This document is classified as ISAE SUPAERO Confidential

Master of Aerospace Engineering Research Project

Optimal control of Trajectory of reusable launcher in OpenMDAO/dymos

S2 Project Report

Authors:
Alberto Fossà
Giuliana Elena Miceli

Tutors:
Laurent Beauregard
Joseph Morlier
Stéphanie Lizy-Destrez

Due date of report: 18 June 2019
Actual submission date: 18 June 2019

Begin of the project: 29 January 2019

Duration: 14 Months

Contents

Declaration of Authenticity

This assignment is entirely our own work. Quotations from literature are properly indicated with appropriated references in the text. All literature used in this piece of work is indicated in the bibliography placed at the end. We confirm that no sources have been used other than those stated.

We understand that plagiarism (copy without mentioning the reference) is a serious examinations offence that may result in disciplinary action being taken.

Date: 18 June 2019 Signatures:

Abstract

The renewed interest in lunar missions and the push on the human exploration of the Solar System has brought the space engineering sector to find feasible and efficient solutions to bring again the mankind in the outer space. The future habitable space station that will orbit around the Moon, the Lunar Orbit Platform Gateway (LOP-G), poses the same challenges to perform complex operations such as transfer of goods and crews to and from the Moon surface.

In this context, the aim of this study is to analyze and propose feasible solutions for the lunar surface to Low Lunar Orbit transfer trajectories, first stage to reach the LOP-G.

The sought-after solution will minimize the propellant consumption while satisfying the physical and geographical constraints imposed on the model. In order to achieve the aforementioned objectives OpenMDAO and dymos, two Python-based libraries, are the main tools employed to carry out the numerical simulations while orbital mechanics and optimal control theory are at the bottom of the developed models.

Keywords: Trajectory Optimization, Optimal Control Theory, Lunar missions, MDO, Python

1 Introduction

New challenges are arising from the space exploration development of our ages. Indeed, now more than ever, both space agencies and private companies are interested in further pushing the limits of the human space exploration beyond the Low Earth Orbit and towards new horizons such as the Moon and Mars. New requirements are then arisen bringing space technology and research to focus their attention on new, low-cost optimal solutions to satisfy the aforementioned demand. In the section hereafter 2.1, the dares of human Moon exploration are pointed out and a context is given for the main topic of this study: the optimal transfer trajectory between the Moon surface and a Low Lunar Orbit.

Recently developed methods such as Multidisciplinary Optimization allow to perform deeper analysis of those complex problems that groups together multiple engineering fields. Once the problem of interest is defined a numerical solution that approximates the true optimum is usually searched through different powerful iterative algorithms. Following the aforementioned approach, powerful numerical methods are developed throughout this study to iteratively solve a Nonlinear Programming (NLP) problem that results from an appropriate transcription of the corresponding continuous-time optimal control problem as described in chapter 3.

Similar solutions to those problems have already been investigated by some authors as reported in section 2.2 while their results are taken as initial reference for the investigations carried out in chapter 4, thus validating the adopted strategy. Finally, in sections 2.2.4 to 2.2.6 a brief introduction to the optimal control theory is also provided in order to justify the theoretical models behind the operating tools implemented to accomplish the subsequent work.

2 Semester 2 Section

2.1 Context and key issues

The development of the Apollo program enabled the humanity to land the first astronauts on the surface of the Moon in 1969. After the last Apollo mission flown in 1972, the human space exploration has been confined in Low Earth Orbit (LEO) with different programs such as the Mir space station, the Space Transportation System (STS), the International Space Station (ISS) and the Tiangong space laboratory.

However, a renewed interest in placing a crewed vehicle on the lunar surface is continuously growing since the last few years as first important step for the upcoming human deep space exploration towards Mars and beyond. Differently from the Apollo program, this time a permanent colony will be established on the Moon for both scientific purposes and *in-situ* resources exploitation such as H_2 and O_2 to be used as rocket propellants. Then, to facilitate the transfer of crews and goods form the LEO to the lunar surface a new outpost, the Lunar Orbital Platform-Gateway (LOP-G) will be placed between the Earth and the Moon in a so called Near Rectilinear Halo Orbit (NRHO).

Since a direct transfer from the Moon surface to the LOP-G orbit and vice-versa is not feasible, the ascent and descent trajectories have to be split in two phases considering an intermediate Low Lunar Parking Orbit (LLO) where the spacecraft will be temporarily placed.

To maximize the payload mass that can be transferred between the Moon and the LOP-G the propellant mass carried by the transfer vehicle should be carefully handled, thus requiring the transfer trajectory between the Moon surface and the intermediate parking orbit to be properly designed to minimize the fuel consumption.

2.2 State of the Art

Following the new trends in the Moon exploration different studies were recently carried out to exploit possible solutions for such a complex mission. Starting from the knowledge acquired during the Apollo program, several authors propose new optimal solutions for two and three-dimensional transfer trajectories to and from the lunar surface.

2.2.1 2D Descent trajectories

Regarding the descent phase, Wilhite [19] analyzes the flight data of the Apollo missions performed until 1972 and proposes an optimized solution that takes into account the safety requirements for a crewed vehicle such as ground visibility and maximum accelerations. A summary of his results is presented in figure 2.1.

More recently, Ramanan [15] and Remesh [16] study different solutions for an automatic probe soft lunar landing with different initial conditions and constraints. In both cases an initial circular

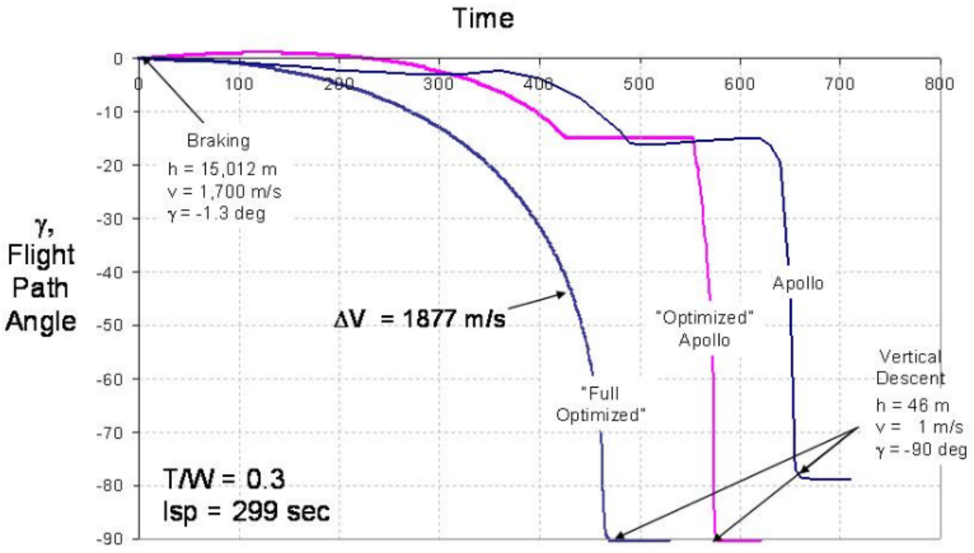

Figure 2.1: Full optimized Apollo descent trajectory
credits: A. Wilhite [19]

parking orbit at 100 km altitude is considered while the descent trajectory is split in multiple steps. At first, an impulsive deorbit burn is performed to inject the spacecraft into an Hohmann transfer trajectory with a predetermined periapsis altitude. At this point a powered descent is performed applying a continuous thrust to reduce the energy of the probe and fulfill the final constraints at touchdown. The optimal solution is a trajectory that reaches the ground with an angle of about 40° respect to the local vertical, thus posing several safety concerns. This phenomenon is avoided adding a constrained vertical phase at the end of the powered descent to force a vertical landing of the spacecraft as shown in figure 2.2.

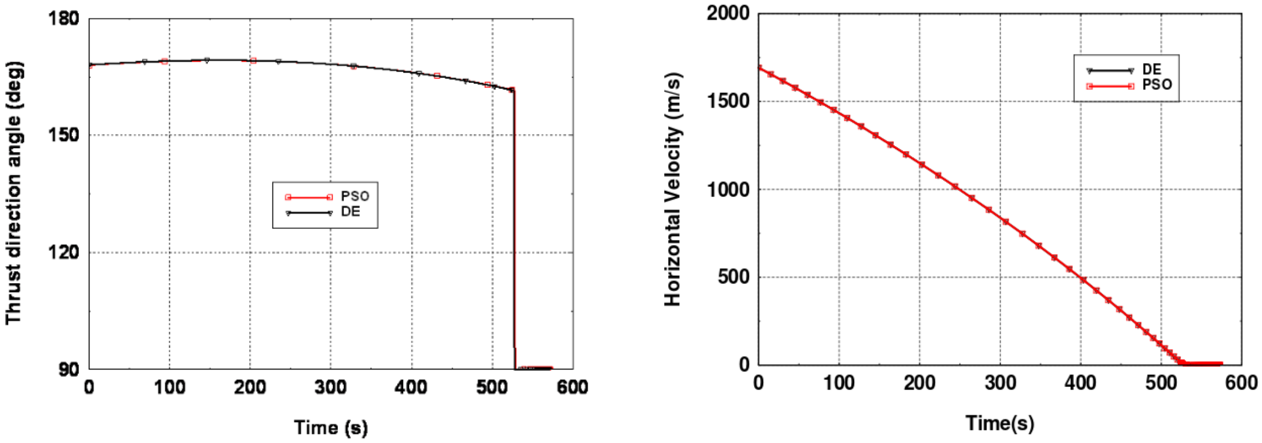

Figure 2.2: Thrust direction and horizontal velocity with vertical constraint
credits: N. Remesh [16]

2.2.2 2D Ascent trajectories

Considering the ascent phase from the Moon surface to a LLO, Zhang [21] allows the thrust magnitude to vary throughout the trajectory from zero to a specified maximum value. Under this assumption different solutions are computed for target orbits at different altitudes proving that the fuel consumption is minimum when an *on/off* control scheme is employed to regulate the thrust magnitude.

Moving further, Ma [10] proposes a constant thrust ascent trajectory with an initial constraint to achieve a safe vertical takeoff. Being more specific, the tangential velocity is constrained to be zero for the first 10 s or 500 m once the ascent phase is initiated. Indeed, without this condition the fuel-optimal solution clears the ground with a smaller angle respect to the surface as demonstrated by the numerical results presented in his work.

2.2.3 3D Ascent trajectory

In other papers [11, 12] the same author solves the optimal control problem for three-dimensional ascent trajectories. In those cases the Moon geographical features are modelled as simple cones around the launch site and a minimum safe distance is imposed throughout the whole ascent phase as depicted in figure 2.3.

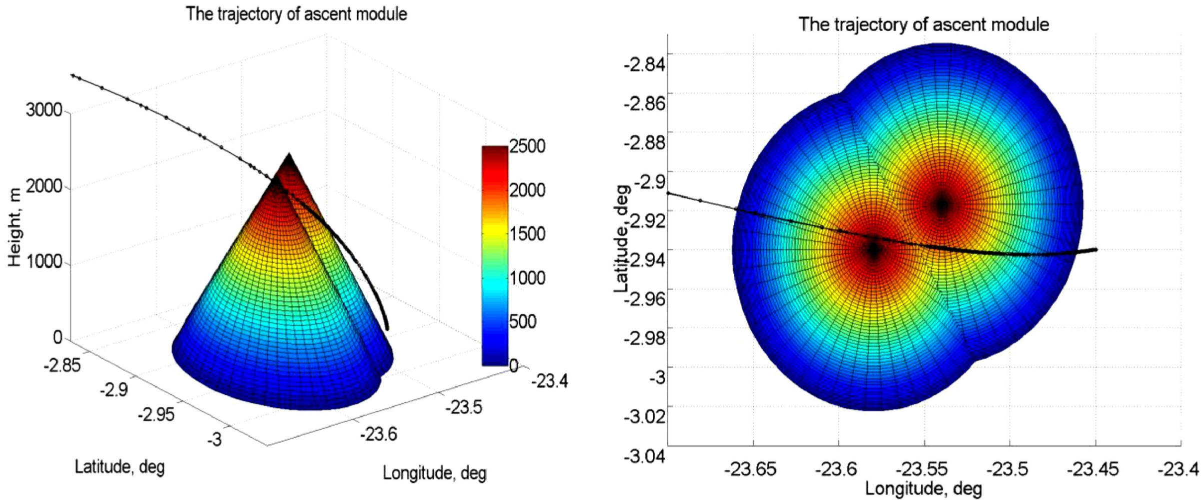

Figure 2.3: Optimized terrain clearance path
credits: L. Ma [10]

2.2.4 Optimal control problem

In sections 2.2.1 to 2.2.3 the optimal transfer trajectory is computed solving an optimal control problem whose mathematical formulation is given in equations 2.1 to 2.6 [9].

Minimize:

$$J = \phi(t_f, \boldsymbol{x}_f) + \int_{t_0}^{t_f} L(t, \boldsymbol{x}, \boldsymbol{u}) dt \tag{2.1}$$

Subject to:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(t, \boldsymbol{x}, \boldsymbol{u}) \tag{2.2}$$

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0 \tag{2.3}$$

$$\boldsymbol{u} \in U \tag{2.4}$$

$$\boldsymbol{\Psi}(t_f, \boldsymbol{x}_f) = \mathbf{0} \tag{2.5}$$

$$S(\boldsymbol{x}) \geq 0 \tag{2.6}$$

Where $\boldsymbol{x}$ are the states variables, $\boldsymbol{u}$ the controls, $\boldsymbol{x}_0$ the initial conditions (ICs) at time t_0, $\boldsymbol{\Psi}$ the terminal constraints, S the path constraints, $\boldsymbol{f}(t, \boldsymbol{x}, \boldsymbol{u})$ the dynamics and J the objective function. The aim is to find the optimal control profile $\boldsymbol{u}(t)$ and the corresponding state history $\boldsymbol{x}(t)$ that lead to a minimum for J.

2.2.5 Indirect methods

The main approach to solve an optimal control problem in the form specified in section 2.2.4 is to derive the first-order necessary conditions for a stationary point of J applying the calculus of variations and then convert the continuous-time problem into a two-point boundary value problem (TPBVP). Those conditions are known as the Euler-Lagrange theorem and they are sufficient to find the optimal solution in the simplest cases. However, for a much more complex formulation another condition known as the Pontryagin Minimum Principle has to be taken into account to find the optimal solution of the original problem.

Obtaining the solution through this approach requires at first a complex algebraic manipulation of equations 2.1 to 2.6 to derive the first-order optimality conditions and obtain the TPBVP formulation [9]. Then, since the equivalent problem has not a closed-form solution a numerical approximation has to be computed starting from a provided initial guess on states, controls and costates variables. The lasts are Lagrange multipliers arisen from the variational formulation and their lack of physical meaning causes the generation of the initial guess to become not trivial. Moreover, the numerical solution has a small region of convergence that makes this approach not desirable even if with a close enough initial guess an highly accurate solution can be found in just few iterations.

2.2.6 Direct methods

To overcome most of the issues encountered in section 2.2.5, a different approach can be used to obtain an approximate solution of an optimal control problem. This formulation does not require the derivation of the necessary conditions for a minimum of J and the continuous optimal control problem is directly converted into a Nonlinear Programming problem (NLP) and solved with an iterative routine.

First, the states and control variables are discretized in time and the solution is computed only in a finite number of points or nodes. The continuous solution is then approximated with an interpolating polynomial that fits the discrete data. Different transcription methods are proposed to discretize those variables as described by Garg [5] and Topputo [18], the mostly used being the High-Order Gauss-Lobatto and Radau Pseudospectral.

The Gauss-Lobatto transcription splits the whole phase duration into a specified number of segments and approximates the states and the control variables with a different interpolating polynomial inside each of them. Boundary conditions are then enforced at both endpoints to guarantee the continuity of states and controls throughout the whole phase. Within one segment the continuous variables are evaluated in an odd number of Legendre-Gauss-Lobatto (LGL) nodes which are usually preferred to an equal number of uniformly spaced points to improve the accuracy of the final solution. Two set of points are defined here, the discretization nodes (even-indexed) in which the state variables are evaluated and the collocation nodes (odd-indexed) in which their time derivatives are computed. The control variables are discretized in both set of points. At this stage the optimal solution is obtained through and iterative algorithm that drives towards zero the collocation defects defined as the difference between the states derivatives computed in the collocation nodes through equation 2.2 and the same quantities obtained after an interpolation step from the discretization nodes using Heremite interpolating polynomials.

The Radau Pseudospectral transcription adopts a similar approach to obtain a solution of the optimal control problem, the only difference being the number of nodes in which states and controls are evaluated. In this approach the whole phase is covered by a single segment and an higher order Lagrange interpolating polynomial is used to approximate the continuous variables. Both states and controls are evaluated in all the nodes and the defects are computed between the state derivatives obtained through equation 2.2 and the same quantities computed differentiating the Lagrange polynomials that approximate the corresponding states.

When a complex dynamics is involved the direct methods presented in this section are preferred to the indirect ones presented in section 2.2.5 since they do not require any algebraic manipulation of the equations of motions (EOMs) and they have a wider region of convergence. The main drawbacks are a lower accuracy of the obtained solution as well as an increased computational effort due to an higher number of decision variables.

Depending on the programming language, different libraries were developed to convert an optimal control problem into an NLP problem using one of the transcriptions described above. The most widely used is GPOPS-II [13], a proprietary software available for MATLAB, while an open-source alternative coded in Python is provided by dymos [8]. The last implements both the Gauss-Lobatto and the Radau Pseudospectral transcription methods and takes advantage of the OpenMDAO [7] framework to find the solution of an optimal control problem.

Once the continuous problem is transcribed, the resulting NLP problem is solved using external packages for nonlinear optimization that can be distinguished into two main categories: gradient-based and gradient-free. Gradient-based methods such as SNOPT [6], IPOPT [20] and fmincon [1] uses differential information on the design variables to find the optimum of the objective function. Their main advantages are a fast convergence when starting from a good initial guess and a small error on the obtained solution. On the other side, gradient-free algorithms are based on natural selection and converge to the optimal solution evolving an initial population composed by several states and controls profiles. Contrarily to the first family of methods, they do not require any initial guess but the convergence is achieved after an higher number of iterations and the final accuracy is lower.

2.3 Aims and objectives

The main objective of this work is to determine the most fuel-efficient transfer trajectory from a specified point on the lunar surface to a given final orbit and vice-versa suitable for a crewed reusable launcher. This last condition brings several constraints and safety requirements to ensure the success of the mission such as maximum allowed velocities, accelerations and time of flight. Finally, the avoidance of the Moon geographical features has also to be taken into account.

The analysis starts with the two dimensional ascent trajectory to a circular LLO for which multiple solutions are computed enforcing different constraints. Firstly, a parametric study is conducted for different values of specific impulse (Isp) and thrust over initial weight ratio (twr) imposing a fixed parking orbit altitude while the only control variable being the thrust direction. The objective is to find the best combination of the two quantities taking into account the current technological limitations before proceeding with a more accurate analysis of the transfer itself. The obtained results are validated with the ones computed through an indirect approach by one of the project tutors. Secondly, a further validation of the working tools is conducted reproducing the results presented by Ma [10] for a two-dimensional ascent phase. Thirdly, the constant thrust assumption is relaxed in order to fully exploit the range of feasible solutions while looking for the most fuel-efficient trajectory. Finally, the geographical features of the Moon are taken into account and avoided constraining the spacecraft to fly over a minimum safe distance from the surface itself.

As a second step, the results obtained by Ramanan [15] and Remesh [16] for several descent trajectories with different boundary conditions and path constraints are reproduced to demonstrate the versatility of the software used in this study. The last achievements concern the three-dimensional ascent trajectory for which a solution can be computed for every starting point on the Moon surface and target LLO, allowing the spacecraft to perform both in-plane and out-of-plane manoeuvres.

Another objective is to make the results reproducible by anyone interested in this subject. For this reason the final goal is achieved using only open-source software such as the Python libraries OpenMDAO and dymos.

2.4 Justification of the potential degree of novelty

The tools used in this study differs from the ones that can be find in all the previous literature and they include only open-source software freely available online. Being more specific, dymos is used to provide the transcription methods while the underlying NLP problem is solved with the OpenMDAO APIs coupled with the nonlinear optimization package IPOPT. Since the first two libraries were developed quite recently, only few simple examples are provided for two-dimensional ascent trajectories and no solutions have been already computed in the three-dimensional scenario, thus making the conducted work pioneering.

Moreover, the employed direct transcription methods represent a new branch in trajectory optimization and thus the obtained results may validate the advantages of these new approaches respect to the more classical ones as pointed out by several authors [3, 5, 17, 18].

Since this study fits in the context of Multidisciplinary Design Optimization (MDO) of a whole reusable launcher, not only the trajectory parameters are considered but also the engine performances and the mass constraints are taken into account while optimizing the transfer to and from the lunar surface. To fulfill those requirements several simulations are carried out testing the performances of the launcher while powered by different fuel mixtures and equipped with different propellant masses.

Finally, removing the constant thrust assumption while considering a three-dimensional ascent trajectory allows to exploit innovative solutions that can be potentially applied in future space missions.

3 Investigation Methods

Throughout this work the optimal transfer trajectories are obtained as solutions of a continuous-time optimal control problem arisen from the equations of motion (EOMs) that describe the spacecraft dynamics under the restricted two-body problem assumption. A transcription method is then applied to convert the optimal control problem into the corresponding NLP problem which is then numerically solved with an appropriate iterative algorithm as described in the following sections.

3.1 2D Model

In the restricted two-body problem the EOMs of a spacecraft subject to the gravitational pull of a central spherical body as well as its own thrust are expressed in polar coordinates as given by equations 3.1 to 3.5. The corresponding reference frame is depicted in figure 3.1.

$$\dot{r} = u \tag{3.1}$$

$$\dot{\theta} = \frac{v}{r} \tag{3.2}$$

$$\dot{u} = -\frac{\mu}{r^2} + \frac{v^2}{r} + \frac{T}{m} \sin\alpha \tag{3.3}$$

$$\dot{v} = -\frac{uv}{r} + \frac{T}{m} \cos\alpha \tag{3.4}$$

$$\dot{m} = -\frac{T}{Isp \; g_0} \tag{3.5}$$

Here r is the distance from the central body, θ the spawn angle, u and v the radial and tangential velocities, m the spacecraft mass, T the thrust magnitude, α the thrust direction, Isp the specific impulse, g_0 the standard gravitational acceleration and μ the Moon standard gravitational parameter.

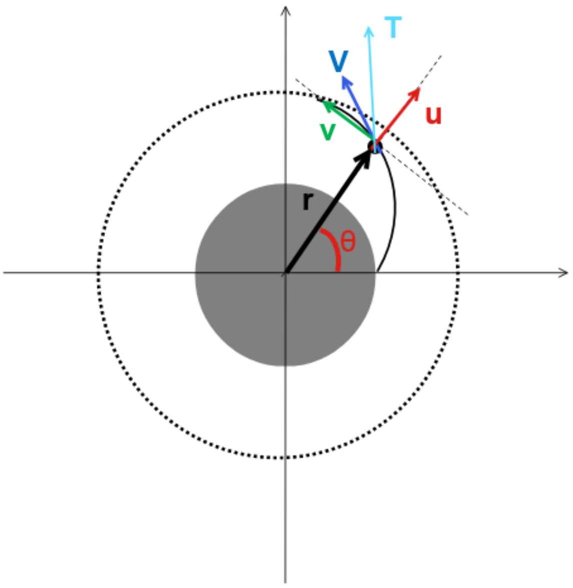

Figure 3.1: Polar coordinates reference frame

3.1.1 Ascent trajectory

Since the main goal is to find the most fuel-efficient ascent trajectory the resulting optimal control problem can be expressed with the formalism presented in section 2.2.4 as follows:

Minimize:

$$J = -m(t_f) \qquad \text{(objective function)} \tag{3.6}$$

Subject to:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(t, \boldsymbol{x}, \boldsymbol{u}) \qquad \text{(EOMs)} \tag{3.7}$$

$$\boldsymbol{x} = [r, \theta, u, v, m] \qquad \text{(state variables)} \tag{3.8}$$

$$\boldsymbol{u} = [T, \alpha] \qquad \text{(control variables)} \tag{3.9}$$

where 3.7 groups together equations 3.1 to 3.5. The initial conditions $t_0, \boldsymbol{x}(t_0)$ are specified as:

$$\begin{cases} t_0 = 0 \\ r(t_0) = R \\ \theta(t_0) = 0 \\ u(t_0) = 0 \\ v(t_0) = 0 \\ m(t_0) = m_0 \end{cases} \tag{3.10}$$

With R lunar radius and m_0 initial spacecraft mass. The final boundaries $\boldsymbol{\Psi}(t_f, \boldsymbol{x}_f)$ are given by:

$$\begin{cases} t_f = free \\ r(t_f) = R + H \\ \theta(t_f) = free \\ u(t_f) = 0 \\ v(t_f) = \sqrt{\mu/(R+H)} \\ m(t_f) = free \end{cases} \tag{3.11}$$

With H target orbit altitude. Finally, the inequality path constraints are specified as follows:

$$\begin{cases} r(t) > R \\ m(t) > 0 \\ T_{min} < T < T_{max} \end{cases} \tag{3.12}$$

When considering a constrained vertical path as presented by Ma [10] one of the two set of path constraints given by equations 3.13 has to be added to the previous formulation.

$$\begin{cases} v(t < 10\ s) = 0 \\ \alpha(t < 10\ s) = 90° \end{cases} \qquad \begin{cases} v(r < R + 500\ m) = 0 \\ \alpha(r < R + 500\ m) = 90° \end{cases} \tag{3.13}$$

A minimum safe altitude can be then imposed throughout the whole trajectory adding the corresponding inequality path constraint expressed as:

$$r > R + \frac{hR\theta}{R\theta + h/s} \tag{3.14}$$

Where h is the asymptotic safe altitude far from the launch site and s the slope of the constraint as depicted in figure 3.2.

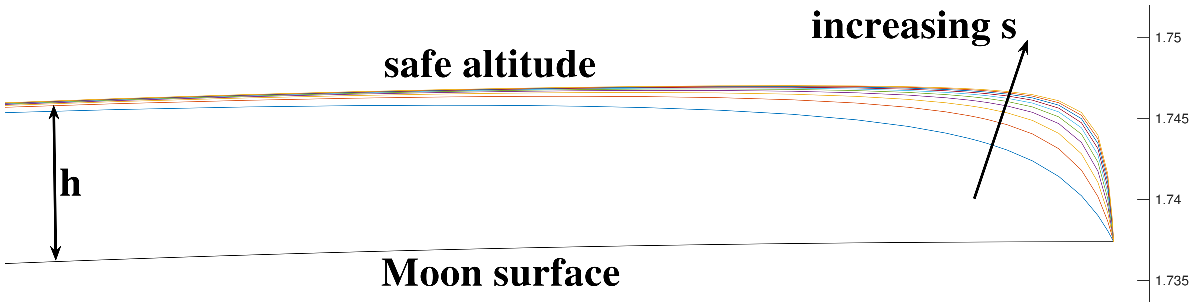

Figure 3.2: Geographical constraint for different slope values

3.1.2 Descent trajectory

The descent trajectory is split in two phases: an Hohmann manoeuvre to transfer the spacecraft from the initial circular parking orbit to a lower periapsis and a powered descent until the final touchdown. The controls are applied only in the second phase, while the initial conditions for the powered descent are computed solving the Hohmann transfer as described below. Finally, the overall fuel consumption is obtained adding the propellant mass required for the impulsive burn that inserts the spacecraft in the transfer trajectory with the one consumed during the powered braking manoeuvre.

Looking at figure 3.3 the Hohmann transfer can be described as follows:

$$a = \frac{r_a + r_p}{2} \tag{3.15}$$

$$\Delta v_a = \sqrt{\frac{\mu}{r_a}}\left[1 - \sqrt{\frac{2r_p}{r_a + r_p}}\right] \tag{3.16}$$

$$v_p = \sqrt{\mu\left(\frac{2}{r_p} - \frac{1}{a}\right)} \tag{3.17}$$

Where $r_a = R + H$ is the initial parking orbit radius, $r_p = R + h_p$ the periapsis radius in which the powered descent is initiated, h_p the corresponding periapsis altitude, a the semimajor axis, Δv_a the impulsive deorbit burn and v_p the periapsis tangential velocity.

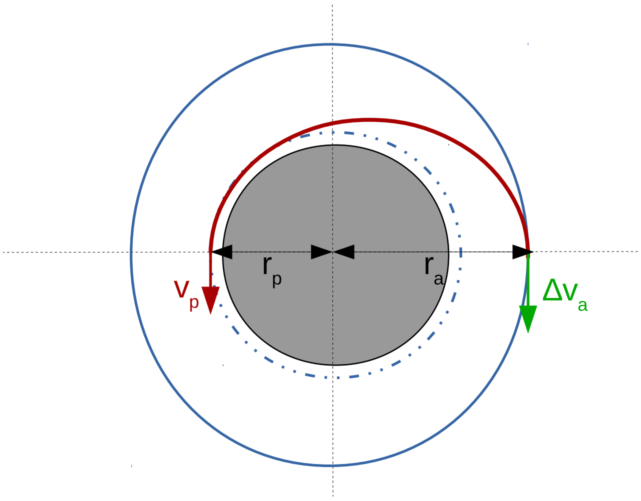

Figure 3.3: Example of an Hohmann transfer

Then, once Δv_a is known from equation 3.16 the corresponding propellant mass can be computed through the Tsiolkovsky rocket equation as follows:

$$\Delta m = m_0\left(1 - e^{-\frac{\Delta v_a}{Isp\ g_0}}\right) \tag{3.18}$$

At this stage the optimal control problem for the powered breaking phase is modelled one more time with equations 3.6 to 3.9 where only the boundary conditions are modified as given by equations 3.19 to 3.20.

Initial conditions:

$$\begin{cases} t_0 = 0 \\ r(t_0) = R + h_p \\ \theta(t_0) = 0 \\ u(t_0) = 0 \\ v(t_0) = v_p \\ m(t_0) = m_0 - \Delta m \end{cases} \tag{3.19}$$

Final conditions:

$$\begin{cases} t_f = free \\ r(t_f) = R \\ \theta(t_f) = free \\ u(t_f) = 0 \\ v(t_f) = 0 \\ m(t_f) = free \end{cases} \tag{3.20}$$

Finally, a vertical land is obtained enforcing additional path constraints on the tangential velocity and thrust direction as follows:

$$\begin{cases} v(r < R + h_v) = 0 \\ \alpha(r < R + h_v) = 90° \end{cases} \tag{3.21}$$

Where h_v is the altitude at the beginning of the vertical descent.

3.1.3 Semianalytic initial guess

As pointed out in section 2.2.6, when an optimal control problem is transcribed and an approximate solution for the resulting NLP problem is searched employing gradient-based methods their convergence is achieved only if a reasonable initial guess is provided. For the first simulations a linear interpolation of the boundary conditions is accurate enough to find the optimal solution, while in the most complex cases a more precise guess has to be provided to obtain a feasible solution. The last cases arise for example when the constraint on the minimum safe altitude described by equation 3.14 is enforced to obtain an ascent trajectory that avoids the Moon geographical features.

In this context an accurate initial guess is computed solving a simplified ascent trajectory divided in three separate phases for which a semianalytic solution can be easily obtained. Being more specific, the aforementioned transfer is split as follows:

- powered phase at constant radius $r \equiv R$ and constant thrust $T \equiv T_{max}$ to acquire the tangential velocity at the periapsis of an Hohmann transfer whose apoapsis altitude coincides with the target LLO altitude
- coasting phase (Hohmann transfer) from the lunar surface up to the target LLO
- impulsive burn at the apoapsis of the previous manoeuvre to inject the spacecraft in the desired circular parking orbit

The EOMs for the first powered phase obtained after imposing $r(t) \equiv R$ in equations 3.1 to 3.5 are given by the followings:

$$\dot{r} = u = 0 \tag{3.22}$$

$$\dot{\theta} = \frac{v}{R} \tag{3.23}$$

$$\dot{u} = 0 = -\frac{\mu}{R^2} + \frac{v^2}{R} + \frac{T}{m}\sin\alpha \tag{3.24}$$

$$\dot{v} = \frac{T}{m}\cos\alpha \tag{3.25}$$

$$\dot{m} = -\frac{T}{Isp\ g_0} \tag{3.26}$$

Combining together equations 3.24 and 3.25:

$$\begin{cases} \dfrac{T}{m}\sin\alpha = \dfrac{\mu}{R^2} - \dfrac{v^2}{R} \\ \dfrac{T}{m}\cos\alpha = \dot{v} \end{cases} \tag{3.27}$$

Applying the Pythagorean identity to 3.27 and solving for $\dot{v}$:

$$\left(\frac{T}{m}\right)^2 = \left(\frac{\mu}{R^2} - \frac{v^2}{R}\right)^2 + \dot{v}^2 \tag{3.28}$$

$$\dot{v}^2 = \left(\frac{T}{m}\right)^2 - \left(\frac{\mu}{R^2} - \frac{v^2}{R}\right)^2 \tag{3.29}$$

While, from equation 3.26:

$$\int_{m_0}^{m} dm = -\frac{T}{Isp\ g_0}\int_{t_0}^{t} dt \tag{3.30}$$

$$m(t) = m_0 - \frac{T}{Isp\ g_0}t \qquad (t_0 = 0) \tag{3.31}$$

Finally, combining both results:

$$\dot{v}(t) = \frac{dv}{dt} = \sqrt{\left(\frac{T}{m_0 - \frac{Tt}{Isp\ g_0}}\right)^2 - \left(\frac{\mu}{R^2} - \frac{v^2}{R}\right)^2} \tag{3.32}$$

$$\frac{dt}{dv} = \frac{1}{dv/dt} = f(t, v, parameters) \tag{3.33}$$

Equation 3.33 can be solved for the required Δt to achieve v_p as follows:

$$dt = f(t, v, parameters)dv \tag{3.34}$$

$$\Delta t = \int_0^{v_p} f(t, v, parameters)dv \tag{3.35}$$

The ODE 3.35 is integrated numerically from 0 to v_p with initial condition $t_0 = 0$. Once Δt is obtained both quantities $\theta(t)$ and $v(t)$ can be computed as described below:

$$\begin{cases} \dot{\theta}(t) = \dfrac{v(t)}{R} \\ \dot{v}(t) = f(t, v, parameters) \end{cases} \tag{3.36}$$

Now the ODEs 3.36 are integrated numerically from 0 to Δt with initial conditions $\theta_0 = v_0 = 0$. At this point the timeseries of $\theta(t), v(t), \dot{v}(t), m(t)$ are known while $r(t) \equiv R$ and $u(t) \equiv 0$.

From equations 3.24 and 3.25:

$$\tan\alpha = \frac{\mu/R^2 - v^2/R}{\dot{v}(t)} \tag{3.37}$$

$$\alpha(t) = \text{atan}\left(\frac{\mu/R^2 - v^2/R}{\dot{v}(t)}\right) \tag{3.38}$$

In this way the initial powered phase is uniquely determined once v_p is known. This last value is computed solving the Hohmann manoeuvre performed in the second phase as follows:

$$v_p = \sqrt{2\mu\left(\frac{R+H}{R(2R+H)}\right)} \tag{3.39}$$

$$\Delta v_a = \sqrt{\frac{\mu}{R+H}}\left[1 - \sqrt{\frac{2R}{2R+H}}\right] \tag{3.40}$$

Where H is the same of the equations 3.11.

In order to determine the values of $r(t), \theta(t), u(t)$ and $v(t)$ during the Hohmann transfer in all the discretization nodes the Kepler's time of flight equation has to be solved for each t_i defined by the adopted transcription. The total mass during the Hohmann transfer is constant and equivalent to the final mass of the first phase: $m_{hohmann} = m(\Delta t)$.

In the third and final phase the injection burn is modelled as follows:

$$m_{final} = m_{hohmann}\ e^{-\frac{\Delta v_a}{Isp\ g_0}} \tag{3.41}$$

The timeseries of the whole trajectory are computed assembling together the results of the 3 phases as plotted in figure 3.4. This solution is then used as initial guess for the actual trajectory optimization.

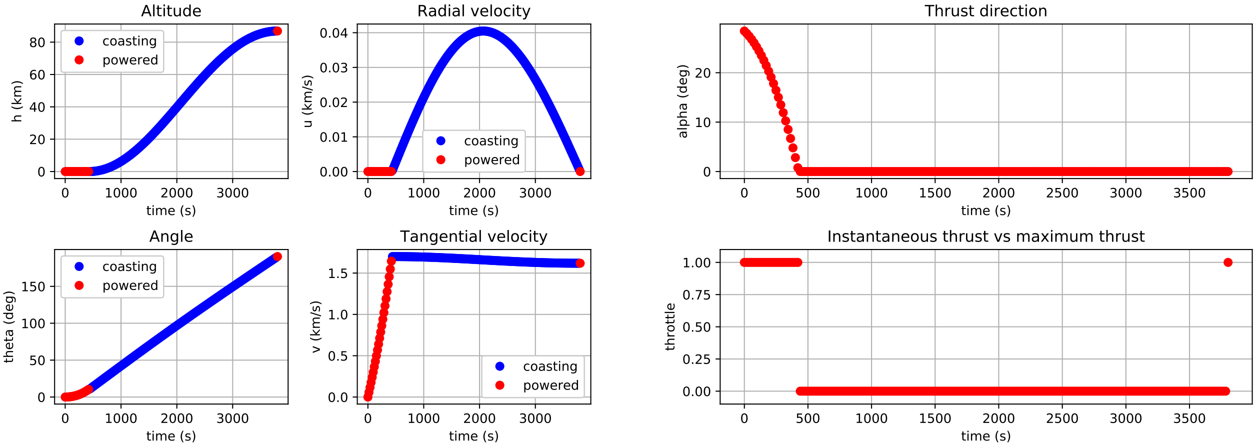

Figure 3.4: States and controls profiles for the semianalytical ascent trajectory

3.2 3D Model

Moving from a two-dimensional to a three-dimensional scenario, the spacecraft dynamics under the keplerian two-body assumption is described in cartesian coordinates respect to an intertial reference frame whose origin coincides with the center of the Moon as depicted on the left of figure 3.5.

Following the example presented by Benson [3] the corresponding EOMs are presented below.

$$\dot{\boldsymbol{r}} = \boldsymbol{v} \tag{3.42}$$

$$\dot{\boldsymbol{v}} = -\frac{\mu}{r^3}\boldsymbol{r} + \frac{T}{m}\hat{u} \tag{3.43}$$

$$\dot{m} = -\frac{T}{Isp\ g_0} \tag{3.44}$$

$$\begin{aligned}\boldsymbol{r} &= [x, y, z]\\ \boldsymbol{v} &= [v_x, v_y, v_z]\\ \hat{u} &= [u_x, u_y, u_z]\end{aligned} \tag{3.45}$$

Where $\boldsymbol{r}$ and $\boldsymbol{v}$ are the spacecraft position and velocity vectors, r the distance from the central body, T the thrust magnitude and $\hat{u}$ a unit vector that represents the thrust direction.

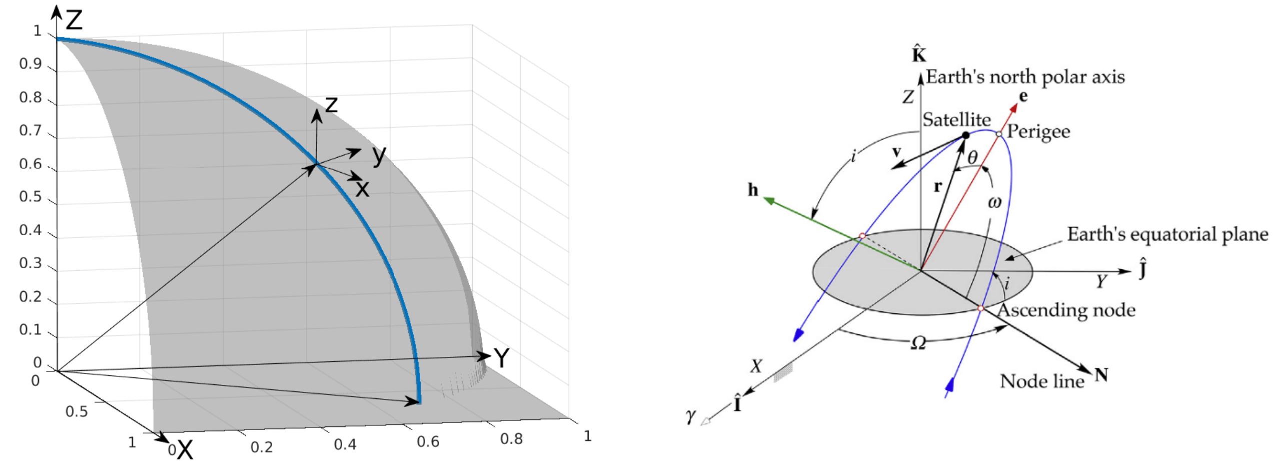

Figure 3.5: Cartesian reference frame and Classical orbital elements
credits: H. D. Curtis [4]

The target orbit is specified through its five Classical Orbital Elements (COEs) a, e, i, Ω, ω with a semimajor axis, e eccentricity, i inclination, Ω Right Ascension of the Ascending Node (RAAN) and ω Argument of periapsis. Here the true anomaly θ is not taken into account since the injection point is left as a free parameter to be optimized.

Thus, after computing the actual COEs from the spacecraft state vector $\boldsymbol{r}(t_f), \boldsymbol{v}(t_f)$ as described by Curtis [4] the final boundary conditions can be imposed as presented in equations 3.47.

$$\boldsymbol{r}(t_f), \boldsymbol{v}(t_f) \longrightarrow a, e, i, \Omega, \omega \tag{3.46}$$

$$|a - a_{tgt}| = 0 \qquad |e - e_{tgt}| = 0 \qquad |i - i_{tgt}| = 0 \qquad |\Omega - \Omega_{tgt}| = 0 \qquad |\omega - \omega_{tgt}| = 0 \tag{3.47}$$

Where $a_{tgt}, e_{tgt}, i_{tgt}, \Omega_{tgt}, \omega_{tgt}$ are the five Classical Orbital Elements of the specified target orbit. Equation 3.46 has not a straightforward implementation and requires the introduction of multiple *if/else* statements to handle both quadrant ambiguities while computing Ω, ω and singularities in case of circular ($e = 0$) or equatorial ($i = 0$) orbits.

Due to this, a different formulation of equations 3.47 based on the specific angular momentum and eccentricity vectors $\boldsymbol{h}, \boldsymbol{e}$ has to be preferred to impose the final boundaries. Those vectors are two constant of motion of the final orbit and can be computed both from the spacecraft final state vector and the target COEs as described by Curtis [4].

From the spacecraft state vector $\boldsymbol{r}, \boldsymbol{v}$ the two quantities are readily computed as follows:

$$\boldsymbol{h} = \boldsymbol{r} \times \boldsymbol{v} \tag{3.48}$$

$$\boldsymbol{e} = \frac{1}{\mu}\left[\left(v^2 - \frac{\mu}{r}\right)\boldsymbol{r} - (\boldsymbol{r} \cdot \boldsymbol{v})\boldsymbol{v}\right] \tag{3.49}$$

While from the target COEs they are obtained as presented below.

$$h = \sqrt{\mu a(1 - e^2)} \tag{3.50}$$

$$\boldsymbol{Q}_{p \longrightarrow i} = \boldsymbol{R}_3^T(\Omega)\boldsymbol{R}_1^T(i)\boldsymbol{R}_3^T(\omega) \tag{3.51}$$

$$\boldsymbol{h}_{tgt} = h \cdot \boldsymbol{Q}_{p \longrightarrow i}\ \hat{k} \tag{3.52}$$

$$\boldsymbol{e}_{tgt} = e \cdot \boldsymbol{Q}_{p \longrightarrow i}\ \hat{\imath} \tag{3.53}$$

Where $\boldsymbol{Q}_{p \longrightarrow i}$ is the rotation matrix between perifocal and inertial reference frame, $\boldsymbol{R}_1$ and $\boldsymbol{R}_3$ the elementary rotation matrices around the X and Z directions while $\hat{\imath} = [1, 0, 0]^T$ and $\hat{k} = [0, 0, 1]^T$ are the unit vectors along the x and z axis of the perifocal reference frame.

Finally, an equivalent formulation of the boundary conditions 3.47 is obtained as follows:

$$\boldsymbol{h} - \boldsymbol{h}_{tgt} = \boldsymbol{0} \qquad \boldsymbol{e} - \boldsymbol{e}_{tgt} = \boldsymbol{0} \tag{3.54}$$

The resulting optimal control problem for the three-dimensional ascent trajectory is then formulated as shown below:

Minimize:

$$J = -m(t_f) \qquad \text{(objective function)} \tag{3.55}$$

Subject to:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(t, \boldsymbol{x}, \boldsymbol{u}) \qquad \text{(EOMs)} \tag{3.56}$$

$$\boldsymbol{x} = [\boldsymbol{r}, \boldsymbol{v}, m] \qquad \text{(state variables)} \tag{3.57}$$

$$\boldsymbol{u} = [T, \hat{u}] \qquad \text{(control variables)} \tag{3.58}$$

where 3.56 groups together equations 3.42 to 3.44. The initial conditions $t_0, \boldsymbol{x}(t_0)$ are specified as:

$$\begin{cases} t_0 = 0 \\ \boldsymbol{r}(t_0) = \boldsymbol{r}_0 \\ \boldsymbol{v}(t_0) = \boldsymbol{0} \\ m(t_0) = m_0 \end{cases} \tag{3.59}$$

With $\boldsymbol{r}_0$ initial position on the Moon surface and m_0 initial spacecraft mass. The final boundaries $\boldsymbol{\Psi}(t_f, \boldsymbol{x}_f)$ are given by:

$$\begin{cases} t_f = free \\ \boldsymbol{h}(t_f) - \boldsymbol{h}_{tgt} = \boldsymbol{0} \\ \boldsymbol{e}(t_f) - \boldsymbol{e}_{tgt} = \boldsymbol{0} \\ m(t_f) = free \end{cases} \tag{3.60}$$

Where the second and third equations in 3.60 are equivalent to 3.54. Finally, the path constraints are specified as follows:

$$\begin{cases} \|\boldsymbol{r}(t)\| > R \\ m(t) > 0 \\ T_{min} < T < T_{max} \\ \|\hat{u}\| = 1 \end{cases} \tag{3.61}$$

3.3 Python implementation

A solution for the optimal control problems presented in sections 3.1 and 3.2 is obtained through a direct transcription method implemented in the OpenMDAO/dymos framework.

The OpenMDAO structure starts from a top-level container, the Problem, which holds the System and the Driver classes. The System can be either a Group or a Component and defines the problem variables, equations and solvers. Moreover, a Group can contain more than one instance of both classes Component and Group and the different input/output can be linked together. On the other side, the Component class can be implemented either as an ExplicitComponent or an ImplicitComponent and provides only the lowest-level functionalities such as basic calculations. Lastly, the Driver defines the optimizer that iteratively solve the Problem [7].

For each dynamic model the equations of motions are implemented in a dedicated ExplicitComponent that provides the EOMs outputs and their partial derivatives wrt the EOMs inputs. When complex constraints are modelled, a dedicated ExplicitComponent is implemented to provide the corresponding outputs and partial derivatives.

At this point the dymos library implements two classes that inherit directly from the OpenMDAO Group class to model a single or a multiple phase trajectory: Phase and Trajectory. The first is used to group the required ExplicitComponents that describe a continuous ascent or descent trajectory, while the second is implemented to combine together multiple Phase instances, for example to handle discontinuities in the states and controls variables introduced by a vertical take-off or landing. Finally, the whole System is composed either by a Phase or a Trajectory class instance.

The Problem is then built grouping together the previously defined System with a Driver class instance in which the optimizer is defined as can be seen in figure 3.6. By default, the only optimizer provided by OpenMDAO able to handle both equality and inequality constraints and thus suitable for the considered NLP problems is SLSQP wrapped from the Scipy library. Since this algorithm is able to converge only the simplest analyzed cases an external library, namely pyOptSparse [14], has been compiled and linked with OpenMDAO to gain access to a wider suite of more powerful NLP solvers. After several tests IPOPT [20] coupled with the sparse symmetric indefinite linear solver MA57 [2] has been finally selected as the most promising solver to obtain an accurate solution in a reasonable computational time.

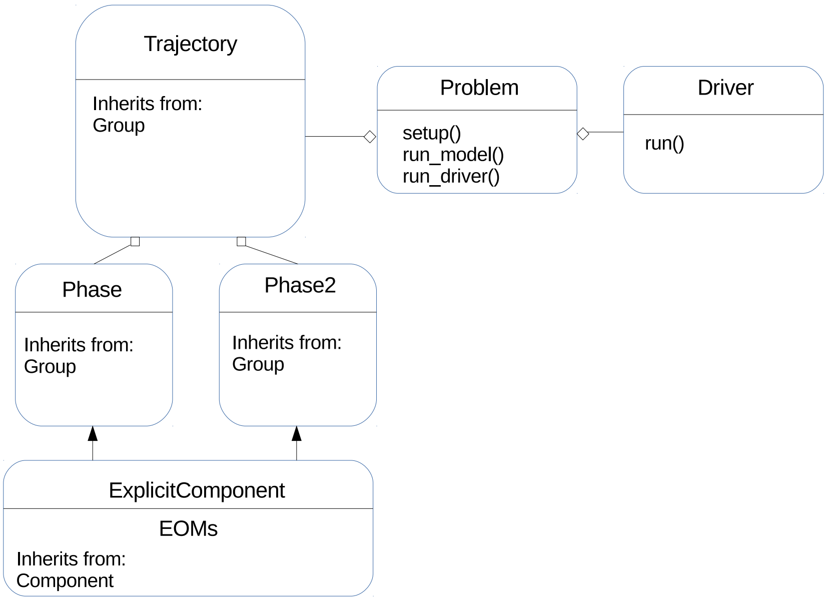

Figure 3.6: OpenMDAO/dymos simplified UML diagram

Once the Problem, Driver and System class instances are defined, the objective function as well as the boundary conditions and the path constraints are readily set through the corresponding methods implemented in both classes Phase and Trajectory. Then, before a solution can be actually computed an initial guess for both the states and the controls variables has to be provided either as a linear interpolation of the previously defined boundaries or computing a simplified solution as described in section 3.1.3.

After that the optimal transfer trajectory is computed running the model and waiting for the optimizer to perform the required iterations and drive the collocation defects below the specified tolerance as seen in section 2.2.6.

A further analysis can be then conducted to assess the numerical accuracy of the obtained solution. Being more specific, the two classes Phase and Trajectory implement the method *simulate* which is aimed to reproduce the obtained solution through an explicit integration step. The method wraps the Scipy *solve_ivp* function that employes a Runge-Kutta integration scheme to solve the initial value problem (IVP) fromulated starting from the ascent trajectory EOMs, initial boundary conditions and previously computed optimal controls profiles. If the error between IVP and optimal solutions is below the specified threshold the obtained transfer trajectory is said to be feasible.

4 Results and Analysis

In this chapter the solutions obtained for different ascent and descent trajectories are presented and their main features are discussed. If not differently specified, the initial guess required to solve the associated NLP problem is obtained from a linear interpolation of the boundary conditions as pointed out in section 3.1.3 while the time of flight is initially estimated on a case-by-case basis. Moreover, the numerical results are validated using the *simulate* method presented in section 3.3 and compared with the ones presented in the literature when possible.

All the solutions are computed setting the following numerical values for the different constant parameters:

Constant	Units	Value
g_0	m/s	9.80665
μ	m^3/s^2	$4.9028 \cdot 10^{12}$
R	m	$1737.4 \cdot 10^3$

Table 4.1: Constant parameters

4.1 2D Ascent trajectory

The first analysis are conducted for a two-dimensional ascent trajectory from the Moon surface to a circular LLO for which the dynamic model is described in section 3.1.1. Firstly, a solution with constant thrust magnitude is obtained and a parametric study is conducted for different values of Isp and thrust over initial weight ratio (twr). Secondly, the constant thrust assumption is relaxed to fully exploit the range of optimal solutions. Finally a path constraint on the minimum safe altitude is added to avoid the Moon geographical features during the ascent phase.

The different optimal transfer trajectories are computed for the Isp, twr, initial spacecraft mass m_0 and final orbit altitude H given in table 4.2. Then, in section 4.1.2 the first two values are varied in the specified intervals to conduct the aforementioned parametric analysis.

	Units	Value	Interval (section 4.1.2)
Isp	s	450.0	[256.98, 513.95]
twr	-	2.1	[1.1, 4.0]
m_0	kg	1.0	constant
H	m	$86.87 \cdot 10^3$	constant

Table 4.2: Isp, twr, initial mass and target orbit altitude for the 2D ascent trajectories

4.1.1 Constant thrust

The first solution is obtained replacing the third inequality constraint in 3.12 with two equality constraints to impose a constant thrust magnitude as described by equation 4.1:

$$T_{min} = T = T_{max} \tag{4.1}$$

In this case a Gauss-Lobatto transcription method with 10 segments and a third-order interpolating polynomial is used to convert the continuous-time optimal control problem into the corresponding NLP problem that is finally solved with the IPOPT routine. The results obtained with an initial guess $t_f = 500\ s$ are depicted in figure 4.1 while the corresponding time of flight (tof) and propellant fraction are given in table 4.3.

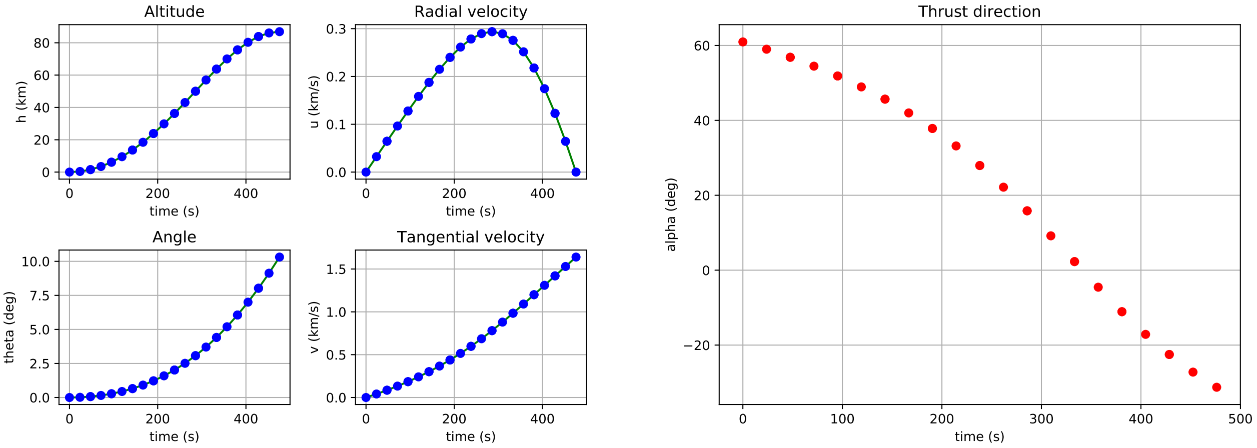

Figure 4.1: States and controls profiles for the 2D ascent trajectory with constant thrust

	Units	Value
tof	s	476.13
m_p/m_0	-	0.3680

Table 4.3: Tof and propellant fraction for the 2D ascent trajectory with constant thrust

4.1.2 Constant thrust response surfaces

In this section the response surfaces that describe the variation of both time of flight and propellant fraction for different values of Isp and twr are computed solving multiple optimal transfer trajectories for every couple (Isp, twr) obtained discretizing the continuous intervals in table 4.2.

Those surfaces are then used in the whole optimization framework that aims to optimize every aspect of a lunar reusable launcher in which this work fits into. Consequently, when a Multidisciplinary Analysis (MDA) is launched on the complete model the trajectory block is already solved since the results can be retrieved from the aforementioned surfaces. With this strategy the computational time of an MDA analysis is considerably reduced since the trajectory block requires the highest computational effort among all the other disciplines.

The numerical results presented in figure 4.2 are obtained applying a Gauss-Lobatto transcription method with 5 segments and an interpolating polynomial of 11^{th} order and solving the corresponding NLP problem with the SLSQP routine available in the Scipy library.

As can be seen from the left plot in figure 4.2, the optimal twr is always between 2.1 and 2.2 independently from the Isp value. On the other side, for a given twr the propellant fraction decreases monotonically while the Isp is increased. Finally, from the plot on the right is possible to conclude that the time of flight decreases with increasing values of twr but it is almost independent from the Isp value when the thrust is lower than two times the initial spacecraft weight.

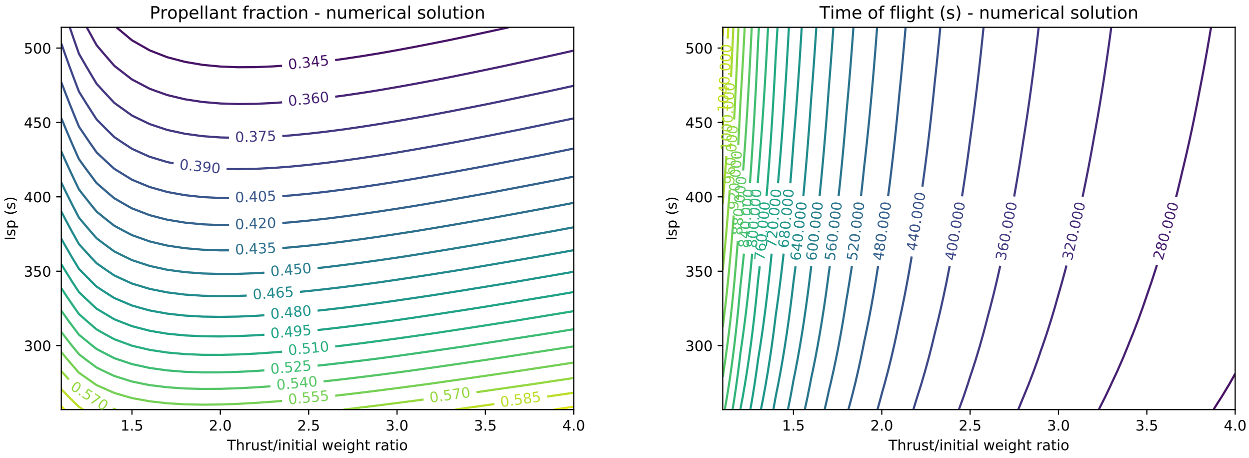

Figure 4.2: Propellant fraction and time of flight response surfaces

Those results are then compared with the ones obtained by one of the project tutors through a indirect approach and an almost perfect match is found, thus validating the investigation methods adopted in this study.

4.1.3 Variable thrust

In this section as well as in 4.1.4 the constant thrust assumption is removed and equation 4.1 is substituted with the following inequality constraints:

$$0 \leq T \leq T_{max} \tag{4.2}$$

Where T_{max} is computed from the twr given in table 4.2 and the time of flight initial guess is set equal to $t_f = 4000\ s$.

The optimal transfer trajectory is obtained computing the solution of the NLP problem that arises from a Gauss-Lobatto transcription with 150 segments and a third-order interpolating polynomial using the IPOPT solver. The corresponding results are depicted in figure 4.3 and the optimal time of flight and propellant fraction are reported in table 4.4.

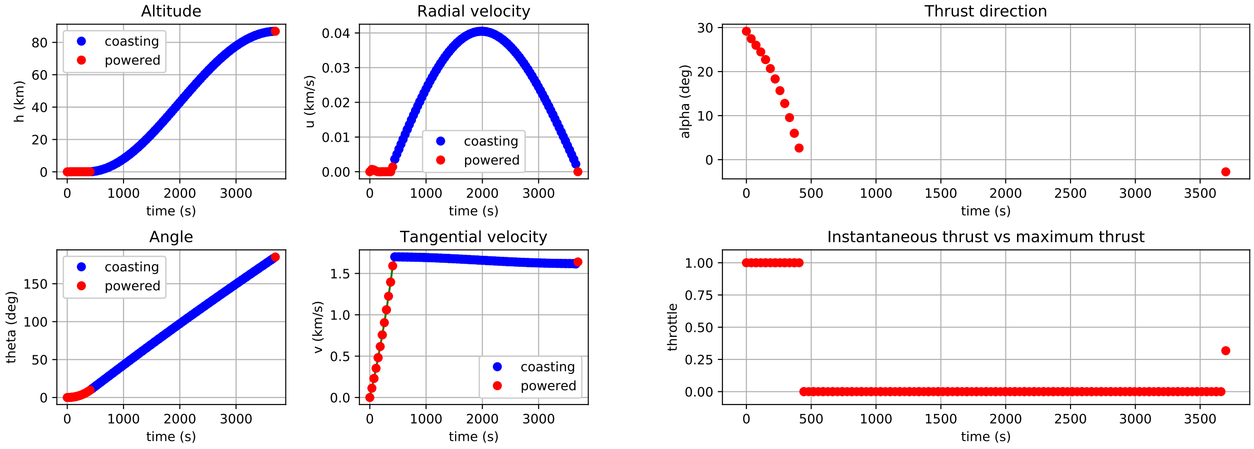

Figure 4.3: States and controls profiles for the 2D ascent trajectory with variable thrust

As demonstrated in both figure 4.3 and table 4.4 a variable thrust ascent trajectory lowers the fuel consumption respect to its counterpart at constant thrust when the same LLO is targeted.

	Units	Value
tof	s	3697.56
m_p/m_0	-	0.3364

Table 4.4: Tof and propellant fraction for the 2D ascent trajectory with variable thrust

The only drawback being a consistent increase in the required time of flight that is now almost 8 times higher. Moreover, as expected from the theoretical background a *bang-bang* control scheme in which the thrust magnitude is either zero or maximum is obtained for the control profile.

Finally, those results are a close approximation of the semianalytic solution presented in section 3.1.3 obtained splitting the whole trajectory in three separate phases. If the spacecraft is allowed to move on the lunar surface at $r \equiv R$ the three-phases trajectory represents the actual optimum with the least possible fuel consumption. As a consequence, the results obtained in this section represent the optimal transfer when the inequality constraint $r > R$ is added to the problem formulation.

4.1.4 Variable thrust with constrained minimum safe altitude

As shown in figure 4.3, when the constant thrust assumption is relaxed the spacecraft clears the ground with a really shallow angle to minimize the fuel consumption, thus posing several safety concerns due to the presence of lunar highlands around the launch site. Consequently, a successful mission can be only guaranteed if an appropriate constraint is added to specify a minimum safe distance between the spacecraft and the ground throughout the whole ascent trajectory. Since both the position of the launch site and an accurate model for the Moon geographical features are unknowns, the lasts are replaced by a simple analytic function and the aforementioned constraint is given by equation 3.14. The numerical values for the asymptotic safe altitude h and the constraint slope s are given in table 4.5.

	Units	Value
h	m	5000
s	-	100

Table 4.5: Asymptotic safe altitude and constraint shape for the 2D ascent trajectory with variable thrust and constrained minimum altitude

The numerical results are obtained applying a Gauss-Lobatto transcription method with 200 segments and a third-order interpolating polynomial and solving the resulting NLP problem with the optimizer IPOPT. In this case the NLP solver is unable to converge starting from an initial guess obtained as a simple linear interpolation of the boundaries, thus requiring the semyanalitic solution described in section 3.1.3 to be computed and set as starting point for the subsequent iterations.

The timeseries of both states and controls variables are depicted in figure 4.4, while the optimal time of flight and propellant fraction are summarized in table 4.6.

	Units	Value
tof	s	3367.77
m_p/m_0	-	0.3550

Table 4.6: Tof and propellant fraction for the 2D ascent trajectory with variable thrust and constrained minimum altitude

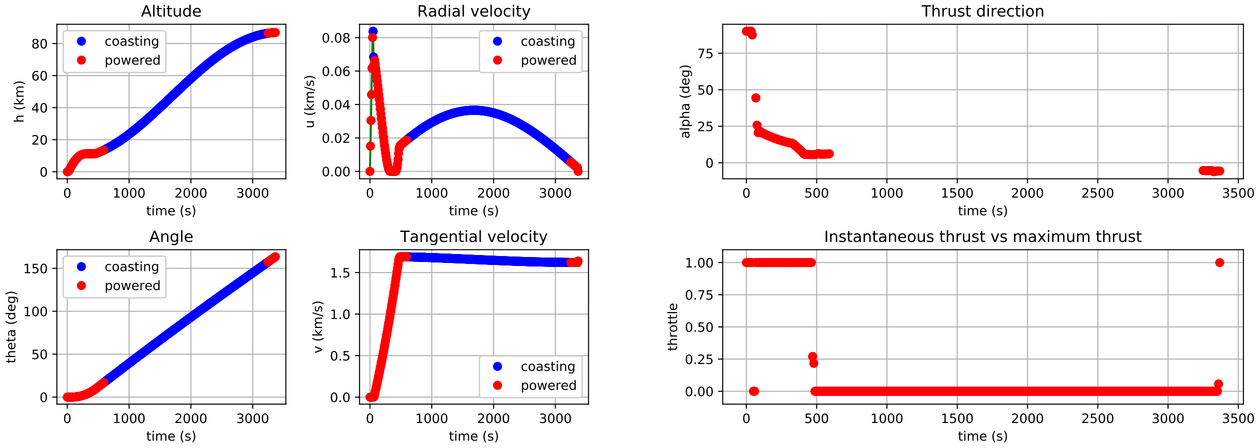

Figure 4.4: States and controls profiles for the 2D ascent trajectory with variable thrust and constrained minimum altitude

As demonstrated by figure 4.4, once a minimum safe altitude is specified the spacecraft clears the ground almost vertically with an initial thrust direction $\alpha(t_0) = 90°$, thus avoiding the surrounding lunar highlands. α is then reduced as soon as a minimum safe altitude is achieved to boost the spacecraft towards the final LLO while minimizing the propellant consumption. After about 500 s the required tangential velocity to reach the parking orbit is achieved and the thrusters are switched off to save as much fuel as possible. Finally, a small burn is performed at the end of the coasting phase to inject the spacecraft in the specified orbit.

Comparing the results in tables 4.4 and 4.6 is possible to conclude that the added constraint causes the propellant fraction to slightly increase as expected. On the other side the time of flight is reduced since the spacecraft is forced to rapidly acquire the safe altitude imposed by the added constraint.

4.2 Ascent with constrained vertical phase

Before computing the optimal trajectory for a generic geographic constraint as seen in section 4.1.4 a less sophisticated approach was implemented adopting the strategy proposed by Ma [10]. In this simulation an appropriate constraint is applied to ensure a vertical take-off. Firstly the thrust direction is forced to remain perpendicular to the Moon surface for a certain amount of time while the second solution is obtained imposing the same limitation up to a predetermined altitude. Those two constraints are described by equations 3.13.

	Units	Value
H	km	51.44
m_0	kg	4869
Isp	s	309
twr	$-$	1.95

Table 4.7: Target orbit altitude, initial spacecraft mass, Isp and twr for the 2D ascent trajectory with initial vertical constraint

The values for target orbit altitude H, initial spacecraft mass m_0, Isp and twr used in the numerical simulations are presented in table 4.7, while the time of flight initial guess is equal to $t_f = 1000\ s$. The ascent trajectory is then modelled with two subsequent phases to describe both the initial vertical ascent and the optimized transfer up to the target LLO. In each phase the equations of motions seen in section 3.1.1 are employed to describe the spacecraft dynamics.

The numerical solutions are then computed after transcribing the optimal control problem into the corresponding NLP formulation applying a Gauss-Lobatto transcription method with 3 and 15 segments and a third-order interpolating polynomial for the first and second phase respectively. The obtained NLP is then handily solved with the IPOPT routine.

The obtained results are plotted in figures 4.5 in which the vertical ascent corresponds to the first points with $\alpha = 90°$. Tables 4.8 and 4.9 show that the propellant consumption is lower in the first case since the vertical constraint is relaxed earlier than in the second solution for which 25.09 s are required to achieve a 500 m altitude. Finally, in accordance to what is said in sections 4.1.3 and 4.1.4 those results demonstrate one more time how the fuel consumption is reduced maintaining a small angle for the initial thrust direction.

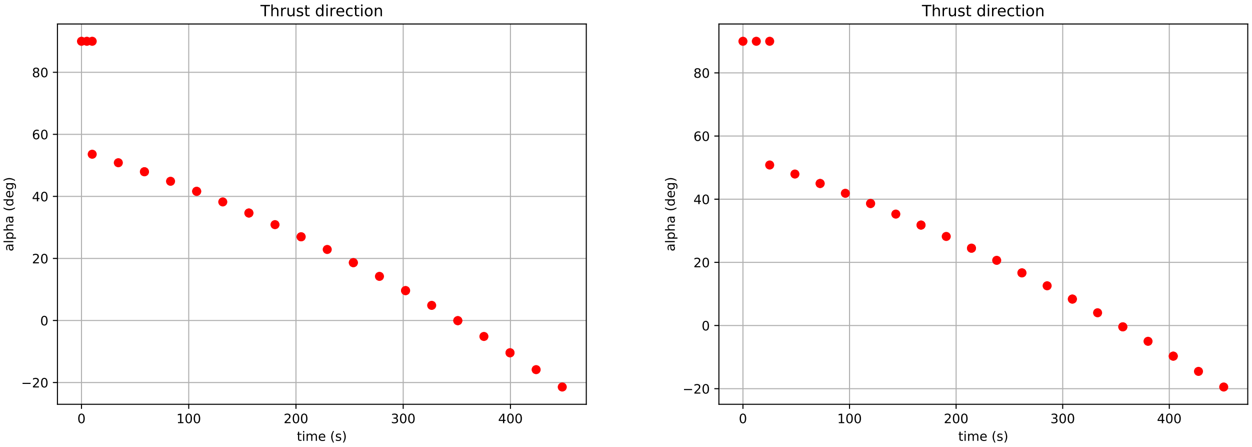

Figure 4.5: Control profiles for the 2D ascent trajectory with initial vertical constraint

	Units	Value
tof	s	448.31
m_f/m_0	$-$	0.4686

Table 4.8: Tof and propellant fraction for the 2D ascent trajectory with initial vertical constraint for $t < 10$ s

	Units	Value
tof	s	451.06
m_f/m_0	$-$	0.4714

Table 4.9: Tof and propellant fraction for the 2D ascent trajectory with initial vertical constraint for $r < R + 500$ m

The present simulations match the work of the authors [10] and are used to test the optimization of a multi-phase trajectory within the OpenMDAO framework, thus validating the investigation methods used throughout this study.

4.3 2D Descent trajectories

In this section the results obtained for a two-dimensional descent trajectory are presented, analyzed and compared to the ones present in the literature. Two particular cases will be treated:

- constant thrust trajectory without constraints on the spacecraft attitude

- similar trajectory with an additional constraint to force the thrust direction to be perpendicular respect to the Moon surface when flying below a specified altitude

For both simulations the Isp, twr, initial spacecraft mass m_0 and periapsis altitude h_p are presented in table 4.10.

	Units	Value
Isp	s	310
twr	-	0.9
m_0	kg	300.0
h_p	km	15

Table 4.10: Isp, twr, intial spacecraft mass and periapsis altitude for the 2D descent trajectories

4.3.1 Constant thrust

The simplest solution is computed with the dynamic model presented in section 3.1.2. The optimization is performed only on the final powered descent phase from the periapsis of the Hohmann transfer until the Moon surface. The optimal transfer trajectory is obtained computing the solution of the NLP problem that arises from a Gauss-Lobatto transcription with 5 segments and a 13^{th}-order interpolating polynomial using the IPOPT solver. The time of flight initial guess is set equal to $t_f = 1000\ s$.

The numerical results are reported in table 4.11 and plotted in figure 4.6. The thrust direction α is not constrained and varies from almost 170° until 130°, meaning that the probe will touch the ground with a 40° angle respect to the local vertical.

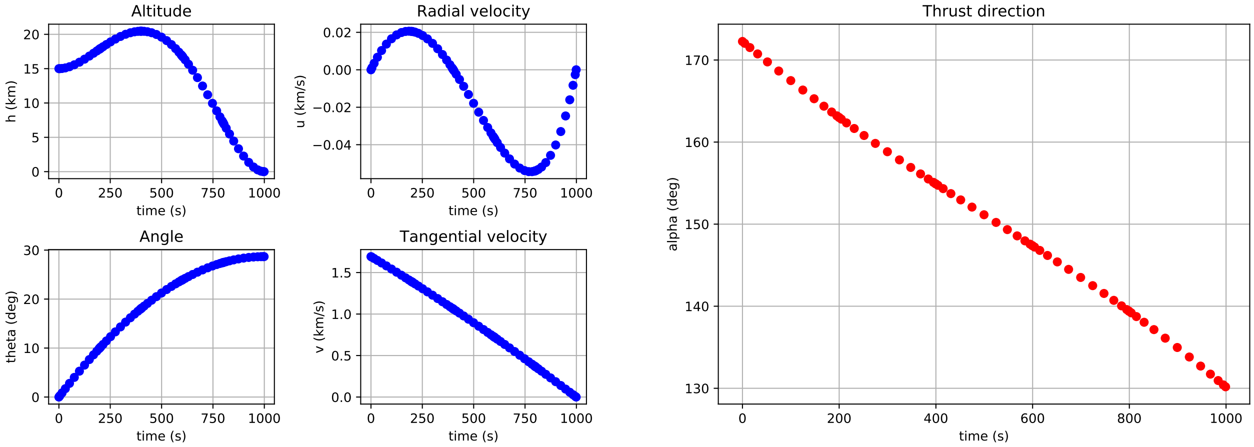

Figure 4.6: States and controls profiles for the 2D descent trajectory without vertical constraints

	Units	Value
tof	s	999.05
m_p/m_0	-	0.4851

Table 4.11: Tof and propellant fraction for the 2D descent trajectory without vertical constraints

The same numerical results are found in the work presented by Ramanan [15], thus validating the ones described in this section.

4.3.2 Constant thrust with constrained vertical landing

In order to satisfy the safety requirements imposed by a crewed mission the previous formulation is modified imcluding a constraint on the final phase of the descend trajectory. Being more specific, a minimum altitude equal to 4 km is a priori chosen as starting point for a constrained vertical landing. At lower altitudes the thrust direction α is set equal to 90° forcing the spacecraft to touch the ground perpendicularly to the surface. The whole transfer is then modelled as a two-phases trajectory as already seen in section 4.2 using the dynamic model presented in section 3.1.2.

The corresponding NLP is obtained using a Gauss-Lobatto transcription method with one segment and a 23^{rd}-order or a 5^{th}-order interpolating polynomial for the first and second phase respectively. The discrete problem is then solved with the SLSQP algorithm with and initial time of flight guess of $t_f = 1100\ s$. The obtained results are shown both in figure 4.7 and in table 4.12.

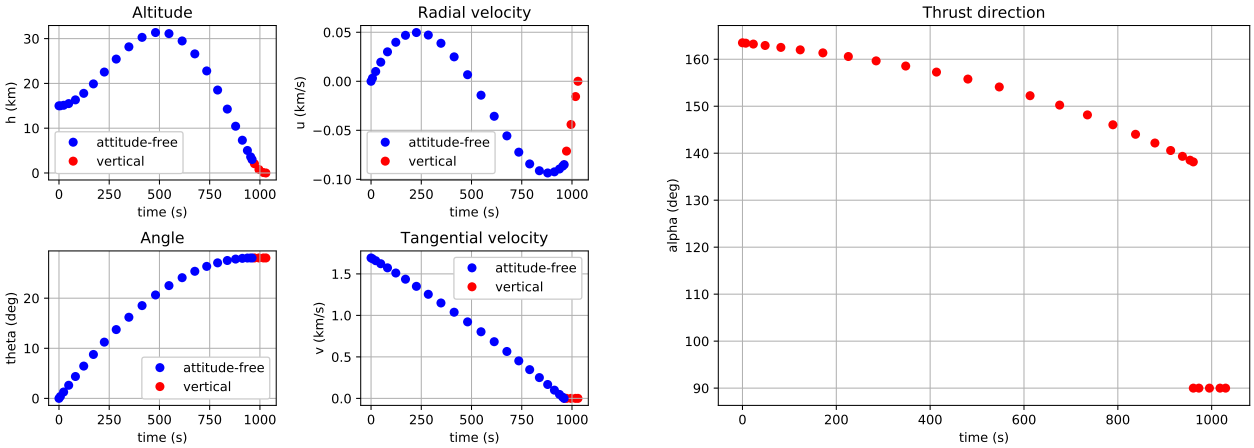

Figure 4.7: States and controls profiles for the 2D descent trajectory with vertical constraints on thrust

	Units	Value
tof	s	1034.79
m_p/m_0	-	0.5024

Table 4.12: Tof and propellant fraction for the 2D descent trajectory with vertical constraints in thrust

From the right plot in figure 4.7 is possible to see the vertical thrust direction at the end of the transfer which is underlined by the sudden change of α from 139.62° to 90° at $t = 955.46\ s$. As expected, the overall propellant fraction is greater when compared with the previous solution and grows from 0.4851 to 0.5124. Notwithstanding the higher value of m_p/m_0, the increased fuel consumption is necessary to ensure a soft and harmless landing.

All the results described above are comparable with the ones obtained by Remesh [16] in his studies.

4.4 3D Ascent trajectory

In this section an optimal solution for different three-dimensional ascent trajectories is searched combining together the dynamic model presented in section 3.2 with a constant thrust assumption $T_{min} = T = T_{max}$. With this formulation is then possible to locate the launch site on the Moon surface as well as accurately specify the target orbit through its five Classical Orbital Elements

(COEs) a, e, i, Ω, ω. As pointed out in section 3.2, the true anomaly θ at injection point is left as a free parameter to be optimized while solving for the corresponding transfer trajectory.

Two optimal solutions are then computed choosing the Moon South Pole as the designed launch site. In the first case the spacecraft is injected in a polar LLO while in the second simulation an highly elliptical orbit with $i = 60°$ is targeted to further assess the capabilities of the operating tools.

The numerical values of Isp, thrust over initial weight ratio (twr) and initial spacecraft mass m_0 used in the subsequent simulations are summarized in table 4.13.

	Units	Value
Isp	s	450.0
twr	-	2.1
m_0	kg	1.0

Table 4.13: Isp, twr and initial spacecraft mass for the 2D ascent trajectories

4.4.1 Polar LLO

The first three-dimensional ascent trajectory is computed as the optimal solution to inject the spacecraft in a polar LLO starting from the Moon South Pole. The initial conditions are imposed specifying the spacecraft state vector $\boldsymbol{r}(t_0), \boldsymbol{v}(t_0)$ while the target orbit is defined by its first five COEs. The final boundary conditions are then imposed on the two vectors $\boldsymbol{h}, \boldsymbol{e}$ as explained in section 3.2. The corresponding numerical values are given in tables 4.14 and 4.15.

	Units	Value
$\boldsymbol{r}_0$	m	$[0, 0, -1737.4 \cdot 10^3]$
$\boldsymbol{v}_0$	m/s	$[0, 0, 0]$

Table 4.14: Initial state vector for the 3D ascent trajectories

a	e	i	Ω	ω	$\theta(guess)$
$1824.27 \cdot 10^3\ m$	0	90°	0°	270°	10°

Table 4.15: COEs for target polar LLO

The required initial guess is one more time computed as a linear interpolation of the boundaries, thus requiring $\boldsymbol{r}, \boldsymbol{v}$ and $\hat{u}$ to be provided at both endpoints. The initial state vector is specified in table 4.14, while $\boldsymbol{r}(t_f), \boldsymbol{v}(t_f)$ are computed from the target COEs given in table 4.15 where θ at injection is guessed depending both on the target orbit and the controls boundaries [4]. Similarly, depending on those boundaries an initial guess on the thrust direction $\hat{u}(t_0), \hat{u}(t_f)$ is provided and then linearly interpolated throughout the whole phase. The corresponding numerical values are given in table 4.16 together with the initial guess on the time of flight.

	Value (guess)
$\hat{u}(t_0)$	$[0.2, 0.0, -0.8]$
$\hat{u}(t_f)$	$[0.5, 0.0, 0.2]$
t_f	$500\ s$

Table 4.16: Thrust direction and time of flight initial guess for the 3D ascent trajectory to a polar LLO

The optimal solution is obtained applying a Gauss-Lobatto transcription method with 10 segments and a third-order interpolating polynomial while the corresponding NLP is handily solved by the IPOPT routine. The obtained results are depicted in figure 4.8 and 4.9 while the required time of flight and propellant fraction are reported in table 4.17.

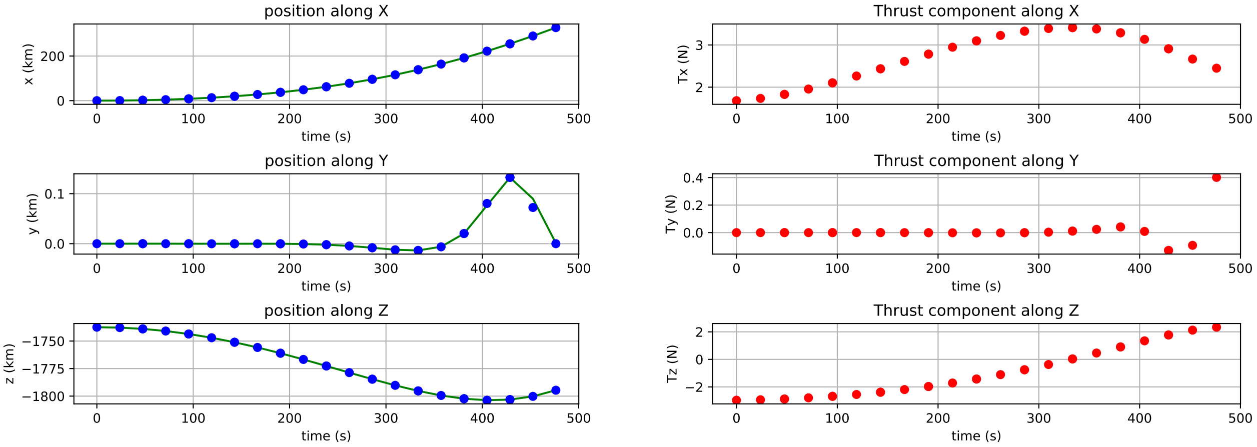

Figure 4.8: Position and thrust profiles for the 3D ascent trajectory to a polar LLO

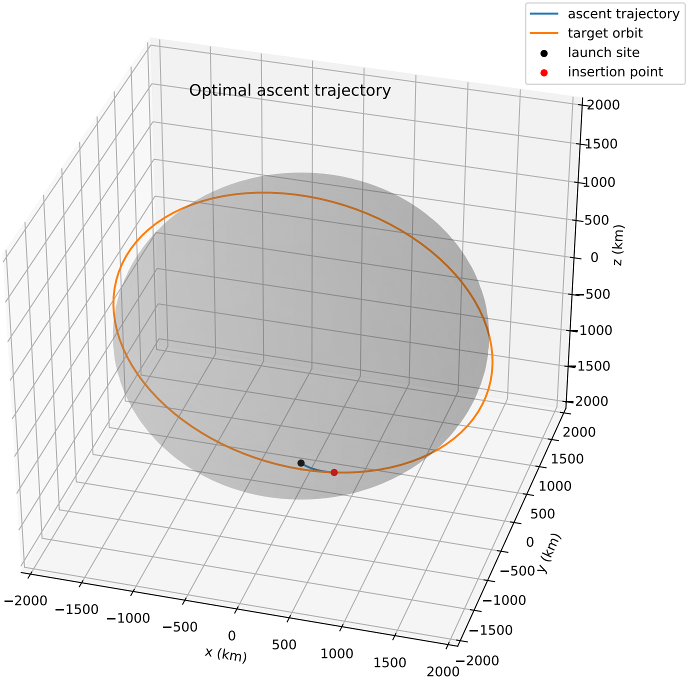

Figure 4.9: 3D ascent trajectory to a polar LLO

	Units	Value
tof	s	476.29
m_p/m_0	-	0.3681

Table 4.17: Tof and propellant fraction for the 3D ascent trajectory to a polar LLO

In the above solution the spacecraft performs only in-plane manoeuvres and the numerical values of Isp, twr, initial spacecraft mass m_0 and final orbit altitude H are selected equal to the ones specified in table 4.3 for the two-dimensional scenario. As expected, the optimal time of flight and propellant fraction almost coincide, thus validating the three-dimensional model presented in section 3.2.

4.4.2 Highly elliptical orbit

In the second simulation an highly elliptical orbit is targeted such that the Moon South Pole, designed to be the launch site, does not belong to the orbit plane. Consequently, the spacecraft is forced to perform both in-plane and an out-of-plane manoeuvres during the transfer. The whole set of Classical Orbital Elements (COEs) that describes the target orbit are given in table 4.18 while the initial conditions are specified in table 4.14.

a	e	i	Ω	ω	$\theta(guess)$
$6080.90 \cdot 10^3\ m$	0.70	60°	0°	270°	10°

Table 4.18: COEs for target highly elliptical orbit

The final boundary conditions are imposed on the two vectors $\boldsymbol{h}, \boldsymbol{e}$ as described in section 3.2 and the initial guess on the states is computed as a linear interpolation between the initial and final spacecraft state vectors as presented in section 4.4.1. Finally, the thrust direction at both endpoints as well as the time of flight are guessed as given in table 4.19. Those values are then interpolated in the discrete nodes to provide a starting point for the subsequent optimization algorithm.

	Value (guess)
$\hat{u}(t_0)$	$[-0.1, -0.1, -0.5]$
$\hat{u}(t_f)$	$[0.2, 0.2, 0.2]$
t_f	$500\ s$

Table 4.19: Thrust direction and time of flight initial guess for the 3D ascent trajectory to an highly elliptical orbit

The continuous time optimal control problem is then transcribed applying a Gauss-Lobatto transcription method with 10 segments and a third-order interpolating polynomial while the resulting NLP problem is solved using the IPOPT routines. The obtained results are displayed in figures 4.10 to 4.12 while the optimal time of flight and propellant fraction are reported in table 4.20.

	Units	Value
tof	s	876.73
m_p/m_0	-	0.6776

Table 4.20: Tof and propellant fraction for the 3D ascent trajectory to an highly elliptical orbit

As expected from the theoretical background, the numerical results in table 4.20 demonstrate that a huge amount of propellant is required when the spacecraft is asked to perform an important out-of-plane manoeuvre. On the other side, figure 4.11 shows the same phenomenon pointed out in section 3.1.3 and verified in section 4.1.3. Being more specific, to optimize the fuel consumption the spacecraft is initially boosted at constant altitude to gain the required tangential velocity to reach the final orbit while the out-of-plane manoeuvre is mainly performed as far as possible from the Moon surface as demonstrated in figures 4.10 to 4.12.

Even if not practically feasible, the solution obtained in this section is used to further validate the dynamic model presented in section 3.2 and as a test bench to asses the capabilities of the operating tools.

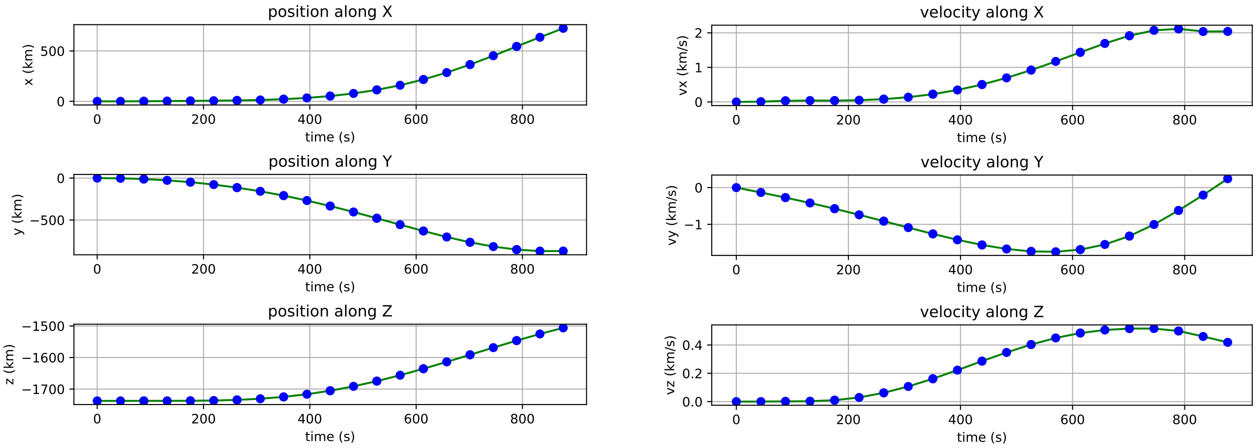

Figure 4.10: Position and velocity profiles for the 3D ascent trajectory to an highly elliptical orbit

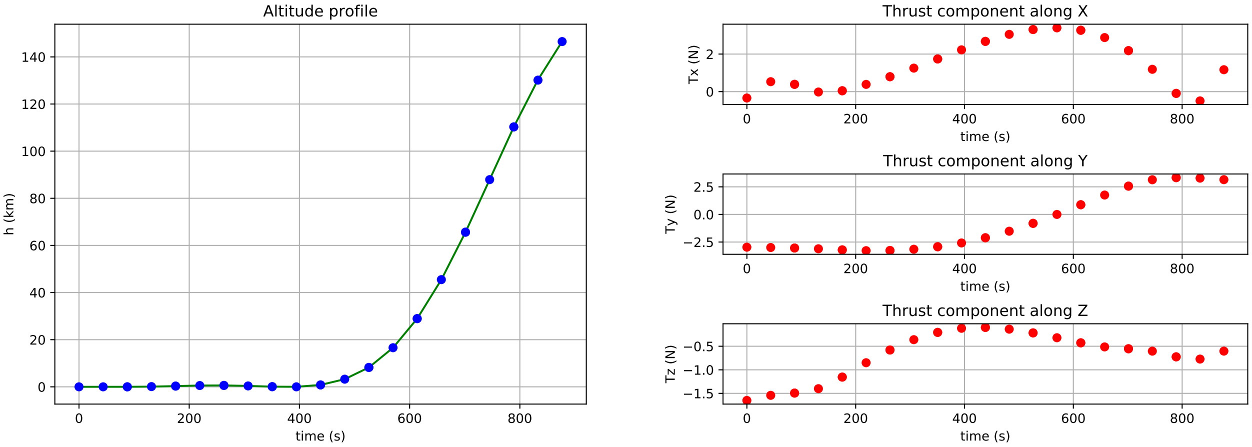

Figure 4.11: Altitude and thrust profiles for the 3D ascent trajectory to an highly elliptical orbit

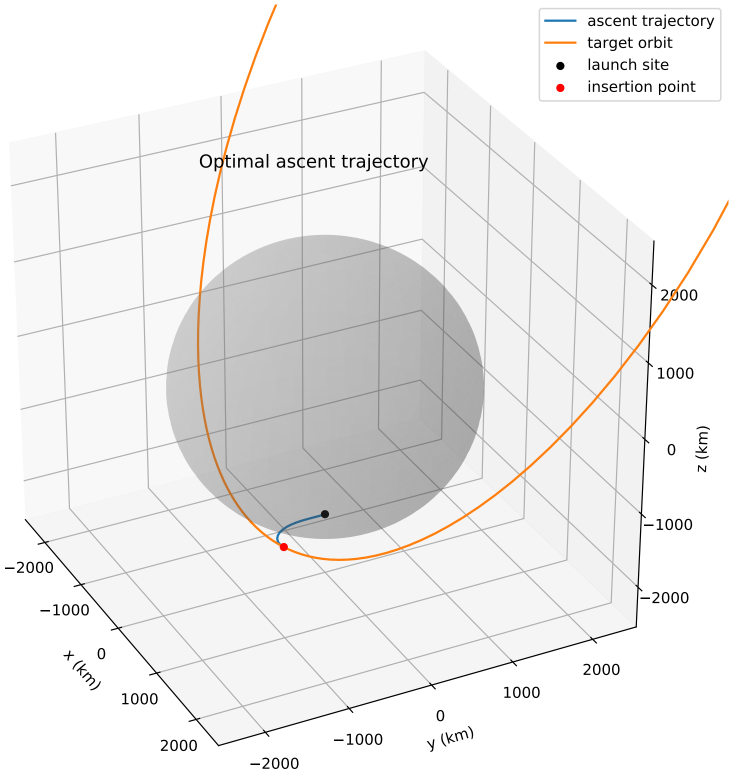

Figure 4.12: 3D ascent trajectory to an highly elliptical orbit

5 Conclusions and Perspectives

The aim of this work is to determine the most fuel-efficient transfer trajectory from the Moon surface to a Low Lunar Orbit (LLO) designed to be an intermediate parking orbit to finally reach the Near Rectilinear Halo Orbit (NRHO) in which the Lunar Orbital Platform-Gateway (LOP-G) will be placed. The study is then split into two main parts: a theoretical analysis that leads to the formulation of the optimal control problem in spaceflight dynamics and a subsequent simulation to generate a numerical solution that approximates the one of the aforementioned continuous-time optimal control problem.

After an accurate analysis of the actual State of the Art conducted in sections 2.2 the dynamic models for a two-dimensional and a three-dimensional transfer trajectories are derived in sections 3.1 and 3.2. In those sections the corresponding optimal control problem is then formulated specifying the final spacecraft mass as the objective function to be maximized. Once the problem is established, an approximate solution is searched discretizing the continuous states and controls variables as described in section 2.2.6. Finally, section 3.3 describes the numerical tools employed in this study to solve the NLP problem arisen from the aforementioned direct transcription.

Chapter 4 presents and analyzes different optimal solutions obtained for both ascent and descent trajectories with different models and constraints. In section 4.1.1 a two-dimensional ascent trajectory performed at constant thrust is derived for a specific value of Isp and thrust over initial weight ratio (twr), while in section 4.1.2 the last two values are varied on a wider interval to determine their influence on the launcher performances. As expected, the propellant consumption decreases monotonically with an increasing Isp while the optimal twr is slightly higher than 2.

Then, in section 4.1.3 the constant thrust assumption is relaxed to further decrease the required propellant to achieve the target orbit. A 3% fuel saving is obtained while the corresponding time of flight is increased by almost 8 times. Moreover, a *bang-bang* control scheme is observed as expected from the theoretical background. Since the launcher clears the ground with an angle less than 30° respect to the surface, several safety concerns arise due to the possible presence of lunar highlands around the launch site preventing the transfer to be practically feasible. To overcome this issue an appropriate path constraints is added to specify a minimum safe altitude throughout the whole phase and guarantee the avoidance of the aforementioned obstacles. The corresponding results are described in section 4.1.4 and represent a great advancement respect to previous approaches found in the literature such as the one reproduced in section 4.2 [10].

Section 4.3 replicates the results obtained by Ramanan [15] for a two-dimensional descent trajectory with an optional vertical landing aiming to further validate the dynamic models and numerical tools employed throughout this work.

Finally, the last simulations are performed adopting the three-dimensional dynamic model developed in section 3.2 to further increase the accuracy of the optimal solution specifying not only the target orbit altitude but its actual three-dimensional shape defined by the first five Classical Orbital Elements (COEs) a, e, i, Ω, ω. At the same time the launch site can be also precisely located on the Moon surface while the unit vector $\hat{u}$ replaces the thrust direction α as the second control variable together with the thrust magnitude T. After the optimization framework is set up, two solutions at constant thrust are computed targeting first a polar LLO and then an highly elliptical orbit starting from the Moon South Pole. During the ascent towards the polar LLO the spacecraft performs only in-plane manoeuvres and the numerical results resemble the ones obtained in section 4.1 since the same altitude is chosen for the final target orbit. As a consequence, these results validate the optimization framework and the dynamic model developed to handle three-dimensional transfer trajectories. The second solution is obtained forcing the spacecraft to perform continuous plane change manoeuvres throughout the whole ascent to achieve the final orbit whose plane does not include the initial launch site. The solution is handily computed and then validated performing an explicit integration with the dymos method *simulate* described in section 3.3. No solutions are found in the literature for such complex trajectories thus making this work pioneering.

Starting form the results presented in this work, several improvements are planned to further advance in this study and provide more accurate answers for the challenging design of an optimal transfer from the Moon surface to a specified LLO. Firstly, a parametric study similar to the one presented in section 4.1.2 will be conducted for a two-dimensional ascent trajectory with variable thrust. As pointed out in section 4.1.2, the availability if such information is crucial to reduce the CPU time when a Multidisciplinary Analysis (MDA) has to be performed on the complete model of the launcher due to the high computational effort required by the trajectory block alone. Moving further, a constraint analogous to the one applied in section 4.1.4 will be added to the model of a two-dimensional descent trajectory to guarantee a final vertical landing while reducing the overall propellant consumption required by the simpler approach proposed by Ramanan [15] and reproduced in section 4.3.2.

The next developments are related to the three-dimensional model derived in section 3.2 for which several opportunities are envisioned to simulate even more accurate trajectories. Firstly, the constant thrust assumption will be removed to further exploit the range of feasible solutions while searching for the most fuel efficient ascent trajectory. Secondly, an automated algorithm will be implemented for the initial guess generation thus removing the needs of a case-by-case provision of both time of flight and thrust direction $\hat{u}$ to be set as starting points for the subsequent optimization as seen in section 4.4. Thirdly, an appropriate constraint on the minimum safe altitude will be added to the problem formulation thus avoiding the phenomenon pointed out in both sections 4.1.3 and 4.4.2. Finally, appropriate boundaries on the thrust direction rates will be implemented to avoid sudden changes in the launcher attitude constraining its maximum pace.

After further validating the optimization framework for a three-dimensional ascent trajectory the same tools will be adapted to accommodate the needs that arise when a three-dimensional descent phase has to be simulated. Being more specific, the characterization of both the initial parking orbit through its first five orbital elements a, e, i, Ω, ω as well as the landing ellipse around the designated landing site has to be provided as the required boundaries for the optimal control problem formulation. Similarly to what has been done for the ascent, the position of the spacecraft in its parking orbit at the first deorbit burn is treated as a free parameter to be determined while solving for the most fuel efficient transfer trajectory. As a consequence, with the employed algorithms the best reciprocal position between the two transfer endpoints will be finally fixed to satisfy the optimality conditions.

The last improvement will be the development of an high fidelity dynamic model to replace the restricted two-body problem assumption made at the beginning of the current implementation and obtain a more accurate description of the spacecraft behaviour. Firstly, the spherical harmonics of the Moon gravitational potential will be included in the formulation to relax the assumption of an homogeneous central attracting body. Secondly, the gravitational pull of Earth, Sun and possibly other planets in the Solar System will be taken into account moving from a two-body to a multi-body dynamic model. Finally, other perturbations such as the solar radiation pressure will be considered to obtain an optimal transfer trajectory as close as possible to its practical implementation on a future lunar mission.

Bibliography

[1] Find minimum of constrained nonlinear multivariable function - MATLAB fmincon. URL `https://www.mathworks.com/help/optim/ug/fmincon.html?s_tid=srchtitle`.

[2] HSL, A collection of Fortran codes for large scale scientific computation. URL `http://www.hsl.rl.ac.uk/`.

[3] D. Benson. *A Gauss pseudospectral transcription for optimal control*. Thesis, Massachusetts Institute of Technology, 2005. URL `https://dspace.mit.edu/handle/1721.1/28919`.

[4] H. D. Curtis. *Orbital mechanics for engineering students*. Elsevier aerospace engineering series. Elsevier, BH, Butterworth-Heinemann is an imprint of Elsevier, Amsterdam; Boston, third edition, 2014. ISBN 978-0-08-097747-8.

[5] D. Garg, M. Patterson, C. Darby, C. Francolin, G. Huntington, W. Hager, and A. Rao. Direct Trajectory Optimization and Costate Estimation of General Optimal Control Problems Using a Radau Pseudospectral Method. In *AIAA Guidance, Navigation, and Control Conference*, Chicago, Illinois, Aug. 2009. American Institute of Aeronautics and Astronautics. ISBN 9781600869785. doi: 10.2514/6.2009-5989.

[6] P. Gill, W. Murray, and M. Saunders. SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization. *SIAM Review*, 47(1):99–131, Jan. 2005. ISSN 0036-1445. doi: 10.1137/S0036144504446096. URL `https://epubs.siam.org/doi/10.1137/S0036144504446096`.

[7] J. S. Gray, J. T. Hwang, J. R. R. A. Martins, K. T. Moore, and B. A. Naylor. OpenMDAO: an open-source framework for multidisciplinary design, analysis, and optimization. *Structural and Multidisciplinary Optimization*, 59(4):1075–1104, Apr. 2019. ISSN 1615-1488. doi: 10.1007/s00158-019-02211-z. URL `https://doi.org/10.1007/s00158-019-02211-z`.

[8] E. S. Hendricks, R. D. Falck, and J. S. Gray. Simultaneous Propulsion System and Trajectory Optimization. In *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. American Institute of Aeronautics and Astronautics, June 2017. doi: 10.2514/6.2017-4435. URL `https://arc.aiaa.org/doi/abs/10.2514/6.2017-4435`.

[9] J. M. Longuski, J. J. Guzmán, and J. E. Prussing. *Optimal Control with Aerospace Applications*. Space Technology Library. Springer-Verlag, New York, 2014. ISBN 9781461489443. URL `https://www.springer.com/fr/book/9781461489443`.

[10] L. Ma, W. Chen, Z. Song, and Z. Shao. A unified trajectory optimization framework for lunar ascent. *Advances in Engineering Software*, 94:32–45, Apr. 2016. ISSN 0965-9978. doi: 10.1016/j.advengsoft.2016.01.002. URL `http://www.sciencedirect.com/science/article/pii/S0965997816300102`.

[11] L. Ma, Z. Shao, W. Chen, X. Lv, and Z. Song. Three-Dimensional Trajectory Optimization for Lunar Ascent Using Gauss Pseudospectral Method. In *AIAA Guidance, Navigation, and Control Conference*, AIAA SciTech Forum. American Insti-

tute of Aeronautics and Astronautics, Jan. 2016. doi: 10.2514/6.2016-1372. URL `https://arc.aiaa.org/doi/10.2514/6.2016-1372`.

[12] L. Ma, K. Wang, Z. Xu, Z. Shao, Z. Song, and L. T. Biegler. Trajectory optimization for lunar rover performing vertical takeoff vertical landing maneuvers in the presence of terrain. *Acta Astronautica*, 146:289–299, May 2018. ISSN 0094-5765. doi: 10.1016/j.actaastro.2018.03.013. URL `http://www.sciencedirect.com/science/article/pii/S0094576516314114`.

[13] M. A. Patterson and A. V. Rao. GPOPS-II: A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using hp-Adaptive Gaussian Quadrature Collocation Methods and Sparse Nonlinear Programming. *ACM Trans. Math. Softw.*, 41(1):1:1–1:37, Oct. 2014. ISSN 0098-3500. doi: 10.1145/2558904. URL `http://doi.acm.org/10.1145/2558904`.

[14] R. E. Perez, P. W. Jansen, and J. R. R. A. Martins. pyOpt: a Python-based object-oriented framework for nonlinear constrained optimization. *Structural and Multidisciplinary Optimization*, 45(1):101–118, Jan. 2012. ISSN 1615-1488. doi: 10.1007/s00158-011-0666-3. URL `https://doi.org/10.1007/s00158-011-0666-3`.

[15] R. V. Ramanan and M. Lal. Analysis of optimal strategies for soft landing on the Moon from lunar parking orbits. *Journal of Earth System Science*, 114(6):807–813, Dec. 2005. ISSN 0973-774X. doi: 10.1007/BF02715967. URL `https://doi.org/10.1007/BF02715967`.

[16] N. Remesh, R. V. Ramanan, and V. R. Lalithambika. Fuel Optimum Lunar Soft Landing Trajectory Design Using Different Solution Schemes. *International Review of Aerospace Engineering (IREASE)*, 9(5):131–143–143, Oct. 2016. ISSN 2533-2279. doi: 10.15866/irease.v9i5.10119. URL `https://bit.ly/2KSZMdk`.

[17] A. Shirazi, J. Ceberio, and J. A. Lozano. Spacecraft trajectory optimization: A review of models, objectives, approaches and solutions. *Progress in Aerospace Sciences*, 102:76–98, Oct. 2018. ISSN 0376-0421. doi: 10.1016/j.paerosci.2018.07.007. URL `http://www.sciencedirect.com/science/article/pii/S0376042118300198`.

[18] F. Topputo and C. Zhang. Survey of Direct Transcription for Low-Thrust Space Trajectory Optimization with Applications. *Abstract and Applied Analysis*, 2014. doi: 10.1155/2014/851720. URL `https://www.hindawi.com/journals/aaa/2014/851720/`.

[19] A. Wilhite, R. Tolson, M. Mazur, and J. Wagner. Lunar Module Descent Mission Design. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Guidance, Navigation, and Control and Co-located Conferences. American Institute of Aeronautics and Astronautics, Aug. 2008. doi: 10.2514/6.2008-6939. URL `https://arc.aiaa.org/doi/10.2514/6.2008-6939`.

[20] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106 (1):25–57, Mar. 2006. ISSN 0025-5610, 1436-4646. doi: 10.1007/s10107-004-0559-y. URL `https://link.springer.com/article/10.1007/s10107-004-0559-y`.

[21] Z. Zhang, S. Gong, and J. Li. The fuel-optimal trajectory for finite-thrust LUNAR ASCENT. *Aerospace Science and Technology*, 39:675–684, Dec. 2014. ISSN 1270-9638. doi: 10.1016/j.ast.2014.06.011. URL `http://www.sciencedirect.com/science/article/pii/S1270963814001308`.