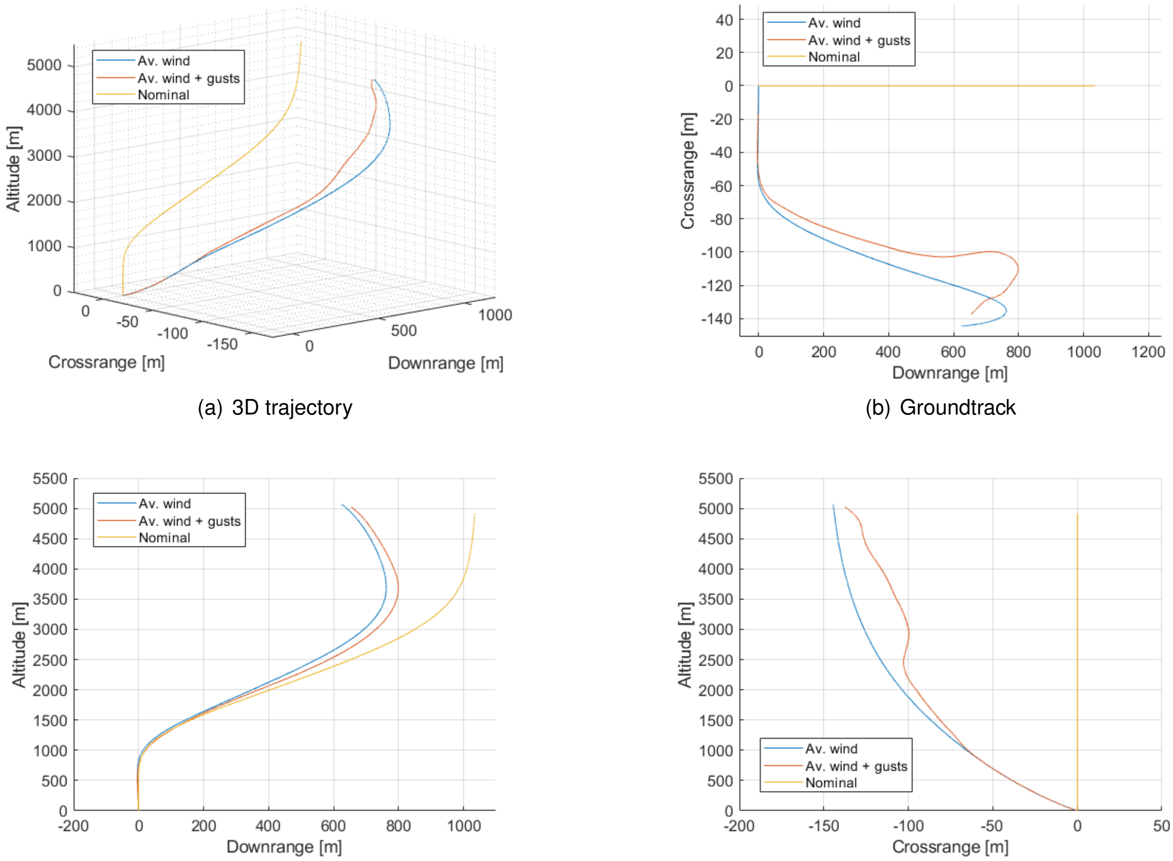
To better understand the simulation results, the sum squared error of the pitch and yaw angle, and the squared mean root of the control inputs, throughout both trajectories, are shown in Tab. 8.1 using degrees as unit. In addition, a solution with one PID controller per degree of freedom (one for pitch and one for yaw) was implemented to compare the results.

Table 8.1: Detailed simulation results.

	Average wind		Average wind + gusts	
	LQI	PID	LQI	PID
$\Sigma{\theta_e}^2$	3.15E-04	0.0173	2.6083	15.0879
$\Sigma{\psi_e}^2$	0.0016	1631.3	0.2457	1632.7
$\delta_{\mu_1,rms}$	0.2336	0.2299	0.6109	0.6125
$\delta_{\mu_2,rms}$	0.1512	177.88	0.3273	177.91

The designed LQI controller shows overall better performance than the PID controller. For the yaw angle and control input, the PID has significantly large values, which are due to the error caused by the initial wind perturbation in the yaw plane that the PID controller has difficulty in rejecting. For the pitch angle, the tracking performance is closer but the LQI is still a better option, specially considering the similar control effort given by the root mean square of the input. It is noted that with the addition of wind gusts the reference tracking performance is decreased and the control effort increases. Given the additional perturbations and added noise caused by the wind gusts, this was an expected result.

Additionally, the three-dimensional trajectory, alongside the two-dimensional representation in each plane, is shown (Fig.8.4).



(a) 3D trajectory

(b) Groundtrack

(c) Pitch plane

(d) Yaw plane

Figure 8.4: Trajectory.

As expected, the trajectories in the presence of wind don't match the nominal trajectory in terms of position. The controller tracks the desired attitude evolution with respect to the inertial frame, however, wind causes the rocket to drift in terms of position as it is moving with it. To ensure position tracking, an additional position controller that dictates the real-time attitude reference for the attitude control system would be necessary. The fact that the control system still ensures satisfactory attitude tracking when the rocket has considerably deviated from the nominal trajectory in terms of position, demonstrates the robustness of the controller to variations in spatial dependent parameters, like the air density.

After verifying the correct functioning of the control system in the presence of perturbations, the response to non-zero attitude commands, with respect to the nominal evolution, can be studied. As done in Chapter 5, for the linear domain performance analysis, 3° steps were requested to the system, both for pitch and yaw, in different points of the trajectory. Table 8.2 details the obtained step response performance for two distinct points of the trajectory, obtained for the case with wind gusts. The results obtained with the implemented PID are also shown for comparison.

Table 8.2: Step response performance.

	$t = 5$ s				$t = 40$ s			
	LQI		PID		LQI		PID	
	θ	ψ	θ	ψ	θ	ψ	θ	ψ
Rise time (s)	0.26	0.26	0.77	0.50	0.32	0.25	0.70	0.58
Settling time (s)	0.48	0.48	3.54	2.85	0.56	0.46	5.92	4.65
Overshoot (%)	0.21	0.14	8.69	28.17	1.23	0.69	10.29	18.98

Once again, the LQI controller outperforms the PID controller, demonstrating significantly better step response characteristics. More specifically, the settling time is much lower, as well as the overshoot, which in the case of the PID is concerningly high. The results obtained for the LQI in simulation are slightly better than the ones obtained in the linear domain analysis, which might indicate that the controller performs better when the time varying dynamics are being considered. Lastly, the expected decrease in performance for the pitch angle control during the pitch manoeuvre is present, demonstrating higher settling time and overshoot.

8.1.1 Robustness analysis

The control system has demonstrated satisfactory performance in simulation so far. However, all the parameters used in the non-linear model are in agreement with the ones assumed for the control system design in the linear domain. To test the robustness of the controller to model uncertainty, it is necessary to characterize its performance when the parameters vary from the assumed values. The parameters selected for the analysis were the dry mass (M_{dry}), the centre of mass position (x_{cm}), which will influence the control torque arm (l) and the static margin (SM), the transverse inertia J_t, the aerodynamic force coefficients (C_A and C_N), which directly influence the aerodynamic moment coefficients (C_m and C_n), and the thrust force (T). Each parameter was varied independently, and the variation was done in percentage of the original value. The results for the tracking performance and control effort were obtained including wind gusts and are present in Tables 8.3 to 8.8, where the wind with gusts included was used.

Table 8.3: M_{dry} variation.

(x) M_{dry}	0.9	0.95	1	1.05	1.1
$\Sigma{\theta_e}^2$	3.45	3.02	2.60	2.11	1.44
$\Sigma{\psi_e}^2$	0.51	0.37	0.25	0.14	0.06
$\delta_{\mu_1,rms}$	0.71	0.67	0.61	0.55	0.47
$\delta_{\mu_2,rms}$	0.42	0.38	0.33	0.26	0.20

Table 8.4: J_t variation.

(x) J_t	0.8	0.9	1	1.1	1.2
$\Sigma{\theta_e}^2$	2.45	2.53	2.60	2.70	2.79
$\Sigma{\psi_e}^2$	0.23	0.24	0.25	0.26	0.27
$\delta_{\mu_1,rms}$	0.61	0.61	0.61	0.61	0.61
$\delta_{\mu_2,rms}$	0.33	0.33	0.33	0.33	0.33

Table 8.5: x_{cm} variation.

(x) x_{cm}	0.8	0.9	1	1.1	1.2
$\Sigma{\theta_e}^2$	0.61	1.31	2.60	4.85	9.00
$\Sigma{\psi_e}^2$	0.07	0.14	0.25	0.45	0.84
$\delta_{\mu_1,rms}$	0.33	0.46	0.61	0.78	0.98
$\delta_{\mu_2,rms}$	0.19	0.25	0.33	0.42	0.54

Table 8.6: T variation.

(x) T	0.96	0.98	1	1.02	1.04
$\Sigma{\theta_e}^2$	2.10	2.36	2.60	2.74	2.84
$\Sigma{\psi_e}^2$	0.12	0.19	0.25	0.31	0.36
$\delta_{\mu_1,rms}$	0.54	0.57	0.61	0.64	0.66
$\delta_{\mu_2,rms}$	0.25	0.29	0.33	0.35	0.37

Table 8.7: C_A variation.

(x) C_A	0.8	0.9	1	1.1	1.2
$\Sigma{\theta_e}^2$	2.79	2.70	2.60	2.52	2.43
$\Sigma{\psi_e}^2$	0.28	0.26	0.25	0.23	0.22
$\delta_{\mu_1,rms}$	0.68	0.70	0.65	0.59	0.57
$\delta_{\mu_2,rms}$	0.35	0.38	0.33	0.36	0.31

Table 8.8: C_N variation.

(x) C_N	0.8	0.9	1	1.1	1.2
$\Sigma{\theta_e}^2$	1.83	2.18	2.60	3.03	3.46
$\Sigma{\psi_e}^2$	0.23	0.24	0.25	0.25	0.26
$\delta_{\mu_1,rms}$	0.55	0.57	0.61	0.66	0.71
$\delta_{\mu_2,rms}$	0.31	0.32	0.33	0.33	0.34

First of all, the controller has demonstrated to be very robust to model uncertainty as it functioned with satisfactory performance for all the parameter variations, inside the defined ranges of percentage variation.

Starting from the dry mass (M_{dry}), the defined range was $\pm 5\,\%$, since higher variations would compromise the thrust-to-weight ratio. Nevertheless, it was one of the parameters causing higher performance variation. The tracking performance increases with mass and the control effort decreases.

The transverse inertia value (J_t) caused small performance variation, even with an higher range of variation with respect to the dry mass. In this case, the performance of the control system decreases with the increase in inertia. This can be justified by the direct influence of the transverse inertia in the control authority, since an higher inertia will require an higher control torque for the same rotational motion.

The centre of mass (x_{cm}) is the parameter that most influenced the control system performance. A lower value, meaning a position closer to the tip of the rocket, causes the moment arm for the thrust vector actuation to be higher, which increases the control authority. At the same time, the natural instability of the rocket reduces. In this way, the tracking performance increases when the centre of mass moves closer to the tip, while the control effort decreases.

Similarly to the dry mass, the thrust (T) was varied in a lower percentage to allow for a reasonable thrust-to-weight ratio. With the increase in thrust, the performance of the control system decreases. A possible justification is the increase of the velocity of the rocket, which poses an harder task for the attitude tracking by the control system. Looking at the results for the axial force coefficient (C_A), a decrease in the coefficient causes a worse performance, which also indicates that an higher velocity is the cause. However, the variation in performance for the axial force coefficient is reduced, as it does not influence the rotational motion.

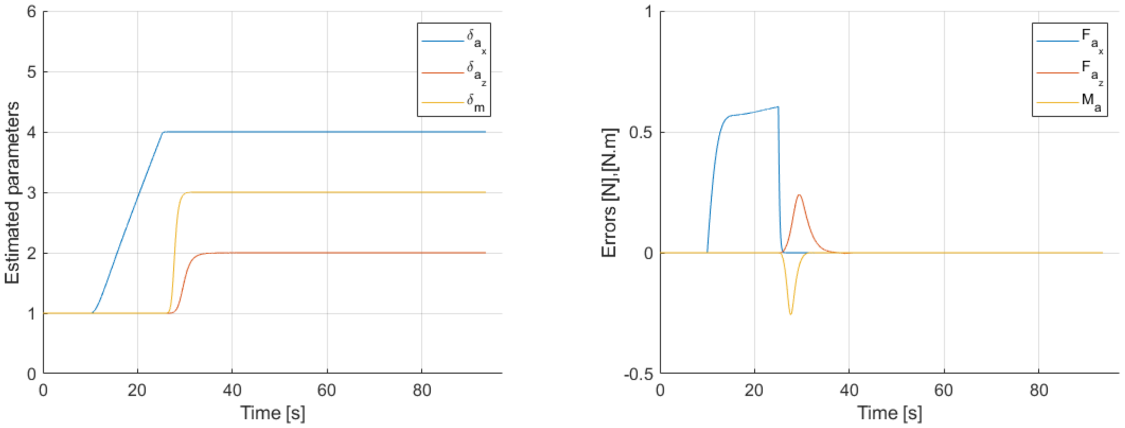
Finally, the normal force coefficient (C_N) is seen to cause a considerable variation in performance. In this case, the coefficient directly influences the rotational motion as it will impact the aerodynamic moment. An higher value of the coefficient causes a worse performance by the control system since the external aerodynamic force and moment will have an higher influence, also amplifying the effect of wind gusts.

8.2 Adaptive LQI control

As already described, the adaptive controller computes the gains in real time, using the estimates on the aerodynamic correction factors, coming form the linear parameter estimator, to update the state-space dynamics matrix.

8.2.1 LPE

Starting by the Linear Parameter Estimator, in order to test its performance, a set of simulations with induced errors on the parameters were performed, outputting satisfactory results. Initially, only the nominal trajectory was used, without adding perturbations. The results here shown (Fig. 8.5) are for a simulation where the real C_A value was multiplied by a ramp with $0.2/sec$ slope, starting at 1 up to 4, a constant multiplication factor of 2 on C_N was added, and a constant multiplication factor of 3 was used on C_m. The selected aerodynamic parameters are related to pitch plane motion since it is the only plane of motion during the nominal trajectory.



(a) Estimated parameters (δ_{a_x}, δ_{a_x}, and δ_m)

(b) Aerodynamic froces and moments estimation errors

Figure 8.5: LPE simulation results.

The simulation results for the LPE were positive, the estimates on the correction factors allowed for a limited error in the aerodynamic forces and moments estimation. Even though the linear model assumed constant parameters, the estimator performed well under varying conditions, as for instance with the axial aerodynamic force where an approximately constant reduced error was obtained during the ramp multiplication. As expected, the estimates on the normal aerodynamic force and pitching aerodynamic moment correction factors (δ_{a_z}, and δ_m) only starts to converging when the pitch manoeuvre initiates, as no information regarding side motion is available up to then. Figure 8.6 displays the errors in the aerodynamic forces and moments when the estimator is turned-off. It is possible to see that without the correction provided by the estimator the errors are quite large, once again demonstrating the good performance of the estimator.

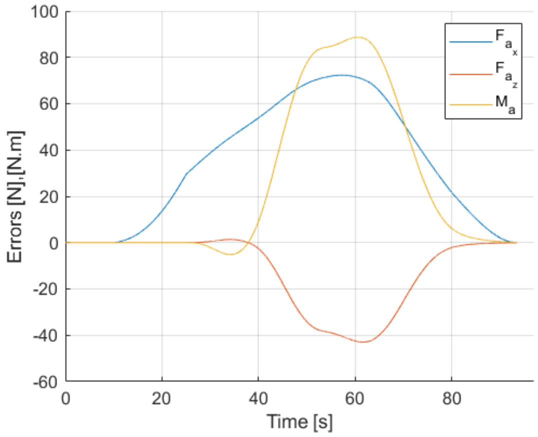


Figure 8.6: Aerodynamic forces and moments errors without estimation.

8.2.2 Adaptive control

The adaptive controller was initially tested without adding parameter uncertainty. As expected, the results were similar to the ones obtained for the regular LQI controller. Even though the adaptive controller is using the exact values for all parameters, the improvement in results is marginal. This is due to the already demonstrated robustness of the off-line LQI control, which has a very constant performance even when the parameters are varied.

Secondly, errors were introduced in the aerodynamic coefficients, as done in the LPE testing, to determine if the adaptive controller is able to achieve better performance by using the aerodynamic correction factors estimates to update the dynamics matrix. For these tests, it was also verified that the adaptive controller wasn't able to produce better results than the regular LQI controller with gain scheduling. Once again, the robustness of the original system is pointed as the cause of this similarity in results.

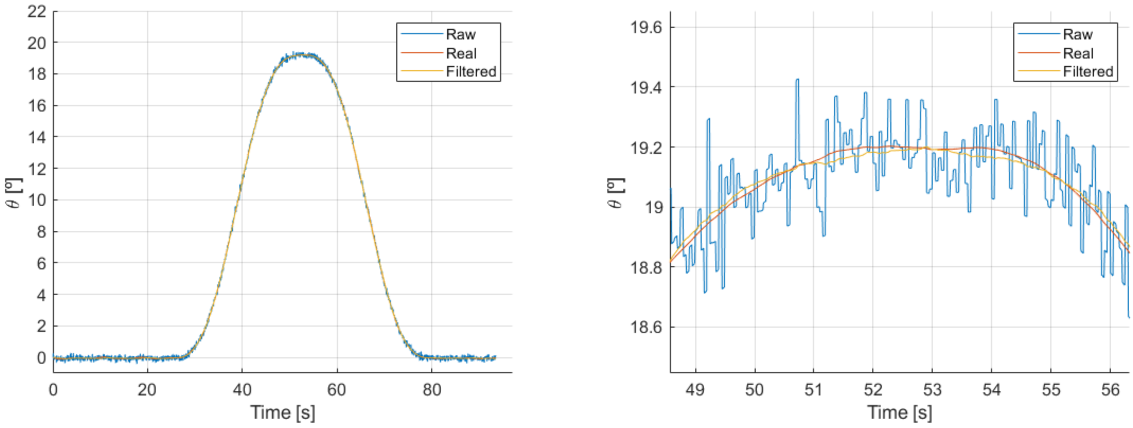
Knowing that the adaptive controller is not able to produce better results in simulation, the off-line LQI control is selected as the best option since, on one side, it demands less computational power for a real implementation and, on the other, is less subjected to errors associated with parameter estimation, that can cause undesirable controller gains which might destabilize the system.

8.3 Navigation System

The performance of the Navigation system was also verified in simulation by adding the sensors and respective noise in the loop. The results are presented for the nominal trajectory, and the performance of each filter is separately evaluated.

8.3.1 ACF

The ACF is responsible for filtering the Euler angles and estimating the bias of the gyroscope. As pitch is the only angle varying, it is selected to display the performance of the ACF. Figure 8.7 shows the raw, real, and filtered values for the pitch angle throughout the trajectory.



(a) Full trajectory

(b) Zoomed in the maximum pitch interval

Figure 8.7: ACF attitude estimation.

It is possible to conclude that the ACF is able to reject the noise introduced by the sensors and to provide an accurate estimate on the attitude of the rocket. To test the bias estimation, a constant 0.1° /s bias was added. Figure 8.8 displays the biased reading of the pitch rate from the gyroscope and the unbiased estimate obtained after correcting it with the bias estimate. The bias is properly correct by the ACF, however, the pitch rate measurement still carries some noise as no direct filtering is being applied to the angular rates readings.

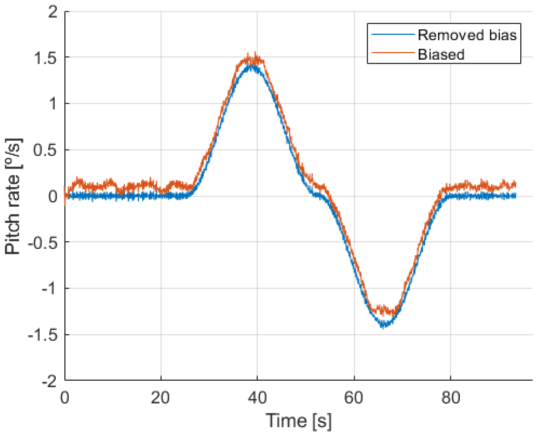
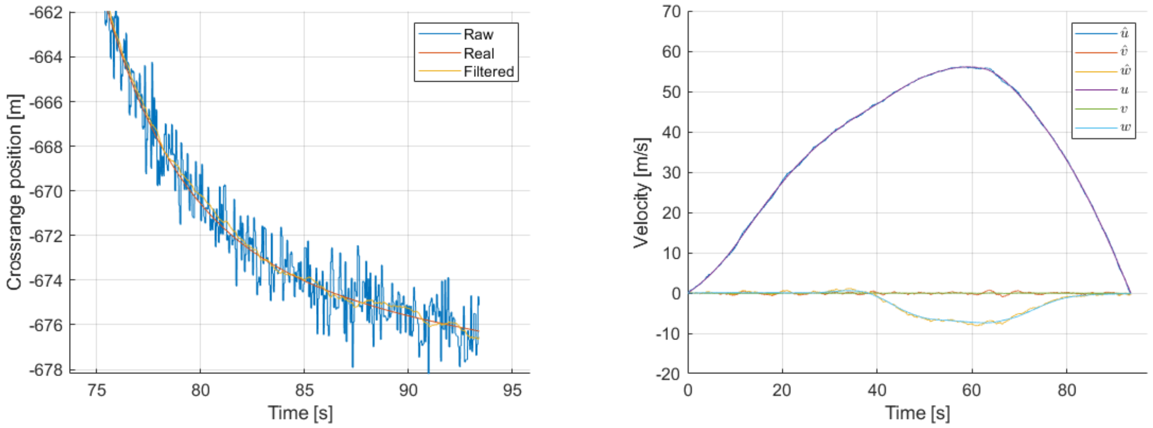


Figure 8.8: ACF bias estimation.

8.3.2 PCF

Regarding the PCF, it was also possible to verify its correct functioning by analysing the position and velocity estimates. Figure 8.9 shows a zomed section of the crossrange position estimation to better understand the filtering done by the PCF, and the linear velocity components estimation given in the body frame. The position filtering is able to reject the noise from the measurements while maintaining good accuracy with respect to the true value. The filter also provides accurate measurements on the velocity, a variable that is not directly measured by any sensor.



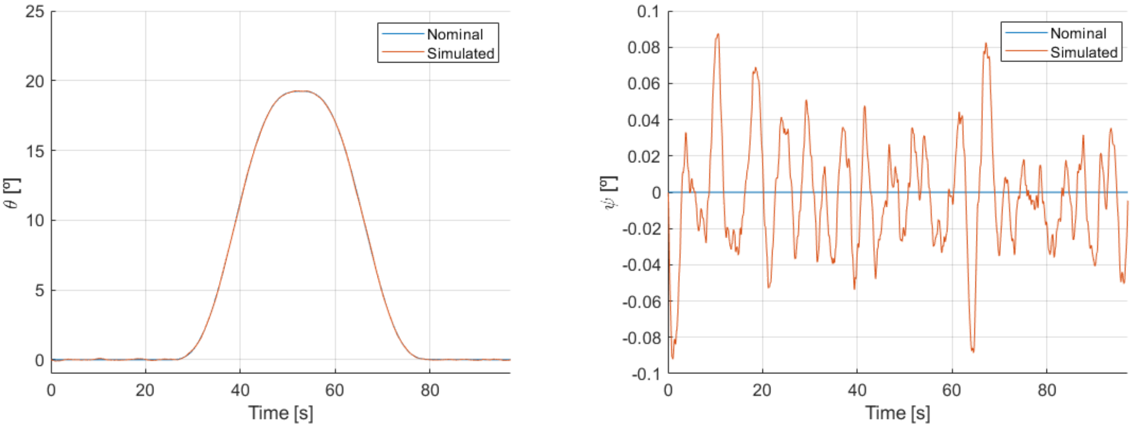
(a) Position estimation

(b) Velocity estimation

Figure 8.9: PCF position and velocity estimation.

8.4 Complete ADCS

Having analysed all the individual systems, it is possible to detail the performance of the designed ADCS, using the off-line LQI controller and the developed navigation system. With the inclusion of the navigation system, the effect of the sensor's noise on the control system can be studied. The results here shown were obtained considering as external perturbation the wind with gusts included.

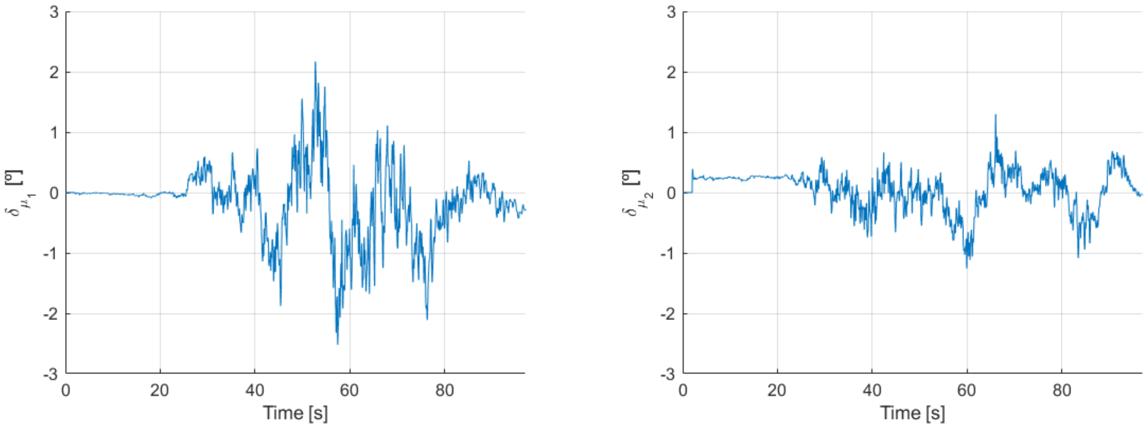


(a) Pitch angle (θ)

(b) Yaw angle (ψ)

Figure 8.10: Attitude over time.

It is possible to conclude that the control system still correctly functions with the addition of the navigation system and respective sensors. As predicted, a decrease in the attitude tracking performance caused by the addition of noise is noted. The control effort has not significantly changed since the wind



(a) Pitch control input adjustment (δ_{μ_1})

(b) Yaw control input adjustment (δ_{μ_2})

Figure 8.11: Control inputs over time.

gusts were already a noise source in the system for the actuators even before adding the noise from the sensors. Table 8.9 details the performance of the ADCS in comparison with the previously obtained results for the control system alone.

Table 8.9: ADCS simulation results.

	Control system alone	Complete ADCS
$\Sigma{\theta_e}^2$	2.6083	15.3665
$\Sigma{\psi_e}^2$	0.2457	11.9733
$\delta_{\mu_1,rms}$	0.6109	0.6309
$\delta_{\mu_2,rms}$	0.3273	0.3176

Chapter 9

Conclusions

9.1 Achievements

With the conclusion of this work, it is possible to state that the primary goal has been achieved: the successful design of an attitude determination and control system applicable to sounding rockets with thrust vectoring. The design process was described in a generic way to ensure that the system can be easily applied to different vehicles under the same category. Nevertheless, the future implementation of the system in a student-built sounding rocket was always taken into account, as it was the initial motivation behind this work.

The first major achievement was the development of a six-degrees-of-freedom non-linear model for the dynamics and kinematics of a generic sounding rocket. The same model was translated into simulation environment that served as tool for the ADCS design. After the implementation in simulation, the model was validated by comparing the results to a standard open-source sounding rocket simulator, with a very close agreement being verified. The generic non-linear model was then particularized for a finless thrust-vector-controlled sounding rocket

With the non-linear model already developed, the following milestone was the derivation of a generic linear version using Taylor series expansion at multiple operating points. With the linearization, an original linear, time-varying, state-space representation was obtained, which would be used for the linear control system design.

To test the derived linear models and, subsequently, the ADCS, it was necessary to use a reference sounding rocket to provide the system parameters. In this way, a preliminary design to a future sounding rocket with TVC, to be built by RED, was performed. To serve as testing platform for TVC systems, certain requirements were imposed on the vehicle, such as to have a fully controlled ascent. Moreover, the design was done considering solid propulsion, the technology that is currently used by RED. All these factors lead to an original and, at the same time, practical design to facilitate the future implementation. Moreover, a nominal trajectory in terms of attitude over time was defined to meet certain mission requirements and be the foundation of the control system testing.

The following achievement was the successful control system design using a linear, optimal control tool, the linear quadratic regulator. The controller design was based on the time-varying state-space representation of the system and included an integral action. This controller proved to be robust and significantly outperformed the PID control classical solution.

As a method to increase the robustness of the control system, a linear parameter estimator that relies on the developed rocket dynamics model was designed. The objective was to estimate a correction factor for the onboard calculations of the aerodynamic forces and moments, and then to used said estimates to feed an adaptive version of controller with online gain computation. Although the linear parameter estimator has demonstrated good estimation performance, the adaptive controller was not able to improve the results obtained with its non-adaptive counterpart. The identified reason was the robustness of the original control system to errors in the aerodynamic parameters.

Another achievement was the design and implementation of a navigation system that includes the linear parameter estimator and two additional filters to estimate the rocket's state vector. The underlying technique behind the estimators was the Kalman filtering, and the results obtained in simulation demonstrated satisfactory estimation accuracy and noise rejection.

The final achievement was the successful interconnection of the control and navigation systems composing the complete attitude determination and control system.

9.2 Future Work

After verifying the proposed system in simulation, the next step is the implementation in a real vehicle, which might include more detailed verification and validation procedures such as software and hardware in the loop testing. RED is currently developing small scale prototypes to test the TVC technology and the associated navigation and control systems. In this way, it is intended to implement the techniques in here developed to such prototypes and to analyse all the results coming from test campaigns.

Regarding the possible improvements on the system, it would be of interest to develop non-linear controllers for the attitude control problem in order to compare the performance of said controllers with the developed ones. Particularly, the designed linear parameter estimator could be used for a non-linear control system that requires accurate information on the aerodynamic forces and moments to guarantee its correct functioning.

Moreover, both the developed simulation model and navigation system can be verified and validated using real flight data from sounding rockets launched by RED.

Bibliography

[1] T. Barrows and J. Orr. *Dynamics and Simulation of Flexible Rockets*. Academic Press, 2021. ISBN: 978-0-12-819994-7.

[2] J. W. Cornelisse, H. F. R. Schöyer, and K. F. Wakker. *Rocket Propulsion and Spaceflight Dynamics*. Pitman, 1979. ISBN:0-273-01141-3.

[3] J. Barrowman. Stability of a Model Rocket in Flight. Technical report, Centuri, 1970.

[4] V. I. Feodosiev and G. B. Siniarev. *Introduction to Rocket Technology*. Academic Press, 1959.

[5] J. W. Jang, A. Alaniz, R. Hall, N. Bedrossian, C. Hall, and M. Jackson. Design of launch vehicle flight control systems using ascent vehicle stability analysis tool. Aug. 2011. doi: 10.2514/6.2011-6652.

[6] A. Tewari. *Advanced Control of Aircraft, Spacecraft and Rockets*. John Willey & sons, 2011. ISBN: 978-1-119-97120-7.

[7] L. Ljunge. Guidance, Navigation and Control Systems. Review, Analysis and Future Design. Technical report, European Space Agency (ESA), 2009.

[8] G. P. Sutton and O. Biblarz. *Rocket Propulsion Elements*. John Willey & Sons, 9^{th} edition, 2017. ISBN:9781118753880.

[9] B. Wie, W. Du, and M. Whorton. Analysis and design of launch vehicle flight control systems. *AIAA Guidance, Navigation and Control Conference and Exhibit*, Aug. 2008. doi: 10.2514/6.2008-6291.

[10] R. Sumathi and M. Usha. Pitch and yaw attitude control of a rocket engine using hybrid fuzzy- pid controller. *The Open Automation and Control Systems Journal*, 6:29–39, Apr. 2014. doi: 10.2174/1874444301406010029.

[11] L. Sopegno, P. Livreri, M. Stefanovic, and K. P. Valavanis. Linear quadratic regulator: A simple thrust vector control system for rockets. In *2022 30th Mediterranean Conference on Control and Automation (MED)*, pages 591–597, 2022. doi: 10.1109/MED54222.2022.9837125.

[12] Y. Lei, Z. Yong-li, L. Teng-hui, and S. Bin. A novel robust time-varying modal parameter estimation method based on absolute deviation for missile and rocket. In *2020 3rd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM)*, pages 458–462, 2020. doi: 10.1109/WCMEIM52463.2020.00102.

[13] M. Lungu, R. Lungu, and L. Sepcu. Nonlinear adaptive system for the automatic control of rockets' motion in vertical plane. Feb. 2013.

[14] E. M. Marconi. What is a Sounding Rocket? URL: `https://www.nasa.gov/missions/research/f_sounding.html`. Online (accessed: 29.12.2021).

[15] N. Sentse and M. Sabbatini. ESA User Guide to Low Gravity Platforms. Technical report, European Space Agency (ESA), Dec. 2014.

[16] F. Collette, Y. Kempf, and M. Karas. Sounding Rockets: Principle, Functioning and Applications. Technical report, Warsaw School of Economics, 2007.

[17] National Aeronautics and Space Admnistration (NASA). Forces on a rocket. URL: `https://www.grc.nasa.gov/www/k-12/rocket/rktfor.html`, . Online (accessed: 20.01.2022).

[18] J. D. Anderson. *Fundamentals of Aerodynamics*. McGraw-Hill, 5^{th} edition, 2010. ISBN:978-0-07-339810-5.

[19] National Aeronautics and Space Admnistration (NASA). What is Lift? URL: `https://www.grc.nasa.gov/www/k-12/airplane/lift1.html`, . Online (accessed: 20.01.2022).

[20] Tec-science. Drag coefficient (friction and pressure drag). URL: `https://www.tec-science.com/mechanics/gases-and-liquids/drag-coefficient-friction-and-pressure-drag/#Profile_drag_coefficient_overall_drag_coefficient`. Online (accessed: 20.01.2022).

[21] National Aeronautics and Space Admnistration (NASA). What is Drag? URL: `https://www.grc.nasa.gov/www/k-12/airplane/drag1.html`, . Online (accessed: 20.01.2022).

[22] J. Ettl and J. Pfänder. Rate control system for sounding rockets. *SP-671*, Sept. 2009.

[23] National Aeronautics and Space Admnistration (NASA). Examples of Controls. URL: `https://www.grc.nasa.gov/WWW/K-12/rocket/rktcontrl.html`, . Online (accessed: 10.01.2022).

[24] C. B. Ensworth. Thrust vector control for nuclear thermal rockets. *American Institute of Aeronautics and Astronautics (AIAA)*, Oct. 2013. doi: 10.2514/6.2013-4075.

[25] Aerospaceweb. Missile Control Systems. URL: `http://www.aerospaceweb.org/question/weapons/q0158.shtml`. Online (accessed: 15.01.2022).

[26] T. Vu, N. Rongione, and A. Rahmani. Minimalist control design of canard-actuated micro-projectiles. In *SoutheastCon 2015*, pages 1–5. IEEE, 2015. doi: 10.1109/SECON.2015.7133052.

[27] A. Butt, C. G. Popp, H. M. Pitts, and D. J. Sharp. Nasa ares i launch vehicle roll and reaction control systems design status. In *45th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, Aug. 2009. doi: 10.2514/6.2009-5130.

[28] P. Fortescue, J. Stark, and G. Swinerd. *Spacecraft Systems Engineering*. John Willey & sons, 3^{rd} edition, 2004. ISBN: 0-171-61951-5.

[29] N. S. Nise. *Control Systems Engineering*. John Willey & sons, 6^{th} edition, 2011. ISBN: 978-0470-54756-4.

[30] A. B. Kisabo and A. F. Adebimpe. State-space modeling of a rocket for optimal control system design. June 2019. doi: 10.5772/intechopen.82292.

[31] W. Du, B. Wie, and M. Whorton. Dynamic modeling and flight control simulation of a large flexible launch vehicle. In *AIAA Guidance, Navigation and Control Conference & Exhibit*, Aug. 2008. doi: 10.2514/6.2008-6620.

[32] G. F. McDonough, W. D. Murphree, J. C. Blair, J. R. Scoggins, T. G. Reed, E. L. Linsley, V. S. Verderdaime, J. A. Lovingood, M. Rheinfurth, and R. Ryan. AGARD 115 - Wind Effects on Launch Vehicles. Technical report, Technivision Services, Feb. 1970.

[33] B. Etkin. *Dynamics of Atmospheric Flight*. Dover Publications, 2005. ISBN: 9780486445229.

[34] B. Friedland. *Control System Design: An Introduction to State-Space Methods*. Dover Publications, 1^{st} edition, 2005. ISBN:9780486135113.

[35] J. M. Lemos. *Controlo no Espaço de Estados*. IST press, 1^{st} edition, 2019. ISBN:978-989-8481-70-2.

[36] L. Martins, C. Cardeira, and P. Oliveira. Linear quadratic regulator for trajectory tracking of a quadrotor. *IFAC-PapersOnLine*, 52(12):176–181, 2019. ISSN 2405-8963. doi: https://doi.org/10.1016/j.ifacol.2019.11.195. URL https://www.sciencedirect.com/science/article/pii/S2405896319311450. 21st IFAC Symposium on Automatic Control in Aerospace ACA 2019.

[37] D. Cabecinhas, P. Batista, P. Oliveira, and C. Silvestre. Hovercraft control with dynamic parameters identification. *IEEE Transactions on Control Systems Technology*, 26(3):785–796, 2018. doi: 10.1109/TCST.2017.2692733.

[38] A. Antunes, P. Outeiro, C. Cardeira, and P. Oliveira. Implementation and testing of a sideslip estimation for a formula student prototype. *Robotics and Autonomous Systems*, 115:83–89, 2019. ISSN 0921-8890. doi: https://doi.org/10.1016/j.robot.2019.01.018.

[39] Glenn Research Center. Earth Atmosphere Model. URL: https://www.grc.nasa.gov/www/k-12/airplane/atmosmet.html. Online (accessed: 06.01.2022).

[40] Mathworks. Horizontal Wind Model 07. URL: https://www.mathworks.com/help/aeroblks/horizontalwindmodel07.html, . Online (accessed: 08.02.2022).

[41] T. M. I. Hakim and O. Arifianto. Implementation of dryden continuous turbulence model into simulink for lsa-02 flight test simulation. *Jounal of Physics: Conference series*, 1005, Sept. 2017. doi: 10.1088/1742-6596/1005/1/012017.

[42] Mathworks. Dryden Wind Turbulence Model (Continuous). URL: https://www.mathworks.com/help/aeroblks/drydenwindturbulencemodelcontinuous.html, . Online (accessed: 10.02.2022).

[43] M. I. Ribeiro. Kalman and extended kalman filters: Concept, derivation and properties. ISR, Feb 2004.

[44] M. Grewal and A. Andrews. Kalman filtering: theory and practice using matlab. *New York: John Wiley and Sons*, 14, 01 2001. doi: 10.1002/9780470377819.

Appendix A

Decoupled State-Space Representations

In this Appendix, the decoupled state-space representations, resulting from applying the nominal conditions to the generic state-space representation of the system, are shown.

Starting from the varying pitch section, the state-space representations for the longitudinal and lateral modes are:

$$\delta x_{lon} = [\delta u, \delta w, \delta q, \delta\theta]^T, \qquad \delta u_{lon} = [\delta\mu_1] \tag{A.1a}$$

$$A_{lon}(t) = \begin{bmatrix} 0 & -q_0 & -w_0 & g\,\sin\theta_0 \\ q_0 + \dfrac{\overline{q}\,S\,C_{N_\alpha}\,w_0}{m\,({u_0}^2+{w_0}^2)} & -\dfrac{\overline{q}\,S\,C_{N_\alpha}\,u_0}{m\,({u_0}^2+{w_0}^2)} & u_0 & -g\,\cos\theta_0 \\ \dfrac{\overline{q}\,S\,d\,SM\,C_{N_\alpha}\,w_0}{J_t\,({u_0}^2+{w_0}^2)} & -\dfrac{\overline{q}\,S\,d\,SM\,C_{N_\alpha}\,u_0}{J_t\,({u_0}^2+{w_0}^2)} & \dfrac{\overline{q}\,S\,d^2\,(C_{m_q}+C_{m_{\dot\alpha}})}{2\,J_t\,V_0} & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{A.1b}$$

$$B_{lon}(t) = \begin{bmatrix} -\dfrac{T}{m}\,\sin\mu_{1_0} \\ -\dfrac{T}{m}\,\cos\mu_{1_0} \\ -\dfrac{T\,l}{J_t}\,\cos\mu_{1_0} \\ 0 \end{bmatrix} \tag{A.1c}$$

$$\delta x_{lat} = [\delta v, \delta r, \delta \phi, \delta \psi]^T, \qquad \delta u_{lat} = [\delta \mu_2] \tag{A.1d}$$

$$A_{lat}(t) = \begin{bmatrix} \dfrac{\overline{q}\, S\, C_{Y_\beta}}{m\, V_0} & -u_0 & -g\, \sin\theta_0 & g\, \cos\theta_0 \\[2ex] -\dfrac{\overline{q}\, S\, d\, SM\, C_{Y_\beta}}{J_t\, V_0} & \dfrac{\overline{q}\, S\, d^2\, (C_{n_r} + C_{n_{\dot\beta}})}{2\, J_t\, V_0} & 0 & 0 \\[2ex] 0 & \tan\theta_0 & q_0\, \tan\theta_0 & 0 \\[2ex] 0 & \dfrac{1}{\cos\theta_0} & \dfrac{q_0}{\cos\theta_0} & 0 \end{bmatrix} \tag{A.1e}$$

$$B_{lat}(t) = \begin{bmatrix} -\dfrac{T}{m}\, \cos\mu_{1_0} \\[2ex] \dfrac{T\, l}{J_t}\, \cos\mu_{1_0} \\[2ex] 0 \\[2ex] 0 \end{bmatrix} \tag{A.1f}$$

On the other hand, the decoupled state-space representations for vertical flight are:

$$\delta x_{lon} = [\delta w, \delta q, \delta \theta]^T, \qquad \delta u_{lon} = [\delta \mu_1] \tag{A.2a}$$

$$A_{lon}(t) = \begin{bmatrix} -\dfrac{\overline{q}\, S\, C_{N_\alpha}}{m\, u_0} & u_0 & -g \\[2ex] -\dfrac{\overline{q}\, S\, d\, SM\, C_{N_\alpha}}{J_t\, u_0} & \dfrac{\overline{q}\, S\, d^2\, (C_{m_q} + C_{m_{\dot\alpha}})}{2\, J_t\, u_0} & 0 \\[2ex] 0 & 1 & 0 \end{bmatrix} \qquad B_{lon}(t) = \begin{bmatrix} -\dfrac{T}{m} \\[2ex] -\dfrac{T\, l}{J_t} \\[2ex] 0 \end{bmatrix} \tag{A.2b}$$

$$\delta x_{lat} = [\delta v, \delta r, \delta \psi]^T, \qquad \delta u_{lat} = [\delta \mu_2] \tag{A.2c}$$

$$A_{lat}(t) = \begin{bmatrix} \dfrac{\overline{q}\,S\,C_{Y_\beta}}{m\,u_0} & -u_0 & g \\[2ex] -\dfrac{\overline{q}\,S\,d\,SM\,C_{Y_\beta}}{J_t\,u_0} & \dfrac{\overline{q}\,S\,d^2\,(C_{n_r}+C_{n_{\dot\beta}})}{2\,J_t\,u_0} & 0 \\[2ex] 0 & 1 & 0 \end{bmatrix} \quad B_{lat}(t) = \begin{bmatrix} -\dfrac{T}{m} \\[2ex] \dfrac{T\,l}{J_t} \\[2ex] 0 \end{bmatrix} \tag{A.2d}$$

Appendix B

Kalman Filter Theory

In this Appendix, the Kalman filter theory is briefly explained for the continuous time case, mainly relying on [35, 43, 44]

The Kalman filter is a widely used observer to tackle the estimation problem for linear dynamic systems. When both the process and measurement associated with the estimated state are corrupted by random, independent, zero mean Gaussian white noise, the solution provided by the Kalman filter is statistically optimal with respect to any quadratic function of the estimation error. For this reason, it is also referred to as Linear Quadratic Estimator (LQE), and represents the dual of the LQR to the estimation problem.

In continuous time, the random process and observation are given by

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\,\mathbf{x}(t) + \mathbf{B}(t)\,\mathbf{u}(t) + \mathbf{G}(t)\,\mathbf{w}(t)\,, \tag{B.1a}$$

$$\mathbf{y}(t) = \mathbf{C}(t)\,\mathbf{x}(t) + \mathbf{v}(t)\,, \tag{B.1b}$$

where $\mathbf{w}(t)$ is the process noise (associated with the model), $\mathbf{v}(t)$ is the measurement noise (associated with the sensors) and $\mathbf{G}(t)$ is the process noise coupling matrix. The aforementioned noise properties translate into the following conditions

$$E[\mathbf{w}(t)] = E[\mathbf{v}(t)] = 0\,, \tag{B.2a}$$

$$E[\mathbf{w}(t)\,\mathbf{w}^T(t+\tau)] = \mathbf{Q}\,\delta(\tau)\,, \tag{B.2b}$$

$$E[\mathbf{v}(t)\,\mathbf{v}^T(t+\tau)] = \mathbf{R}\,\delta(\tau)\,, \tag{B.2c}$$

$$E[\mathbf{w}(t)\,\mathbf{v}^T(\tau)] = 0\,, \tag{B.2d}$$

where $E[\mathbf{a}(t)]$ is the expected value of the random variable $\mathbf{a}(t)$, $\mathbf{Q}$ and $\mathbf{R}$ are the noise covariance matrices associated with $\mathbf{w}$ and $\mathbf{v}$, respectively, and $\delta(.)$ is the Dirac delta function. The $\mathbf{Q}$ and $\mathbf{R}$ matrices are positive semi-definite.

Given the defined process, observation and noise properties, the Kalman filter is capable of providing an optimal state estimation according to the differential equation

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}(t)\,\hat{\mathbf{x}}(t) + \mathbf{B}(t)\,\mathbf{u}(t) + \mathbf{L}(t)\,\left(\mathbf{y}(t) - \mathbf{C}(t)\,\hat{\mathbf{x}}(t)\right)\,, \tag{B.3}$$

in which $\hat{\mathbf{x}}(t)$ is the state estimate and $\mathbf{L}(t)$ is the Kalman gain. Given an initial condition $\hat{\mathbf{x}}(0)$, the state estimate derivative $\dot{\hat{\mathbf{x}}}$ is recursively propagated by correcting the process with the state estimation error $(\mathbf{y}(t) - \mathbf{C}(t)\,\hat{\mathbf{x}}(t))$ multiplied by the Kalman gain. Similarly to the LQR, the Kalman gain is given by

$$\mathbf{L}(t) = \mathbf{P}(t)\,\mathbf{C}^T(t)\,\mathbf{R}^{-1}\,, \tag{B.4}$$

where $\mathbf{P}$ is the solution to the matrix Riccati differential equation

$$\dot{\mathbf{P}}(t) = \mathbf{A}(t)\,\mathbf{P}(t) + \mathbf{P}(t)\,\mathbf{A}^T(t) + \mathbf{G}(t)\,\mathbf{Q}(t)\,\mathbf{G}^T(t) - \mathbf{P}(t)\,\mathbf{C}^T(t)\,\mathbf{R}^{-1}\,\mathbf{C}(t)\,\mathbf{P}(t)\,. \tag{B.5}$$

As the process is time-varying, the steady-state solution to the Riccati equation cannot be used and has to be continuously solved, as opposed to what was derived for the LQR. The state estimate $\hat{\mathbf{x}}$ given by the Kalman filter satisfies the following conditions:

1. It is centred, which means that $E[\mathbf{x}(t) - \hat{\mathbf{x}}(t)] = 0$;

2. The estimation error has minimal energy, which means that $\int_0^\infty \|\mathbf{x}(t) - \hat{\mathbf{x}}(t)\|^2\,dt = 0$.

Equivalently to the LQR, the tuning parameters will be the $\mathbf{Q}$ and $\mathbf{R}$ matrices, which are now the noise covariance matrices. The $\mathbf{R}$ matrix can be tuned according to the specifications of the on-board sensors, while determining the model noise covariance ($\mathbf{Q}$) might represent a harder task. Resorting to simulation in order to properly tune the $\mathbf{Q}$ matrix is a good initial method, which can later be updated using results coming from a real implementation scenario.