Optimal Lunar Landing Trajectory Design for Hybrid Engine

SAPIENZA
UNIVERSITÀ DI ROMA

Project of Optimal Control.
Prof. Daniela Iacoviello
A.A. 2020/2021

Leonardo Salustri (M1762556)
Francesco Scotti (M1758391)

SAPIENZA
UNIVERSITÀ DI ROMA

Introduction

- The lunar landing trajectory is a good example for the optimal problem because the dynamics is very simple, and the various energy-like values can be selected as a candidate for the cost function of the optimization problems.

- In general, the lunar landing is divided into two phases:

  - Deorbit burn (the Hohmann transfer method)
  - Powered Descent Phase

- There are different approach to investigate the Powered Descent Phase

  - **Continous Thruster with a two-dimensional (2D) dynamics**
  - Variable thrus level
  - Vertical lunar landing by using spacecraft rotational motions

Assumptions

- The gravity field of the moon is **uniform** through the whole path of the lunar lander, and the moon is entirely spherical. So we can apply the **two-body dynamics** to this problem.

- The moon rotates on its own axis with the constant angular velocity. Moreover the lunar parking orbit, the lunar landing trajectory, and the lunar equator are placed in the same plane, so one can easily apply the **2D dynamics**.

- The initial trajectory of the lunar parking orbit is the circular orbit, and **Hohmann transfer method** is used for the deorbit burn phase. Therefore, the **initial velocity** of the powered descent phase can be calculated.

- During the powered descent phase, the thrust of the lander is **constant** and there are no other **perturbations**.

- We assume that the lunar lander retrofired the **Impulsive thruster** to reduce the horizontal velocity rapidly at the initiated time for the powered descent phase.

Governing Equations

- ***r*** and $\boldsymbol{\phi}$ are the radial distance and the position angle from the center of the moon.
- ***u*** and ***v*** are the transverse and the radial velocity.
- **m** is the proper mass of the lander and **μ** is the standard gravitational parameter.
- $\mathbf{I_{sp}}$ is the specific impulse, while **g** is the gravitational acceleration on the Earth.
- **T** and $\boldsymbol{\beta}$ are used to express the thruster's magnitude and its phase angle.

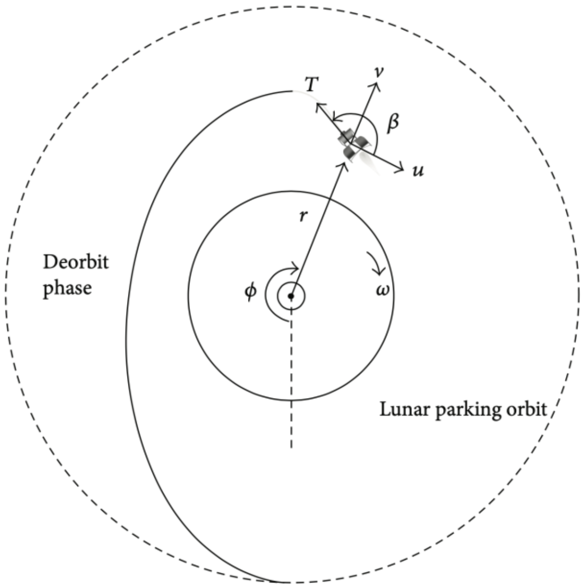
$$\dot{r} = v,$$

$$\dot{\phi} = \frac{u}{r},$$

$$\dot{u} = -\frac{uv}{r} + \frac{T}{m}\cos\beta,$$

$$\dot{v} = \frac{u^2}{r} - \frac{\mu}{r^2} + \frac{T}{m}\sin\beta,$$

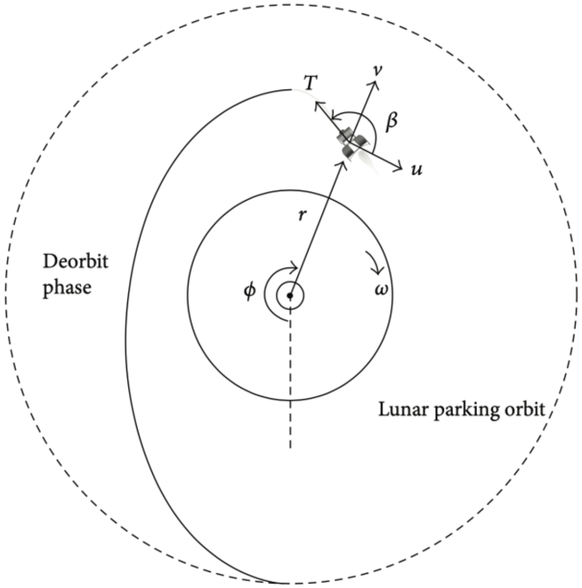
$$\dot{m} = -\frac{T}{I_{sp} g},$$



Cost Function

$$J = -m_f$$

Boundary Conditions



Hohmann transfer method is used during the deorbit phase, thus:

$$V_0 = \sqrt{\mu\left(\frac{2}{r_0} - \frac{2}{r_0 + r_p}\right)}$$

The impulsive thruster retrofired at the initial time of the powered descent phase. Therefore, the initial horizontal velocity (u_0) is not fixed and it is one of the optimal parameters.

However, there is a relationship between the initial horizontal velocity and the initial mass.

$$m_0 = M \exp\left(\frac{u_0 - V_0}{I_{\text{sp}} g}\right)$$

Boundary Conditions

The initial states of the lunar lander at the initiated time of the powered descent phase and the final state constraints can be described as follows:

$$
\begin{aligned}
r_0 &= 15 + r_{\text{moon}}, & r_f &= r_{\text{moon}}, \\
\phi_0 &= 180^\circ, & \phi_f &= \text{free}, \\
u_0 &= \text{unkown}, & u_f &= r_{\text{moon}}\omega, \\
v_0 &= 0, & v_f &= 0, \\
m_0 &= m \exp\left(\frac{u_0 - V_0}{I_{\text{sp}} g}\right) & m_f &= \text{free},
\end{aligned}
$$

Optimal Trajectory Design

The Hamiltonian function is established from the system dynamics and cost function in the previous slides as follows:

$$H = \lambda_r \dot{r} + \lambda_\phi \dot{\phi} + \lambda_u \dot{u} + \lambda_v \dot{v} + \lambda_m \dot{m}$$

where λ_r, λ_ϕ, λ_u, λ_v and λ_m represent the costate variables for each state variable.

$$\Downarrow$$

$$\dot{\lambda} = -H_x^T$$

SAPIENZA
UNIVERSITÀ DI ROMA

Optimal Trajectory Design

$$\dot{\lambda}_r = \lambda_\phi \frac{u}{r^2} - \lambda_u \frac{uv}{r^2} + \lambda_v \left(\frac{u^2}{r^2} - 2\frac{\mu}{r^3} \right),$$

$$\dot{\lambda}_\phi = 0,$$

$$\dot{\lambda}_u = -\frac{\lambda_\phi}{r} + \lambda_u \frac{v}{r} - \lambda_v \frac{2u}{r},$$

$$\dot{\lambda}_v = -\lambda_r + \lambda_u \frac{u}{r},$$

$$\dot{\lambda}_m = \frac{T}{m^2} \left(\lambda_u \cos\beta + \lambda_v \sin\beta \right).$$

Then, the control command can also be obtained from the optimal control theory:

$$\frac{\partial H}{\partial \beta} = -\frac{T}{m} \left(\lambda_u \sin\beta - \lambda_v \cos\beta \right) = 0$$

$$\frac{\partial^2 H}{\partial \beta^2} = -\frac{T}{m} \left(\lambda_u \cos\beta + \lambda_v \sin\beta \right) > 0$$

$$\Downarrow$$

$$\boxed{\beta = \tan^{-1}\left(\frac{-\lambda_v}{-\lambda_u} \right)}$$

Optimal Trajectory Design

The augumented constraint function is written as follows:

$$G = \underbrace{-m_f}_{\text{Perfomance}} + \nu^T \underbrace{\psi\left(t_f, x_f\right)}_{\substack{\text{Final State}\\ \text{Constraints}}} + \xi^T \underbrace{\sigma\left(t_0, x_0\right)}_{\substack{\text{Initial State}\\ \text{Constraints}}}$$

Where the final and initial state constraints matrices are described as follows:

$$\psi = \left[r_f - r_{\text{moon}} \quad u_f - r_{\text{moon}}\omega \quad v_f\right]^T \qquad \sigma = \begin{bmatrix} r_0 - (15 + r_{\text{moon}}) \\ \phi_0 - 180^\circ \\ v_0 \\ m_0 - M \exp\left(\dfrac{u_0 - V_0}{I_{\text{sp}} g}\right) \end{bmatrix}$$

Optimal Trajectory Design

So, the boundary conditions for the costate and the Hamiltonian are:

$$H_f = -G_{t_f}$$

$$\lambda_f = G_{x_f}^T,$$

$$\lambda_0 = -G_{x_0}^T$$

$$H_f = 0,$$

$$\lambda_\phi\left(t_f\right) = 0,$$

$$\lambda_m\left(t_f\right) = -1,$$

$$\lambda_u(0) + \lambda_m(0)\frac{M}{I_{sp}g}\exp\left(\frac{u_0 - V_0}{I_{sp}g}\right) = 0.$$

Two-point boundary value problem

- This optimal control problem can be reformulated to the two-point boundary value problem, and it is usually solved by using parameter optimization methods like:
  - Sequential dynamic programming
  - Evolutionary algorithm
  - Genetic algorithm
  - Particle swarm optimization
- It is usually difficult to select a suitable search boundary of the solution. Therefore, the **shooting method** is applied to find the optimal solution.

Shooting Method

For the shooting method, the constraints matrix (h) must be described at first.

$$h = \begin{bmatrix} r_0 - 15 - r_{\text{moon}} \\ \phi_0 - \pi \\ v_0 \\ m_0 - M\exp\left(\dfrac{u_0 - V_0}{I_{sp}g}\right) \\ \lambda_u(t_0) + \lambda_m(t_0)\dfrac{M}{I_{sp}g}\exp\left(\dfrac{u_0 - V_0}{I_{sp}g}\right) \\ r_f - r_{\text{moon}} \\ u_f - r_{\text{moon}}\omega \\ v_f \\ \lambda_\phi(t_f) \\ \lambda_m(t_f) + 1 \\ H_f \end{bmatrix}$$

Shooting Method

The derivative of the constraint matrix is written as follows:

$$dh = \dot{h}_f dt_f + h_{z_0}\delta z_0 + h_{z_f}\delta z_f + \text{H.O.T}$$

Where $z = [x^T\ \lambda^T]^T$ represents the augumented state matrix and t_f represents the final time. Also, the subscription *0* and *f* mean the values at initial and final times.

Here the higher orders are neglected, and the equation is rewritten for only the initial augumented state matrix and final time by using the transition matrix **Φ**:

$$dh = \begin{bmatrix} \dot{h}_f & h_{z_0} + h_{z_f}\Phi_f \end{bmatrix} \begin{bmatrix} dt_f \\ \delta z_0 \end{bmatrix}$$

Shooting Method

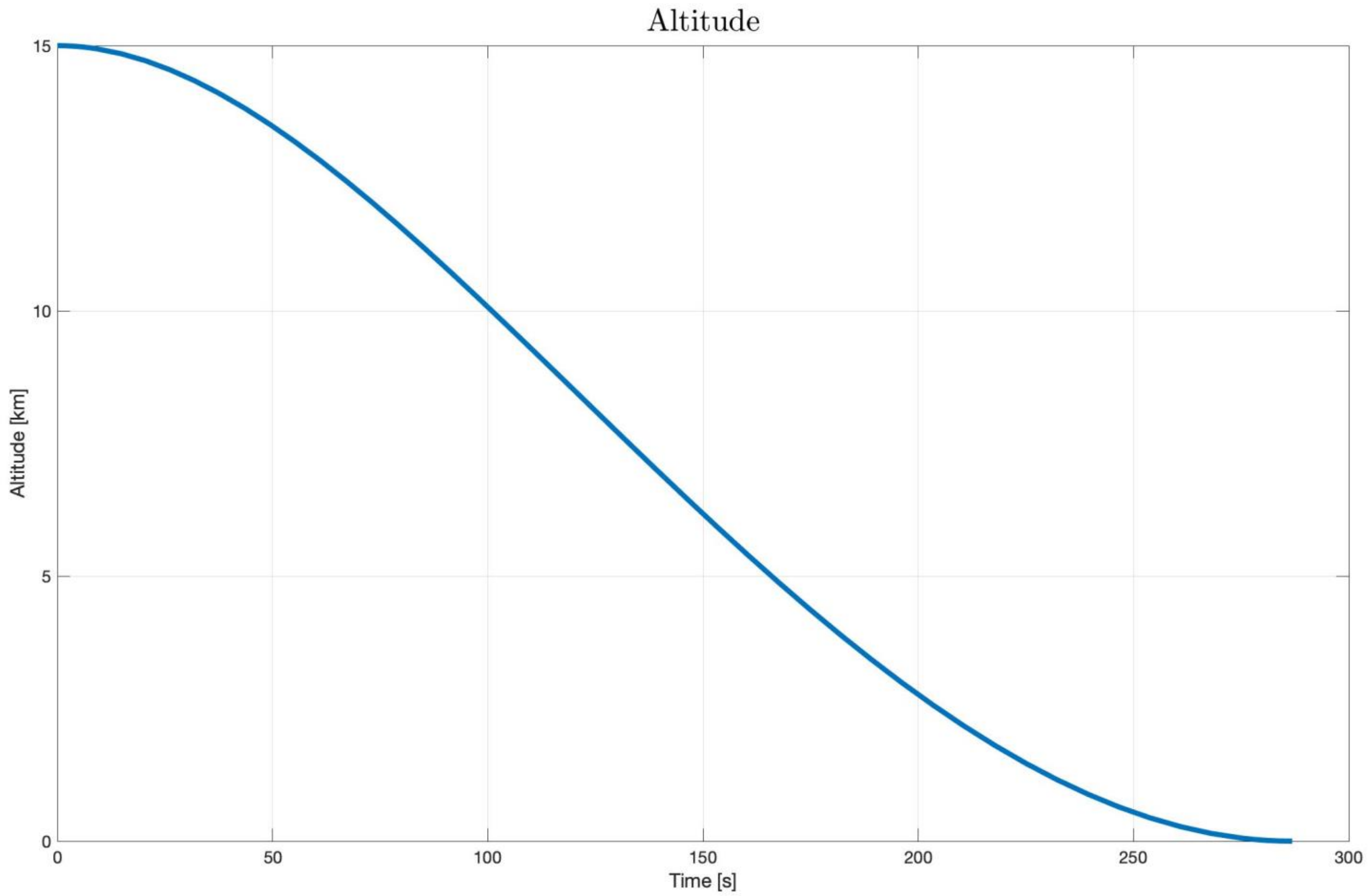
To reduce the constraints, one can choose the derivative of this constraints matrix as:

$$dh = -\alpha h, \quad 0 < \alpha \leq 1 \quad \Rightarrow \quad \begin{bmatrix} dt_f \\ \delta z_0 \end{bmatrix} = -\alpha \left[\dot{h}_f \quad h_{z_0} + h_{z_f} \Phi_f \right]^{-1} h \qquad (24)$$

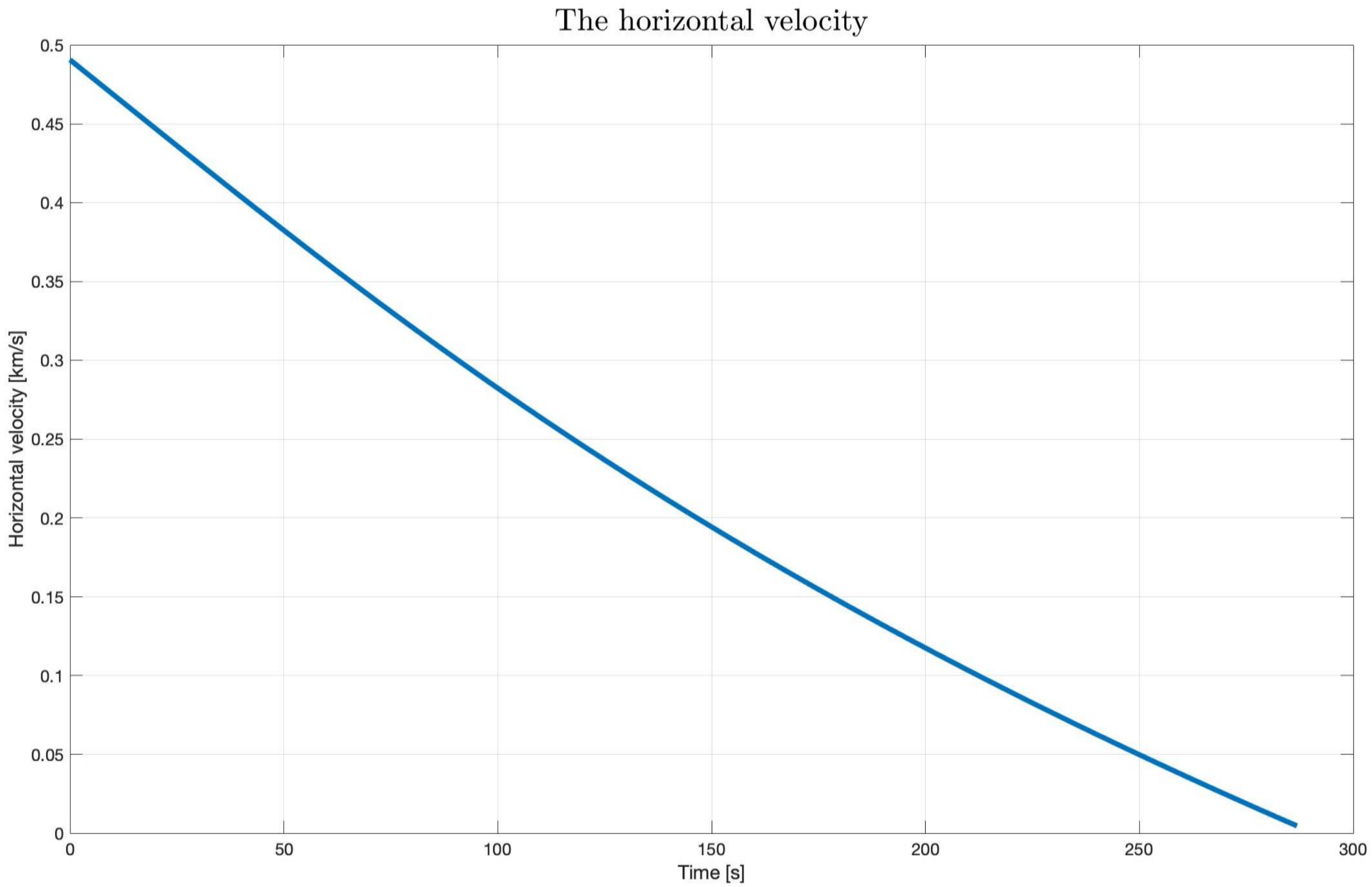
Algorithm:

- Initially guess the augumented state vector (z) and total flight time (t_f).
- Propagate the system and the costates dynamics with the optimal control law for β.
- Calculate the constraint matrix (h) and check it.
  - If ||h|| is less than tolerance value, then this shooting methos terminates.
  - If ||h|| is not less than tolerance value, then calculate the update value from (24) and add this value to the initial guessing one. Then go to the second step, with these updated values.

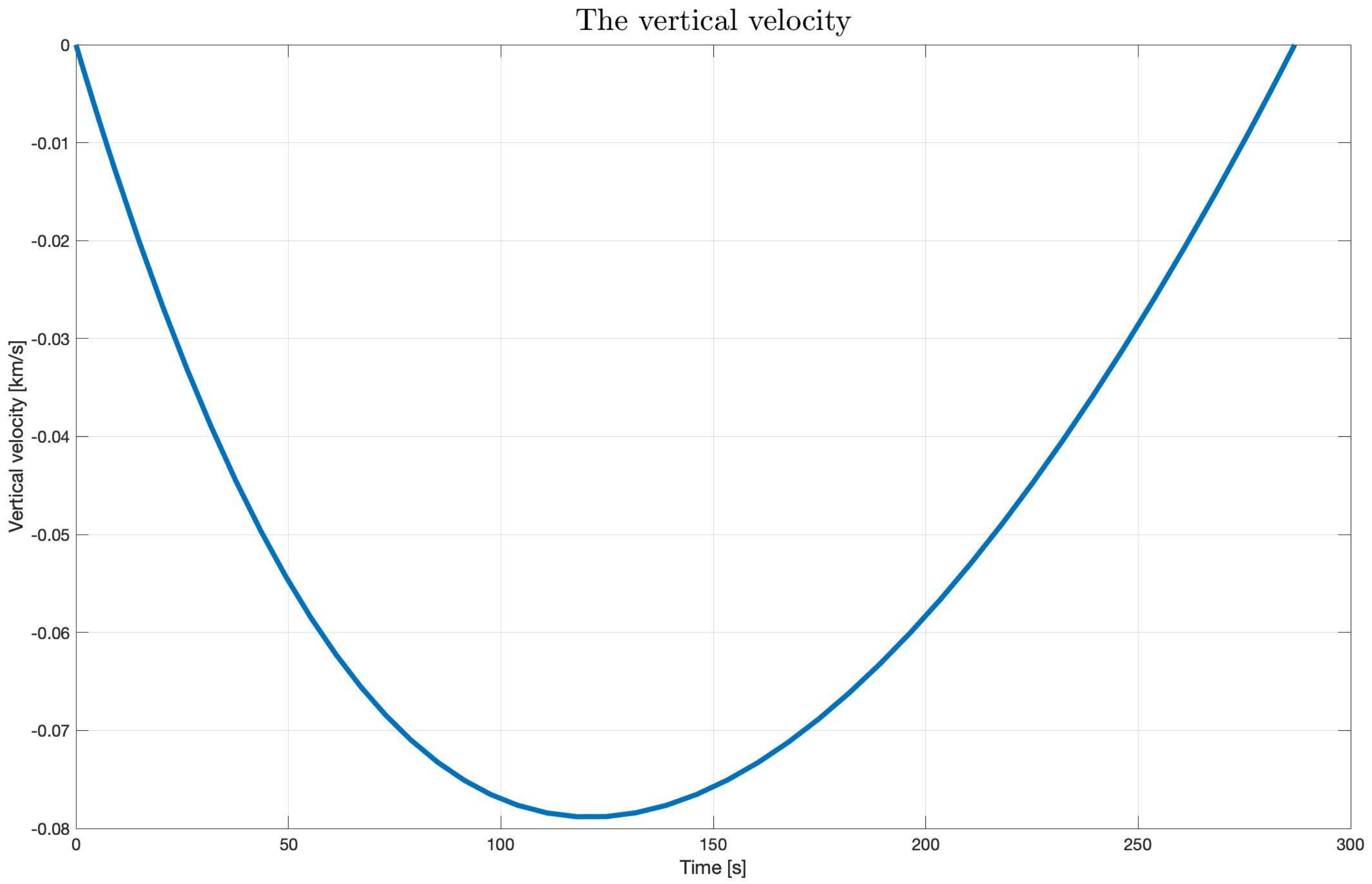
Simulations and results



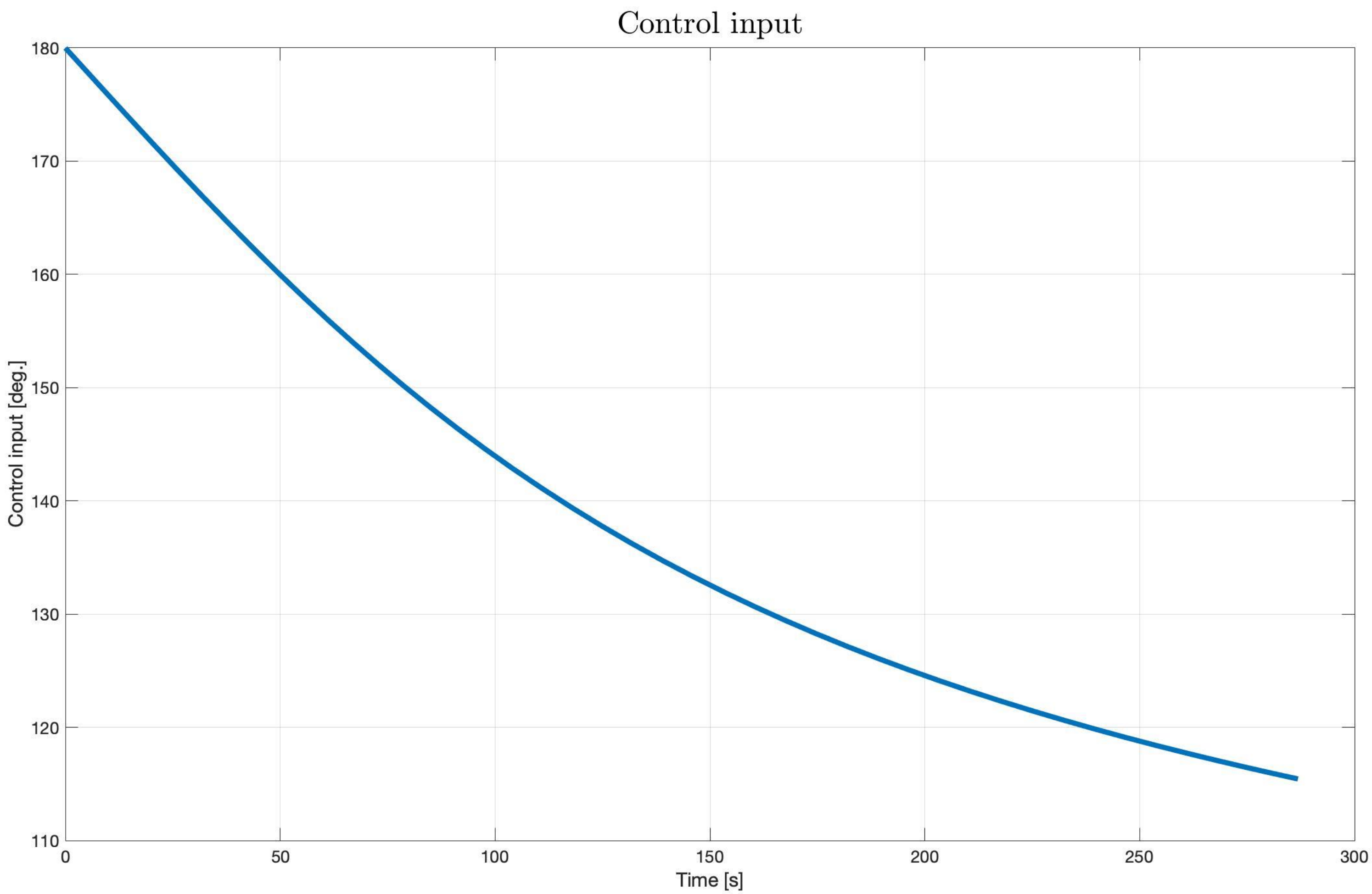
Simulations and results



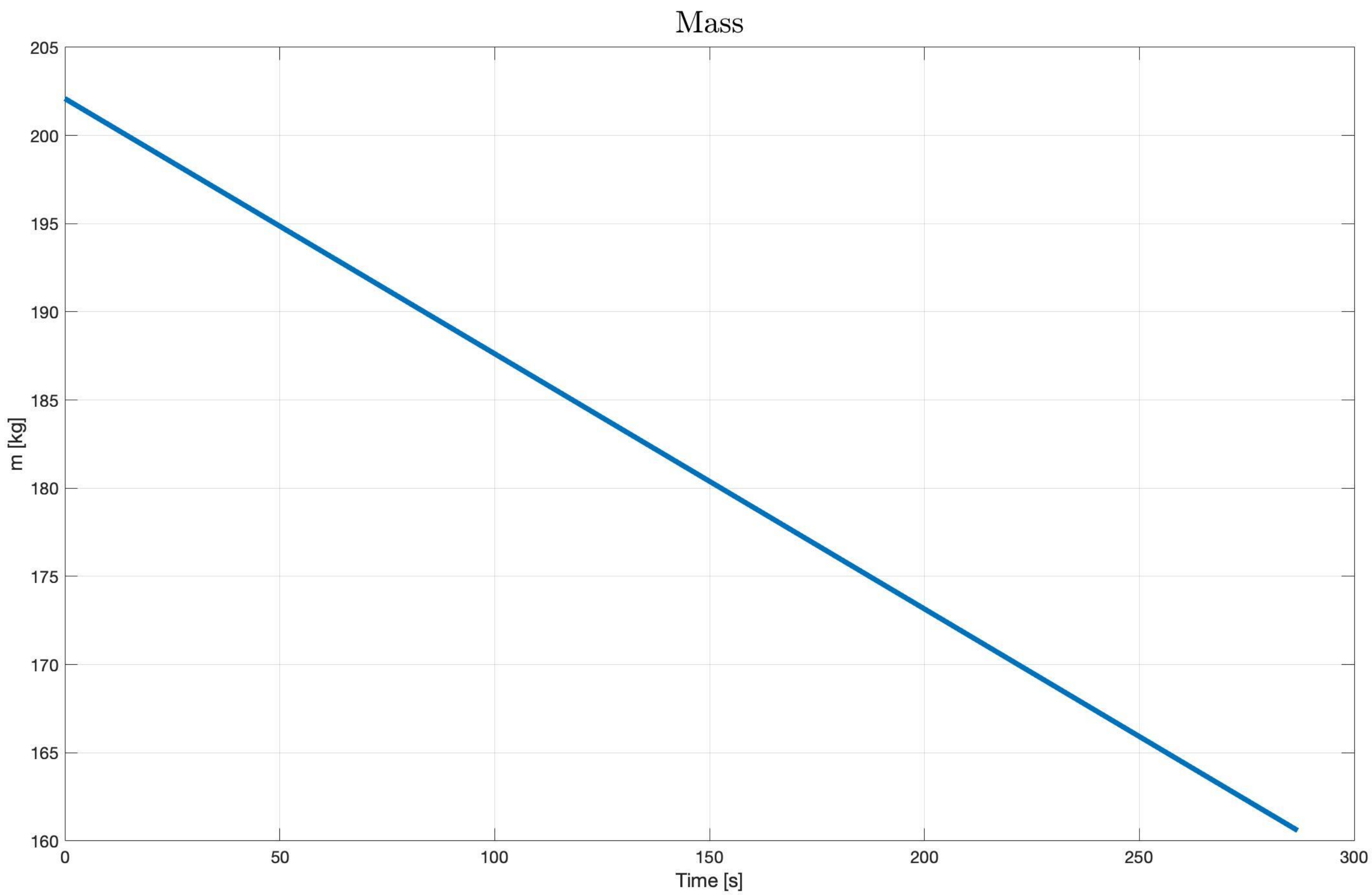
Simulations and results



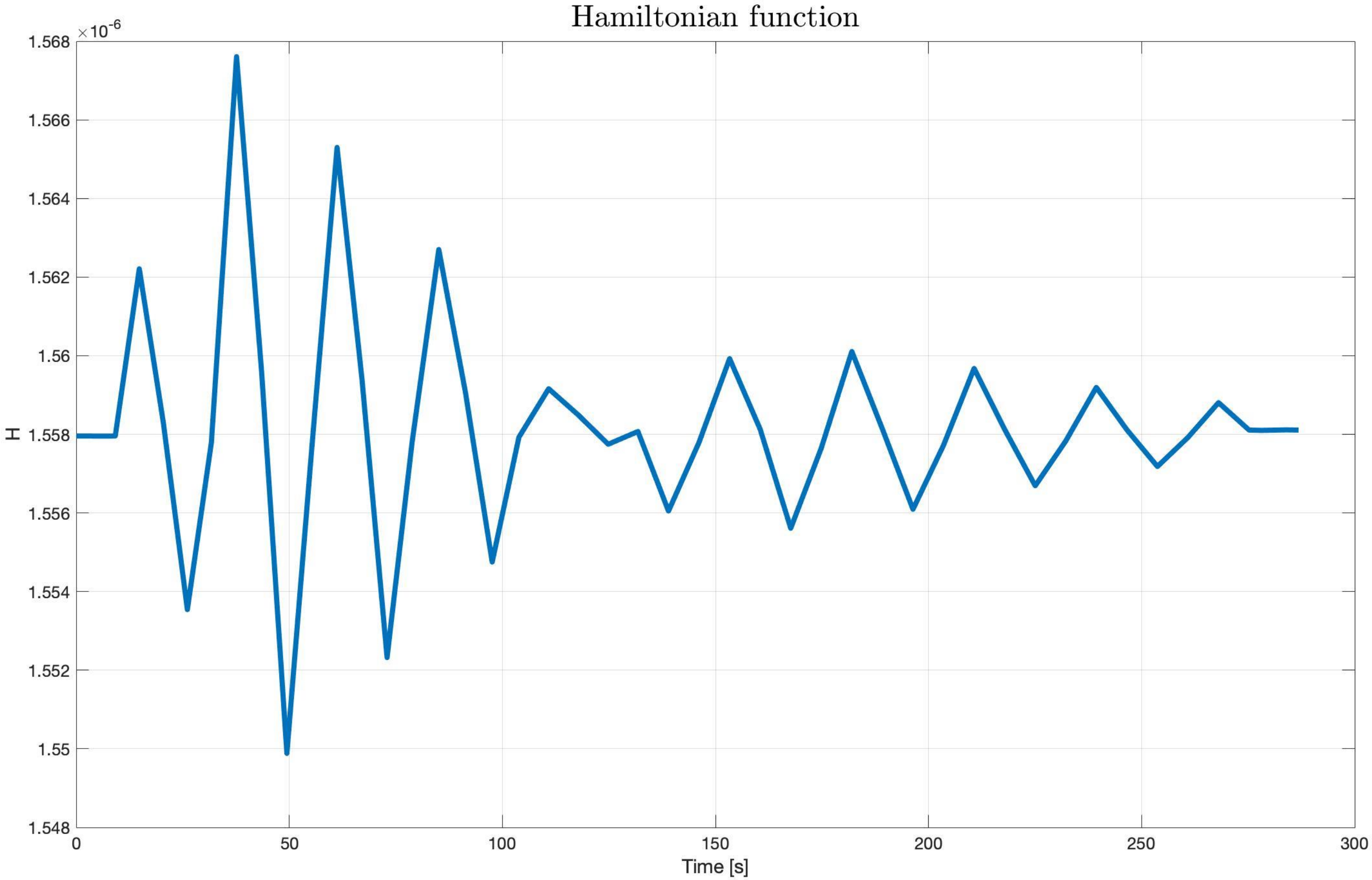
Simulations and results



Simulations and results



Simulations and results



Grazie per la Vostra Attenzione!

SAPIENZA
UNIVERSITÀ DI ROMA

Leonardo Salustri (M1762556)

Francesco Scotti (M1758391)