SGN – Assignment #1

Davide Lanza, 995599

1 Periodic orbit

Exercise 1

Consider the 3D Sun–Earth Circular Restricted Three-Body Problem with $\mu = 3.0359 \times 10^{-6}$.

1) Find the x-coordinate of the Lagrange point L_2 in the rotating, adimensional reference frame with at least 10-digit accuracy.

Solutions to the 3D CRTBP satisfy the symmetry

$$\mathcal{S} : (x, y, z, \dot{x}, \dot{y}, \dot{z}, t) \to (x, -y, z, -\dot{x}, \dot{y}, -\dot{z}, -t).$$

Thus, a trajectory that crosses perpendicularly the $y = 0$ plane twice is a periodic orbit.

2) Given the initial guess $\mathbf{x}_0 = (x_0, y_0, z_0, v_{x0}, v_{y0}, v_{z0})$, with

$$\begin{aligned} x_0 &= 1.008296144180133 \\ y_0 &= 0 \\ z_0 &= 0.001214294450297 \\ v_{x0} &= 0 \\ v_{y0} &= 0.010020975499502 \\ v_{z0} &= 0 \end{aligned}$$

Find the periodic halo orbit that passes through z_0; that is, develop the theoretical framework and implement a differential correction scheme that uses the STM either approximated through finite differences or achieved by integrating the variational equation.

The periodic orbits in the CRTBP exist in families. These can be computed by continuing the orbits along one coordinate, e.g., z_0. This is an iterative process in which one component of the state is varied, while the other components are taken from the solution of the previous iteration.

3) By gradually increasing z_0 and using numerical continuation, compute the families of halo orbits until $z_0 = 0.0046$.

(8 points)

1.1 Second Lagrange Libration Point

L_2 is one of the collinear equilibrium points in the circular restricted three-body problem. It is located along the line connecting the two primary masses. Its position was obtained by setting the gradient of the potential function to zero. Since it is a collinear point, in the rotating adimensional reference frame, it can be imposed that the y and z coordinates are zero.

$$\frac{\partial U}{\partial x}(x, 0, 0) = x - (1 - \mu)\frac{x + \mu}{|x + \mu|^3} - \mu\frac{x + \mu - 1}{|x + \mu - 1|^3} \tag{1}$$

To obtain the solution, the zero-finding problem was solved using the *fzero* Matlab function since it is a 1D problem. To ensure a 10-digit accuracy on the solution, an absolute tolerance of 10^{-10} was imposed. The initial interval for the *fzero* function was determined graphically by searching for two endpoints with opposite signs to the right of the Earth, knowing that L_2 is located in that direction.

In Fig. 1 the results are shown graphically and the second libration point vector in the Sun-Earth barycenter adimensional rotating frame is

$$\boldsymbol{L}_2 = \begin{bmatrix} 1.0100701875 \\ 0 \\ 0 \end{bmatrix} \tag{2}$$

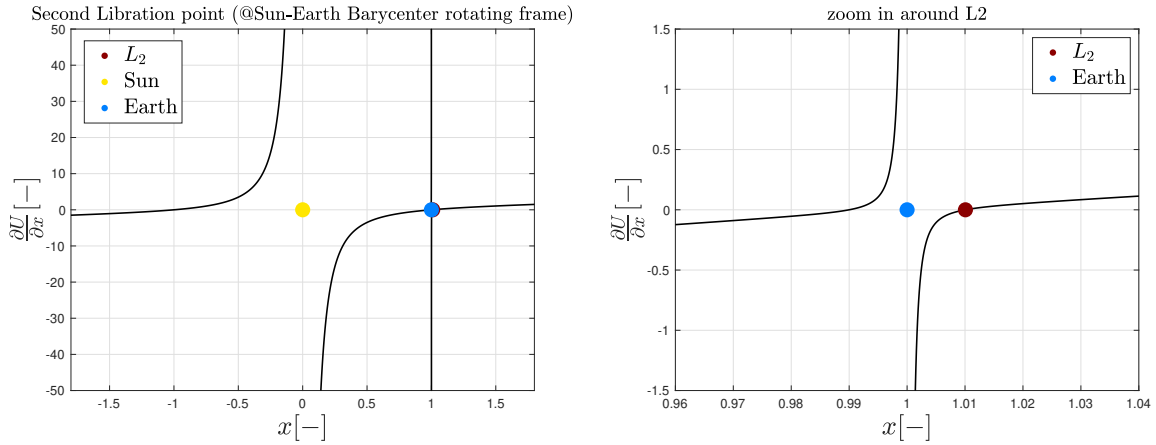


Figure 1: Second Libration Point (@Sun-Earth barycenter adimensional rotating frame).

1.2 Periodic Halo Orbit

The complete dynamics of the 3D Circular Restricted Three Body Problem (CRTBP), consisting of a system of 42 ordinary differential equations comprising 6 equations of motion and 36 variational equations, along with their initial conditions, can be described as follows:

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{f}(\mathbf{x}, t) \\ \dot{\boldsymbol{\Phi}} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} \Phi \\ \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \\ \boldsymbol{\Phi}(t_0) = I^{(36)} \end{cases} \tag{3}$$

Here, $\boldsymbol{x(t)}$ and $\boldsymbol{\Phi(t)}$ are propagated starting from the initial conditions until the event condition $y = 0$ plane is reached. To satisfy the perpendicularity condition, an iterative differential correction scheme is implemented based on the theoretical framework:

$$\phi(\boldsymbol{x}_0 + \Delta\boldsymbol{x}_0, t_e) = \phi(\boldsymbol{x}_0, t_e) + \Phi(t_e)\Delta\boldsymbol{x}_0 \tag{4}$$

In order to achieve v_x^e and v_z^e close to zero, the resulting correction given by

$$\begin{bmatrix} \Delta x_0 \\ \Delta v_{y_0} \end{bmatrix} = - \begin{bmatrix} \Phi_{41} & \Phi_{45} \\ \Phi_{61} & \Phi_{65} \end{bmatrix}^T \begin{bmatrix} v_x^e \\ v_z^e \end{bmatrix} \tag{5}$$

is iterated until the error is less then a given tolerance:

$$\text{err} = \max\left(\left| \frac{\Delta x_0}{x^e} \right|, \left| \frac{\Delta v_{y_0}}{v_y^e} \right| \right) \leq 10^{-8} \tag{6}$$

After 52 iterations, the final result shown in Fig. 2 was obtained. The complete halo orbit was reconstructed by leveraging its symmetry properties. An important observation is that this periodic orbit is around L_2, as x_0 is close to the x-coordinate of L_2, which was determined previously.

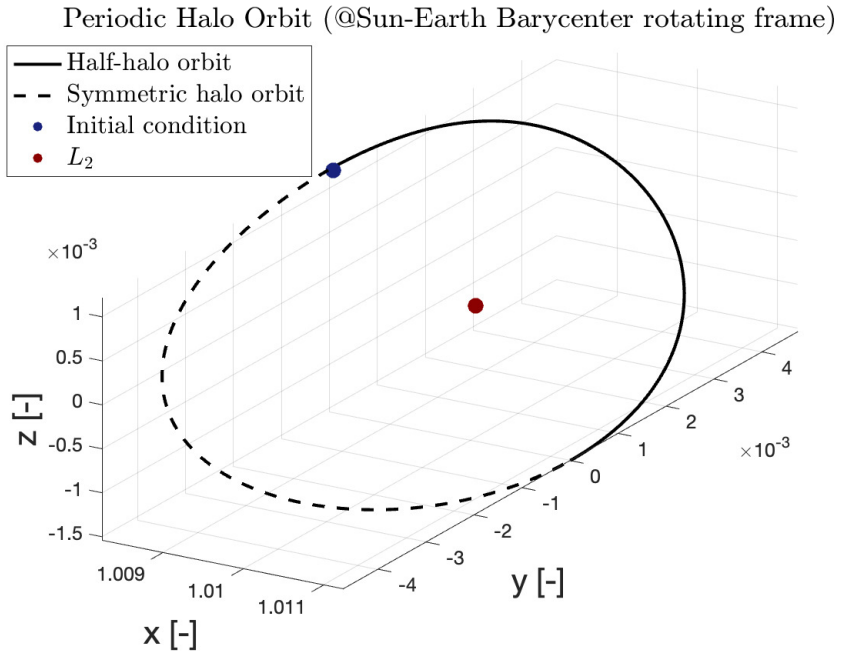
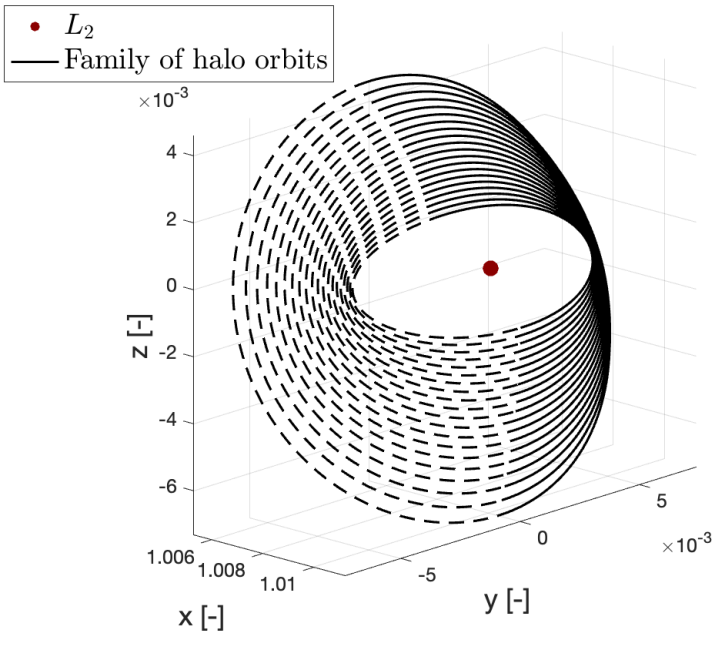


Figure 2: Periodic halo orbit passing through z_0 (@Sun-Earth barycenter adimensional rotating frame)

1.3 Family of Halo Orbits

The recovery of the family of periodic halo orbits around L_2 was done by gradually increasing z_0 until 0.0046. The other initial conditions were obtained from the solution of the previous iteration, and the differential correction scheme was applied at each step to ensure that the resulting orbit satisfied the required periodicity conditions. As a result of this iterative process, the family of periodic halo orbits was obtained, as depicted in Fig. 3.



cha

Figure 3: Family of 20 periodic halo orbits from z_0 to $z_f = 0.0046$ (@Sun-Earth barycenter adimensional rotating frame)

2 Impulsive guidance

Exercise 2

Consider the two-impulse transfer problem stated in Section 3.1 (Topputo, 2013)*.

1) Using the procedure in Section 3.2, produce a first guess solution using $\alpha = 1.5\pi$, $\beta = 1.41$, $\delta = 7$, and $t_i = 0$. Plot the solution in both the rotating frame and Earth-centered inertial frame (see Appendix 1 in (Topputo, 2013)).

2) Considering the first guess in 1) and using $\{\mathbf{x}_i, t_i, t_f\}$ as variables, solve the problem in Section 3.1 with simple shooting in the following cases

   a) without providing any derivative to the solver, and

   b) by providing the derivatives and by estimating the state transition matrix with variational equations.

3) Considering the first guess solution in 1) and the procedure in Section 3.3, solve the problem with multiple shooting taking $N = 4$ and using the variational equation to compute the Jacobian of the nonlinear equality constraints.

(11 points)

2.1 First guess solution

In order to generate a first guess solution for the Planar Bicircular Restricted Four-Body Problem (PBRFBP), the procedure described in Section 3.2 (Topputo, 2013) was followed and for the sake of consistency, the same nomenclature was used throughout the exercise. Given $\{\alpha, \beta, t_i, \delta\}$:

$$\begin{aligned} x_0 &= r_0 cos\alpha - \mu \\ y_0 &= r_0 sin\alpha \\ \dot{x}_0 &= -(v_0 - r0) sin\alpha \\ \dot{y}_0 &= (v_0 - r_0) cos\alpha \\ t_f &= t_i + \delta \end{aligned} \tag{7}$$

where $v_0 = \beta\sqrt{\frac{1-mu}{r_0}}$ and $r_0 = r_i = \frac{R_e+h_i}{DU}$.

The initial NLP variable vector is

$$\mathbf{y} = [\mathbf{x}_i, t_i, t_f] \tag{8}$$

where $\mathbf{x}_i = [x_0, y_0, \dot{x}_0, \dot{y}_0]$ is the initial state vector.

The transfer orbit was computed by numerically integrating the differential equations that govern the dynamics (PBRFBP), propagating the initial state vector from the initial to the final time. The resulting solution was plotted in two different reference frames; the transformation from the rotating frame to the Earth-centered inertial frame was performed using the procedure presented in Appendix 1 (Topputo, 2013).

The graphical analysis presented in Fig.4 indicates that a single initial guess is inadequate to accurately reach the Moon, confirming previous expectations. This underscores the importance of performing multiple iterations to converge to the optimal solution.

*F. Topputo, "On optimal two-impulse Earth–Moon transfers in a four-body model", Celestial Mechanics and Dynamical Astronomy, Vol. 117, pp. 279–313, 2013, DOI: 10.1007/s10569-013-9513-8.

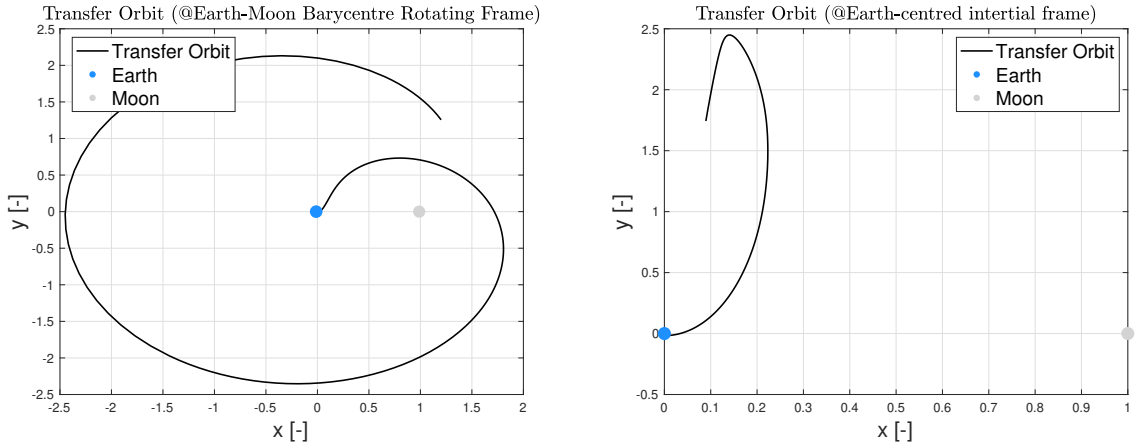


Figure 4: Earth-Moon transfer orbit with one guess solution (@Eart-Moon barycenter rotating frame and Earth-centred inertial frame). This is generated with $\alpha = 1.5\pi$, $\beta = 1.41$, $\delta = 7$, $t_i = 0$.

2.2 Simple Shooting

As a first optimization method, the Simple Shooting method was implemented to solve the problem stated in Section 3.1 (Topputo, 2013). Initially, the solver was employed without any derivative information, and subsequently, the gradient was provided to the optimization algorithm.

2.2.1 Statement of the optimization problem

Let the NLP variable vector be defined as

$$\mathbf{y} = [\mathbf{x}_i, t_i, t_f] \tag{9}$$

The initial and final transfer states $\mathbf{x}_i$ and $\mathbf{x}_f$ are at a physical distance from the Earth and the Moon, and have velocities aligned with the local circular velocity. These initial and final conditions are indicated as two-dimensional vectors valued functions (see A.1 for the extended expressions)

$$\mathbf{\Psi}_i = \mathbf{\Psi}_i(\mathbf{x}_i) = 0 \tag{10}$$

$$\mathbf{\Psi}_f = \mathbf{\Psi}_f(\mathbf{x}_f) = 0 \tag{11}$$

Then the vector of nonlinear equality constraints is

$$\mathbf{c}(\mathbf{y}) = [\mathbf{\Psi}_i, \mathbf{\Psi}_f]^T \tag{12}$$

Additionally, the transfer duration must be strictly positive

$$\tau = t_i - t_f < 0 \tag{13}$$

Therefore the linear inequality constraint is

$$g(\mathbf{y}) = \tau \tag{14}$$

The cost for the initial and final maneuvers are

$$\Delta V_i = \sqrt{(\dot{x}_i - y_i)^2 + (\dot{y}_i + x_i + \mu)^2} - \sqrt{\frac{1-\mu}{r_i}} \tag{15}$$

$$\Delta V_f = \sqrt{(\dot{x}_f - y_f)^2 + (\dot{y}_f + x_f + \mu - 1)^2} - \sqrt{\frac{\mu}{r_f}} \tag{16}$$

where $\mathbf{x}_f = \phi(\mathbf{x}_i, t_i; t_f)$.

At this point, the optimization problem can be formalized as

$$\min_{\boldsymbol{y}} f(\boldsymbol{y}) := |\Delta V_i| + |\Delta V_f| \quad \text{s.t.} \quad \begin{cases} \mathbf{c}(\boldsymbol{y}) = 0 \\ g(\boldsymbol{y}) < 0 \end{cases}$$

The search for solutions can be further refined by setting upper and lower boundaries. In this problem, the following conditions have been selected:

$$\begin{cases} -r_0 - \mu \leq x_i \leq r_0 - \mu \\ -r_0 \leq y_i \leq r_0 \\ -\sqrt{2}v_c + r_0 \leq v_{x_i} \leq \sqrt{2}v_c - r_i \\ -\sqrt{2}v_c + r_0 \leq v_{y_i} \leq \sqrt{2}v_c - r_i \\ 0 \leq t_i \leq |\frac{2\pi}{\omega_{sun}}| \\ 0 \leq t_f \leq 23 \end{cases} \tag{17}$$

where v_c is the local circular velocity.

2.2.2 Gradient Computation

To fulfill requirement (b) of the problem, the gradients of the cost function and all the constraints had to be calculated. Providing the gradients to the *fmincon* function in MATLAB is important because it allows the optimization algorithm to make more informed decisions about the search direction and step size, leading to faster convergence and potentially better solutions. The complete formulation of the gradients and additional formulas are provided in Appendix A.1 of the report to avoid cluttering the main text.

The cost function's gradient is a 6x1 vector, as the cost function is a scalar function. It can be expressed as

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial \boldsymbol{x}_i} \\ \frac{\partial f}{\partial t_i} \\ \frac{\partial f}{\partial t_f} \end{bmatrix}_{6\times 1} \tag{18}$$

By decoupling the equality constraints vector into four scalar constraints, a Jacobian matrix of size 4x6 for the equality constraints is obtained, which can be expressed as

$$\nabla \boldsymbol{c} = \begin{bmatrix} \frac{\partial c_1}{\partial \boldsymbol{x_i}} & \frac{\partial c_2}{\partial \boldsymbol{x_i}} & \frac{\partial c_3}{\partial \boldsymbol{x_i}} & \frac{\partial c_4}{\partial \boldsymbol{x_i}} \\ \frac{\partial c_1}{\partial t_i} & \frac{\partial c_2}{\partial t_i} & \frac{\partial c_3}{\partial t_i} & \frac{\partial c_4}{\partial t_i} \\ \frac{\partial c_1}{\partial t_f} & \frac{\partial c_2}{\partial t_f} & \frac{\partial c_3}{\partial t_f} & \frac{\partial c_4}{\partial t_f} \end{bmatrix}^T_{4x6} \tag{19}$$

The inequality constraint gradient vector can be expressed as

$$\nabla g = \begin{bmatrix} \frac{\partial g}{\partial \boldsymbol{x}_i} \\ \frac{\partial g}{\partial t_i} \\ \frac{\partial g}{\partial t_f} \end{bmatrix}_{6\times 1} \tag{20}$$

As shown in Appendix A.1, the independence of certain functions from the variables results in numerous zero elements in the matrices presented above.

2.2.3 Results

To solve the optimization problem, the *fmincon* function of Matlab was used. Two cases were considered: the first without providing gradients, the second providing them. Both solutions were calculated using the same relative and absolute tolerances. In the first case, the computational times were significantly longer due to the multiple approximations that needed to be made during the optimization process. On the other hand, when gradients were provided, the optimization process was significantly faster because the derivatives provide information about the behavior of the objective function and the constraints, allowing for more efficient optimization.

	ΔV $[\frac{m}{s}]$	**ToF** [days]	**Iterations**
(a) Without derivatives	4148.9903	35.9009	120
(b) With derivatives	4309.5051	34.0160	24

Table 1: Simple Shooting: Optimization Results

The primary numerical results of the optimization process are presented in Table 1. Additionally, the plots of the transfer orbit for both solutions -with and without the provided gradient- are graphically illustrated in Fig. 5.

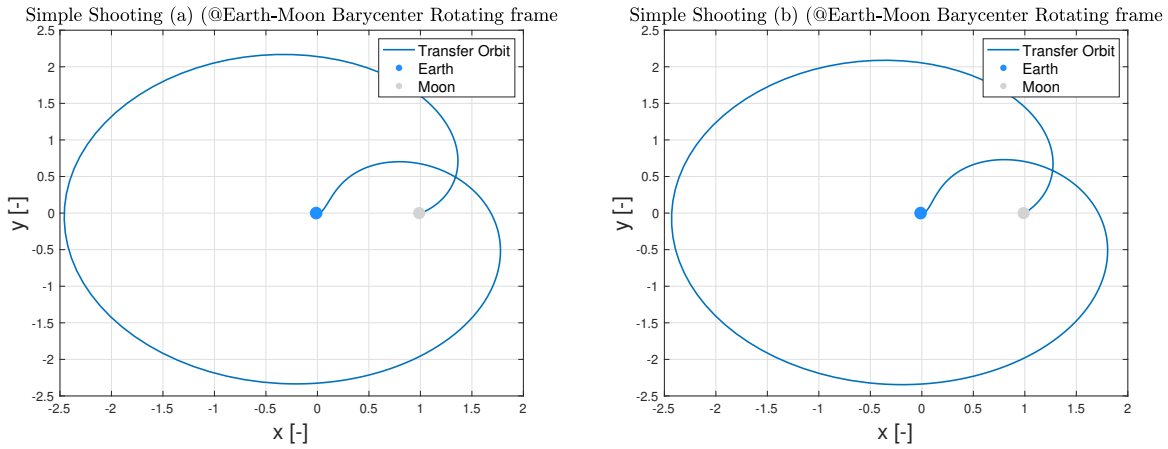


Figure 5: Simple Shooting: Earth-Moon optimal transfer orbit with and without gradient provided (@Earth-Moon barycenter rotating frame).

2.3 Multiple Shooting

The Multiple Shooting method will be used in this section, which involves the use of N=4 nodes to discretize the trajectory. This results in three distinct segments and 18 NLP variables, compared to the 6 variables used in the Simple Shooting method.

2.3.1 Statement of the optimization problem

Let the NLP variable vector be defined as

$$\boldsymbol{y} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4, t_i, t_f] \tag{21}$$

where $\boldsymbol{x}_j = [x_j, y_j, \dot{x}_j, \dot{y}_j]$ for $j = 1, \dots, 4$ represents the solution at the j-th node.

Let also create a uniform time grid which is made up by 4 points such that1 $t_1 = t_i$ and $t_4 = t_f$ and

$$t_j = t_i + \frac{j-1}{N-1}(t_f - t_i) \tag{22}$$

The continuity of position and velocity is ensured by imposing the defect constraints

$$\boldsymbol{z}_j = \boldsymbol{\phi}(\boldsymbol{x}_j, t_j; t_{j+1}) - \boldsymbol{x}_{j+1} \qquad \text{for } j = 1, \dots, N-1 \tag{23}$$

Furthermore, as in the simple shooting method, the initial and final states are located at a physical distance from the Earth and the Moon, and have velocities aligned with the local circular velocity. Therefore the resulting vector of nonlinear equality constraints is

$$\boldsymbol{c}(\boldsymbol{y}) = [\boldsymbol{\Psi}_i, \boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{z}_3, \boldsymbol{\Psi}_f] \tag{24}$$

Besides, to avoid solutions that impact the Moon or Earth, the following conditions are imposed:

$$\begin{aligned} &\left(\frac{R_e}{DU}\right)^2 - (x_j + \mu)^2 - y_j^2 < 0 \\ &\left(\frac{R_e}{DU}\right)^2 - (x_j + \mu - 1)^2 - y_j^2 < 0 \quad \text{for } j = 1, \dots, N \end{aligned} \tag{25}$$

which can be written more compactly in vector form as $\boldsymbol{\eta}_j < \mathbf{0}$. Therefore, the vector of nonlinear inequality constraints, taking into account that the time of flight must be positive, is

$$\boldsymbol{g}(\boldsymbol{y}) = [\boldsymbol{\eta}_1, \boldsymbol{\eta}_2, \boldsymbol{\eta}_3, \boldsymbol{\eta}_4, \tau] \tag{26}$$

The objective function is

$$f(\boldsymbol{y}) = \Delta V_i(\boldsymbol{x}_i) + \Delta V_f(\boldsymbol{x}_f) \tag{27}$$

Where the impulses are defined in Eqs. 15 and 16.

At this point, the optimization problem can be formalized as

$$\min_{\boldsymbol{y}} f(\boldsymbol{y}) := |\Delta V_i| + |\Delta V_f| \quad \text{s.t.} \quad \begin{cases} \mathbf{c}(\boldsymbol{y}) = 0 \\ g(\boldsymbol{y}) < 0 \end{cases}$$

The upper and lower boundaries used for the Multiple Shooting method were the same as those used for the Simple Shooting method. However, no additional limitations were imposed on the internal nodes since the continuity and defect constraints were already sufficient to ensure a feasible solution.

2.4 Gradient Computation

In the Multiple Shooting method, similar to the Simple Shooting approach, the optimization algorithm requires the gradient of the cost function and the Jacobian of the constraints. The complete formulation of the derivatives and additional formulas are provided in Appendix A.2.

The gradient of the cost function is an 18x1 vector and is given by

$$\boldsymbol{\nabla} f = \begin{bmatrix} \frac{\partial f}{\partial \boldsymbol{x}_1} \\ \vdots \\ \frac{\partial f}{\partial \boldsymbol{x}_4} \\ \frac{\partial f}{\partial t_i} \\ \frac{\partial f}{\partial t_f} \end{bmatrix}_{18\times 1}$$

By decoupling the equality constraints vector into 16 scalar constraints, a Jacobian matrix of size 18x16 for the equality constraints is obtained, which is

$$\nabla \boldsymbol{c} = \begin{bmatrix} \frac{\partial c_1}{\partial \boldsymbol{x_1}} & \dots & \dots & \frac{\partial c_{16}}{\partial \boldsymbol{x_1}} \\ \vdots & \ddots & & \vdots \\ \frac{\partial c_1}{\partial \boldsymbol{x_4}} & & & \frac{\partial c_{16}}{\partial \boldsymbol{x_4}} \\ \frac{\partial c_1}{\partial t_i} & & & \frac{\partial c_{16}}{\partial t_i} \\ \frac{\partial c_1}{\partial t_f} & \dots & \dots & \frac{\partial c_{16}}{\partial t_f} \end{bmatrix}^T_{16x18} \tag{28}$$

Finally, the Jacobian of nonlinear inequality constraints is given by

$$\nabla \boldsymbol{g} = \begin{bmatrix} \frac{\partial g_1}{\partial \boldsymbol{x_1}} & \dots & \dots & \frac{\partial g_9}{\partial \boldsymbol{x_1}} \\ \vdots & \ddots & & \vdots \\ \frac{\partial g_1}{\partial \boldsymbol{x_4}} & & & \frac{\partial g_{16}}{\partial \boldsymbol{x_4}} \\ \frac{\partial g_1}{\partial t_i} & & & \frac{\partial g_{16}}{\partial t_i} \\ \frac{\partial c_1}{\partial t_f} & \dots & \dots & \frac{\partial g_9}{\partial t_f} \end{bmatrix}^T_{9x18} \tag{29}$$

2.5 Results

The optimization problem was solved using the *fmincon* Matlab function. Computationally, the Multiple Shooting method is similar to the Single Shooting approach with provided gradient. Due to the subdivision into segments, Multiple Shooting is generally expected to yield more accurate results than Single Shooting, which is attributed to the nature of the method and its formulation. The table shows some numerical results of the optimization process. Fig. 6

ΔV $[\frac{m}{s}]$	**ToF** [days]	**Iterations**
4175.3411	36.3452	37

graphically illustrates the optimal transfer orbit, which was obtained by propagating the optimal states and resulted in three distinct orbit arcs.

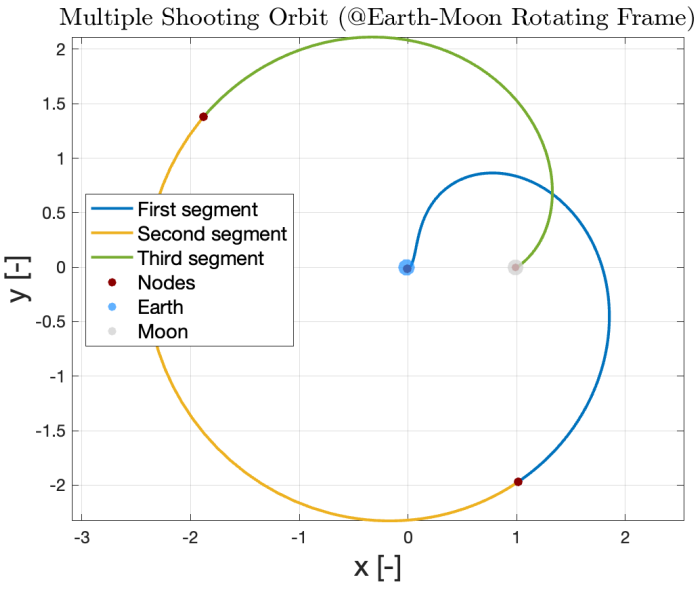


Figure 6: Multiple Shooting: Earth-Moon optimal transfer orbit (@Earth-Moon barycenter rotating frame).

3 Continuous guidance

Exercise 3

A low-thrust option is being considered for an Earth-Mars transfer[†]. Provide a *time-optimal* solution under the following assumptions: the spacecraft moves in the heliocentric two-body problem, Mars instantaneous acceleration is determined only by the Sun's gravitational attraction, the departure date is fixed, and the spacecraft initial and final states are coincident with those of the Earth and Mars, respectively.

1) Write down the spacecraft equations of motion, the costate dynamics, and the zero-finding problem for the unknowns $\{\boldsymbol{\lambda}_0, t_f\}$ with the appropriate transversality condition.

2) Solve the problem considering the following data:

   - Launch date: `2022-08-03-12:45:20.000 UTC`
   - Spacecraft mass: $m_0 = 1500$ kg
   - Electric properties: $T_{\max} = 150$ mN, $I_{sp} = 3000$ s
   - Number of thrusters: 4

   Report the obtained solution in terms of $\{\boldsymbol{\lambda}_0, t_f\}$ and the error with respect to the target. Validate your results exploiting the properties of time-optimal solutions.

3) Solve the problem for a degraded configuration with only 3 thrusters available, assuming that the failure occurs immediately after launch. Plot the thrust angles and compare them to the nominal case in 2).

(11 points)

3.1 Statement of the problem

The equations of motion for the satellite, subject to the assumptions specified in the problem description, can be represented as a system of seven first-order differential equations, expressed as follows:

$$\begin{cases} \dot{\boldsymbol{r}} = \boldsymbol{v} \\ \dot{\boldsymbol{v}} = -\frac{\mu}{r^3}\boldsymbol{r} + u\frac{T_{max}}{m}\hat{\boldsymbol{\alpha}} \\ \dot{m} = -u\frac{T_{max}}{I_{sp}g_0} \end{cases} \tag{30}$$

The initial conditions are

$$\begin{cases} \boldsymbol{r}(t_0) = \boldsymbol{r_0} \\ \boldsymbol{v}(t_0) = \boldsymbol{v_0} \\ m(t_0) = m_0 \end{cases} \tag{31}$$

To describe the dynamics of the system in a more concise manner, the state-space representation can be utilized:

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \\ \boldsymbol{x}(t_0) = \boldsymbol{x_0} \end{cases} \tag{32}$$

It should be noted that the initial conditions consist of the Earth's position and velocity in the Heliocentric reference frame at the initial time, as well as the spacecraft's initial mass. The

[†]Read the necessary gravitational constants and planets positions from SPICE. Use the kernels provided on WeBeep for this assignment.

variable u represents the throttling factor, which is equal to 1 in time optimal control problems. The unit vector $\hat{\boldsymbol{\alpha}}$ represents the thrust pointing, and T_{max} is the maximum thrust available from all of the thrusters.

The costate variables dynamics can be described by the following ordinary differential equation:

$$\dot{\boldsymbol{\lambda}} = -\frac{\partial H}{\partial \boldsymbol{x}} \tag{33}$$

Given the Hamiltonian defined in time optimal problems as:

$$H(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{u}, t) := l(\boldsymbol{x}, \boldsymbol{u}, t) + \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) = 1 + \boldsymbol{\lambda_r} \cdot \boldsymbol{v} - \frac{\mu}{r^3} \boldsymbol{r} \cdot \dot{\boldsymbol{\lambda}}_{\boldsymbol{v}} + \frac{T_{max}}{I_{sp} g_0} u \left(-\boldsymbol{\lambda_v} \frac{I_{sp} g_0}{m} - \lambda_m \right)$$

the costates dynamics are:

$$\begin{cases} \dot{\boldsymbol{\lambda}}_{\boldsymbol{r}} = -\frac{3\mu}{r^5} (\boldsymbol{r} \cdot \boldsymbol{\lambda_v}) \boldsymbol{r} + \frac{\mu}{r^3} \boldsymbol{\lambda_v} \\ \dot{\boldsymbol{\lambda}}_{\boldsymbol{v}} = -\boldsymbol{\lambda_r} \\ \dot{\lambda}_m = -u \frac{\lambda_v T_{max}}{m^2} \end{cases} \tag{34}$$

The initial conditions for the costates are:

$$\boldsymbol{\lambda}(t_0) = \boldsymbol{\lambda_0} \tag{35}$$

To conclude, the zero-finding problem for the unknowns, subject to the transversality condition, is given by:

$$\boldsymbol{F}(\boldsymbol{z}) = \begin{cases} \boldsymbol{r}(t_f) - \boldsymbol{r_M}(t_f) \\ \boldsymbol{v}(t_f) - \boldsymbol{v_M}(t_f) \\ \lambda_m(t_f) \\ H(t_f) - \boldsymbol{\lambda_r}^T \cdot v_M(t_f) - \boldsymbol{\lambda_v}^T \cdot a_M(t_f) \end{cases} \tag{36}$$

where $\mathbf{r}_M$ and $\mathbf{v}_M$ are the position and velocity vectors of Mars at the final time t_f.

3.2 Quadruple Thrusters Configuration

The problem was solved by adimensionalizing it to improve the robustness of the numerical solution. This allowed the use of the same nondimensional parameters across different unit systems without affecting the outcome.

To solve the TPBVP, the *fsolve* function in Matlab was utilized to find the zeros of the zero-finding problem. In order to achieve convergence, initial guesses were required. These guesses were generated by randomly selecting values for costates and final times within appropriate ranges. The while loop continued to generate new numbers until the error of the zero-finding problem was less than a given tolerance, as expressed by the equation:

$$error = ||F_{z_{guess}}|| \leq 10^{-9} \tag{37}$$

In Tab. 2, the resulting optimal initial guesses are reported, while in Tab. 3, the final state error with respect to the target is presented.

$\boldsymbol{\lambda}_{0,r}$	-54.9868	39.0403	1.7644
$\boldsymbol{\lambda}_{0,v}$	-67.9039	-27.3768	-0.0725
$\lambda_{0,m}$	0.0053		
t_f	2023-06-04-05:46:40 UTC		
TOF [days]	306.2085		

Table 2: Time-optimal Earth-Mars transfer solution.

$\|\|\mathbf{r}_f(t_f) - \mathbf{r}_M(t_f)\|\|$	[km]	0.0029
$\|\|\mathbf{v}_f(t_f) - \mathbf{v}_M(t_f)\|\|$	[m/s]	5.7432e-7

Table 3: Final state error with respect to Mars' center.

Subsequently, given the initial guesses, the dynamics of the system were numerically integrated between the initial and final times to propagate the transfer orbit. The resulting transfer orbit, together with the two planet orbits obtained through SPICE, are shown in Fig. 7

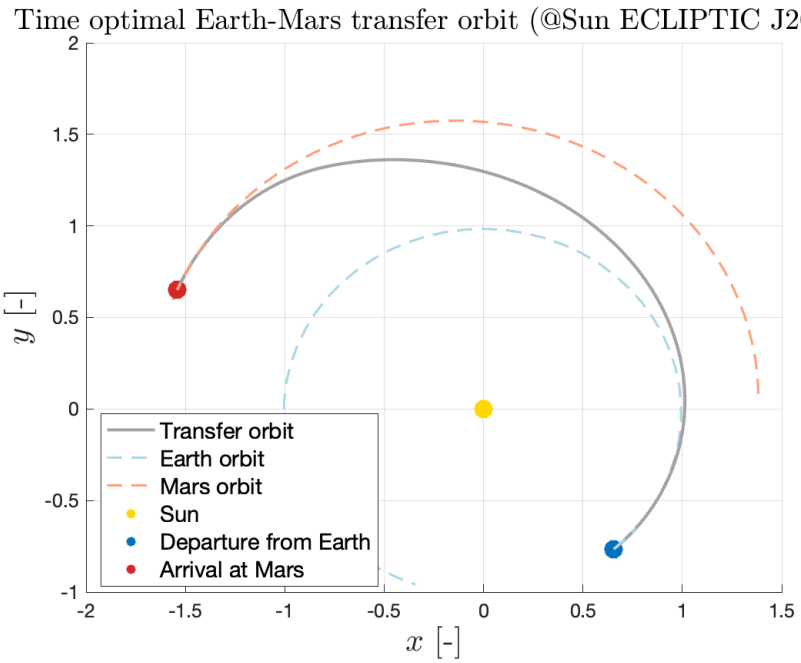


Figure 7: Time-optimal Earth-Mars transfer solution (4 thrusters configuration) (@Sun ECLIPTIC J2000)

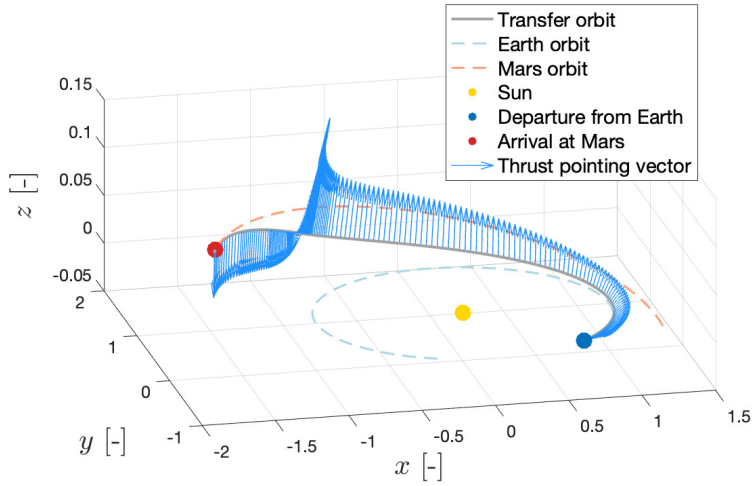


Figure 8: Time-optimal Earth-Mars transfer solution (@Sun ECLIPTIC J2000)

In the figure above, the orbits are plotted on the x-y plane, which is the dominant plane in terms of maneuvers, as the planes of both Mars and Earth's orbits are tilted only by a few degrees. However, in the Fig. 8, the z-axis has been scaled appropriately to highlight the non-planar trajectory of the satellite, which would not be easily visible on the previous scale. Additionally, the unit vectors of the thrust pointing have been added to the graph to provide more insight into the transfer trajectory.

To further validate the results obtained, the switching function and Hamiltonian were analyzed. In a time-optimal problem, it is expected that the Hamiltonian is constant as it does not directly depend on time. On the other hand, the switching function is expected to be always less than zero in the case of a time-optimal transfer. In Fig. 9 are shown the results obtained for each of these functions.

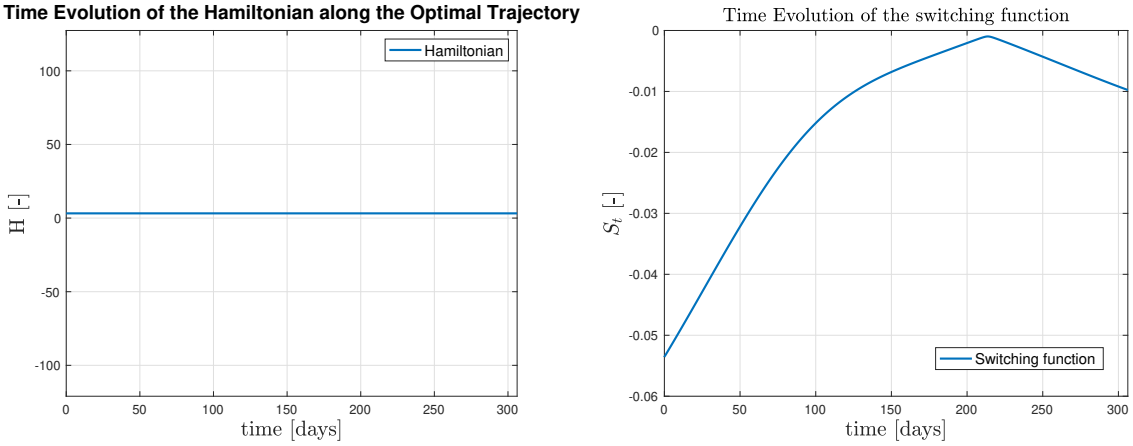


Figure 9: Time-evolution of Hamiltonian and switching function

3.3 Triple Thrusters Configuration

In this section, the resolution procedure for the triple thrusters configuration remained the same as in the previous section, with an adjustment on the range of the final time random generation since it is expected to be higher in this case, given the reduced thrust.

In Tab. 4, the resulting optimal initial guesses are reported, while in Tab. 5, the final state error with respect to the target is presented.

$\boldsymbol{\lambda}_{0,r}$	-61.1708	54.5531	0.5575
$\boldsymbol{\lambda}_{0,v}$	-80.5285	-45.8227	0.6023
$\lambda_{0,m}$	0.0072		
t_f	2023-11-19-20:17:48 UTC		
TOF [days]	474.8127		

Table 4: Time-optimal Earth-Mars transfer solution (N=3)

$\|\|\mathbf{r}_f(t_f) - \mathbf{r}_M(t_f)\|\|$	[km]	0.0764
$\|\|\mathbf{v}_f(t_f) - \mathbf{v}_M(t_f)\|\|$	[m/s]	9.2940e-6

Table 5: Final state error with respect to Mars' center(N=3)

Fig. 10 shows the graphical solution of the orbits, where it is evident that the transfer orbit is more extended than in the previous case. This is due to the reduced thrust configuration, which leads to an increase in the flight time. This also results in a lower final mass compared to the regular configuration. Specifically, the triple thrusters configuration results in a final mass of approximately 870 kg, compared to the regular configuration's final mass of approximately 960 kg.

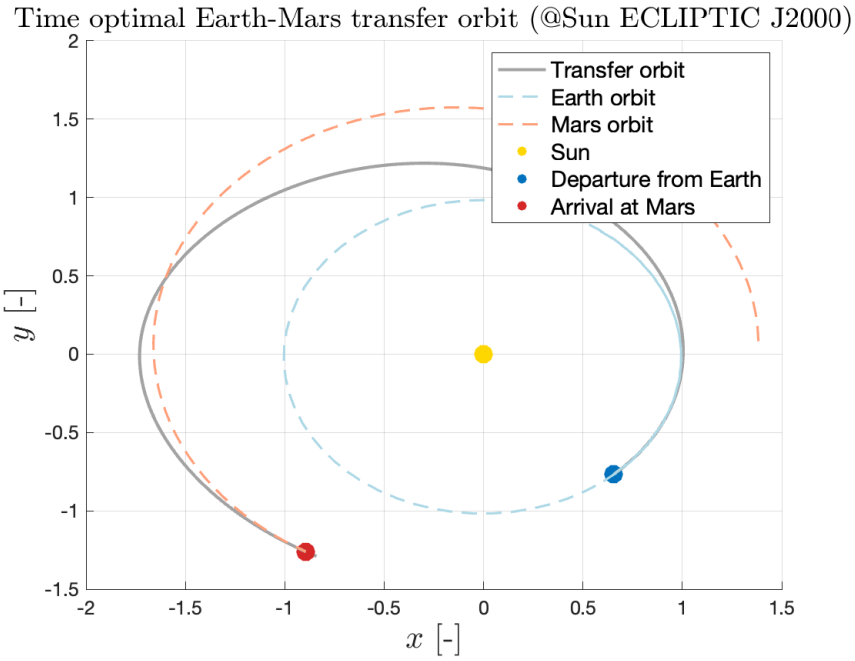


Figure 10: Time-optimal Earth-Mars transfer solution (3 thrusters configuration) (@Sun ECLIPTIC J2000)

In Fig. 11, as well as in Fig. 8, the thrust direction along the entire transfer orbit can be seen, which is a useful result that can be integrated and analyzed together with results in Fig. 12

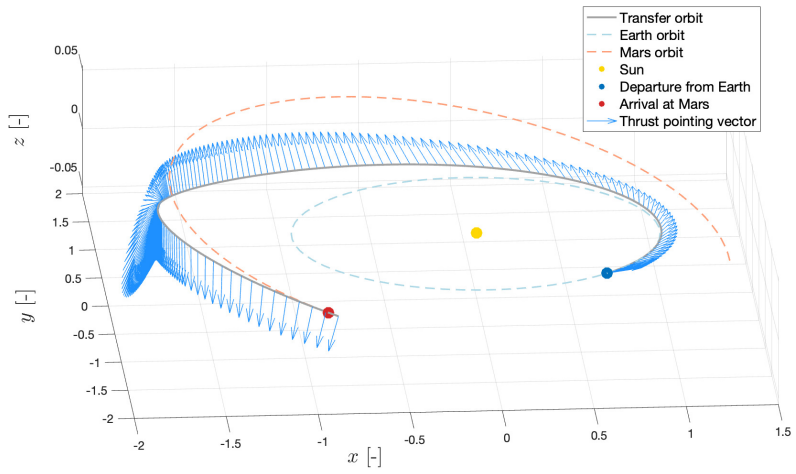


Figure 11: Time-optimal Earth-Mars transfer solution (3 thrusters configuration) (@Sun ECLIPTIC J2000)

In order to better understand the results of the thrust direction, the vector was decomposed into spherical coordinates, specifically the in-plane angle (ϕ) defined between 0 and 2π and the out-of-plane angle (θ) defined between $-\frac{\pi}{2}$ and $\frac{\pi}{2}$. Figure 12 shows the in-plane and out-of-plane angles of the thrust profile for both the regular and degraded configurations.

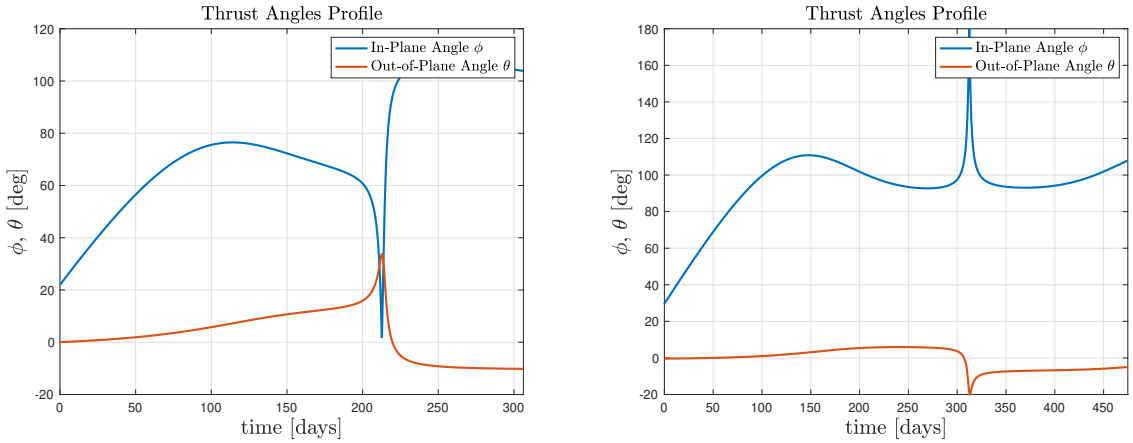


Figure 12: Comparison between thrust profile angles in the regular (left) and degraded (right) configuration.

It can be observed that a peak in the thrust profile angles is present in both configurations of the problem at the point of thrust inversion. This inversion, that occurs earlier in the regular case compared to the degraded one, is due to the need to change the satellite's orbital plane from Earth's to Mars'.

As done in the previous section, the temporal evolution of the Hamiltonian and the switching function have been plotted in Fig. 13 to validate the results of the problem. The Hamiltonian is found to be constant over time, while the switching function is always negative, as expected for a time-optimal solution.

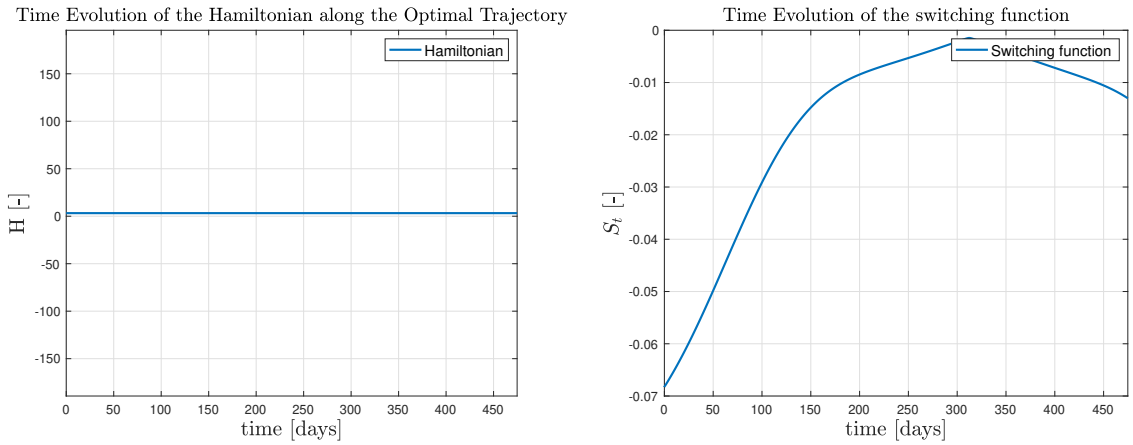


Figure 13: Time-evolution of Hamiltonian and switching function (N=3)

A Appendix: Gradient Computation - Exercise 2

A.1 Simple Shooting

The objective function is

$$f(\boldsymbol{y}) = \Delta V_i(\boldsymbol{x}_i) + \Delta V_f(\phi(\boldsymbol{x}_i, t_i; t_f)) \tag{38}$$

where

$$\begin{cases} \Delta V_i = \sqrt{(\dot{x}_i - y_i)^2 + (\dot{y}_i + x_i + \mu)^2} - \sqrt{\frac{1-\mu}{r_i}} \\ \Delta V_f = \sqrt{(\dot{x}_f - y_f)^2 + (\dot{y}_f + x_f + \mu - 1)^2} - \sqrt{\frac{\mu}{r_f}} \end{cases} \tag{39}$$

Making the following observations:

1. ΔV_1 depends only on $\boldsymbol{x}_i$
2. $\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{x}_i} = \boldsymbol{\Phi}(t_i, t_f) = \begin{pmatrix} \Phi_{rr} & \Phi_{rv} \\ \Phi_{vr} & \Phi_{vv} \end{pmatrix}$ which is a 4x4 STM
3. $\frac{\partial \phi}{\partial t_f} = f(\boldsymbol{x}(t_f), t_f)$
4. $\frac{\partial \phi}{\partial t_i} = -\boldsymbol{\Phi}(t_i, t_f) f(\boldsymbol{x}_i, t_i)$

the resulting gradient of the objective function is

$$\boldsymbol{\nabla} f = \begin{bmatrix} a \begin{pmatrix} v_{y_i} + x_i + \mu \\ y_i - v_{x_i} \\ v_{x_i} - y_i \\ v_{y_i} + x_i + \mu \end{pmatrix}_{4x1} + b\boldsymbol{\Phi}^T \begin{pmatrix} v_{y_f} + x_f + \mu - 1 \\ y_f - v_{x_f} \\ v_{x_f} - y_f \\ v_{y_f} + x_f + \mu - 1 \end{pmatrix}_{4x1} \\ -b \begin{pmatrix} v_{y_f} + x_f + \mu - 1 \\ y_f - v_{x_f} \\ v_{x_f} - y_f \\ v_{y_f} + x_f + \mu - 1 \end{pmatrix}^T_{1x4} \boldsymbol{\Phi} f(\boldsymbol{x}_i, t_i)_{4x1} \\ b \begin{pmatrix} v_{y_f} + x_f + \mu - 1 \\ y_f - v_{x_f} \\ v_{x_f} - y_f \\ v_{y_f} + x_f + \mu - 1 \end{pmatrix}_{1x4} f(\boldsymbol{x}_f, t_f)_{4x1} \end{bmatrix}_{6\times 1} \tag{40}$$

where

$$a = \frac{1}{\sqrt{(v_{x_i} - y_i)^2 + (v_{y_i} + x_i + \mu)^2}}$$

$$b = \frac{1}{\sqrt{(v_{x_f} - y_f)^2 + (v_{y_f} + x_f + \mu - 1)^2}}$$

With regards to the equality constraint, $\boldsymbol{\Psi}_i$ and $\boldsymbol{\Psi}_f$ can be explicitly written as

$$\boldsymbol{c} = \begin{bmatrix} (x_i + \mu)^2 + y_i^2 - r_i^2 \\ (x_i + \mu)(v_{x_i} - y_i) + y_i(v_{y_i} + x_i + \mu) \\ (x_f + \mu - 1)^2 + y_f^2 - r_f^2 \\ (x_f + \mu - 1)(v_{x_f} - y_f) + y_f(v_{y_f} + x_f + \mu - 1) \end{bmatrix}_{4x1} \tag{41}$$

and so the resulting Jacobian is

$$\boldsymbol{\nabla c} = \begin{bmatrix} \begin{pmatrix} 2x_i+\mu \\ 2y_i \\ 0 \\ 0 \end{pmatrix}_{4x1} & \begin{pmatrix} v_{x_i} \\ v_{y_i} \\ x_i+\mu \\ y_i \end{pmatrix}_{4x1} & \left(\begin{pmatrix} 2(x_f+\mu-1) \\ 2y_f \\ 0 \\ 0 \end{pmatrix}_{1x4}^T \boldsymbol{\Phi} \right)_{4x1}^T & \left(\begin{pmatrix} v_{x_f} \\ v_{y_f} \\ x_f+\mu-1 \\ y_f \end{pmatrix}_{1x4}^T \boldsymbol{\Phi} \right)_{4x1}^T \\ 0 & 0 & -\left(\begin{pmatrix} 2(x_f+\mu-1) \\ 2y_f \\ 0 \\ 0 \end{pmatrix}_{1x4}^T \boldsymbol{\Phi} f(\boldsymbol{x}_i, t_i)_{4x1} \right)_{1x1} & -\left(\begin{pmatrix} v_{x_f} \\ v_{y_f} \\ x_f+\mu-1 \\ y_f \end{pmatrix}_{1x4}^T \boldsymbol{\Phi} f(\boldsymbol{x}_i, t_i)_{4x1} \right)_{1x1} \\ 0 & 0 & \left(\begin{pmatrix} 2(x_f+\mu-1) \\ 2y_f \\ 0 \\ 0 \end{pmatrix}_{1x4}^T f(\boldsymbol{x}_f, t_f)_{4x1} \right)_{1x1} & \left(\begin{pmatrix} v_{x_f} \\ v_{y_f} \\ x_f+\mu-1 \\ y_f \end{pmatrix}_{1x4}^T f(\boldsymbol{x}_f, t_f)_{4x1} \right)_{1x1} \end{bmatrix}_{4x6}^T$$

Finally, the equality constraint is

$$g(t_i, t_f) = t_i - t_f \tag{42}$$

and so the resulting gradient is

$$\boldsymbol{\nabla} g = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ -1 \end{bmatrix}_{6x1} \tag{43}$$

A.2 Multiple Shooting

The gradient of the objective function is

$$\boldsymbol{\nabla} f = \begin{bmatrix} a \begin{pmatrix} v_{y_i} + x_i + \mu \\ y_i - v_{x_i} \\ v_{x_i} - y_i \\ v_{y_i} + x_i + \mu \end{pmatrix}_{4x1} \\ (\mathbf{0})_{8x1} \\ b \begin{pmatrix} v_{y_f} + x_f + \mu - 1 \\ y_f - v_{x_f} \\ v_{x_f} - y_f \\ v_{y_f} + x_f + \mu - 1 \end{pmatrix}_{4x1} \\ 0 \\ 0 \end{bmatrix}_{18x1} \tag{44}$$

The vector of nonlinear equality constraints is

$$\boldsymbol{c} = \begin{bmatrix} \boldsymbol{\phi}(\boldsymbol{x}_1, t_1; t_2) - \boldsymbol{x}_2 \\ \boldsymbol{\phi}(\boldsymbol{x}_2, t_2; t_3) - \boldsymbol{x}_3 \\ \boldsymbol{\phi}(\boldsymbol{x}_3, t_3; t_4) - \boldsymbol{x}_4 \\ (x_1+\mu)^2 + y_1^2 - r_i^2 \\ (x_1+\mu)(v_{x_1} - y_1) + y_1(v_{y_1} + x_1 + \mu) \\ (x_4+\mu-1)^2 + y_4^2 - r_f^2 \\ (x_4+\mu-1)(v_{x_4} - y_4) + y_4(v_{y_4} + x_4 + \mu - 1) \end{bmatrix}_{16x1} \tag{45}$$

and so its Jacobian matrix is

$$\nabla \boldsymbol{c} = \begin{bmatrix} (\boldsymbol{\Phi}_1)_{4x4} & (-\boldsymbol{I})_{4x4} & (\boldsymbol{0})_{4x4} & (\boldsymbol{0})_{4x4} & \left(\frac{\partial \boldsymbol{c}_1}{\partial t_1}\right)_{4x1} & \left(\frac{\partial \boldsymbol{c}_1}{\partial t_4}\right)_{4x1} \\ (\boldsymbol{0})_{4x4} & (\boldsymbol{\Phi}_2)_{4x4} & (-\boldsymbol{I})_{4x4} & (\boldsymbol{0})_{4x4} & \left(\frac{\partial \boldsymbol{c}_2}{\partial t_1}\right)_{4x1} & \left(\frac{\partial \boldsymbol{c}_2}{\partial t_4}\right)_{4x1} \\ (\boldsymbol{0})_{4x4} & (\boldsymbol{0})_{4x4} & (\boldsymbol{\Phi}_3)_{4x4} & (-\boldsymbol{I})_{4x4} & \left(\frac{\partial \boldsymbol{c}_3}{\partial t_1}\right)_{4x1} & \left(\frac{\partial \boldsymbol{c}_3}{\partial t_4}\right)_{4x1} \\ \begin{pmatrix} 2(x_i+\mu) & 2y_i & 0 & 0 \\ v_{x_i} & v_{y_i} & x_i+\mu & y_i \end{pmatrix}_{2x4} & (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x1} & (\boldsymbol{0})_{2x1} \\ (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x4} & \begin{pmatrix} 2(x_f+\mu-1) & 2y_f & 0 & 0 \\ v_{x_f} & v_{y_f} & x_f+\mu-1 & y_f \end{pmatrix}_{2x4} & (\boldsymbol{0})_{2x1} & (\boldsymbol{0})_{2x1} \end{bmatrix}_{16x18}$$

where

$$\boldsymbol{\Phi}_i = \boldsymbol{\Phi}(t_i, t_{i+1}) \qquad \text{for } i = 1, \dots, N-1$$

$$\frac{\partial \boldsymbol{c}_i}{\partial t_1} = -\boldsymbol{\Phi}(t_i, t_{i+1}) f(\boldsymbol{x}_i, t_i) \frac{N-i}{N-1} + f(\phi(\boldsymbol{x}_i, t_i, t_{i+1}), t_{i+1}) \frac{N-i+1}{N-1} \qquad \text{for } i = 1, \dots, N-1$$

$$\frac{\partial \boldsymbol{c}_i}{\partial t_4} = -\boldsymbol{\Phi}(t_i, t_{i+1}) f(\boldsymbol{x}_i, t_i) \frac{i-1}{N-1} + f(\phi(\boldsymbol{x}_i, t_i, t_{i+1}), t_{i+1}) \frac{i}{N-1} \qquad \text{for } i = 1, \dots, N-1$$

The vector of nonlinear inequality constraints is

$$\boldsymbol{g} = \begin{bmatrix} \left(\frac{R_e}{DU}\right)^2 - (x_1+\mu)^2 - y_1^2 \\ \left(\frac{R_e}{DU}\right)^2 - (x_1+\mu-1)^2 - y_1^2 \\ \left(\frac{R_e}{DU}\right)^2 - (x_2+\mu)^2 - y_2^2 \\ \left(\frac{R_e}{DU}\right)^2 - (x_2+\mu-1)^2 - y_2^2 \\ \left(\frac{R_e}{DU}\right)^2 - (x_3+\mu)^2 - y_3^2 \\ \left(\frac{R_e}{DU}\right)^2 - (x_3+\mu-1)^2 - y_3^2 \\ \left(\frac{R_e}{DU}\right)^2 - (x_4+\mu)^2 - y_4^2 \\ \left(\frac{R_e}{DU}\right)^2 - (x_4+\mu-1)^2 - y_4^2 \\ t_1 - t_4 \end{bmatrix}_{9x1} \tag{46}$$

and its Jacobian matrix is given by

$$\nabla \boldsymbol{g} = \begin{bmatrix} \left(\frac{\partial \boldsymbol{\eta}_i}{\partial \boldsymbol{x}_1}\right)_{2x4} & (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x2} \\ (\boldsymbol{0})_{2x4} & \left(\frac{\partial \boldsymbol{\eta}_i}{\partial \boldsymbol{x}_2}\right)_{2x4} & (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x2} \\ (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x4} & \left(\frac{\partial \boldsymbol{\eta}_i}{\partial \boldsymbol{x}_3}\right)_{2x4} & (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x2} \\ (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x4} & (\boldsymbol{0})_{2x4} & \left(\frac{\partial \boldsymbol{\eta}_i}{\partial \boldsymbol{x}_4}\right)_{2x4} & (\boldsymbol{0})_{2x2} \\ (\boldsymbol{0})_{1x4} & (\boldsymbol{0})_{1x4} & (\boldsymbol{0})_{1x4} & (\boldsymbol{0})_{1x4} & \begin{pmatrix} 1 & -1 \end{pmatrix}_{1x2} \end{bmatrix}_{9x18} \tag{47}$$

where

$$\frac{\partial \boldsymbol{\eta}_i}{\partial \boldsymbol{x}_1} = \begin{bmatrix} -2(x_i+\mu) & -2y_i & 0 & 0 \\ -2(x_i+\mu-1) & -2y_i & 0 & 0 \end{bmatrix} \qquad \text{for } i = 1, \dots, N$$