Study of Feasibility of a Mission to Mars using Aerocapture Technique

Oscar Belart Bayo

Director:
Dr. Marco A. Pérez Martínez
Department of Strength of Materials and Structural Engineering (RMEE)

— Terrassa - July 4, 2012 —
Abstract

Aerocapture is a brand new technique of orbit insertion, which consists in the energy depletion of a spacecraft using atmospheric drag. This decelerates the space vehicle and changes its trajectory from hyperbolic orbit to elliptic orbit. Then two propulsive maneuvers must be performed in order to obtain the desired final orbit.

This project seeks to assess the viability of aerocapture. It includes an implementation of a simulation tool so-called AECASIM\textsuperscript{1} that is able of simulating the aerocapture maneuver. This software contains simple environment and vehicle models that help to describe the desired aerocapture problem for its analysis. PredGuid guidance algorithm was implemented and adapted to this project's objectives and scenario. In order to analyse the performances of this maneuver, four essential parameters were varied and some conclusions about their influence on aerocapture were drawn. Furthermore, capturable corridors were constituted with respect to the different parameter variations and were also compared between them, in order to distinguish the best possible configuration.

Finally, a comparison with regular propulsive methods was performed. It revealed that with aerocapture technique great fuel savings can be achieved. For the studied case, this allows to increase the vehicle mass about a 77\% and hence to rise payload mass or to reduce the initial mass approximately by a half, reducing with this the costs of the mission.

\textsuperscript{1}Corresponding author: Oscar Belart Bayo (oscar.belart@gmail.com).
Acknowledgements

First of all, I would like to thank my parents, Santiago and Lorenza, for their unconditional support and love. They have always been by my side, since the first day I was born, giving me all I needed no matter how hard it was. Thank you wholeheartedly for that. Many thanks to my brother David and to Olga for the support and for remembering me that I also could have some spare time. Thank you David for being always my little big “bro”. I would like to give thanks to my sister Mar, who has always taken care of me since the first day. Thank you all!!!

Next, I want to thank Marco, my project’s director, for the patience he had with me. I’ve send you many e-mails with stupid things! Many thanks for calming me down when my nervous breakdowns showed up. I also want to most sincerely thank you for believing in me and allowing me to realize this fabulous project.

I would like to thank Victor, who had always made me believe that this project was awesome! Your interest in it fed my enthusiasm for keeping on developing it. Thank you for making me realize that I actually have professional ambitions in my life :)

Special thanks to my best friend Alba, who is always willing to unconditionally help me. Thank you so much for being there when I needed you the most, even when hundreds of kilometres separated us. I would like to thank Carla for her help, when she had better things to do rather than helping me out. Thank you so much!

I want to give eternal gratitude and to dedicate this project to my beloved Josep, who always was and I hope, always will be by my side. Many many many thanks for taking care of me, supporting me, loving me and above all for being who you are! I love you!!

Finally, I would like to thank Nuria R., Adrián, Alvaro, David G., Nuria A. and Pol for their trust, support and love and all the people that helped me out and I cannot list. Thank you very much!!
# Contents

1 INTRODUCTION .................................................. 1
   1.1 Motivation .................................................. 1
   1.2 State of the Art ............................................. 3
   1.3 Problem Definition .......................................... 6
   1.4 Aims of the Research ....................................... 11
   1.5 Outline ..................................................... 12

2 NAVIGATION DESIGN ............................................ 13
   2.1 Overview .................................................... 13
   2.2 Coordinate Systems and Transformations .................... 13
       2.2.1 Reference Frames ...................................... 13
       2.2.2 Matrix Transformations ................................. 15
   2.3 Environment Models ......................................... 18
       2.3.1 Atmospheric Model .................................... 18
       2.3.2 Gravitational Model ................................. 20
   2.4 Vehicle Model ............................................... 21
   2.5 Equations of Motion ........................................ 21
   2.6 Aerodynamic Heating ....................................... 27

3 SIMULATION DESIGN ........................................... 29
   3.1 Overview .................................................... 29
   3.2 Simulation tool - AECASIM ................................. 29
3.3 Guidance .............................................. 32
  3.3.1 PredGuid Flow Chart ...................... 32
  3.3.2 Initialization ................................. 35
  3.3.3 Aerodynamic Properties .................. 36
  3.3.4 Atmospheric Model ......................... 38
  3.3.5 Aerodynamic Heating ...................... 39
  3.3.6 Energy Management ......................... 40
  3.3.7 Phase Check .................................. 42
  3.3.8 Targeting ..................................... 43
  3.3.9 Corrector ...................................... 45
  3.3.10 Predictor ..................................... 47
  3.3.11 Integrator ................................... 49
  3.3.12 Lateral Control .............................. 50
  3.3.13 Command Incorporation ................... 51

3.4 Simulation Implementation ..................... 52
  3.4.1 Vehicle Properties for Study ................ 52
  3.4.2 Initial Conditions at Entry Interface .... 52
  3.4.3 Simulation Parameters ....................... 53

4 ANALYSIS OF RESULTS .............................. 55
  4.1 Software Verification .......................... 55
  4.2 Model Assessment ................................ 62
  4.3 Influence of $L/D$ and $\beta$ Variation ........ 72
  4.4 Corridor Determination ....................... 79
  4.5 Lateral Velocity ................................ 89
  4.6 Comparison with propulsive Maneuvers ....... 93
  4.7 Saving Opportunities ........................... 97

5 CONCLUSIONS ......................................... 99
  5.1 General Conclusions ......................... 99
[This page is intentionally left blank]
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Aerocapture Maneuver</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>Maneuver Type Comparison</td>
<td>6</td>
</tr>
<tr>
<td>1.3</td>
<td>Detailed Scheme of the Aerocapture Problem</td>
<td>7</td>
</tr>
<tr>
<td>1.4</td>
<td>Definition of Flight Path Angle</td>
<td>8</td>
</tr>
<tr>
<td>1.5</td>
<td>Bank Angle Modulation Concept</td>
<td>9</td>
</tr>
<tr>
<td>1.6</td>
<td>Lift Modulation Examples</td>
<td>10</td>
</tr>
<tr>
<td>2.1</td>
<td>Local Horizon Coordinate System</td>
<td>14</td>
</tr>
<tr>
<td>2.2</td>
<td>Body to Stability Transformation</td>
<td>16</td>
</tr>
<tr>
<td>2.3</td>
<td>Stability to Wind Transformation</td>
<td>16</td>
</tr>
<tr>
<td>2.4</td>
<td>Wind to Horizontal Transformation</td>
<td>17</td>
</tr>
<tr>
<td>2.5</td>
<td>Horizontal to Rotational Transformation</td>
<td>17</td>
</tr>
<tr>
<td>2.6</td>
<td>Rotational to Inertial Transformation</td>
<td>18</td>
</tr>
<tr>
<td>2.7</td>
<td>Spherical Coordinate System</td>
<td>22</td>
</tr>
<tr>
<td>2.8</td>
<td>Aerodynamic Forces</td>
<td>25</td>
</tr>
<tr>
<td>3.1</td>
<td>AECASIM Logic Flow</td>
<td>30</td>
</tr>
<tr>
<td>3.2</td>
<td>PredGuid Flow Chart</td>
<td>34</td>
</tr>
<tr>
<td>3.3</td>
<td>Initialization Flow Chart</td>
<td>35</td>
</tr>
<tr>
<td>3.4</td>
<td>Aerodynamic Properties Flow Chart</td>
<td>37</td>
</tr>
<tr>
<td>3.5</td>
<td>Atmospheric Model Flow Chart</td>
<td>38</td>
</tr>
<tr>
<td>3.6</td>
<td>Aerodynamic Heating Flow Chart</td>
<td>39</td>
</tr>
<tr>
<td>3.7</td>
<td>Energy Management Flow Chart</td>
<td>41</td>
</tr>
</tbody>
</table>
[This page is intentionally left blank]
## List of Tables

2.1 Mars General Parameters ................................................. 18
2.2 Density Model Parameters ............................................... 19
2.3 Vehicle Model Parameters ............................................... 21
3.1 Initial Conditions at Entry Interface .............................. 52
4.1 Plane Change Maneuver Properties ................................. 92
## List of Symbols and Acronyms

### Roman Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>Semi-major axis</td>
</tr>
<tr>
<td>$\ddot{a}$</td>
<td>Acceleration</td>
</tr>
<tr>
<td>$C$</td>
<td>Coefficient</td>
</tr>
<tr>
<td>$\vec{D}$</td>
<td>Drag</td>
</tr>
<tr>
<td>$e$</td>
<td>Orbit eccentricity</td>
</tr>
<tr>
<td>$\vec{F}$</td>
<td>Force</td>
</tr>
<tr>
<td>$\vec{g}$</td>
<td>Gravity</td>
</tr>
<tr>
<td>$HS$</td>
<td>Scale height</td>
</tr>
<tr>
<td>$h$</td>
<td>Altitude</td>
</tr>
<tr>
<td>$I$</td>
<td>Impulse</td>
</tr>
<tr>
<td>$\hat{i}$</td>
<td>Unit vector</td>
</tr>
<tr>
<td>$J_2$</td>
<td>$J_2$ perturbation constant</td>
</tr>
<tr>
<td>$\vec{L}$</td>
<td>Lift</td>
</tr>
<tr>
<td>$L/D$</td>
<td>Lift-to-drag ratio</td>
</tr>
<tr>
<td>$m$</td>
<td>Mass</td>
</tr>
<tr>
<td>$\hat{n}$</td>
<td>Normal unit vector</td>
</tr>
<tr>
<td>$P$</td>
<td>Orbital period</td>
</tr>
<tr>
<td>$Q$</td>
<td>Heat load</td>
</tr>
<tr>
<td>$\dot{Q}$</td>
<td>Heat flux</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>$q$</td>
<td>Dynamic pressure</td>
</tr>
<tr>
<td>$R$</td>
<td>Radius</td>
</tr>
<tr>
<td>$\vec{r}$</td>
<td>Position</td>
</tr>
<tr>
<td>$S$</td>
<td>Surface</td>
</tr>
<tr>
<td>$t$</td>
<td>Time</td>
</tr>
<tr>
<td>$\vec{V}$</td>
<td>Relative velocity</td>
</tr>
</tbody>
</table>

**Greek Symbols**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>Angle of attack</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Ballistic coefficient</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>Flight path angle</td>
</tr>
<tr>
<td>$\Delta$</td>
<td>Increment</td>
</tr>
<tr>
<td>$\delta$</td>
<td>Sideslip angle</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>Wedge angle</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Longitude</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Standard gravitational parameter</td>
</tr>
<tr>
<td>$\rho$</td>
<td>Density</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>Bank angle</td>
</tr>
<tr>
<td>$\phi$</td>
<td>Latitude</td>
</tr>
<tr>
<td>$\psi$</td>
<td>Heading angle</td>
</tr>
<tr>
<td>$\vec{\Omega}$</td>
<td>Angular velocity vector of the rotational coordinate system</td>
</tr>
<tr>
<td>$\vec{\omega}$</td>
<td>Rotation vector</td>
</tr>
</tbody>
</table>

**Subscripts**

<table>
<thead>
<tr>
<th>Subscript</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0$</td>
<td>Reference</td>
</tr>
<tr>
<td>$a$</td>
<td>Apoapsis or apoareion</td>
</tr>
<tr>
<td>$aerodyn$</td>
<td>Aerodynamic</td>
</tr>
<tr>
<td>$corr$</td>
<td>Corridor</td>
</tr>
</tbody>
</table>
$D$  Drag

$dp$  Desired plane

$ext$  External

$f$  Final

$g$  Perturbed position

$grav$  Gravitation

$i$  Initial

$inertial$  Inertial

$lat$  Lateral

$n$  Spacecraft's nose

$p$  Periapsis or periareion

$pc$  Pericenter

$pole$  Mars north pole

$r$  Position

$ref$  Reference

$SL$  Sea level

$sp$  Specific

$target$  Target

$\oplus$  Earth

$\oplus$  Mars

$\infty$  Infinity

**Superscripts**

$E$  Earth

$M$  Mars

**Acronyms**

AECASIM  Aerocapture Simulator
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFE</td>
<td>Aeroassist Flight Experiment</td>
</tr>
<tr>
<td>APC</td>
<td>Analytic Predictor-Corrector</td>
</tr>
<tr>
<td>CNES</td>
<td>Centre National d’Études Spatiales</td>
</tr>
<tr>
<td>COSPAR</td>
<td>Committee on Space Research</td>
</tr>
<tr>
<td>EC</td>
<td>Energy Controller</td>
</tr>
<tr>
<td>ESA</td>
<td>European Space Agency</td>
</tr>
<tr>
<td>MARS - GRAM</td>
<td>Mars Global Reference Atmosphere Model</td>
</tr>
<tr>
<td>MSL</td>
<td>Mars Science Laboratory</td>
</tr>
<tr>
<td>NASA</td>
<td>National Aeronautics and Space Administration</td>
</tr>
<tr>
<td>NPC</td>
<td>Numeric Predictor-Corrector</td>
</tr>
<tr>
<td>TCP</td>
<td>Terminal Point Controller</td>
</tr>
</tbody>
</table>
Chapter 1

INTRODUCTION

1.1 Motivation

Space exploration has led humanity to increase significantly the limits of science and technology. The curiosity and enthusiasm that human beings own has made them to perform milestones like the stepping on the moon by Neil Armstrong in 1969 or the landing of the Mars Pathfinder on Mars in 1997. These events make the society more likely to interest itself in science and to invest more resources to the national space administrations. The scientific labs send on each mission carry out many investigations in important scientific fields such as medicine, material science, biology or physics, which provide life down on Earth with the state-of-the-art technology. Every step made on space exploration is an improvement and a gain of knowledge, not just about the Universe, but about where human innovation and technology can reach. The mankind uses on their everyday routine many developed spacial systems as Earth observation, navigation, defense and specially, communication systems.

Human curiosity and willingness to learn about the surroundings, its past and the mankind’s as well, is the main leading force to high achievements.

About Aerocapture

Aerocapture is a specific type of aeroassisted maneuver for orbital insertion. It is based on a reduction from hyperbolic velocities to low orbit velocities by means of the drag force produced by the atmosphere of the destination planet. The enormous kinetic energy is transformed into heat, which is insulated from the spacecraft with a protection aeroshell made of ablative materials.
1.1. Motivation

With this technique the amount of chemical propellant that would have been used is no longer required. Therefore, the vehicle mass or the scientific payload can be incremented beyond the actual limits providing the spacecraft the possibility of carrying out more investigations per mission, increasing the structural limitations of the spacecraft or flying the same payload still farther. One major application would be the returning of samples from closer planets or reaching the distant ones and their moons.

The sequence of the whole maneuver is divided in some phases (see figure 1.1). First of all, there is an hyperbolic approximation until the vehicle enters the atmosphere. Next, the most important part is the guidance of the spacecraft trough the atmosphere due to the uncertainties that the atmosphere can show. The spacecraft is guided until it exits the atmosphere in a controlled form and before it reaches the target altitude, the aeroshell protection must be jettisoned. Once the vehicle is close to the target an impulsive maneuver is performed in order to raise the periapsis. Finally, a circularization maneuver is needed in order to correct the errors in the final altitude.

Another aeroassisted well-known method of orbit insertion is aerobraking. The reduction of velocity due to drag is similar to aerocapture maneuver but the spacecraft must be inserted in a high elliptical orbit. The periapsis of the orbit lies within the atmosphere which will produce orbital decay each time the spacecraft approximates the closes point of the orbit to the planet. This technique requires an initial propulsive maneuver so as to put the spacecraft into the elliptical orbit, and a lot of time for the whole maneuver to be completed, e.g. 3 months versus 3 hours by aerocapture for the Mars Orbiter [1].

![Figure 1.1: Aerocapture Maneuver](image)

Hypersonic Atmospheric Circularization
Aerocapture
Periapsis
Atmospheric Entry
Atmospheric Exit
Circularization Maneuver
Aerocapture Maneuver

Figure 1.1: Aerocapture Maneuver
1. INTRODUCTION

1.2 State of the Art

The concept of aeroassisted transfer orbits was first mentioned in the 60’s [2]. Some proposals for implementation of experiments of aerocapture verification, such as the Aeroassist Flight Experiment (AFE) in the 90’s and moreover the Mars Sample Return Mission by NASA and the Centre National d’Études Spatiales (CNES) were performed. Those led to important algorithms for use to implement of aerocapture verification simulations. These algorithms are the following [1]:

- **Analytic Predictor-Corrector**: the bank angle is modulated with a linear second-order differential equation of the altitude until the vehicle has lost its velocity below a specific-mission limit. The fact that the altitude rate is the only control variable makes it possible to integrate it. The vehicle relative velocity at exit is predicted by the analytical equations assuming a constant altitude rate.

- **Numeric Predictor-Corrector**: the orientation of the lift vector is controlled about the relative velocity vector by means of modifying the bank angle. The algorithm numerically integrates the vectors of position and velocity to the forward state, assuming constant bank angle, until the atmospheric exit is reached.

- **Terminal Point Controller**: the vehicle is intended to follow a fixed reference trajectory until it reaches a concrete terminal point or a group of terminal conditions.

- **Energy Controller**: the vehicle energy is controlled until a reference energy state with the modulating of the energy gain. This gain refers an altitude rate and it is modulated using equations of vertical acceleration.

John Higgins developed in 1984 the original numeric predictor-corrector algorithm for Earth applications [1]. The algorithm, named PredGuid, has several limitations in high energy orbit transfers due to its inability to handle hyperbolic trajectories. On the one hand, the bank angle guess is calculated according to the slope from previous guesses. Hyperbolic trajectories have negative for apoapsis and semi-major axis which are incorrectly interpreted. On the other hand, the constant bank angle assumption limits the ways that algorithm has to reduce energy and this brings on excess values of acceleration loads.

In 1986 Doug Fuhrly created a new functionality to PredGuid [3]. It consists of an energy management that brings the spacecraft to a constant altitude cruise. When the velocity decays underneath a certain value, the Higgins’ algorithm continues the targeting phase. The software is relatively complex and specialized for aerocapture at Mars.
In 2001 Jennifer Di Carlo introduced a management phase prior to targeting in order to enhance PredGuid algorithm [1]. The generic method of transitioning between energy management and targeting phases allowed high energy trajectories to be analysed without major errors. Furthermore, other heuristic features were replaced by more generic ones. The result was an enhanced coverage for aerocapture at Earth compared to Higgins’.

CNES and NASA assessed in 2002 the robustness, accuracy, capability to limit the load, and the complexity of APC, NPC, TPC and EC algorithms [4]. The evaluation demonstrated the numerical guidance principle not to be competitive compared to the analytical concepts. The rest of the algorithms were well adapted to guarantee the success of aerocapture.

In 2005 many authors have carried out different studies about the concept of aerocapture with Monte Carlo simulations. M. Rozanov and M. Guelman studied the aeroassisted orbital maneuvering using variable structure control and only kinetic variable measurements, which eliminated the need of density estimation [5]. M. Kumar and A. Tewari created a model that used a combination of aerocapture and aerobraking maneuvers. Such model was able to achieve a circular orbit \(e = 0.017\) from an open orbit of high eccentricity \(e = 1.55\) within a short span of 20 hours and with scant propellant consumption [6]. The same year J. Hall, M. Noca and R. Bailey studied the cost benefit of the aerocapture maneuver [7]. The paper shows different results depending on the destination and target orbit, showing greater cost-over-mass reduction on farther planets. As for Mars orbits (circular at 300 km altitude), the aerocapture-improved mission’s normalized delivery cost is approx. $0.05M/kg. The corresponding percentage of cost-to-mass benefit for this mission is about 12%.

Although aerocapture technique has not been tested yet, the technology is being matured by the In-Space Propulsion Technology Program. The program objective is to develop in-space propulsion technologies that can enable or benefit near and mid-term NASA space science missions by significantly reducing cost, mass and travel times. NASA researchers are considering aerocapture technologies for a broad range of future mission objectives including orbiters at Titan, a moon of Saturn, Venus, Mars, and Neptune [8, 9]. Furthermore, NASA plans to perform a flight test for the Lunar Return Orion mission, which will use a numerical guidance of thousands of lines. The flight test is scheduled for about 2015 [10].

The European Space Agency (ESA) is also planning future missions with aerocapture techniques. Within the Aurora Programme there are two main mission sets: the Arrow missions and the Flagship missions. Firstly, one of the Arrow missions is a Mars aerocapture demonstrator, whose aim is to validate the technology needed to enable a
spacecraft to perform the maneuver [11]. Secondly, in the Mars Sample Return Mission
an orbiter will be inserted in a low-altitude orbit around Mars, then a descent module
will be released and descend to Mars' surface. On board the landing platform of the de-
scent module, there will be a device to collect samples and an ascent vehicle to return
back up to the orbiter in order to return the samples back to Earth [12]. This mission
was initially scheduled to be launched in 2011 and now it is rescheduled to be set in
between the time frame 2020-2022.
1.3 Problem Definition

The aerocapture technique was thought to replace the propulsive maneuvers to a certain extent. The main disadvantage of these is the cost of the consumed fuel in their execution. As shown in the Rocket Equation\(^1\) (eq. 1.1), mass fraction grows exponentially with delta-V (\(\Delta V\)). The mass fraction is defined as initial mass \(m_i\) over final mass \(m_f\). Therefore, if missions require more delta-V, the initial mass increases up to so high values, that some missions simply can not be executed due to the involved costs, the impossibility to launch extremely high-weighed vehicles or very long mission times.

\[
\frac{m_i}{m_f} = \exp\left(\frac{\Delta V}{I_{sp}g_0_{SL}}\right)
\]  

(1.1)

Aerocapture provides an important amount of speed reduction, since the deceleration is caused by atmospheric friction. That reduces considerably the need for high initial masses (see figure 1.2) \[9\]. Once the aerocapture performance is perfectly settled, the humankind will be able to execute missions that nowadays are not viable.

![Figure 1.2: Maneuver Type Comparison](image)

Although aerocapture is fuel-efficient, there are still two discrete burns needed. As shown in figure 1.3, once the spacecraft has exited the atmosphere, it flies through a post-aerocapture elliptic orbit. Then, when it reaches the apoapsis\(^2\) \(r_{ap}\), which is the farthest locus point of an orbit, a first burn \((\Delta V_1)\) must be performed in order to raise the

---

\(^1\)Where \(I_{sp}\) and \(g_0_{SL}\) are the specific impulse of the vehicle that performs the burn and the gravitational acceleration of Earth at sea level, respectively.

\(^2\)This can also be called apoareion for a Marian orbit.
periapsis or periareion \( (r_p) \), the closes locus point of an orbit, so as to avoid re-entering the atmosphere again. Doing so, the spacecraft is conducted to the target periapsis \( (r_{p_{\text{target}}}) \). In such point, a second burn \( (\Delta V_2) \) is required in order to circularize the orbit, which means to give the orbit its target shape. This delta-V can either be in one direction or the opposite depending on whether the post-aerocapture orbit’s apoapsis is greater or smaller than the target’s. Notice that if the spacecraft would reach the apoapsis target before the first delta-V, it would only require one single shot to rise the periapsis and circularize the orbit at the same time, since the actual apoapsis and the target apoapsis would be the same.

![Figure 1.3: Detailed Scheme of the Aerocapture Problem](image)

The aerocapture problem has four important parameters with a major influence on the performance of a mission. Variations of these parameters can change considerably the performance of the flight and hence the result of the mission. Such parameters are velocity, flight path angle, ballistic coefficient and lift-to-drag ratio and are commented below:

- Velocity \( V \): the velocity is a significant parameter since it plays an important role in the aerothermodynamic phenomena that regards aerocapture. The inertial velocity indicates how fast the spacecraft approaches the destination planet and, once the vehicle is in the atmosphere, determines the relative velocity, which origins the aerodynamic forces. The higher the velocity is, the greater the aerodynamic forces, the dynamic pressure and the heat fluxes will be.
• Flight path angle $\gamma$: this is the angle between the velocity vector and the local horizon plane and is defined as negative when the velocity vector lays underneath the local horizon. That means, the spacecraft flies towards the planet (see figure 1.4). This parameter is used to define the behaviour of the space vehicle in the atmospheric entry and it helps to predict the physical phenomena that will affect the vehicle. As flight path angle becomes steeper, the vehicle enters straight to the planet’s atmosphere. This increases the risk of ground collision and an increment of the aerothermodynamic effects, providing high heat and dynamic pressure, which could bring the spacecraft to failure. Otherwise, if flight path angle were too shallow, the vehicle would hardly be affected by drag so it would need to remain longer within the atmosphere in order to reduce its velocity. This would cause the spacecraft to have significantly high heat loads.

![Figure 1.4: Definition of Flight Path Angle](image)

• Ballistic coefficient $\beta$: this is a measure of the ability of an object to overcome air resistance in flight. It is calculated as equation 1.2 shows. The higher the ballistic coefficient is, the lower the resistance to flight is. A higher ballistic coefficient also means a higher mass, which brings the spacecraft to be less affected by aerodynamic and gravity forces.

$$\beta = \frac{m}{S_{ref} C_D} \quad (1.2)$$

• Lift-to-drag ratio $L/D$: this is calculated by the division of lift by drag. Both forces are generated by the spacecraft as it flies through a fluid that, in this case, is the atmosphere of Mars. This aerodynamic parameter is normally desired to be very high for all types of airplanes at Earth in order to reduce fuel consumption. However, as for aerocapture the vehicle is designed to reduce its velocity by means of drag, then lift-to-drag ratio must not be high. Furthermore, the very high velocities in which the space vehicle is involved do not usually require it to have large lift coefficients, since enough lift can be easily generated to control the spacecraft.
This thesis contains the study in detail of the aerocapture problem, which involves the period of time from the moment the spacecraft enters the atmosphere until it approaches the apoapsis. The most important phase is control and guidance of the space vehicle throughout the atmosphere, which is performed with the bank angle modulation.

Figure 1.5 is a three-dimensional representation of lift modulation. The change in bank angle ($\sigma$) changes the orientation of lift ($\vec{L}$) but not drag ($\vec{D}$).

The bank angle controls the orientation of the lift vector providing separate control of vertical and lateral velocities. Since the bank angle is the only control variable, it is impossible to correct both lateral position and velocity errors [1]. Therefore, this thesis chooses to control only the lateral velocity (see Lateral Control in subsection 3.3.12). As seen in figure 1.6, the combination of value and sign of the bank angle enables enough flexibility for controlling the spacecraft in both directions. For example, in case A the lift vector points upwards to the right. Changing the sign of the bank angle (the so-called bank reversal or roll reversal maneuver) produces a change in the lateral, but not in the vertical direction. In order to always maintain certain lateral control, the bank angle is limited from 15 degrees to 165 degrees, which correspond to full lift up and full lift down, respectively.

$^3\alpha$ corresponds to the angle of attack.
In addition, one feature that defines the performance of aerocapture is the so-called corridor. It describes the relationship between the inertial velocity, the flight path angle and the success of the executed maneuvers. The capturable corridor represents the range of velocities and flight path angles, with which the aerocapture succeeds within certain predefined constraints and target tolerance. This corridor and the constraints will be further described in chapter 4.
1.4 Aims of the Research

The main goal of this project is to perform a preliminary study about the feasibility of the aerocapture technique at Mars. This must be done by running simulations, where the parameters $V_{\text{inertial}}$, $\gamma$, $L/D$ and $\beta$ are varied within certain ranges with respect to a predefined mission. Afterwards, the results must be analysed in order to find how these variations affect aerocapture in matter of mission performance and fuel expenditure. To do so, the outcomes must be collected into $V_{\text{inertial}} - \gamma$ corridors and then analysed.

The selected mission consists of a travel from Earth to Mars and an orbit insertion into a circular equatorial orbit of 500 km altitude. The mission is considered success if the target altitude is achieved within a threshold of $\pm$ 50 km.

Furthermore, an enough robust simulation environment must be implemented, in order to perform the needed simulations. It must show sufficient flexibility to allow multiple-case simulating. The simulation tool must have the series of dynamic equations that are able to describe correctly the physics of the problem, as well as simple environment models that describe the planet with enough accuracy. Furthermore, PredGuid guidance algorithm must be implemented and adapted for Mars application. This algorithm is responsible of controlling the spacecraft within the Martian atmosphere and bringing it to the desired target altitude.

Finally, aerocapture and propulsive maneuvers will be compared to in order to assess the benefits of aerocapture.
1.5 Outline

The remaining chapters of this project are followed as below described:

- **Chapter 2** contains theoretical descriptions of the different parts that compose the equations of motion. The different coordinate systems, in which the spacecraft is referred, and the corresponding transformation matrices are firstly presented. Then, the description of the Martian features. This describes the planet with three constants and two simple environment models: atmospheric and gravitational. It follows the vehicle model, which fully describes the vehicle with a table of values for different parameters. Afterwards, the cinematic and dynamic equations are derived from the previous models. In the end, there is an aerodynamic heating model that will be used to calculate the heat transfer phenomena related to the spacecraft.

- **Chapter 3** provides explanations about how the main simulating tool was implemented and how it works. PredGuid guidance algorithm is also described, as well as all subroutines that form it, with the help of logic flow charts. Finally, there is a guide of implementation, which presents the studied vehicle and initial conditions parameters, describe how those were modified and justifies the selected simulation parameters.

- **Chapter 4** firstly presents a verification section that justifies the validity of the software by means of examples. A single reference case is assessed to give explanations and justifications about aerocapture performances. In the end, the results of the simulated cases are presented and analysed.

- **Chapter 5** includes a brief reminder of the exposed objectives. It presents a collection of all the conclusions and a set of future work opportunities.
Chapter 2

NAVIGATION DESIGN

2.1 Overview

In order to be able to simulate the problem defined in chapter 1, it is necessary to derive the corresponding equations of motion, which describe the dynamic behaviour of a spacecraft in a predefined modelled scenario, in this case Mars. Besides, the coordinate systems, where vehicle’s moves can be expressed from and the matrix transformations between them will also be described in this chapter.

2.2 Coordinate Systems and Transformations

2.2.1 Reference Frames

The elements that describe the different used reference frames are below listed (see figures 2.1 to 2.6).

Inertial Coordinate System

- Origin: Mars’ center of masses
- $\hat{i}_I$: direction to the zero meridian
- $\hat{j}_I$: right-hand triad
- $\hat{k}_I$: direction through Mars north pole
2.2. Coordinate Systems and Transformations

Rotational Coordinate System

- Origin: Mars’ center of masses
- \( \hat{i}_R \): direction to the position of the spacecraft. This is the inertial position vector \( \vec{r} \).
- \( \hat{j}_R \): lies in the equatorial plane of the planet \( \hat{i}_I - \hat{j}_I \) plane.
- \( \hat{k}_R \): right-hand triad, in direction upwards to Mars’ north pole

Local Horizon Coordinate System

- Origin: spacecraft center of masses
- \( \hat{i}_H \): lies in the intersection of a parallel plane to \( \hat{i}_I - \hat{j}_I \) with the perpendicular plane to the inertial position unit vector, in direction to Mars’ rotation sense (see figure 2.1).
- \( \hat{j}_H \): right-hand triad
- \( \hat{k}_H \): direction to the center of masses of the planet, along the inertial position vector

Figure 2.1: Local Horizon Coordinate System
Wind Coordinate System

- Origin: spacecraft center of masses
- \( \hat{i}_W \): direction along the relative velocity vector
- \( \hat{j}_W \): lies in the local horizon plane
- \( \hat{k}_W \): right-hand triad, in direction downwards to Mars

Stability Coordinate System

- Origin: spacecraft center of masses
- \( \hat{i}_S \): direction along the projection of relative velocity onto the \( \hat{i}_B - \hat{k}_B \) plane
- \( \hat{j}_S \): right-hand triad
- \( \hat{k}_S \): direction to planet’s center

Body Coordinate System

- Origin: spacecraft center of masses
- \( \hat{i}_B \): along the ‘nose’ of the spacecraft
- \( \hat{j}_B \): along the ‘right wing’ of the spacecraft
- \( \hat{k}_B \): right-hand triad

2.2.2 Matrix Transformations

Body to Stability

Considering no sideslip (\( \delta = 0 \)), the relative velocity vector is the same as the \( \hat{i}_S \) unit vector (see figure 2.2).

\[
T_{B2S} = \begin{bmatrix}
\cos \alpha & 0 & \sin \alpha \\
0 & 1 & 0 \\
-\sin \alpha & 0 & \cos \alpha
\end{bmatrix}
\] (2.1)
2.2. Coordinate Systems and Transformations

Stability to Wind

\[ T_{S2W} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \sigma & -\sin \sigma \\ 0 & \sin \sigma & \cos \sigma \end{bmatrix} \]  
(2.2)

Wind to Horizontal

\[ T_{W2H} = \begin{bmatrix} \cos \gamma \cos \psi & \sin \psi & \sin \gamma \cos \psi \\ -\cos \gamma \sin \psi & \cos \psi & -\sin \gamma \sin \psi \\ -\sin \gamma & 0 & \cos \gamma \end{bmatrix} \]  
(2.3)
2. NAVIGATION DESIGN

Figure 2.4: Wind to Horizontal Transformation

Horizontal to Rotational

\[ T_{H2R} = \begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \]  \hspace{1cm} (2.4)

Figure 2.5: Horizontal to Rotational Transformation

Rotational to Inertial

\[ T_{R2I} = \begin{bmatrix} \cos \phi \cos \theta & -\sin \theta & -\sin \phi \cos \theta \\ \cos \phi \sin \theta & \cos \theta & -\sin \phi \sin \theta \\ \sin \phi & 0 & \cos \phi \end{bmatrix} \]  \hspace{1cm} (2.5)
2.3 Environment Models

Mars can be, for the purpose of this project, fully described by three general parameters, which are gathered in table 2.1, an atmospheric model and a gravitational model. Furthermore, Mars will be considered geometrically spheric with a constant radius, the equatorial radius. For the case of the rotation of Mars, it is assumed no deviation between the rotation vector and the north pole vector.

<table>
<thead>
<tr>
<th>Table 2.1: Mars General Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>$R_d$</td>
</tr>
<tr>
<td>$\mu_d$</td>
</tr>
<tr>
<td>$\omega_d$</td>
</tr>
</tbody>
</table>

2.3.1 Atmospheric Model

The atmospheric model used in the simulation is the COSPAR northern summer atmosphere model. This model was already used by Fuhry in his master thesis in 1988 [3] and is simple, since it describes the density with the altitude.

The present thesis does not account for an epoch attached to a relative position to Mars. So there is no day or night that could be interpreted by other complexer atmospheric models. The atmospheric model is considered exponential, so an altitude-dependent temperature profile, which the perfect-gas equations could be used with, will
not be provided. There is no need either to consider an atmosphere in relative rotation with the planet. For Mars, this would mean only a 7% compared to, e.g. the Saturn’s atmosphere, which turns at approximately the 40% of the planet’s rotational speed. Furthermore, Saturn and Jupiter as well would need a special mention due to the turbulent atmospheres, the great rotation speed and the absence of well-defined surface. This is not the case for Mars [13]. Obviously, that causes the atmosphere to rotate with the same rate as Mars.

Although the model COSPAR is simple, the use of the Mars Global Reference Atmosphere Model (MARS - GRAM) is rejected. The Mars-GRAM is an engineering-level atmospheric model, which has many powerful simulation features such as dust particles, wind, temperature profiles... For that reason and with the above mentioned simplifications, the use of Mars-GRAM is considered not necessary since it exceeds the objectives of this work. Furthermore, the limit of the atmosphere is considered to be at an altitude of 131.1 km [14].

Table 2.2 collects the density model parameters needed to reproduce the profile of densities. Hence, densities are calculated using the actual altitude in the exponential density equation (eq. 2.6), which uses the values for scale height ($H_S$), reference altitude ($h_0$) and reference density ($\rho_0$) from table 2.2:

$$\rho = \rho_0 \exp \left( - \frac{h - h_0}{H_S} \right)$$

(2.6)
2.3. Environment Models

where the scale height fitting constants ($C$) are obtained from the same table:

$$HS = C_0 + C_1 h + C_2 h^2 + C_3 h^3$$  \hspace{1cm} (2.7)

### 2.3.2 Gravitational Model

Previously, it has been mentioned that Mars would be considered geometrically spheric. However, it will not be considered to have spheric symmetry in relation to the distribution of masses. That means, the planet has a gravitational asymmetry which causes a perturbation in the acceleration. This model takes $J_2$ perturbation into account. The value of the $J_2$ constant of Mars is 0.00196045 and its application is described below [1].

The acceleration due to gravity is derived from Newton’s law of universal gravitation, it is:

$$\vec{g} = -\frac{\mu \mathbf{r}}{||\mathbf{r}||^3} \hat{g}$$  \hspace{1cm} (2.8)

The $\hat{g}$ is the ‘perturbed’ inertial position unit vector. This vector is constituted as:

$$\hat{g} = \hat{i}_r + \frac{3}{2} J_2 \frac{R^2}{||\mathbf{r}||^3} ((1 - 5z^2) \hat{i}_r + 2z \hat{i}_\text{pole})$$  \hspace{1cm} (2.9)

where $\hat{i}_r$ and $\hat{i}_\text{pole}$ are the ‘not perturbed’ inertial position unit vector and the north pole unit vector of Mars, respectively (see equations 2.10 and 2.11).

$$\hat{i}_r = \frac{\mathbf{r}}{||\mathbf{r}||} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$  \hspace{1cm} (2.10)

$$\hat{i}_\text{pole} = \begin{bmatrix} -\sin \phi \cos \theta \\ -\sin \phi \sin \theta \\ \cos \phi \sin \psi \end{bmatrix}$$  \hspace{1cm} (2.11)

Introducing the equations 2.10, 2.11 and 2.12 into 2.9, one can achieve the following result:

$$\vec{g} = -\frac{\mu \mathbf{r}}{||\mathbf{r}||^2} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + \frac{3}{2} J_2 \frac{R^2}{||\mathbf{r}||^2} \left(1 - 5(\sin \phi \cos \theta)^2\right) \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$+ 2(\sin \phi \cos \theta) \begin{bmatrix} -\sin \phi \cos \theta \\ -\sin \phi \sin \theta \\ \cos \phi \sin \psi \end{bmatrix}$$  \hspace{1cm} (2.13)
For convenience, the resultant array at above may be replaced by three variables. This is done in order to ease subsequent operations.

\[
\mathbf{g} = \begin{bmatrix} g_x \hat{i}_R \\ g_y \hat{j}_R \\ g_z \hat{k}_R \end{bmatrix}
\]  

(2.14)

### 2.4 Vehicle Model

This thesis chooses Mars Science Laboratory (MSL) mission as a preliminary source of spacecraft’s parameters. The table below shows the general parameters of the vehicle, as well as the initial values for \(\beta\) and \(L/D\) before they are varied [3, 14, 15].

<table>
<thead>
<tr>
<th>Model Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entry Mass</td>
<td>2431 kg</td>
</tr>
<tr>
<td>Nose Radius</td>
<td>1.125 m</td>
</tr>
<tr>
<td>Base Radius</td>
<td>2.25 m</td>
</tr>
<tr>
<td>(S_{ref})</td>
<td>13.77 (m^2)</td>
</tr>
<tr>
<td>(\beta)</td>
<td>1025 (kg/m^2)</td>
</tr>
<tr>
<td>(C_D)</td>
<td>0.1722</td>
</tr>
<tr>
<td>(L/D)</td>
<td>1.5</td>
</tr>
</tbody>
</table>

Note: the spacecraft has a 70-deg cone in his front so the reference surface is calculated with the projection of its surface. Drag coefficient is calculated afterwards, with the equation 1.2.

### 2.5 Equations of Motion

The aerocapture maneuver can be described by a set of dynamical equations. These equations will be referenced into the previous defined rotational reference frame. This is a non-inertial frame. Hence, Coriolis effect must be taken into account.

There are only six variables needed to fully describe the trajectory at any coordinate system. This thesis chooses a spherical coordinate system (see figure 2.7). These variables are: longitude (\(\theta\)), latitude (\(\phi\)), radius (\(r\)), velocity (\(V\)), flight path angle (\(\gamma\)) and heading angle (\(\psi\)).
2.5. Equations of Motion

Figure 2.7: Spherical Coordinate System

Before the calculations, it is of convenience to assume some considerations that will simplify the equations:

- The vehicle is symmetric.
- The vehicle has constant mass during the whole aerocapture process.
- There is no thrust.
- There are no aerodynamic moments. That is, the vehicle is statically trimmed.
- There are neither side forces nor sideslip $\delta$.
- The angle of attack is constant and fixed at 10 degrees.

The equation 2.15 is the acceleration in the rotational reference frame [13]. With the objective to solve it, each one of the terms in the summation must be firstly found in the rotational coordinate system.

$$\vec{a} = \frac{\vec{F}_{\text{ext}}}{m} - 2(\vec{\omega} \times V) - \vec{\omega} \times (\vec{\omega} \times \vec{r})$$

(2.15)

First of all, it is required to find $\vec{r}$, $\vec{V}$ and $\vec{\omega}$. Then, the other can be found by multiplying the corresponding sequences.

$$\vec{r} = r \hat{i}_R$$

(2.16)

$$\vec{V} = T_{H2R} T_{W2H} \begin{bmatrix} V \hat{i}_W \\ 0 \hat{j}_W \\ 0 \hat{k}_W \end{bmatrix} = \begin{bmatrix} V \sin \gamma & \hat{i}_R \\ V \cos \gamma \cos \psi & \hat{j}_R \\ V \cos \gamma \sin \psi & \hat{k}_R \end{bmatrix}$$

(2.17)
\[ \vec{\omega} = T_{R2I}' \begin{bmatrix} 0 & \hat{i}_I \\ 0 & \hat{j}_I \\ \omega & \hat{k}_I \end{bmatrix} = \begin{bmatrix} \omega \sin \phi & \hat{i}_R \\ 0 & \hat{j}_R \\ \omega \cos \phi & \hat{k}_R \end{bmatrix} \] (2.18)

Operating the equations 2.16, 2.17 and 2.18:

\[
\vec{\omega} \times \hat{V} = \begin{bmatrix} \omega \sin \phi & \hat{i}_R \\ 0 & \hat{j}_R \\ \omega \cos \phi & \hat{k}_R \end{bmatrix} \times \begin{bmatrix} V \sin \gamma & \hat{i}_R \\ V \cos \gamma \cos \psi & \hat{j}_R \\ V \cos \gamma \sin \psi & \hat{k}_R \end{bmatrix} = \begin{bmatrix} -\omega V \cos \phi \cos \gamma \cos \psi & \hat{i}_R \\ -\omega V \sin \phi \cos \gamma \cos \psi & \hat{j}_R \\ \omega V \sin \phi \cos \gamma \cos \psi & \hat{k}_R \end{bmatrix}
\]

\[
\vec{\omega} \times \hat{r} = \begin{bmatrix} \omega \sin \phi & \hat{i}_R \\ 0 & \hat{j}_R \\ \omega \cos \phi & \hat{k}_R \end{bmatrix} \times \begin{bmatrix} 0 & \hat{i}_R \\ r & \hat{j}_R \\ 0 & \hat{k}_R \end{bmatrix} = \begin{bmatrix} 0 & \hat{i}_R \\ \omega r \cos \phi & \hat{j}_R \\ 0 & \hat{k}_R \end{bmatrix}
\]

\[
\vec{\omega} \times (\vec{\omega} \times \hat{r}) = \begin{bmatrix} \omega \sin \phi & \hat{i}_R \\ 0 & \hat{j}_R \\ \omega \cos \phi & \hat{k}_R \end{bmatrix} \times \begin{bmatrix} 0 & \hat{i}_R \\ \omega r \cos \phi & \hat{j}_R \\ 0 & \hat{k}_R \end{bmatrix} = \begin{bmatrix} -\omega^2 r \cos^2 \phi & \hat{i}_R \\ 0 & \hat{j}_R \\ \omega^2 r \sin \phi \cos \phi & \hat{k}_R \end{bmatrix}
\]

Now, since the axis of the rotational system also are in constant movement (because of \( \theta \) and \( \phi \)), it is necessary to evaluate the angular velocity vector of this coordinate system (\( \vec{\Omega} \)). The rotational frame firstly turns a rotation \( \theta \) about the positive \( \hat{k}_I \) axis and then turns a rotation \( \phi \) about the negative \( \hat{j}_R \) axis. Bearing this in mind, the resultant angular velocity vector is as follows:

\[
\vec{\Omega} = \left( \sin \phi \frac{d\theta}{dt} \right) \hat{i}_R - \left( \frac{d\phi}{dt} \right) \hat{j}_R + \left( \cos \phi \frac{d\theta}{dt} \right) \hat{k}_R \] (2.19)

The derivation of the axes in the inertial frame is equal to the vectorial product of the axes and its angular velocity in space, since they have not a longitudinal velocity.

\[
\frac{d\hat{i}_R}{dt} = \vec{\Omega} \times \hat{i}_R = \begin{bmatrix} 0 & \hat{i}_R \\ \cos \phi \frac{d\phi}{dt} & \hat{j}_R \\ 0 & \hat{k}_R \end{bmatrix}
\]

\[
\frac{d\hat{j}_R}{dt} = \vec{\Omega} \times \hat{j}_R = \begin{bmatrix} -\cos \phi \frac{d\phi}{dt} & \hat{i}_R \\ 0 & \hat{j}_R \\ \sin \phi \frac{d\phi}{dt} & \hat{k}_R \end{bmatrix}
\]

23
2.5. Equations of Motion

\[
\frac{d\hat{k}_R}{dt} = \hat{\Omega} \times \hat{k}_R = \begin{bmatrix}
-\frac{d\phi}{dt} & \frac{d\hat{i}_R}{dt} \\
-\sin \phi \frac{d\theta}{dt} & \frac{d\hat{j}_R}{dt} \\
0 & \frac{d\hat{k}_R}{dt}
\end{bmatrix}
\]

Taking the derivative from equation 2.16 one obtain:

\[
\vec{V} = \frac{d\vec{r}}{dt} = \frac{dr}{dt} \hat{i}_R + r \frac{d\hat{R}}{dt} = \begin{bmatrix}
\frac{dr}{dt} & \frac{\hat{i}_R}{dt} \\
r \cos \phi \frac{d\theta}{dt} & \frac{\hat{j}_R}{dt} \\
r \frac{d\phi}{dt} & \frac{\hat{k}_R}{dt}
\end{bmatrix}
\]  
(2.20)

If equations 2.17 and 2.20 are compared and combined, three new equations can be drawn. These new ones are the cinematic equations.

\[
\frac{dr}{dt} = V \sin \gamma
\]  
(2.21)

\[
\frac{d\theta}{dt} = \frac{V \cos \gamma \cos \psi}{r \cos \phi}
\]  
(2.22)

\[
\frac{d\phi}{dt} = \frac{V}{r} \cos \gamma \sin \psi
\]  
(2.23)

It follows, the equation 2.17 will be further derived and evaluated.

\[
\frac{d\vec{V}}{dt} = \begin{bmatrix}
\frac{d}{dt}(V \sin \gamma) \hat{i}_R + (V \sin \gamma) \frac{d\hat{i}_R}{dt} \\
\frac{d}{dt}(V \cos \gamma \cos \psi) \hat{j}_R + (V \cos \gamma \cos \psi) \frac{d\hat{j}_R}{dt} \\
\frac{d}{dt}(V \cos \gamma \sin \psi) \hat{k}_R + (V \cos \gamma \sin \psi) \frac{d\hat{k}_R}{dt}
\end{bmatrix}
\]  
(2.24)

\[
\hat{i}_R : \sin \gamma \frac{dV}{dt} + V \cos \gamma \frac{d\gamma}{dt} - \frac{V^2}{r} \cos^2 \gamma \cos^2 \psi
\]  
(2.25)

\[
\hat{j}_R : \cos \gamma \cos \psi \frac{dV}{dt} - V \sin \gamma \cos \psi \frac{d\gamma}{dt} - V \cos \gamma \sin \psi \cos \phi \frac{d\gamma}{dt}
\]  
(2.26)

\[
\hat{k}_R : \cos \gamma \sin \psi \frac{dV}{dt} - V \sin \gamma \sin \psi \frac{d\gamma}{dt} + V \cos \gamma \cos \psi \frac{d\psi}{dt}
\]  
(2.27)

The external forces that act on the spacecraft are the aerodynamic and gravitation ones. Aerodynamic force is split into lift and drag forces. Drag lies in the relative velocity vector but in backwards direction, and lift is perpendicular to the relative velocity vector and at the same time modulated by the bank angle (see figure 2.8). Independently to the bank angle, lift always lies in the opposite direction along the \(\hat{k}_S\) axis.

\[
\vec{F}_{ext} = \vec{F}_{aerodyn} + \vec{F}_{grav}
\]  
(2.28)
2. NAVIGATION DESIGN

Figure 2.8: Aerodynamic Forces

\[
\vec{F}_{\text{aerodyn}} = \vec{L} + \vec{D}
\]  
(2.29)

\[
\vec{L} = T_{H2R} T_{W2H} T_{S2W} \begin{vmatrix}
0 & \hat{i}_S \\
0 & \hat{j}_S \\
-L & \hat{k}_S
\end{vmatrix} = \begin{vmatrix}
L \cos \sigma \cos \gamma & \hat{i}_R \\
-L(\cos \sigma \sin \gamma \cos \psi - \sin \sigma \sin \psi) & \hat{j}_R \\
-L(\cos \sigma \sin \gamma \sin \psi + \sin \sigma \cos \psi) & \hat{k}_R
\end{vmatrix}
\]  
(2.30)

\[
\vec{D} = T_{H2R} T_{W2H} \begin{vmatrix}
-D & \hat{i}_W \\
0 & \hat{j}_W \\
0 & \hat{k}_W
\end{vmatrix} = \begin{vmatrix}
-D \sin \gamma & \hat{i}_R \\
-D \cos \gamma \cos \psi & \hat{j}_R \\
-D \cos \gamma \sin \psi & \hat{k}_R
\end{vmatrix}
\]  
(2.31)

The gravity force can be calculated as the product of mass with equation 2.14. Then, the external force is the sum of the previous found expressions.

\[
\vec{F}_{\text{grav}} = m\vec{g} = \begin{vmatrix}
m g_x & \hat{i}_R \\
m g_y & \hat{j}_R \\
m g_z & \hat{k}_R
\end{vmatrix}
\]  
(2.32)

\[
\vec{F}_{\text{ext}} = \begin{vmatrix}
L \cos \sigma \cos \gamma - D \sin \gamma + m g_x & \hat{i}_R \\
-L(\cos \sigma \sin \gamma \cos \psi - \sin \sigma \sin \psi) - D \cos \gamma \cos \psi + m g_y & \hat{j}_R \\
-L(\cos \sigma \sin \gamma \sin \psi + \sin \sigma \cos \psi) - D \cos \gamma \sin \psi + m g_z & \hat{k}_R
\end{vmatrix}
\]  
(2.33)
In order to simplify the above equation, the dynamic pressure, the ballistic coefficient (equation 1.2) and lift-to-drag ratio are going to be used to replace variables and shorten the external forces expression.

Drag can be developed as following equation shows:

\[
D = \frac{1}{2} \rho V^2 S_{ref} C_D
\]  

(2.34)

Now, using the ballistic coefficient equation (eq. 1.2) one can obtain:

\[
\frac{D}{m} = \frac{q}{\beta}
\]  

(2.35)

where \( q \) is the dynamic pressure. Then, using the lift-to-drag ratio and operating, one obtains the following equations:

\[
\frac{L}{m} = \frac{L}{D} \frac{D}{m} = \frac{L}{D} \frac{q}{\beta}
\]  

(2.36)

\[
\begin{align*}
\frac{\vec{F}_{ext}}{m} &= \left\{ \begin{array}{l}
\frac{L}{D} \frac{q}{\beta} \cos \sigma \cos \gamma - \frac{q}{\beta} \sin \gamma + g_x \\
- \frac{L}{D} \frac{q}{\beta} (\cos \sigma \sin \gamma \cos \psi - \sin \sigma \sin \psi) - \frac{q}{\beta} \cos \gamma \cos \psi + g_y \\
- \frac{L}{D} \frac{q}{\beta} (\cos \sigma \sin \gamma \sin \psi + \sin \sigma \cos \psi) - \frac{q}{\beta} \cos \gamma \sin \psi + g_z
\end{array} \right\} \\
\end{align*}
\]  

(2.37)

Introducing the previous equations into equation 2.15, a set of three simultaneous equations containing three variables is provided (equations 2.38, 2.39 and 2.40) Next step is to find the variables \( dV/\!\!\!\!/dt \), \( d\gamma/\!\!\!\!/dt \) and \( d\psi/\!\!\!\!/dt \).

\[
\begin{align*}
\hat{i}_R: \sin \gamma \frac{dV}{dt} + V \cos \gamma \frac{d\gamma}{dt} - \frac{V^2}{r} \cos^2 \gamma &= - \frac{q}{\beta} \sin \gamma \\
&+ \frac{L}{D} \frac{q}{\beta} \cos \sigma \cos \gamma + g_x + 2w V \cos \gamma \cos \psi \cos \phi + w^2 r \cos^2 \phi
\end{align*}
\]  

(2.38)

\[
\begin{align*}
\hat{j}_R: \cos \gamma \cos \psi \frac{dV}{dt} - V \sin \gamma \cos \psi \frac{d\gamma}{dt} - V \cos \gamma \sin \psi \frac{d\psi}{dt} \\
&+ \frac{V^2}{r} \cos \gamma \cos \psi (\sin \gamma - \cos \gamma \sin \psi \tan \phi) \\
&= - \frac{q}{\beta} \cos \gamma \cos \psi - \frac{L}{D} \frac{q}{\beta} (\cos \sigma \sin \gamma \cos \psi - \sin \sigma \sin \psi) + g_y \\
&+ 2w V (\cos \gamma \sin \psi \sin \phi - \sin \gamma \cos \phi)
\end{align*}
\]  

(2.39)
\[ \dot{k}_R : \cos \gamma \sin \psi \frac{dV}{dt} - V \sin \gamma \sin \psi \frac{d\gamma}{dt} + V \cos \gamma \cos \psi \frac{d\psi}{dt} + \frac{V^2}{r} \cos \gamma (\sin \gamma \sin \psi + \cos \gamma \cos^2 \psi \tan \phi) \]
\[ = -\frac{q}{\beta} \cos \gamma \sin \psi - \frac{L q}{D \beta} (\cos \sigma \sin \gamma \sin \psi + \sin \sigma \cos \psi) + g_z \]
\[ - 2wV \cos \gamma \cos \psi \sin \phi - w^2 r \sin \phi \cos \phi \]

Once the simultaneous equations are resolved, the resulting are the dynamic equations (eq. 2.41, 2.42 and 2.43). These are ordinary differential equations and are going to be solved by the simulation model written in Simulink®, since this programming language is specialized in simulating dynamic systems. Note that \( w^2 r \) is much smaller than the others and can generally be eliminated. However, in this thesis it will be conserved in order to gain accuracy in the monitoring of the inertial position around Mars. Without it, for example, it would cause the inertial trajectory to rotate slowly over time.

\[ \frac{dV}{dt} = -\frac{q}{\beta} \cos \gamma \sin \psi + g_x \sin \gamma \cos \gamma \cos \psi + g_z \cos \gamma \sin \psi + \frac{w^2 r}{r} \cos \phi (\sin \gamma \cos \phi - \cos \gamma \sin \psi \sin \phi) \]
\[ (2.41) \]

\[ V \frac{d\gamma}{dt} = \frac{L q}{D \beta} \sin \sigma + g_x \cos \gamma \cos \gamma \cos \psi + g_y \sin \gamma \cos \psi - g_z \sin \gamma \sin \psi + \frac{V^2}{r} \cos \gamma \]
\[ + 2wV \cos \psi \cos \phi + w^2 r \cos \phi (\cos \gamma \cos \phi + \sin \gamma \sin \psi \sin \phi) \]
\[ (2.42) \]

\[ V \frac{d\psi}{dt} = -\frac{L q \sin \sigma}{D \beta} \cos \gamma - g_y \sin \psi \cos \gamma \cos \gamma + g_z \frac{\cos \psi}{\cos \gamma} - \frac{V^2}{r} \cos \gamma \cos \psi \tan \phi \]
\[ + 2wV (\tan \gamma \sin \psi \cos \phi - \sin \phi) - \frac{w^2 r}{\cos \gamma} \cos \psi \sin \phi \cos \phi \]
\[ (2.43) \]

### 2.6 Aerodynamic Heating

The heating of a spacecraft by friction with the surrounding air can cause the spacecraft literally to burnout. It is important to monitor heating flux rate and total heat load in order to investigate which are the best paths to be flown. That means, the trajectories that do not introduce much heat into the spacecraft, which can damage the structure of the vehicle and the onboard systems. If the spacecraft flies through a very steep trajectory the heat flux will be greater and the vehicle will gain an important amount of heat in a short time span. If the trajectory is too shallow, the heat flux will be lower but the vehicle may remain more time in the atmosphere and hence receive a higher quantity of total
heat load. A limitation in both variables is to be introduced. The values were arbitrary set to 250 W/cm\(^2\) for heat flux, which does not exceed the applicable maximum of 260 W/cm\(^2\) for the MSL mission [8], and 35 kJ/cm\(^2\) for heat load.

The following equation describes the heating rate behaviour of the spacecraft through the Martian atmosphere [3]:

\[
\dot{Q} = \frac{865.0 \sqrt{\rho}}{R_n} \rho_{0_{SL}} \left( \frac{V}{10000} \right)^{3.5} 
\]  \hspace{1cm} (2.44)

where \(R_n\) is the nose radius of the vehicle and \(\rho_{0_{SL}}\) is the sea-level density at Earth, which is 1.225 kg/m\(^3\).

Once the heat flux is calculated it must be integrated over time in order to obtain the actual amount of heat load. As heat flux is a time-variable function and there is any available analytical function, the integration is simplified into a simple product of heat flux and time span and summation of all previous heat load values.

\[
Q(t) = \int_0^t \dot{Q}(\tau) d\tau 
\]  \hspace{1cm} (2.45)

\[
Q(t) = \dot{Q} \Delta t
\]

The units of the heat flux equation (eq. 2.44) are Btu/ft\(^2\)s and need to be converted into International System. As opposite, nose radius and relative velocity must be first converted into Imperial Units. Below are announced the used change factor values:

\[
1 \frac{\text{Btu(th)}}{\text{ft}^2\text{s}} = 1.134893179 \frac{\text{W}}{\text{cm}^2} \hspace{1cm} (2.46)
\]

\[
1 \text{ m} = 3.280839895 \text{ ft} \hspace{1cm} (2.47)
\]

\[
1 \frac{\text{m}}{\text{s}} = 3.280839895 \frac{\text{ft}}{\text{s}} \hspace{1cm} (2.48)
\]
Chapter 3

SIMULATION DESIGN

3.1 Overview

Once the equations of motion and the different models have been settled, it is necessary to translate them into a group of computing processes that will allow the problem to be simulated as planned. This chapter describes the functionality of the implemented simulation system, starting from the main program so-called AECASIM (Aerocapture Simulator) and introducing all subroutines that form the whole program.

3.2 Simulation tool - AECASIM

AECASIM was written with a combination of Matlab® and Simulink® sources. The top-level routine, which is responsible to process all data and integrate it, was performed with Simulink®, since this is a specific tool for simulating dynamical systems. However, all the low-level subroutines that feed it were written in Matlab®.

The flow diagram of AECASIM is represented in figure 3.1. It starts with three inputs: spacecraft properties, planet properties and initial conditions. Spacecraft properties and planet properties correspond to table 2.3 and section 2.3, respectively. The initial condition parameters will be later presented in subsection 3.4.2. In order to be able to simulate different conditions, some of those parameters were programmed to be initialized within a first routine, which must prepare the simulation engine and call AECASIM. This means, there are some constants that can be changed before the simulation begins, allowing multiple-case results.
For example, lift-to-drag ratio can be changed each time and so produce different outcomes but will remain constant throughout a single simulation. These variable parameters are: relative velocity, flight path angle, lift-to-drag ratio and drag coefficient (see section 3.4).

The information from input data goes through different paths feeding different subprocesses but in the end, all the information is gathered into Dynamics process. There are two special routines designed in Simulink that need to be run before, in order to generate some information that Dynamics also need. This information is the components of the perturbed gravity vector and the dynamic pressure. The application Atmosphere Model must be run to generate the actual density and so compute the dynamic pressure.
Once the original and the lastly generated information go to the Dynamics module, the equations of motion are integrated forward in time to provide the new state of the spacecraft. When Dynamics has calculated the new state, it sends the navigation and aerodynamics information to PredGuid module and to output routine as well. PredGuid must then calculate the new bank angle command in order to guide the spacecraft to the desired target. The guidance program sends the commanded bank angle to Dynamics, in order to calculate the new state\(^1\). The commanded bank angle goes to output routine and the computed aerodynamic heating too. Nevertheless, it is remarkable that the navigation information that PredGuid receives from Dynamics has not been altered. It would be necessary to create a tool in order to simulate the acquirement of the information that the vehicle’s sensors would do. Hence, the global simulation would be more realistic and the guidance scheme would provide more information about its robustness and accuracy.

This whole process is iterative, as it runs periodically from the start until the simulation time has reached the end. The dashed lines in figure 3.1, that stand for initial condition parameters, mean that they do not participate actively in the cyclical run of AECASIM. They only give information in the first step of the simulation. Afterwards, that information is substituted by new one, which comes from Dynamics routine.

\(^1\)There is a special feature in AECASIM that should be noticed. It is a manual switch that allows to simulate flight conditions with constant bank angle during the entire simulated time. This object switches between an input parameter, which can provide constant bank angle or an array of pre-defined bank angle values, and the constantly-changing bank angle commands that PredGuid generates. That means, it is possible to simulate flights with a constant bank angle along the whole simulation time.
3.3 Guidance

The guidance application used in the simulation is a numeric predictor-corrector so-called PredGuid. It was widely used in many investigations about aerocapture. This thesis uses a self-version of PredGuid, which is a combination of two principal works. The resulting software takes the whole content that DiCarlo already improved [1], whose certain items were modified for convenience, but instead of the energy management phase based on a reference drag, it uses Fuhry’s version, which is based on a defined constant cruise altitude of 34 km [3].

This software works with three different types of parameters: local variables, global variables and constants. The main difference between the first two is that local variables are exchanged between neighbour or consequent subroutines. Global variables are being used by different subroutines, without them being subsequent in order of execution. The most common variables of these type are flags which are activated or deactivated in one function and have an application later in the software. Hence, one must not carry these parameters throughout the whole program.

Below, the logic flow of PredGuid (see figure 3.2) and its components are described to provide a deeper comprehension of the guidance functionality.

3.3.1 PredGuid Flow Chart

As mentioned before, PredGuid has the due to periodically determine the bank angle commands that control the spacecraft to a defined objective altitude. It takes the actual state of the spacecraft and predicts the next state integrating forward in time, considering always a constant bank angle.

This program is thought to calculate the desired bank angle at a frequency of 0.5 Hz. This angle is the estimated angle that will bring spacecraft to its next target. Also, it must provide, at a frequency of 1 Hz, the commanded bank angle. This is the real bank angle the spacecraft flies with and represents the non-instantaneous change between desired bank angles [1].

Figure 3.2 shows a depicted view of the algorithm’s logic. In the first pass of execution all global variables are initialized and initialization flag set to off, in order to avoid the program to initialize these variables over again and lose information with it. Then, if aerodynamic acceleration (measured in Earth g) is high enough, the aerodynamic properties are updated and whether this process has been performed or not, the following subroutine, Aerodynamic Heating, computes the actual heat flux rate and then calculates the accumulated amount of heat load by the spacecraft.
Next, if the total aerodynamic load is higher than a certain limit, which is higher than the previous, guidance can be applied. If not, it calculates neither the commanded bank angle nor the desired bank angle and exits this cycle with any new command. On the other hand, if the aerodynamic load is high enough to apply guidance, the program will execute Bank Angle Determination, Lateral Control and Command Incorporation functions sequentially.

Bank Angle Determination is a process divided in two phases: the first is an Energy Management phase, which controls the spacecraft to a constant cruise altitude until sufficient energy has been dissipated and some conditions to change to next phase are fulfilled. The second is the Targeting phase, which controls the spacecraft until it performs a controlled exit from the atmosphere, where aerodynamic forces are not existent and hence spacecraft can no longer be controlled. Thus, it is important to exit the atmosphere in a controlled way, in order to be able to reach the target altitude.

Lateral Control checks whether spacecraft has gone beyond lateral limits or not and actuates consequently correcting the sign of the bank angle, which is the one responsible of the lateral velocity. Command Incorporation provides the new commanded bank angle.

Finally, since the generation of the desired bank angle must run at a lower frequency, there is a counter so-called Guidance Pass that allows PredGuid to compute that angle or not, depending on its value.
3.3. Guidance

Aerodynamic Heating

Yes

First Pass

Aerodynamic Properties

No

Yes

g_{load} > atmos. min.

Yes

No

Guidance Pass = 1

Yes

No

Bank Angle Determination

Yes

No

Lateral Control

Commanded bank angle

Increment Guidance Pass

Guidance Pass = lim

Reset Guidance Pass

Command Incorporation

Output

- Commanded bank angle

Figure 3.2: PredGuid Flow Chart
3. SIMULATION DESIGN

3.3.2 Initialization

This function initializes the global guidance parameters on the first guidance pass. It attributes previous-saved values to all global parameters that will be used along the whole PredGuid functionality (see figure 3.3). An important parameter is the unit normal to desired orbit plane ($\hat{i}_{dp}$), which is used by Lateral Control procedure to calculate the lateral velocity with respect to the desired plane and makes possible the lateral guidance of the spacecraft. In this project, the desired orbital plane is calculated instead of being initialized from a predefined value, and it corresponds to the orbit plane where the spacecraft starts the simulation in.

![Initialization Flow Chart](image-url)

Figure 3.3: Initialization Flow Chart
3.3.3 Aerodynamic Properties

This function has the aim to update the aerodynamic properties to the actual state if drag acceleration is greater than zero in module. It starts with a computation of unit vectors, which are then used to compute the aerodynamic accelerations (see figure 3.4). Later, drag acceleration is checked to be greater than zero. If true, angle of attack is calculated and drag coefficient updated with it. The real density is then calculated with the Atmospheric Model function. Next, with the acceleration input and the recent calculated drag coefficient, the measured density at current altitude is estimated. Afterwards, the subroutine calculates with both densities the estimated density bias, which is filtered with a constant filter gain. In the end, the estimated lift-to-drag ratio is calculated with the aerodynamic accelerations and is filtered too.

Notice that since this works assumes constant drag coefficient and lift-to-drag ratio all simulation long and there are no errors in the density estimation, this function would not be necessary. However, it was included from other works for possible future applications.
3. SIMULATION DESIGN

Compute unit vectors
Compute aerodynamic accelerations
Compute angle of attack
Compute estimated drag coefficient

\[ a_{\text{drag}} > 0 \]

No

Yes

Compute angle of attack
Compute estimated drag coefficient

Atmospheric Model

Compute estimated density at current altitude
Compute and filter density bias est. and lift to drag est.

Output
- Global variables updated

Input
- Input Variables
- Global variables
- Constants

Input Variables
Global variables
Constants

Figure 3.4: Aerodynamic Properties Flow Chart
3.3.4 Atmospheric Model

The Atmospheric Model calculates the actual density based on the atmospheric model from subsection 2.3.1. It first checks whether the actual altitude is higher than the atmosphere limit and if it is true, it calculates the scale height using a fourth order curve fit. Lastly, density is calculated with the exponential density equation (eq. 2.6).

Figure 3.5: Atmospheric Model Flow Chart
3. SIMULATION DESIGN

3.3.5 Aerodynamic Heating

This subroutine uses the Atmospheric Model to generate the actual density value and then calculates the heating properties corresponding to equations 2.44 and 2.45. The resulting heat load is summed to the accumulated term (see figure 3.6).

![Aerodynamic Heating Flow Chart](image)

Figure 3.6: Aerodynamic Heating Flow Chart
3.3.6 Energy Management

This function has as objective to deplete sufficient energy in order to allow targeting of the exit conditions in targeting phase. To do so, this phase is modelled as a second-order spring-mass-damper system [1].

Energy Management is executed as depicted in figure 3.7. It starts with the calculation of the ballistic coefficient. Afterwards, an altitude rate damping term, an altitude deviation term and the cosine of the bank angle for zero altitude rate are calculated. With all this information, the commanded bank angle for constant cruise is computed. In the end, the program checks whether is time to check phase or not and executes Phase Check function if needed. Time to check phase corresponds to a frequency of 0.1667 Hz or every 6 seconds.

Notice that compared to targeting phase, this function calculates the commanded bank angle instead of the desired. Therefore, only Energy Management is executed at 1 Hz, which is the frequency corresponding to the commanded bank angle.
3. SIMULATION DESIGN

Figure 3.7: Energy Management Flow Chart
3.3.7 Phase Check

Phase Check function (see figure 3.8) has the due to check whether the targeting phase should be initialized or the energy management phase should continue. There are two conditions to be fulfilled in order to allow that change. First: the spacecraft can fly a constant angle of 110 degrees for the remaining of the trajectory and still have the resulting apoareion within a given tolerance. Second: energy is negative, which indicates that spacecraft is no longer on an hyperbolic trajectory. If both conditions are true the targeting phase is initialized.

![Phase Check Flow Chart](image)

Figure 3.8: Phase Check Flow Chart
3. SIMULATION DESIGN

3.3.8 Targeting

Targeting function alternately runs Predictor and Corrector functions to determine the desired bank angle needed to guide the spacecraft to the target altitude.

The sequentially order of this algorithm is as below described (see figure 3.9). Firstly, counting and interpolation variables must be initialized every time Targeting function is called. Then, apoareion miss tolerance is calculated taking a certain constant percent of the actual inertial velocity, ensuring an increasing accuracy as velocity falls down.

A part of the Targeting function is repeated cyclically a finite number of times within a Targeting single execution. If it is not capable of finding an accurate value for the bank angle before the maximum number of runs is reached, Corrector is called to generate a last desired bank angle guess. Otherwise, Corrector is called to produce a new bank angle guess, which is next used in Predictor to predict the resulting apoareion. This outcome is compared to target apoapsis to compute the miss. Afterwards, the function checks whether this miss is greater or smaller than the before-calculated tolerance. If true, this means the try was acceptable and Targeting finishes. If false, the guess must be characterized between multiple possibilities. If the miss is high and the bank angle is too shallow the spacecraft would continue going higher so a full lift down is required. It also can inversely occur that the miss is low and the bank too steep, so a full lift up is required. If any of these two possibilities happen the outcome can be good (but still high or low) or capture (see Predictor algorithm in subsection 3.3.10). The respective variables must be updated for further application. Next, if the solution is bracketed, this means that for example there is one high and low result or high and capture, the function checks whether the bank miss is smaller than the bank angle tolerance.

Finally, if one or both last two conditions are not fulfilled, the Targeting algorithm restarts from the beginning of the cyclical part correcting the bank angle to try with the corrector until one properly bank angle is found or there are not more possible runs of this cycle.
Figure 3.9: Targeting Flow Chart
3. SIMULATION DESIGN

3.3.9 Corrector

This algorithm generates a new bank angle to try in Predictor. There are several methods to find the bank angle based on the characterization of previous states or solutions from Targeting (see figure 3.10).

1. If this is the first run, it means that the Targeting algorithm has just begun (see Targeting algorithm). Then, it is legal to think that the last used bank angle could be still good. Therefore, Corrector chooses here the last used guess.

2. If the solution is bracketed (at least one high and low or capture) and the number of low guesses is greater than zero, the program interpolates a high and a low guess to target.

3. If the solution is bracketed, there are any low guesses and the number of high guesses is one, the program interpolates a high and a capture to target.

4. If the solution is bracketed, there are any low guesses and the number of high guesses is different from one, the program extrapolates two high guesses. However, it checks whether the resulting bank is greater or equal to the one that had produced a capture. If so, it interpolates a high and a capture.

5. If the solution is not bracketed and the number of good guesses is one, a “Smart Guess” is executed [1]. It consists of increasing or decreasing the last tried bank angle one degree if the solution is high or low, respectively.

6. If the solution is not bracketed and there are two or more good guesses, the program extrapolates two high or two low guesses.

7. If the solution is not bracketed and there are no good guesses, it means that there are only captures. The spacecraft must then exit the capture so a march out of capture region is achieved by increasing the cosine of the bank angle.
Figure 3.10: Corrector Flow Chart
3. SIMULATION DESIGN

3.3.10 Predictor

The Predictor function calculates the predicted apoareion assuming the current bank angle guess is constant.

As flow chart shows in figure 3.11, Predictor starts initializing the predicted variables, because every time it is executed it generates a new predicted apoareion. Afterwards, it calculates the ballistic coefficient, the relative velocity and the current density to find the aerodynamic accelerations.

Next, the perturbed gravitational acceleration is computed. Integrator algorithm is called to use the total acceleration and the previous information in order to find the future state of the vehicle (see Integrator algorithm in subsection 3.3.11). Then, it calculates the centrifugal velocity (or altitude rate) and the flight path angle and checks whether the vehicle has exited the atmosphere. If so, the program computes the predicted apoareion\(^2\) and exits. Otherwise, it calculates the centrifugal acceleration and then checks whether there is an atmospheric capture. In case the vehicle has been captured, the predicted apoareion is set to negative infinity. In case there is no atmospheric capture, it means this is not an undesirable case. The prediction has neither gone too high, outside the atmosphere, nor is being dragged down to Earth.

If any unwanted case has occurred, the algorithm restarts over again, from the calculation of the relative velocity, until something wrong happens or the integration is completed. The last condition is fulfilled when fourth integration step is finished.

---

\(^2\)This calculation is performed by a set of equations, that where modified from DiCarlo’s work [16].
3.3. Guidance

**Input**
- Input Variables
- Global variables
- Constants

Input

- Initialize predicted variables
- Calculate ballistic coefficient
- Calculate relative velocity
- Atmospheric Model
  - Compute aerodynamic acceleration
  - Compute gravitational acceleration

**Integrator**
- Calculate total acceleration

**Atmosphere exit**
- Calculate centrifugal velocity and flight path
- Integration completed

**Atmosphere capture**
- Set apoareion to negative infinity

**Output**
- Predicted apoareion
- Global variables updated

Figure 3.11: Predictor Flow Chart
3. SIMULATION DESIGN

3.3.11 Integrator

When Prediction send the accumulated state variables to the Integrator algorithm, this uses a fourth order Runge-Kutta method to explicitly integrate the predicted state vectors one time step forward. Integrator must only perform the correct integration step from the set of equations that compound this method.

![Integrator Flow Chart](image)

Figure 3.12: Integrator Flow Chart
3.3.12 Lateral Control

The main objective of this module is to determine the need for roll reversals that keep the spacecraft in the desired orbital plane. To do so, it first calculates the maximum allowed lateral corridor velocity, which is a constant percentage of the actual inertial velocity. Then, the real out of plane velocity is calculated with the unit normal to the desired orbit plane and the inertial velocity. If lateral velocity is greater than the maximum, Lateral Control performs a roll reversal by changing the sign of the bank angle.

![Lateral Control Flow Chart](image)

Figure 3.13: Lateral Control Flow Chart
3.3.13 Command Incorporation

Command Incorporation updates the bank angle command. Firstly, it calculates the commanded bank angle based on the desired bank and the previous commanded bank angle. Since this procedure is executed at a doubled frequency than guidance’s, there will be two different commanded bank angles for each desired bank angle. The first will be the midpoint between the last commanded bank angle and the desired, and the second will be equal to the desired. This represents a discontinuous change between desired bank angles.

Finally, the calculated bank angle is limited to its maximum and minimum values and the sign of bank is incorporated. The outcome, in sign and value, is the bank angle the spacecraft to flies with.

Figure 3.14: Command Incorporation Flow Chart
3.4 Simulation Implementation

3.4.1 Vehicle Properties for Study

As commented in section 3.2, there are some parameters that will remain constant for the whole study, and other that will be changed in order to perform a study that fits the project’s objectives. However, these parameters will firstly have a constant value that will be analysed in the assessment of the simulation model. The commented possibilities by varying those parameters will be studied and analysed in chapter 4.

Lift-to-drag ratio and ballistic coefficient are two vehicle properties that will change. They will be incremented and decremented in order to study its repercussion on the aerocapture maneuver. The first values of them are those from table 2.3. The values for the variations were chosen arbitraly, however the variations lay within the ranges of the reference. The values for lift-to-drag ratio are 1 and 2 and the values for ballistic coefficient are 683.33 $\text{kg/m}^2$ and 1366.67 $\text{kg/m}^2$.

3.4.2 Initial Conditions at Entry Interface

The parameters corresponding to the initial conditions lay on table 3.1, but only the inertial velocity and the flight path angle are going to be changed.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>131.1 km</td>
</tr>
<tr>
<td>Inertial Velocity</td>
<td>6 km/s</td>
</tr>
<tr>
<td>Longitude</td>
<td>0 deg</td>
</tr>
<tr>
<td>Latitude</td>
<td>0 deg</td>
</tr>
<tr>
<td>Flight Path Angle</td>
<td>$-15 \text{ deg}$</td>
</tr>
<tr>
<td>Heading Angle</td>
<td>0 deg</td>
</tr>
</tbody>
</table>

For convenience, the longitude, latitude and heading angle values at entry interface will be zero. The value of the altitude is set like this due to the assumption that the spacecraft enters the atmosphere at that height [14], and the selected inertial velocity of 6 km/s is a widely used value for robotic Martian missions [17]. Finally, the value for the flight path angle was also selected arbitrarily.

As Dynamics module of AECASIM calculates the new state of the spacecraft in a non-inertial reference frame and uses the relative velocity, this must be calculated. This
would be done with equation 3.1 but, as there is no vectorial treatment in AECASIM inputs, it is performed with equation 3.2.

\[
V = \left| \vec{V}_{\text{inertial}} - \vec{\omega}_d \times \vec{r} \right| \tag{3.1}
\]

\[
V = \sqrt{(V_{\text{inertial}} \cos \gamma \sin \psi)^2 + \left( V_{\text{inertial}} \cos \gamma \cos \psi - \omega_d r \cos \phi \right)^2 + (V_{\text{inertial}} \sin \gamma)^2}; \tag{3.2}
\]

Lastly, the inertial velocity will range from 5 to 7 km/s with an increment of 0.25 km/s. This means, the final inertial velocities to study will be nine: 5, 5.25, 5.5, 5.75, 6, 6.25, 6.50, 6.75 and 7 km/s. For the flight path angle, the studied values will vary from -30 to -5 degrees with an increment of 0.5 degrees.

### 3.4.3 Simulation Parameters

The selected integration engine from the Simulink\(^\text{©}\) possibilities is “ode45”. It corresponds to Runge-Kutta integration method, which matches the Predictor method. Since Runge-Kutta is an explicit integration method, a fixed time step is preferred as the variable step is much more useful for high frequency dynamical problems. Then, the value of the time span must allow all subroutines to run properly under their respective frequencies. Therefore, the selected simulation frequency must be greater than the maximal one, which is 1 Hz. So, the resulting frequency for AECASIM process is 2 Hz and corresponds to a time step of 0.5 seconds.

Finally, the maximum time for simulation that best fits the results was found to be 3000 seconds. This is time enough to find the highest values of altitude of all studied cases.
3.4. Simulation Implementation

[This page is intentionally left blank]
Chapter 4

ANALYSIS OF RESULTS

4.1 Software Verification

The AECASIM software is thought to be built by different modules. First, Dynamics module, which is constituted by the equations of motion. This is the core of the simulation tool. Second, PredGuid algorithm, which provides guidance of the spacecraft. Third, the $J_2$ perturbation calculator or gravitational module. And fourth, the aerodynamic module, which computes the actual density and dynamic pressure.

The process of verification should concern all the commented modules in order to proof the correct operation of the whole system. However, there are some obstacles that prevent all modules to be verified. Unfortunately, there are not enough parallelisms in the works of DiCarlo [1] and Fuhry [3] in order to perform a rigorous analysis of verification for PredGuid algorithm. As for the $J_2$ perturbation calculator and the aerodynamic module, it has not been possible to perform the appropriate verification due to the difficult access of the information it requires. Nevertheless, as these modules are based on known equations or information that was already published in other works. Furthermore, the Model Assessment section describes the aerocapture problem giving explanations and justifications for the obtained results. Therefore, some minor verification could be found on that section.

It follows, two different cases will be analysed without $J_2$ effect and atmosphere in order to check whether the orbital mechanics, without either perturbations and aerodynamic forces, are correctly implemented.
The simplest example of an orbit is that with an elliptical form since it is closed and has more remarkable features than the circle orbit. The two simulated cases have identical ellipses but in a different arrangement. The first will lay within the equatorial plane of Mars and the second will have more inclination, in order to check whether the elliptic conditions are also fulfilled in a 3rd dimensional space. Both ellipses have the same periapsis value of 6792.4 km, which is two times the equatorial radius of Mars, but different values for latitude and heading angle. The selected apoapsis is three times the periapsis value.

Since AECASIM needs six spherical parameters as initial conditions, these need to be calculated with the previous information. The program also needs the maximum time for simulation and the time step for integration. Equations 4.1 to 4.4 are used to calculate the needed data.

The altitude can be easily calculated as:

\[ h_p = r_p - R_D \]  (4.1)

Next, the semi-major axis is computed in equation 4.2 and applied in the Vis-Viva Equation (eq. 4.3) to calculate the inertial velocity at periapsis. At this point, the flight path angle is by definition zero.

\[ a = \frac{r_p + r_a}{2} \]  (4.2)

\[ V_{\text{inertial}} = \sqrt{\frac{2\mu_D}{r} - \frac{\mu_D}{a}} \]  (4.3)

AECASIM needs the relative velocity though, which is calculated with equation 3.2. Longitude, latitude and heading angle are arbitrary defined. Furthermore, the period is calculated in equation 4.4 and has a value of approximately 48072 seconds or 13.3534 hours. Since this time is enormous compared to the aerocapture maneuver, the simulation time step has been increased to 10 seconds, which is small enough to provide good accuracy and large enough to avoid long times of simulation. In order to simulate a whole orbit cycle, the maximum simulation time must be set to the period value. However, this time must be divisible by the step time, so the final selected maximum time for simulation is 48070 seconds.

\[ P = 2\pi \sqrt{\frac{a^3}{\mu_D}} \]  (4.4)
Example 1

As this is the first example, the values for longitude, latitude and heading angle will be set to zero for simplicity. The following list resumes the values that should be directly introduced in AECASIM, in order to obtain the results of figures 4.1 to 4.3.

- Altitude = 3396.2 km
- Relative Velocity = 2.5939 km/s
- Longitude = 0 deg
- Latitude = 0 deg
- Flight Path Angle = 0 deg
- Heading Angle = 0 deg
- Simulation Time = 48070 s
- Time Step = 10 s

Figure 4.1: Top View of Inertial and Relative Positions for Example 1
Figure 4.1 shows two different line types. The normal line is the inertial position, which corresponds to the calculated elliptic orbit. One can see that the periapsis position is two (the units of the chart are distance over Mars radius) and the apoapsis distance is three times greater than the first. Nevertheless, the red-dashed line does not seem to represent at all the same ellipse. This is because it is the relative position. This means, that this is what would be seen from an observer on Mars’ surface. If the relative position would be rotated with Mars’ rotational rate, it would look like the inertial position.

Figures 4.2 and 4.3 represent the distance from the spacecraft to the center of Mars vs time and inertial velocity vs time, respectively. As expected, the distance and velocity behave oppositely. As distance grows up, velocity falls and the other way round. The time for the maximum and minimum values are also the same, which is normal.

Figure 4.2: Distance over Mars Radius vs Time for Example 1
Example 2

The second example proofs that the AECASIM mechanics also work for other initial conditions, so the same behaviours for position and inertial velocity are expected. This time, longitude, latitude and heading angle will not be zero.

- Altitude = 3396.2 km
- Relative Velocity = 2.7349 km/s
- Longitude = 90 deg
- Latitude = 45 deg
- Flight Path Angle = 0 deg
- Heading Angle = 15 deg
- Simulation Time = 48070 s
- Time Step = 10 s

From figure 4.4 one can see that the changed variables produced a change in the orbit orientation, but as seen in figures 4.5 and 4.6 the orbit forms are the same.
4.1. Software Verification

Figure 4.4: Three-dimensional View of Inertial and Relative Positions for Example 2

Figure 4.5: Distance over Mars Radius vs Time for Example 2
Figure 4.6: Inertial Velocity vs Time for Example 2
4.2 Model Assessment

The assessment of a single case helps to better understand how the problem works. The selected case is described, giving explanations and justifications about the occurred events. With this, a deeper comprehension of the aerocapture problem can be achieved. The analysis of a single case also helps to verify the guidance scheme.

The selected and studied case has the following configuration: inertial velocity of 6 km/s, flight path angle of -15 degrees, ballistic coefficient of 1025 $kg/m^2$ and lift-to-drag ratio of 1.5. The performed simulation outputs are represented in figures 4.7 to 4.17, which are commented at below.

![Figure 4.7: Model Assessment - Altitude vs Time](image)

Figure 4.7, shows the variation of the altitude with the time. At time 0 the spacecraft enters the atmosphere with 6 km/s of inertial speed and, as the flight path angle is negative, descends down to Mars but quickly is controlled to a stable altitude. Then, once the energy management phase gives way to the targeting phase, the spacecraft tries to reach the target altitude of 500 km. The vehicle might not reach 500 km exactly, because of uncertainties, so there is an acceptable threshold of ± 50 km. At the end of the graphic the altitude decreases over again, this is because the spacecraft is already on an elliptical orbit. Therefore, the propulsive maneuver to raise the periapsis of the orbit must be applied here.
4. ANALYSIS OF RESULTS

A zoomed view of the altitude vs time chart is represented in figure 4.8. Here one can see a magnification of the constant altitude cruise phase. The spacecraft sinks down the atmosphere and tries to stabilize to a cruise altitude of 34 km. Obviously, the characteristic swing of the equivalent second order spring-mass-damper system depends on its constants, which were selected empirically from a Mars scenario [3].

![Figure 4.8: Model Assessment - Altitude vs Time (magnification)](image)

As spacecraft depletes sufficient energy, the guidance method changes the phase at about 305 seconds (red circle) to targeting phase. This phase controls the vehicle until it exits the atmosphere, 643 seconds later from entry.

Figure 4.9 shows the commanded bank angle versus time. It first starts at 90 degrees and it holds this value for a while, because this is the initial bank angle at entry interface and it lasts until the acceleration is high enough to allow guidance. Then it quickly changes to 15 degrees in order to provide enough lift to avoid crash. After approximately 50 seconds, guidance rapidly decreases the bank angle in order to avoid a premature exit of the atmosphere and tries to control the spacecraft into the 34 km constant altitude cruise. Afterwards, the phase change occurs and 5 seconds later guidance sets the bank angle to the minimum in order to elevate the spacecraft most closely to the desired altitude. The commanded bank angle rests at this value for the remainder, therefore the bank angle values are not plotted for the maximum observable time.
The visible abrupt changes in the commanded bank angle sign are produced by the Lateral Control algorithm, which tries to maintain some control on the out of plane velocity. As seen in figure 4.10, the lateral velocity continuously increases\(^1\) until it exceeds the limit lateral velocity, which is proportional to the actual inertial velocity. Then, Lateral Control changes the sign of the commanded bank angle producing a roll reversal and the vehicle starts to fly to the opposite direction, until the maximum is reached again. This process is repeated over and over again until the spacecraft is already in outer space, where the aerodynamic forces can no longer be generated and hence provide control of the spacecraft.

\(^1\)A positive value of lateral velocity means that the spacecraft is perpendicularly flying away from the right side of the desired orbital plane.
At the end of the trajectory, when the desired target is achieved, there is still a lateral velocity. This is caused by the change in the orbit’s inclination, which should be corrected in order to fulfill the mission’s objectives.

In figure 4.11, there is a chart of the inertial velocity with dependency on time. It shows how the spacecraft starts with 6 km/s and raises just for a few seconds. This is produced because the gravity acceleration is greater than drag at the upper layers of the atmosphere, which is extremely low. Then, the speed starts decaying rapidly as the vehicle falls down to Mars. The greatest decrease slope is produced in the next 80 seconds because of the combination of high velocity and increasing density. The second slope corresponds to the cruise phase and is less acute than the first, since the velocity is lower here and drag depends on the square of the velocity. The rest of the inertial velocity chart is a normal part of an elliptical orbit. As the spacecraft reaches his apoapsis, the maximum altitude, at about 2500 seconds, his velocity falls to his lowest value.

The energy state of an orbit defines its shape. It is widely known that the energy for elliptical orbits is negative while for hyperbolic orbits is positive. The energy of an orbit is calculated with the following equation:

$$E = \frac{V^2}{2} - \frac{\mu}{r}$$  \hfill (4.5)
Normally, an orbit whose part of trajectory does not lay within a planet’s atmosphere has a constant value of energy. As represented in figure 4.12, this is not the case of this simulation. One can see that the energy is first positive and then falls into negative values. This fits with the simulation as the vehicle starts in an hyperbolic orbit and ends in an elliptic orbit. Notice that the upper layers of the atmosphere provide almost no energy reduction, since the chart does not seem to decrease at the corresponding period of time.
Density vs time, dynamic pressure vs time, aerodynamic load vs time and heat flux vs time charts (4.13, 4.14, 4.15 and 4.16, respectively) have a similar outline. This is because the last three depend on the first in some manner. Figure 4.13 represents, with respect to the time, the real density that the spacecraft finds at the different flown altitudes. As previously mentioned, one can clearly see the density is almost zero at the upper layers of the Martian atmosphere, around second 0th to 40th and 480th to 640th.
The remaining figures (4.15, 4.14 and 4.16) have not exactly the same form than figure 4.13, because each one is also dependent on other parameters. It is important to mention that the maximum achieved g-load is lower than 5 g (see figure 4.14), which is the maximum for human missions. Depending on the mission, robotic missions can have a limit up to 10 g [1].

Figure 4.13: Model Assessment - Density vs Time
4. ANALYSIS OF RESULTS

Figure 4.14: Model Assessment - Aerodynamic Load vs Time

Figure 4.15: Model Assessment - Dynamic Pressure vs Time
Figure 4.16: Model Assessment - Heat Flux vs Time

The most important thing in the flux heat graphic (see figure 4.16) is the maximum that it presents. A very high peak of heat flux provides large amounts of heat load every second causing the spacecraft to warm up extremely fast. This time, the maximum lays under 150 $W/cm^2$, which is lower than the limit of 250 $W/cm^2$. It can be seen from figure 4.17, that the greatest heat load slope corresponds to the maximum peak of heat flux. Here is where the biggest quantity of energy is supplied to the spacecraft increasing its temperature. The amount of heat load augments until about 15.75 $kJ/cm^2$. This value is smaller than the selected limit of 35 $kJ/cm^2$.

Finally, figure 4.18 shows the path that the spacecraft reproduces during the aero-capture maneuver. That is, from the moment the vehicle enters the atmosphere until it approximately reaches its apoapsis.
4. ANALYSIS OF RESULTS

Figure 4.17: Model Assessment - Heat Load vs Time

Figure 4.18: Model Assessment - Top View of Inertial Position

71
4.3 Influence of $L/D$ and $\beta$ Variation

The results from varying lift-to-drag ratio and ballistic coefficient are gathered together in figures 4.19 to 4.25. The simulations were performed according to the basic case: 6 km/s of inertial velocity and -15 degrees of flight path angle.

One can see in figure 4.19 that the performed variations still allow the spacecraft to reach the target at the conditions of the studied case. However, there is a difference in the time of phase change and in the time the vehicle reaches the maximum altitude. The zoomed view of the same chart (see figure 4.20) helps to understand why this occurs.

![Figure 4.19: Multiple-Case - Altitude vs Time](image)

The fact of increasing or decreasing the lift-to-drag ratio affect the initial swing and the final rate of climb, the one the spacecraft has once exited from the atmosphere.
4. ANALYSIS OF RESULTS

The blue, red and green lines represent, respectively, the normal, the low $L/D$ and high $L/D$ cases. With more lift-to-drag ratio the spacecraft has a greater climb rate and then arrives earlier at its maximum height (green line) compared to lower lift-to-drag value, which makes the space vehicle spend more time to reach the same point (blue and red lines). Furthermore, a greater value of $L/D$ shows a bit higher apoapsis altitude. As for the previous swing, corresponding to the second-order spring-mass-damper system, since the spacecraft seeks to reach the altitude of 34 km, a vehicle with better lift capabilities will reach this altitude earlier and then must remain longer times in cruise phase. For example, red line has a lower lift performance, so it sinks deeper into the atmosphere and hence loses much energy. Therefore, when it reaches the 34 km, it has already depleted enough energy and starts with the targeting phase.

Figure 4.20: Multiple-Case - Altitude vs Time (magnification)
4.3. Influence of $L/D$ and $\beta$ Variation

Regarding to ballistic coefficient, an increase of $\beta$ makes the spacecraft less susceptible to drag. Less drag involves less lift performance and less depletion of energy. Thus, the case with high ballistic coefficient (black line) sinks deeper into the atmosphere and also needs more time to deplete enough energy. This also explains why the low $\beta$ case (magenta line) shows a higher rate climb, which is connected with better lift performance. Furthermore, an increment of ballistic coefficient can also be seen as an increment of mass, which would cause the spacecraft to have more inertia and be less affected by external forces.

The density profiles shown in figure 4.21, are only dependent on the altitude profile of each case. One can see, for example, that density maximums do correspond with the minimums in the altitude chart.

![Figure 4.21: Multiple-Case - Density vs Time](image-url)
As well as in the Model Assessment, all density-based figures have a similar outline. The dynamic pressures chart (see figure 4.22) also shows the same arrangement of the maximum values as the density chart. However, these maximums are not proportionally spaced, since they are also influenced by the inertial velocity.

![Dynamic Pressure vs Time](image)

**Figure 4.22: Multiple-Case - Dynamic Pressure vs Time**

As for the aerodynamic load, the results are shown in figure 4.23. It is function of the dynamic pressure but also lift and drag. The configurations that have higher lift-to-drag ratio and low ballistic-coefficient show a greater g load, since these are the ones that generate more aerodynamic forces.
4.3. Influence of $L/D$ and $\beta$ Variation

Figure 4.23: Multiple-Case - Aerodynamic Load vs Time

Figure 4.24 shows the heat flux chart, which is also a density-based graphic and so has a similar outline as density has. The relative positions of the maximums are not comparable with density or dynamic pressure ones, since the calculation of heat flux uses different powers of velocity and density (see equation 2.44). For example, the black line has a lower value of density and dynamic pressure than red line but a greater value of heating flux. In general, configurations that present better lift performance do not swing to low altitude levels and hence do not present high maximum values of dynamic pressure and heat flux. However, configurations that present a high speed decrease also present a decrease in heat flux, even if their maximums previously lay over other configurations. These configurations are low lift-to-drag ratio (red line) and low ballistic coefficient (magenta line).
4. ANALYSIS OF RESULTS

One can obtain the total amount of heat load of a case by taking the area below heat flux vs time outline, so at a glance it is possible to guess which of the studied configurations will present the highest or lowest heat load value. Figure 4.25 contains the heat load values of the different configurations over time. According to heat flux chart, the major value of heat load is for the high ballistic coefficient (black line) and inversely, the minor value is for the low ballistic coefficient. The observable fact that high and low $L/D$ heat load values cross in a specific moment (lines green and red) is caused by the following: the low lift-to-drag ratio case quickly sinks into the atmosphere, increasing with this its density, dynamic pressure and heat flux but also decreasing the velocity. Much energy has been depleted and then the spacecraft exits before, so it remains less time within the atmosphere. The inverse occurs for the high lift-to-drag ratio case, which ends with a higher heat load transferred to the spacecraft.
4.3. Influence of $L/D$ and $\beta$ Variation

The heat load value is a direct consequence of the time spent in depleting energy in the constant cruise phase, so it might be desirable to quickly lose energy in order to avoid the spacecraft to burnout.

\[ \text{Heat Load value} \]

Figure 4.25: Multiple-Case - Heat Load vs Time
4. ANALYSIS OF RESULTS

4.4 Corridor Determination

As mentioned previously, the study was performed with a variation of inertial velocities from 5 to 7 km/s and a variation of flight path angles from -30 to -5 degrees for each of the lift-to-drag ratio and ballistic coefficient conditions. The success of the mission depends on the final apoapsis altitude the spacecraft reaches, with a threshold of ±50 km, but also on other conditions. There are four constraints, whose conditions should not be met in order to consider the mission success, even if the spacecraft would have reached an apoapsis of 500 km. These constraints are: crash, failure of the structure, maximum heat flux and maximum heat load or burnout.

The conditions for the applicability of these constraints are the following:

1. Crash: if the spacecraft trajectory has a minimum altitude\(^2\) of 0 km.
2. Structure failure: if the spacecraft undergoes an aerodynamic load of 10 g or higher.
3. Maximum heat flux: if the spacecraft undergoes a heat flux of 250 \(W/cm^2\) or higher.
4. Burnout: if the spacecraft undergoes a heat load of 35 \(kJ/cm^2\) or higher.

Furthermore, these constraints will be applied in order. This means, for example, if the vehicle crashes and also its structure fails, the result of the mission is considered as a crash. On the other hand, if the vehicle burns out, it means that all previous constraints did not occur.

As an illustrative example for the different possibilities of the result of the mission, a simple case has been studied. This has an inertial velocity of 7 km/s and the flight path angle ranges from -30 degrees to -5 degrees but with an increment of one degree, in order to eliminate a half of the results and hence make it more clear. This case also has the normal values for lift-to-drag ratio and ballistic coefficients.

Figure 4.26 shows the resulting possibilities of the example. One can see that within a specific velocity, the change of flight path angle produces many different results. If the flight path angle is very small (high modulus and negative in sign), the descend rate increases and the vehicle crashes into the ground (black line). As flight path angle increases, the vehicle does not crash but it still undergoes high aerodynamic loads, so the structure fails (see zoomed view of the same chart in figure 4.27). Once flight

\(^2\)This can be unrealistic due to Mars’ topography features, which can arise up to 21 km like Olympus Mons
path angle is high enough, structure failure result does no longer occur is no longer fulfilled but there is still a risk of exceeding the maximum heat flux before success can be considered.

Notice that yellow and magenta lines represent a mission failure. If the levels of aerodynamic load and heat flux are not high enough to consider their respective constraints, some of these lines would reach the target altitude and other would not. This difference lays on the cruise stage. Once the spacecraft has entered this phase, it flies in a stable regime, so it has much more control at the exit of the same. Otherwise, if the spacecraft loses much energy before this stage, it must exit the atmosphere prematurely in much poor stability and control conditions.

Figure 4.26: Altitude vs Time for 7 km/s of Inertial Velocity
4. ANALYSIS OF RESULTS

As expected, burnout case corresponds to the case in which the spacecraft remains more time in the atmosphere. However, it can also happen that the latest case to exit the atmosphere does not exceed the maximum heat load limit.

![Altitude vs Time for 7 km/s of Inertial Velocity](image)

Figure 4.27: Altitude vs Time for 7 km/s of Inertial Velocity (magnification)

All possible results for each configuration are gathered together from figure 4.28 to 4.32. These figures are colour-mapping charts of the different result possibilities. Each coloured square represents one $V_{inertial-\gamma}$ simulation. Therefore, the whole range of velocities and flight path angles is filled with squares. The colour areas of the figures reveal information about how the destiny of the mission changes based on velocity, flight path angle, lift-to-drag ratio and ballistic coefficient parameters.

Figure 4.28 corresponds to the basic case ($L/D = 1.5$ and $\beta = 1025 \text{ kg/m}^2$). As expected, the lower flight path angles are more likely to provoke a crash (grey squares) rather than the higher. For a single flight path angle, the increase of the inertial velocity...
helps the vehicle to avoid crash, since speed increments the aerodynamic forces but also increases the structure failure results.

As for thermal constraints, only at higher velocities do these phenomena happen. In this case, at 6.75 and 7 km/s happens that the spacecraft undergoes a higher heat flux than the limit and only at 7 km/s the spacecraft burns out.

Green squares represent that the space vehicle has not exceeded any constraint but either reached the target. This can occur in two different situations as previously mentioned. On the one hand, it can happen that the spacecraft simply does not reach the target because it exits the atmosphere from an unstable situation. This is very common at low velocities, since the spacecraft must deplete lower energies so it remains less time within the atmosphere. On the other hand, it can also happen the vehicle to have so high velocity that is no more able to penetrate the atmosphere and then performs
an hyperbolic exit. Hence, at higher velocities and high flight path angles the mission results in a target miss.

The mapping result for the configuration of same ballistic coefficient and lower lift-to-drag ratio is shown in figure 4.29. At a glance, one can see that there is a decrease in terms of performance. The crash border has moved to the right providing worse results than for normal lift-to-drag ratios. There are also more target misses at low-mid values for inertial velocity and mid flight path angles. With respect to thermal limits, there is an increment of maximum heat flux results. Since the spacecraft has a lower lift performance, it must sink at lower altitudes, which cause higher heat fluxes. However, this makes the vehicle to quickly lose energy and then reduce its time into the atmosphere. Therefore, burnout results have been reduced.

Figure 4.29: Mapping of Results for \( L/D = 1 \) and \( \beta = 1025 \, kg/m^2 \)
4.4. Corridor Determination

As shown in figure 4.30, an improved lift performance reverses the latest events: crash corridor is brought to the left, maximum heat flux results have been reduced, burnout results have been increased again. The left blue border has moved to the left, providing more mission-success results.

Figure 4.30: Mapping of Results for \( L/D = 2 \) and \( \beta = 1025 \ kg/m^2 \)

Now, lift-to-drag ratio is 1.5 again and ballistic coefficient has been decreased to 683.33 \( kg/m^2 \) (see figure 4.31). As commented in section 4.3, a decrease in ballistic coefficient produces a higher altitude minimum and a rapid exit of the atmosphere. Therefore, crash boundary has also moved to the left and there are no results for thermal constraints. Furthermore, since the spacecraft exits the atmosphere before the cruise phase, there are more target miss results over more velocities.
Lastly, figure 4.32 collects the results for the high ballistic coefficient configuration ($\beta = 1366.67 \text{ kg/m}^2$). Over again, at a glance one can see that there is a reduction of performances. This configuration has a lower altitude minimum than the normal case and also stays longer times in the atmosphere. Therefore, it has a poor crash and worse thermal behaviours and there are less target miss results.
4.4. Corridor Determination

Figure 4.32: Mapping of Results for $L/D = 1.5$ and $\beta = 1366.67 \text{ kg/m}^2$

The success results from each chart are gathered in figures 4.33 and 4.34. The blue area is the one considered as the capturable corridor. Note that the target miss results within blue area are not considered part of this corridor, so it only includes groups of consecutive target match results.

Figure 4.33 only considers different lift cases in order to analyse the impact of lift variation in the capturable corridor.
The left-sided limit of the corridors varies according to the lift capabilities of the spacecraft. A high lift configuration widens the corridor and provides more stability at the exit of the atmosphere in more velocities. In figure 4.33 one can see that at low-mid velocities the normal case (blue line) has more zigzags than the high lift case (green line). Low lift configuration narrows the corridor and also has less zigzags, since there are more consecutive target miss results (see figure 4.29). Furthermore, regarding to the right-sided limit of the corridor, there is not much difference between the varied configurations. This means that there are hardly differences related to burnout constraint. In conclusion, a higher lift-to-drag ratio is more desirable rather than a lower ratio because a higher ratio widens the corridor, stabilizes the results thanks to a longer cruise stage at a cost of very little increase of heat load. However, for reasons of cost, it may be desirable to perform aerocapture with a relatively simple vehicle, which has a low lift-to-drag ratio [2].
Figure 4.34 shows the capturable corridors for different ballistic coefficient values. A lower ballistic coefficient provides better thermal performance but a very poor stability at low-mid velocities and the opposite happens for a higher value, which undergoes high thermal fluxes and loads but has a better behaviour in mid velocities. It also shows little improvement on stability performance at low velocities. In conclusion, a high ballistic coefficient might be desirable but only for low-mid velocities. However, it would be sensible to make this decision based on a detailed analysis of the desired mission.

![Figure 4.34: Capturable Corridor for Different Ballistic Coefficient Values](image-url)
4.5 Lateral Velocity

The Model Assessment already showed that there is an existent amount of lateral velocity at the end of the trajectory. A correction maneuver must performed to correct the inclination error that lateral velocity caused. In order to do so, a simple plane change maneuver must be performed. A delta-V applied outside the orbital plane and towards the right direction will produce the orbit to change its inclination but will not change the inertial velocity magnitude (the sketch in figure 4.35 shows the initial and final velocities are equal).

\[ \Delta V = 2V_i \sin \frac{\epsilon}{2} \]  

(4.6)

where \( \epsilon \) is the wedge angle, which must be first found and coincides with the inclination of the orbit because the target orbit is equatorial. This angle can be found by taking the inverse cosine of the dot product of specific angular momentum unit vector and the unit vector normal to desired plane orbit, as equation 4.7 shows:

\[ \epsilon = \arccos \hat{i}_h \cdot \hat{n}_{dp} \]  

(4.7)

where the specific angular momentum unit vector is calculated as:

\[ \hat{i}_h = \frac{\vec{r} \times \vec{V}_{inertial}}{|\vec{r} \times \vec{V}_{inertial}|} \]  

(4.8)

The actual lateral control system, presented previously on section 3.3, bases its calculations on the maximum allowed lateral speed. This is calculated with a percent margin of the inertial velocity. In Model Assessment, the value for the margin was 0.1, that means a 10 percent. Below, two more cases will be studied with values 0.4 and 0.025 for the margin (this is 4 times greater and smaller, respectively).
Figure 4.36 collects the results of both cases and the original. It can be seen that, on the one hand, for the 0.4 margin case, lateral velocity is hardly controlled. Higher margins allow greater lateral velocities and that involves lower frequency response of the lateral control, because lateral velocity takes some time to increase up to these levels. On the other hand, a four-times smaller margin (0.025) is more precise as the opposite occurs. The maximum allowed lateral velocity is much lower, so roll reversals are executed at a higher frequency and then the spacecraft is being controlled more time.

![Lateral Velocity for Different Margin Values](image)

One must also consider the need for a higher roll reversal frequency involves a complexer spacecraft. This can cause the cost of the vehicle to increase and even exceed the mission’s cost limit. Fortunately, the difference between the two small margins is small, compared to the normal and the higher cases.

Figure 4.37 shows the final inclinations of the different cases. Lower margins involve lower inclination values compared to higher.

Each one of these orbits must apply a different delta-V burn in order to change its plane. This burn cannot be performed before the periapsis raise maneuver. In order
4. ANALYSIS OF RESULTS

Figure 4.37: Front View of Inertial Position for Different Margin Values

Table 4.1 collects the resulting inclinations and the needed delta-V according to equations 4.6, 4.7 and 4.8. It was necessary to calculate the velocity for the final orbit, assuming that circularization maneuver was already done. This velocity was computed with the following equation:

$$V_{\text{circular}} = \sqrt{\frac{\mu d}{r}}$$  \hspace{1cm} (4.9)

As wedge angle or inclination raises up, the plane change maneuver becomes more $\Delta V$-expensive, so it is very important to design a good enough lateral control algorithm, which enables the spacecraft to reduce the lateral velocity errors. One derivable con-
4.5. Lateral Velocity

Table 4.1: Plane Change Maneuver Properties

<table>
<thead>
<tr>
<th>Margin</th>
<th>$\epsilon$</th>
<th>$\Delta V$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.025</td>
<td>0.47 deg</td>
<td>0.0272 km/s</td>
</tr>
<tr>
<td>0.1</td>
<td>1.73 deg</td>
<td>0.1001 km/s</td>
</tr>
<tr>
<td>0.4</td>
<td>13.69 deg</td>
<td>0.7903 km/s</td>
</tr>
</tbody>
</table>

Conclusion is that a better lateral guidance system increases the mission performance and with it, the costs of the mission can be reduced.
4.6 Comparison with propulsive Maneuvers

The aim of aerocapture was to reduce the required propulsive maneuver to the minimum by means of usage of a planet’s atmosphere to reduce the velocity. This enables orbit insertion missions with lower fuel consumption. In order to demonstrate so, a comparison with a propulsive maneuver was performed based on the same mission of this project.

In figure 4.38, there is a scheme of an interplanetary transfer from Earth to Mars using the Hohmann transfer. This is a well-known low-cost method that uses an elliptical orbit to transfer between two coplanar circular orbits. The values for the hyperbolic excess velocity \( V_8 \) for Earth and Mars are 2.95 km/s and 2.65 km/s, respectively [18]. This is the velocity, relative to the planet, that a space vehicle has when it is at the border of the sphere of influence, the zone where the gravitational attraction of the planet is the greatest.

It is assumed that spacecraft was launched from a low parking orbit at Earth (see figure 4.39). The delta-V needed to launch it depends on the hyperbolic excess velocity and the velocity of the circular orbit. Therefore, as \( V_8 \) is already known, \( \Delta V_E \) only depends on the radius of the circular orbit. It is also assumed that the low parking orbit altitude is exactly the same for the propulsive case and for aerocapture’s and is 200 km.

In order to calculate this delta-V, the circular orbit velocity and the hyperbolic velocity at pericenter radius \( r_{pc} \), which is the closes point of an hyperbolic orbit to its focus, must be first calculated. The first one directly comes from equation 4.9 and has a value of 7.7843 km/s. The second is calculated by assuming equal hyperbolic energy states.
4.6. Comparison with propulsive Maneuvers

Figure 4.39: Departure from Earth at pericenter and at the limit of the sphere of influence (at infinity). If one uses the energy equation (eq. 4.5) and compares both points, one obtains:

\[
\frac{V_{pc}^2}{2} - \frac{\mu \Phi}{r_{pc}} = \frac{V_{\infty}^E}{2} - \frac{\mu \Phi}{r_{\infty}}
\]

(4.10)

and operating:

\[
V_{pc} = \sqrt{V_{\infty}^E + 2 \frac{\mu \Phi}{r_{pc}}}
\]

(4.11)

so finally the hyperbolic velocity at pericenter is 11.3970 km/s. The delta-V is found according to next equation and is \(\Delta V^E\) is 3.6127 km/s.

\[
\Delta V^E = |V_{\infty}^E - V_{pc}|
\]

(4.12)

As for arrival at Mars, the opposite occurs (see figure 4.40). The spacecraft approaches the planet with an hyperbolic trajectory. The maneuver for speed reduction is normally performed at pericenter radius, which coincides with the target circular orbit radius. The circular velocity for the target orbit, which has an altitude of 500 km, has already been calculated and has a value of 3.3155 km/s. Next, the hyperbolic pericenter velocity is calculated with equation 4.11 but with values for Mars:

\[
V_{pc} = \sqrt{V_{\infty}^M + 2 \frac{\mu \Phi}{r_{pc}}}
\]

(4.13)

This results in a velocity of 5.3858 km/s. Then, the delta-V for orbit insertion at Mars \((\Delta V^M)\) is 2.0703 km/s.

In order to calculate the total delta-V expense for the full propulsive Earth-Mars transfer, the separate delta-V must be summed but firstly two assumptions must been taken into account. The first is to assume that the spacecraft approaches Mars though the equatorial plane. This means that there is no need to change the plane of the orbit.
4. ANALYSIS OF RESULTS

The second is to assume that the required delta-V in the in-space cruise phase, which is used to correct the trajectory deviations, is negligible. Then, the resulting total amount for full-propulsive methods is the sum of both delta-V at launch and arrival (see equation 4.14).

\[ \Delta V_{\text{propulsive}} = \Delta V^E + \Delta V^M \]  (4.14)

which is 5.6830 km/s.

Since aerocapture guidance algorithm allows the spacecraft to reach the target within a certain range of inertial velocities and flight path angles, one can assume that the studied scenario for aerocapture is comparable to the above described for propulsive maneuvers. Bearing this in mind, the aerocapture case is further analysed at below.

The threshold of ±50 km for the target altitude enables the success of the mission within an altitude range from 550 km to 450 km. Therefore, in order to distinguish the borders of the range of possible solutions there are three cases that need to be separately studied:

- **A**: apoapsis altitude of 450 km.
- **B**: apoapsis altitude of 500 km.
- **C**: apoapsis altitude of 550 km

Aerocapture technique uses two burns, which must be calculated for each of the three commented cases.

The first delta-V is used in order to raise the actual periapsis to the periapsis target. The main problem is that, since the spacecraft loses energy, the orbits change in shape.
4.6. Comparison with propulsive Maneuvers

and when it exits the atmosphere, it has already gone beyond the periapsis. However, this can be still calculated using the state values at apoapsis, since energy is constant. The periapsis is calculated using the information from Model Assessment and assuming this equal for all three cases.

The values of apoapsis altitude and apoapsis velocity are 514.48 km and 3.1485 km/s, respectively. This values are introduced in equation 4.15, which is an arrangement of Vis-Viva equation (eq. 4.3). With equations 4.1, 4.15 and 4.16, the semi-major axis, the actual periapsis distance and periapsis altitude are calculated and their values are 3571.94 km, 3233.20 km and -162.99 km, respectively. Although this last number is negative, it only means that the spacecraft would crash if periapsis raise maneuver would not be done.

\[ a = \frac{1}{\frac{2}{r_a} - \frac{V_a^2}{\mu}} \]  
(4.15)

\[ r_p = 2a - r_a \]  
(4.16)

Now, the actual velocities must be found. To do so, the actual semi-major axis must first be calculated using the common periapsis distance and each apoapsis. Then, applying Vis-Viva equation one obtain the following results: \( V_{a1A} = 3.1892 \) km/s, \( V_{a1B} = 3.1575 \) km/s and \( V_{a1C} = 3.1265 \) km/s. The same process is applied again but this time, the periapsis value is set equal to the target value for periapsis, since it is desired to end at this position. The resulting inertial velocities are: \( V_{a2A} = 3.3477 \) km/s, \( V_{a2B} = 3.3155 \) km/s and \( V_{a2C} = 3.2839 \) km/s and the corresponding delta-V are: \( \Delta V_{1A} = 0.1585 \) km/s \( \Delta V_{1B} = 0.1579 \) km/s and \( \Delta V_{1C} = 0.1573 \) km/s.

Once periapsis raise maneuver has been performed, a second burn is needed to circularize the orbit to the target apoapsis. This burn is performed when the spacecraft reaches the target periapsis. In order to calculate this second delta-V, the same process as above must be followed. First of all, to calculate the actual periapsis velocities considering the transfer orbit. These velocities are: \( V_{a1A} = 3.3047 \) km/s, \( V_{a1B} = 3.3155 \) km/s and \( V_{a1C} = 3.3260 \) km/s. Now, the final velocity that the three cases will possess are the same: 3.3155 km/s, which corresponds to the target circular orbit velocity. Lastly, the seconds delta-V values are: \( \Delta V_{2A} = 0.0107 \) km/s \( \Delta V_{2B} = 0 \) km/s and \( \Delta V_{2C} = 0.0106 \) km/s. Note that case B only needs to perform the periapsis raise maneuver, since it already starts in the target apoapsis.

Finally, the total amount of delta-V of the different cases are: \( \Delta V_A = 0.1692 \) km/s, \( \Delta V_B = 0.1579 \) km/s and \( \Delta V_C = 0.1679 \) km/s. These values reveal the most economic and expensive possible cases. Obviously, the most economic case corresponds to the perfect altitude match. The worst possible case of mission success is an apoapsis
altitude of 450 km/s. This fact is like that because of the high cost of periapsis raise maneuver compared to circularization maneuver and a lower altitude always requires more delta-V to rise periapsis rather than a higher altitude.

Once the propulsive delta-V range for aerocapture has been found (0.1579 - 0.1692 km/s), the delta-V provided by the atmosphere drag can be also found if the full-propulsive transfer and aerocapture technique are compared. See following equation:

\[ \Delta V^E + \Delta V^M = \Delta V^E + \Delta V_{\text{atmospheric}} + \Delta V_1 + \Delta V_2 + \Delta V_{\text{lateral}} \quad (4.17) \]

This equation can be rearranged as:

\[ \Delta V_{\text{atmospheric}} = \Delta V^M - (\Delta V_1 + \Delta V_2) - \Delta V_{\text{lateral}} \quad (4.18) \]

which, taking \( \Delta V_{\text{lateral}} \) equal to 0.1001 km/s from Lateral Velocity section, results in:

\[ \Delta V_{\text{atmospheric}} = 1.8010 - 1.8123 \text{ km/s.} \]

This amount of delta-V means a reduction of propulsive delta-V expenditures for orbit insertion ranging from 86.99% to 87.54%. The consequences of this fact and further saving opportunities are commented in next section.

### 4.7 Saving Opportunities

In previous section it was found that aerocapture technique provides a reduction of the necessary delta-V of 86.99 to 87.54% for orbit insertion. This involves that this burn must not be performed by the spacecraft and so it must not carry with the related fuel. To bring a higher mass for the same mission can be also desirable. Furthermore, an other option is to keep the fuel, travel to destinations beyond Mars and use aerocapture technique to perform an orbit insertion into other celestial bodies with atmosphere.

The concrete mass savings can be calculated using Rocket Equation (eq. 1.1) but a specific impulse must be firstly assumed. The selected specific impulse has a value of 321 seconds, which corresponds to the single 424 N engine assigned to Exomars orbit insertion mission [19].

The mass fraction corresponding to full-propulsive orbit insertion is calculated with equation 4.19 and only using the delta-V expenditure for Mars arrival. It has a value of 1.9229.

\[ \left( \frac{m_s}{m_f} \right)_{\text{propulsive}} = \exp \left( \frac{\Delta V^M}{I_{\text{sp}0_{\text{SL}}}} \right) \quad (4.19) \]
Then, if same calculation is performed but subtracting the non-propulsive delta-V performed by the atmosphere, one obtains the mass fraction of orbit insertion for aerocapture (see equation 4.20), which ranges from 1.0854 to 1.0893.

\[
\left( \frac{m_i}{m_f} \right)_{\text{aerocapture}} = \exp \left( \frac{(\Delta V^M - \Delta V_{\text{atmospheric}})}{I_{sp}g_0SL} \right) \tag{4.20}
\]

Considering equal initial masses for both cases, aerocapture increases the final mass about 76.53% to 77.16%. This means that the scientific payload increases, which is very favourable. Otherwise, a reduction of the initial mass can be chosen, which would decrease a 43.35% to 43.55%. A reduction of propellant, reduces the mass of the spacecraft and also the mission costs.

Aerocapture is a relative quickly technique for orbit insertion compared to other like aerobraking, which is extremely slow, since the spacecraft must travel around some elliptic orbits in order to slow a certain amount of energy at each cycle [1]. Taking the time at which the spacecraft reaches the apoapsis and using equation 4.4 for the Model Assessment values, one can calculate the total passed time since the spacecraft enters the atmosphere and it reaches its final orbit. This time span is about \(5.8731 \times 10^3\) seconds or 1.6214 hours, an appropriate value for an orbit insertion maneuver.
Chapter 5

CONCLUSIONS

5.1 General Conclusions

The objective of this project was to assess the feasibility of using aerocapture to perform an orbit insertion of a Mars mission. In order to do so, AECASIM simulating tool was designed and implemented with models, which help to better describe the problem scenario. These models are an atmospheric model, a gravitational model and a vehicle model. AECASIM is constituted by a set of different modules, which are the Dynamics module, PredGuid module, gravitational module or $J_2$ perturbation calculator and the aerodynamic module. PredGuid was implemented from varying it from different works in order to adapt it for this project.

The main part of the assessment of aerocapture’s feasibility had the objective to vary the parameters inertial velocity, flight path angle, lift-to-drag ratio and ballistic coefficient, and to analyse their influence on the performances. The regarding mission was a 500 km altitude equatorial circular orbit. Inertial velocity values where varied from 5 to 7 km/s with an increment of 0.25 km/s and flight path angles ranged from -30 deg to -5 deg with an increment of 0.5 deg. The values of lift-to-drag ratio and ballistic coefficient were separately increased and decreased from a reference case, which had values of 1.5 and 1025 $kg/m^2$, respectively. For lift-to-drag ratio, the other values were 1 and 2 and for ballistic coefficient, they were 683.33 $kg/m^2$ and 1366.67 $kg/m^2$.

The study revealed that there is a specific range of flight path angle for each velocity, in which the success of the mission is possible. If flight path angle is too small, the space vehicle will crash, and if it is too big, the centrifugal acceleration will make it exit the atmosphere prematurely without having lost enough energy. The inertial velocity also affects this results. A greater velocity provides the spacecraft with more aerodynamic control to avoid collision but also raises the dynamic pressure, the aerodynamic load
and favour heat transfer phenomena.

The variations in lift capabilities can enhance or deteriorate the performances of the spacecraft. On the one hand, a lower lift-to-drag ratio makes the spacecraft less controllable and it can not avoid sinking at deeper altitudes, which increases heat flux, drag and dynamic pressure. Since the vehicle loses energy at faster rates, it exits the atmosphere quickly but in a less stable regime, which provides more probabilities of target miss. However, as it remains less time within the atmosphere, the total amount of heat load is reduced, decreasing the probabilities of burnout results. On the other hand, if lift capability is improved, the vehicle has a higher control of itself, so collision results are less probable and the vehicle does not sink to deep altitudes. Dynamic pressure is not too high and the probability of structure failure is reduced but there is still some risk of burnout. Generally, a higher lift-to-drag ratio is more desirable since it widens the corridor, favours the constant altitude cruise stage, and provides less target miss results at cost of little increase of heat loads. However, cost constraint can limit the selection of high lift configurations.

As for ballistic coefficient, a lower value provides much better thermal performance but a very poor stability at mid-low velocities. This is caused due to lower values of ballistic coefficient mean higher drag coefficients. Drag enhances lift capabilities, as lift performance is represented by a ratio. Then, the spacecraft is more controllable and does not sink to low altitudes. However, as it has more drag coefficient, the spacecraft exits the atmosphere prematurely, causing multiple target misses. At higher ballistic coefficients, thermal fluxes and aerodynamic loads are higher but since drag coefficient is slower, there are more cruise stages which favour better targeting competences. Finally, a higher ballistic coefficient for low-mid velocities might be preferred.

Depending on each mission, a complexer analysis should be performed in order to find the best compromise between lift-to-drag ratio and ballistic coefficient values.

Lateral control of the spacecraft was also studied. It was found that with a worse lateral control system, the inclination of the final orbit diverges with respect to the equatorial plane and then an extra delta-V is needed in order to perform a greater plane change maneuver, which is \(\Delta V\)-expensive. For the different studied margins, the delta-V expenditure can vary from 1.31% to 38.17% respect to full-propulsive orbit insertion. Therefore, the better the lateral control system is, the more economical the mission will be.

A comparison with propulsive methods was performed. It revealed that if aerocapture technique is applied for the selected scenario, delta-V savings of 86.99% to 87.54% are possible. This savings enables the spacecraft’s final mass to be increased between
5. CONCLUSIONS

76.53% to 77.16% and so to increment the mass of scientific payloads or to reduce the initial mass from 43.35% to 43.55%, which reduces the costs of similar missions.

Finally, the total time of the simulated aerocapture process, with values from Model Assessment, has been calculated. The whole process lasts approximately 1.6214 hours, which is more satisfactory for an orbit insert mission compared to other techniques such as aerobraking.
5.2 Future Work

This project consists of a preliminary study of the feasibility of the aerocapture technique. Below, there are some items about further work that could be done in order to improve the performed study or continue it.

- Improve the robustness of the software: AECASIM and PredGuid algorithm should be further verified with more reference documentation, so that the whole software could grant more robustness.

- Enhance the existent environment models: this thesis uses simple gravitational and atmospheric models. The accuracy of the results can be increased by using complexer environment models such as a gravitational model that better fits the real gravitational field of the planet, or an atmospheric model that calculates the density based on thermal profiles and considers day and night possibilities.

- Implement a heating model: a complex and robust heat transfer model could also be implemented in order to study the aerothermodynamic phenomena with more detail.

- Explore other feasibility variables: this thesis analyses the feasibility of aerocapture by studying the effects of the variation of four parameters and also comparing the whole maneuver with full-propulsive methods. There are more feasibility parameters that could be used in order to complete the study. The most important is the cost. Further analyses should be performed based on specific cost expenditures or cost-savings. Other feasibility parameters for studies could be: materials, state-of-the-art technology or spacecraft design.
Bibliography


