SGN – Assignment #1

Torre Federico, 233133

1 Periodic orbit

Exercise 1

Consider the 3D Earth–Moon Circular Restricted Three-Body Problem with $\mu = 0.012150$.

1) Find the x-coordinate of the Lagrange point L_1 in the rotating, adimensional reference frame with at least 10-digit accuracy.

Solutions to the 3D CRTBP satisfy the symmetry

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

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

$$\begin{aligned} x_0 &= 1.08892819445324 \\ y_0 &= 0 \\ z_0 &= 0.0591799623455459 \\ v_{x0} &= 0 \\ v_{y0} &= 0.257888699435051 \\ v_{z0} &= 0 \end{aligned}$$

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 decreasing z_0 and using numerical continuation, compute the families of halo orbits until $z_0 = 0.034$.

(8 points)

---

1. **x-coordinate of the Lagrange point L1**: To calculate the coordinate of the L1, the function `Lagrangianpoint` is utilized. Inside the function, the zero of the partial derivatives of the scalar potential function U is calculated as in Equation 1, where $r_1 = \sqrt{(x+\mu)^2 + y^2 + z^2}$ and $r_2 = \sqrt{(x+\mu-1)^2 + y^2 + z^2}$.

$$U = \frac{1}{2}\left(x^2 + y^2\right) + \frac{1-\mu}{r_1} + \frac{\mu}{r_2} \tag{1}$$

The studied Lagrange libration point is collinear, meaning that $x = 0$ and $z = 0$ this imply to consider only Equation 2. In detail, inside the function `Lagrangianpoint`, the zero of the potential function is calculated with the Matlab function `fzero`. When the zero is reached and the tolerance of 1×10^{-10} is achieved, the function terminates. The initial guess input function is a vector that goes from 0.1 to 1 spaced by 0.1.

$$\frac{\partial U}{\partial x} = x - \frac{(1-\mu)(x+\mu)}{[(x+\mu)^2 + y^2 + z^2]^{3/2}} - \frac{\mu(x+\mu-1)}{[(x+\mu-1)^2 + y^2 + z^2]^{3/2}} \tag{2}$$

Figure 1 illustrates the obtained Lagrangian coordinate with a precision of 10 digits.

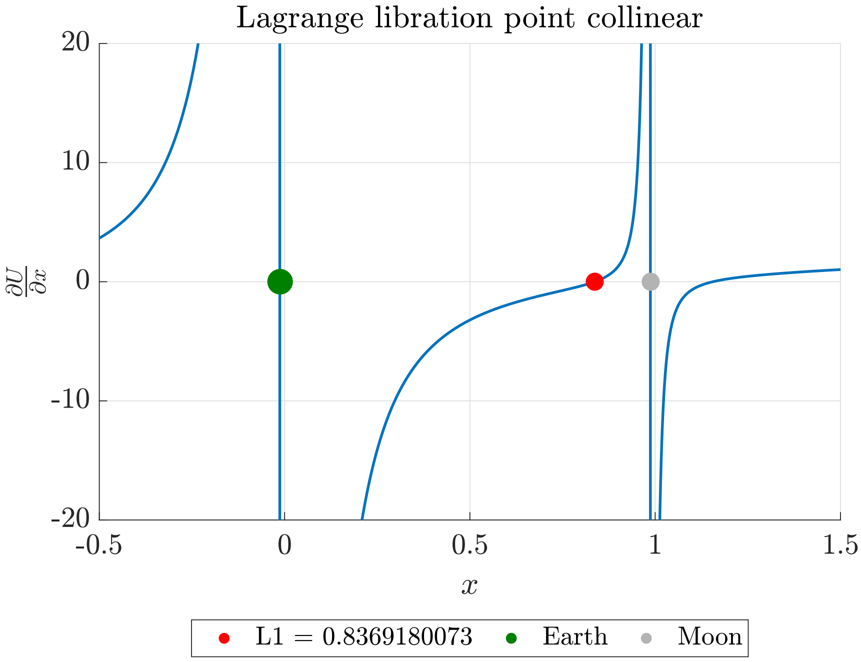

Figure 1: $\frac{\partial U}{\partial x}$ and L1.

2. **Periodic halo orbit**: To determine the periodic halo orbit passing through z_0, the `findPeriodicorbit` function is employed. Within this function, the `propagate` function is utilized to adjust the initial condition. A differential correction scheme is implemented, wherein the values of x_0 and v_{y0} are adjusted while maintaining v_{x0} and v_{z0} at zero at the $y = 0$ plane crossing point. This adjustment is crucial for ensuring the periodicity of the orbit, as it necessitates a perpendicular crossing of the $y = 0$ plane.

   The orbit is propagated from the initial condition up to the $y = 0$ point using the `y_axis_crossing` function, which halts integration when $y = 0$. The states are propagated using the CRTBP dynamics as depicted in Equation 3.

$$\begin{cases} \dot{x} = v_x \\ \dot{y} = v_y \\ \dot{z} = v_z \\ \dot{v}_x = 2v_y + x - \dfrac{(1-\mu)(x+\mu)}{[(x+\mu)^2+y^2+z^2]^{3/2}} - \dfrac{\mu(x+\mu-1)}{[(x+\mu-1)^2+y^2+z^2]^{3/2}} \\ \dot{v}_y = -2v_x + y - \dfrac{1}{[(x+\mu)^2+y^2+z^2]^{3/2}}y - \dfrac{\mu}{[(x+\mu-1)^2+y^2+z^2]^{3/2}}y \\ \dot{v}_z = -\dfrac{1-\mu}{[(x+\mu)^2+y^2+z^2]^{3/2}}z - \dfrac{\mu}{[(x+\mu-1)^2+y^2+z^2]^{3/2}}z \end{cases} \tag{3}$$

   Within the `findPeriodicorbit` function, a Newton method is implemented as illustrated in Equation 4, where x represents the vector of unknowns (x_0 and v_{y0}), f represents the flow of the ODE at the final time $\varphi(x_0, t_0, t_f)$, and f' is its Jacobian.

$$\mathbf{x}_{i+1} = \mathbf{x}_i - \left[\mathbf{f}'(\mathbf{x}_i)\right]^{-1}\mathbf{f}(\mathbf{x}_i) \tag{4}$$

   The core logic within `findPeriodicorbit` is represented by Equation 5, where the correction for both x_0 and v_{y0} is calculated. This equation is derived by considering only the terms of the STM Φ in which variation occurs, in this case variation in the initial state occurring only for $x0$ and $vy0$. The correction is obtained solving the linear system `correction` $= -\Phi \backslash [v_{xf}; v_{zf}]$ in which `mldivide` operator is used to enhance the computational efficiency of the code. The iteration halts when both `err_vxf` and `err_vzf` fall below the chosen tolerance of 1×10^{-14}.

$$\begin{pmatrix} x_0 \\ v_{y0} \end{pmatrix}_{i+1} = \begin{pmatrix} x_0 \\ v_{y0} \end{pmatrix}_{i} - \begin{bmatrix} \Phi_{41} & \Phi_{45} \\ \Phi_{61} & \Phi_{65} \end{bmatrix}^{-1} \begin{pmatrix} \varphi_x \\ \varphi_{v_y} \end{pmatrix} \tag{5}$$

To compute the state transition matrix Φ, the variational approach is employed, ensuring a more accurate STM compared to the finite difference method. The STM dynamics are propagated by solving the following system of ODEs 6, with f serving as the right-hand side of Equation 3.

$$\begin{cases} \dot{\mathbf{\Phi}} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(t)\mathbf{\Phi} \\ \mathbf{\Phi}_0 = \boldsymbol{\varphi}\left(\mathbf{x}_0, t_0, t_0\right) = \mathbf{I}_{6\times 6} \end{cases} \tag{6}$$

Figure 2 depicts the periodic halo orbit alongside the Lagrangian point L2, calculated using the `Lagrangianpoint` function. Table 1 presents the initial state vector.

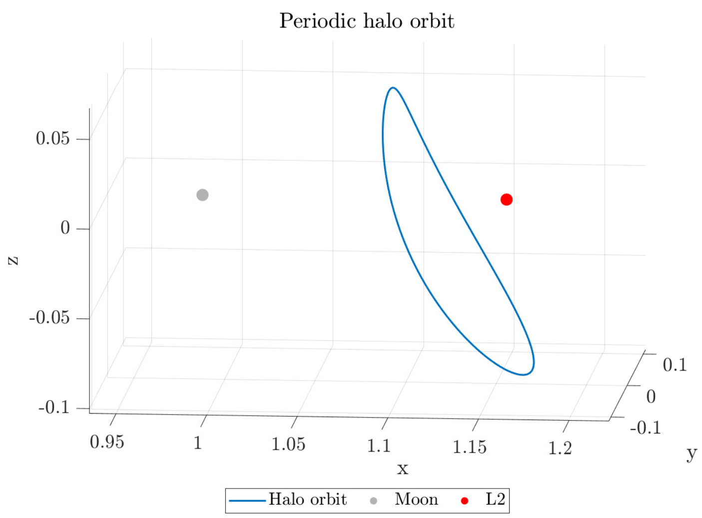

Figure 2: Periodic Halo orbit.

Table 1: Initial conditions periodic halo orbit.

Variable	Value
x_0	1.090278054655008
y_0	0
z_0	0.059179962345546
v_{x0}	0
v_{y0}	0.260349385022439
v_{z0}	0

3. **Numerical continuation**:

   To perform numerical continuation, the z value is iterated from 0.0591799623455459 to 0.034. The `findPeriodicorbit` function is iterated from the initial guess x_0 stated in the text, dividing the z values within a grid with a number of elements chosen to be 10, as a result of a trade-off between computational efficiency and accuracy of the results. Inside the `propagate` function, each orbit at step i is propagated from the initial condition at step $i-1$.

   Figure 3 illustrates the orbits achieved with numerical continuation. Table 2 presents the initial conditions obtained for this case.

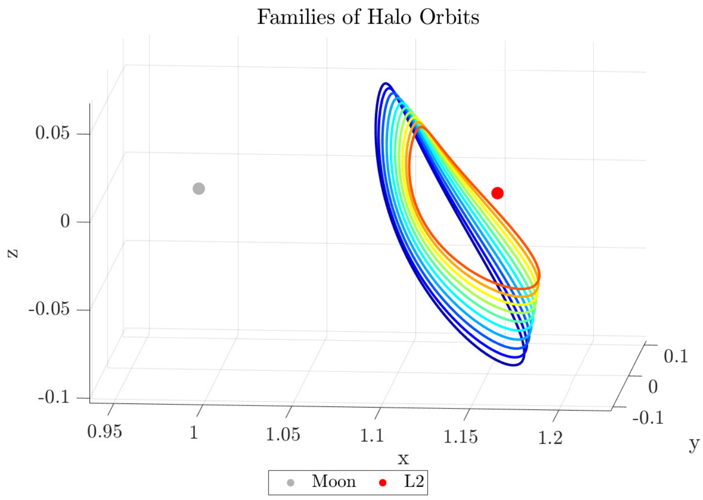

Figure 3: Orbits achieved with numerical continuation.

Table 2: Initial conditions family of halo orbits.

Variable	Value
x_0	1.111669365809449
y_0	0
z_0	0.034
v_{x0}	0
v_{y0}	0.201018707751581
v_{z0}	0

2 Impulsive guidance

Exercise 2

The Aphophis close encounter with Earth will occur on April 2029. You shall design a planetary protection guidance solution aimed at reducing the risk of impact with the Earth.

The mission shall be performed with an impactor spacecraft, capable of imparting a $\Delta\mathbf{v} = 0.00005\,\mathbf{v}(t_{\text{imp}})$, where $\mathbf{v}$ is the spacecraft velocity and t_{imp} is the impact time. The spacecraft is equipped with a chemical propulsion system that can perform impulsive manoeuvres up to a total Δv of 5 km/s.

The objective of the mission is to maximize the distance from the Earth at the time of the closest approach. The launch shall be performed between 2024-10-01 (LWO, Launch Window Open) and 2025-02-01 (LWC, Launch Window Close), while the impact with Apophis shall occur between 2028-08-01 and 2029-02-28. An additional Deep-Space Maneuver (DSM) can be performed between LWO+6 and LWC+18 months.

1) Analyse the close encounter conditions reading the SPK kernel and plotting in the time window [2029-01-01; 2029-07-31] the following quantities:

   a) The distance between Apophis and the Sun, the Moon and the Earth respectively.

   b) The evolution of the angle Earth-Apophis-Sun

   c) The ground-track of Apophis for a time-window of 12 hours centered around the time of closest approach (TCA).

2) Formalize an unambiguous statement of the problem specifying the considered optimization variables, objective function, the linear and non-linear equality and inequality constraints, starting from the description provided above. Consider a multiple-shooting problem with $N = 3$ points (or equivalently 2 segments) from t_0 to t_{imp}.

3) Solve the problem with multiple shooting. Propagate the dynamics of the spacecraft considering only the gravitational attraction of the Sun; propagate the post-impact orbit of Apophis using a full n-body integrator. Use an event function to stop the integration at TCA to compute the objective function; read the position of the Earth at t_0 and that of Apophis at t_{imp} from the SPK kernels. Provide the optimization solution, that is, the optimized departure date, DSM execution epoch and the corresponding $\Delta\mathbf{v}$'s, the spacecraft impact epoch, and time and Distance of Closest Approach (DCA) in Earth radii. Suggestion: try different initial conditions.

(11 points)

---

1. **Plot evolution**:

a) After having read the SPK kernel the `cspice_spkpos` function is used to calculate the position vector of the three celestial bodies with respect Apophis in the `IAU_EARTH` reference frame. Figure 4 show the evolution in the defined time window.

b) The angle between the Earth's Apophis and the Sun can be calculated using the formula `angle_earth_apo_sun` $= \cos^{-1}\left(\frac{\mathbf{earth}\cdot\mathbf{sun}}{r_{\texttt{apo\_sun}}\cdot r_{\texttt{apo\_earth}}}\right)$, where **earth** and **sun** are vectors representing the positions of the Earth and the Sun, and $r_{\texttt{apo\_sun}}$ and $r_{\texttt{apo\_earth}}$ are their distances to the apoapsis, respectively. Figure 5 show the evolution of the Earth-Apophis-Sun angle.

c) The ground track is calculated for 12 hours after the time of closest approach. The distance of closest approach is determined using the `fmincon` function, which searches

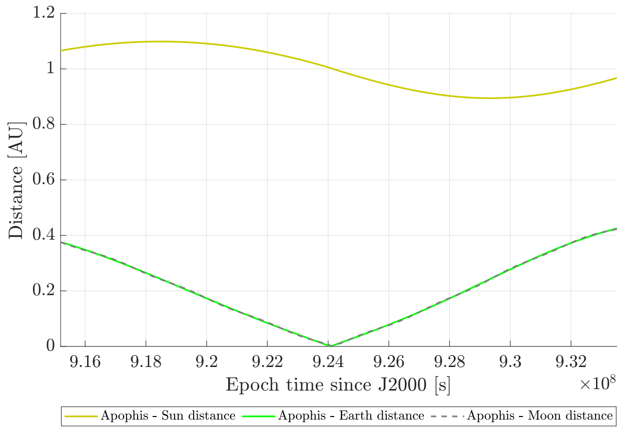

Figure 4: Distance evolution Apophis-Earth,Moon,Sun.

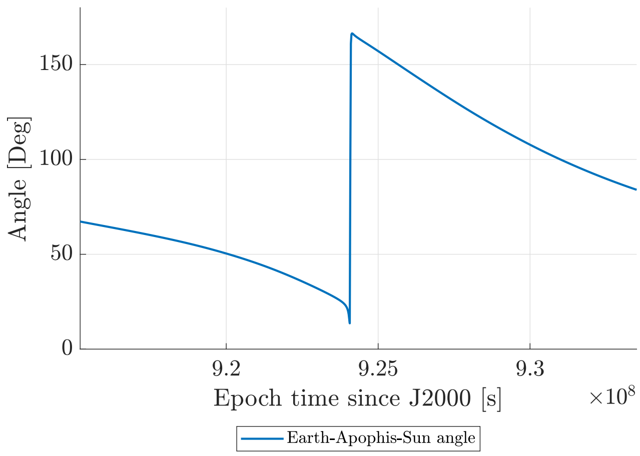

Figure 5: Angle evolution Earth-Apophis-Sun.

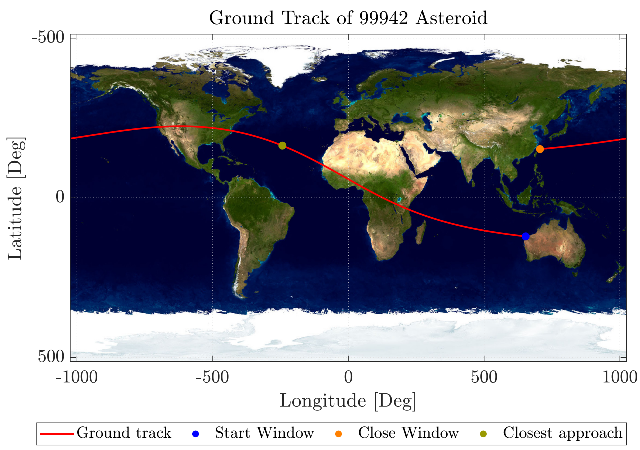

Figure 6: Ground track Apophis at closest approach.

for the minimum distance in the vector `r_apo_earth` within the open and close windows. Once the minimum distance is defined, Apophis is propagated for the subsequent 12 hours. Figure 6 shows the ground track with the closest approach point highlighted in yellow.

2. **Multiple-shooting problem**: The primary objective of this investigation is to devise maneuvers aimed at optimizing the closest approach distance between Earth and Apophis. To achieve this, a methodology employing multiple shooting is adopted, necessitating the resolution of a NLP, as depicted in Equation 7, where `g` denotes the set of inequality constraints and `C` signifies the set of equality constraints.

$$\min_{\mathbf{x}} J \text{ s.t. } \begin{cases} \mathbf{g}(\mathbf{x}) \leq \mathbf{0} \\ \mathbf{C}(\mathbf{x}) = \mathbf{0} \end{cases} \tag{7}$$

The optimization vector $\mathbf{x}$ consists of 21 variables representing the spacecraft's position and velocity at the launch ($\mathbf{x1}$), during the deep space maneuver ($\mathbf{x}2$), and at the point of impact ($\mathbf{x3}$), along with the respective times for these maneuvers. The cost function to be minimized, denoted by $J = -$ DCA, is associated with the distance of closest approach (DCA), which is computed by propagating asteroid dynamics until it reaches its minimal distance from Earth. The boundary conditions of the problem, represented by $\mathbf{r}\text{earth}(t_1)$ and $\mathbf{r}_{\text{sc}}(t_3)$, are implemented in the `constraint` function, indicating Earth's position at departure and Apophis's position at impact. The function $Z_1 = \varphi(\mathbf{x}_1, t_1, t_2)$ describes the spacecraft dynamics from launch to the deep space maneuver time, while $Z_2 = \varphi(\mathbf{x}_2, t_2, t_3)$ represents the spacecraft dynamics from the deep space maneuver to impact. The equality vector is defined as illustrated in Equation 8.

$$\mathbf{C}(\mathbf{x}) = \begin{cases} Z_1 = \varphi(\mathbf{x}_1, t_1, t_2) - \mathbf{x}_2 \\ Z_2 = \varphi(\mathbf{x}_2, t_2, t_3) - \mathbf{x}_3 \\ \xi_1 = \mathbf{r}_{\text{sc}} - \mathbf{r}_{\text{earth}}(t_1) \\ \xi_2 = \mathbf{r}_{\text{Apophis}} - \mathbf{r}_{\text{sc}}(t_3) \end{cases} \tag{8}$$

The inequality constraint equations, as expressed in Equation 9, are related to the window constraint and the maximum deliverable ΔV for the studied mission.

$$\mathbf{g}(\mathbf{x}) = \begin{cases} g_1 := t_{\text{Launch}} - t_{\text{LWC}} \\ g_2 := t_{\text{LWO}} - t_{\text{Launch}} \\ g_3 := t_{\text{DSM}} - t_{\text{DSMC}} \\ g_4 := t_{\text{DSMO}} - t_{\text{DSM}} \\ g_5 := t_{\text{Impact}} - t_{\text{IMPC}} \\ g_6 := t_{\text{IMPO}} - t_{\text{Impact}} \\ \Delta v = \Delta v_{\text{Launch}} + \Delta v_{\text{DSM}} - 5 \end{cases} \tag{9}$$

Solve the problem with multiple shooting: To solve the navigation problem, the delivered kernels are read. After stating the time window of the mission, the construction of the initial guess is done. The algorithm used is shown in Algorithm 1.

Algorithm 1 Procedure to determine the initial condition for `fmincon`.

$rand_{dv}$ = ceil (abs(randn()) $*$ 2.5); % random number from 1 to 5,
$DVL = rand_dv * rand1/norm(rand1)$; % random ΔV launch,
$DSM = (5 - rand_dv) * rand2/norm(rand2)$; % random ΔV DSM,
x_{SC} @ `mid_LAUNCH`;,
$x01 = x_{SC} + [\text{zeros}(3, 1); DVL]$;,
x_{SC12} from `mid_LAUNCH` to `mid_DSM`,
$\mathrm{x02} = \mathrm{x_{SC12}} + [\text{zeros}(3, 1); DSM]$;,
$x03$ from `mid_DSM` to `mid_IMP`,
Input: $xx0 = [x01; x02; x03;$ mid_LAUNCH; mid_DSM; mid_IMP];
$[opt_{sol}]$ = `fmincon`$(objective, xx0, lb, ub, cons, opt)$;

As illustrated in Algorithm 1, the initial condition $xx0$ serves as the input for the `fmincon` function, within which the `objective` function is defined and minimized, resulting in the calculation of $-DCA$. The `cons` function defines both equality and inequality constraints, as depicted in Equations 8 and 9. Table 3 presents the navigation solutions obtained.

The algorithm initializes DVL and DSM to enhance code convergence and narrow down the search space. Specifically, it provides a ΔV range from 1 to 5 for both launch and deep space maneuvers, as outlined in Algorithm 1. Remarkably, the code performs smoothly even with increased flexibility in the initial guess. The lower and upper boundaries, denoted as `lb` and `up`, respectively, define the time windows as the midpoint for each window, while the state vector's boundaries are set to infinite (`Inf(6,1)`).

Launch	2024 DEC 01 23:58:50.317 UTC		
DSM	2025 NOV 30 23:58:50.317 UTC		
Impact	2028 NOV 14 23:58:50.317 UTC		
TCA	2029 APR 14 00:33:50.992 UTC		
$\Delta\mathbf{v}_L$ [km/s]	1.20864	1.4887	0.5682
$\Delta\mathbf{v}_{DSM}$ [km/s]	0.6762	2.7742	0.9198
DCA [Re]	14.8231		

Table 3: Guidance solution for the impactor mission.

Figure 7 illustrates the distance between Apophis and Earth. The blue line represents the nominal solution calculated in 1c), whereas the orange line depicts the scenario where the distance increases following the impact.

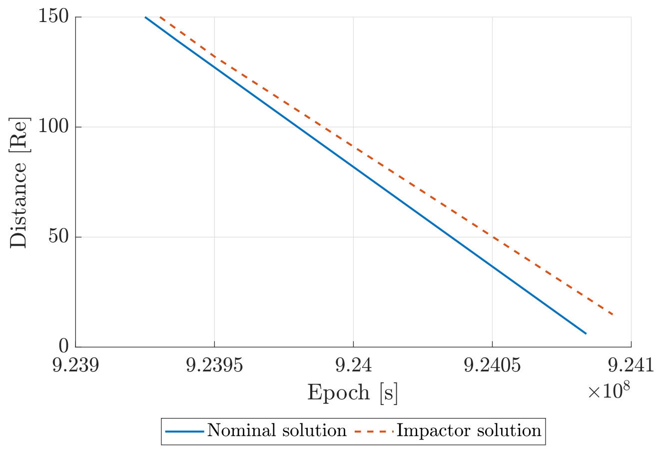

Figure 7: Impactor and nominal solution comparison.

3 Continuous guidance

Exercise 3

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

1) Using the PMP, 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) Adimensionalize the problem using as reference length $LU = 1$ AU† and reference mass $MU = m_0$, imposing that $\mu = 1$. Report all the adimensionalized parameters.

3) Solve the problem considering the following data:

   - Launch date: `2023-05-28-14:13:09.000 UTC`
   - Spacecraft mass: $m_0 = 1000$ kg
   - Electric propulsion properties: $T_{\max} = 800$ mN, $I_{sp} = 3120$ s

   To obtain an initial guess for the costate, generate random numbers such that $\lambda_{0,i} \in [-20; +20]$, while $t_f < 2\pi$. Report the obtained solution in terms of $\{\boldsymbol{\lambda}_0, t_f\}$ and the error with respect to the target. Assess your results exploiting the properties of the Hamiltonian in problems that are not time-dependent and time-optimal solutions.

4) Solve the problem for a lower thrust level $T_{\max} = [500]$ mN. Tip: exploit numerical continuation.

(11 points)

---

1. **Problem Statement**: Using the PMP method, the time-optimal problem, when the departure date is fixed, and the spacecraft's initial and final states coincide with those of the Earth and Venus respectively, is shown in Equation 10.

$$\min_{(u(t),t_f)\in\Upsilon} \int_{t_0}^{t_f} \ell(\mathbf{x},\mathbf{u},t)\ \mathrm{d}t \text{ s.t. } \begin{cases} \dot{\mathbf{x}} & = \mathbf{f}(\mathbf{x},\mathbf{u},t) \\ \mathbf{x}\left(t_0\right) & = \mathbf{x_0} \\ \mathbf{r}\left(t_f\right) & = \mathbf{r}_v(t_f) \\ \mathbf{v}\left(t_f\right) & = \mathbf{v}_v(t_f) \\ \lambda_m\left(t_f\right) & = 0 \end{cases} \tag{10}$$

   Where $\boldsymbol{\ell}(\mathbf{x},\mathbf{u},t)$ is the objective function, and as the problem is time-optimal, $\boldsymbol{\ell}$ can be written as $\min_{(\hat{\boldsymbol{\alpha}},u)\in\Upsilon} \int_{t_0}^{t_f} 1\ \mathrm{d}t$, and Υ is the set of control actions defined as $\Upsilon = \{(u,\hat{\boldsymbol{\alpha}}) : u \in [0,1], \|\hat{\boldsymbol{\alpha}}\| = 1\}$.

   The dynamics used are 2BP and for low thrust propulsion as shown in Equations 11.

---

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

†Read the value from SPICE

$$\begin{cases} \dot{\mathbf{r}} = \mathbf{v} \\ \dot{\mathbf{v}} = -\frac{\mu}{r^3}\mathbf{r} + u\frac{T_{\max}}{m}\hat{\boldsymbol{\alpha}} \\ \dot{m} = -u\frac{T_{\max}}{I_{sp}g_0} \end{cases} \tag{11}$$

Starting from the definition of the Hamiltonian $\mathcal{H} = 1 + \boldsymbol{\lambda} \cdot \mathbf{f}$ where $\boldsymbol{\lambda} = (\boldsymbol{\lambda_r}, \boldsymbol{\lambda_v}, \boldsymbol{\lambda_m})$, the costate dynamics can be written as shown in Equation 12.

$$\begin{cases} \dot{\boldsymbol{\lambda}}_{\boldsymbol{r}} = -\frac{3\mu}{r^5}(\mathbf{r} \cdot \boldsymbol{\lambda}_v)\,\mathbf{r} + \frac{\mu}{r^3}\boldsymbol{\lambda_v} \\ \dot{\boldsymbol{\lambda}}_{\boldsymbol{v}} = -\boldsymbol{\lambda_r} \\ \dot{\lambda}_m = -u\frac{T_{\max}}{m^2}\boldsymbol{\lambda}_v \cdot \hat{\boldsymbol{\alpha}} \end{cases} \tag{12}$$

The TPBVP to be solved is synthesize as in Equation 13.

$$\begin{cases} \dot{x} = \frac{\partial \mathcal{H}}{\partial \lambda} \\ \dot{\lambda} = -\frac{\partial \mathcal{H}}{\partial \mathbf{x}} \\ \mathcal{H}(t_f) - \boldsymbol{\lambda}_r(t_f) \cdot \mathbf{v}(t_f) - \boldsymbol{\lambda}_v(t_f) \cdot \mathbf{a}(t_f) = 0 \end{cases} \tag{13}$$

where $\mathbf{x}$ is the state vector of the spacecraft, and the last Equation in 13 is the transversality condition needed to express the constrained control action. $\mathbf{v}(t_f)$ and $\mathbf{a}(t_f)$ are the velocity and acceleration of Venus, respectively.

To provide the minimum value of the Hamiltonian function, the chosen $u, \hat{\boldsymbol{\alpha}}$ are defined in Equation 14.

$$\begin{cases} u^* = 1 \\ \hat{\boldsymbol{\alpha}}^* = \frac{\boldsymbol{\lambda}_v}{\lambda_v} \end{cases} \tag{14}$$

By substitution of these Equations into $\mathcal{H}$, it is possible to obtain the switching function as shown in Equation 15.

$$S = -\frac{\lambda_v}{m} I_{\mathrm{sp}} g_0 - \lambda_m \tag{15}$$

2. **Adimensionalize the problem**: The adimensionalization process is performed using the `adimensionalize` function, which takes as input a struct containing the dimensional quantities. To impose $\mu = 1$, originally expressed in $\mathrm{km}^3/\mathrm{s}^2$, the time units are calculated as $TU = \frac{\mathrm{AU}}{v_{\mathrm{rcf}}}$, where $v_{\mathrm{rcf}} = \sqrt{\frac{\mathrm{GM}}{\mathrm{AU}}}$ is expressed in km/s. Table 4 shows the result obtained with the adimensionalization.

$\mathbf{r}_0$	-0.40093	-0.93063	4.8795×10^{-5}
$\mathbf{v}_0$	0.90233	-0.39917	4.2073×10^{-5}
m_0	1		
I_{sp}	6.2119×10^{-4}		
$T_{\max}$	0.13491		
g_0	1653.712		
GM	1		

Table 4: Adimensionalized quantities ($T_{\max} = 800$ mN).

3. **Resolution for 800 mN thrust**: To solve the problem, Algorithm 2 outlines the procedure followed. For this problem, the initial guess is chosen such that λ_0 is within ± 20, and t_f is generated between 0 and 2π. The function `Orbitsol` finds the orbit solution, within which the `fsolve` function is implemented. Table 5 shows the result of the time-optimal Earth-Venus transfer. Table 6 shows the final error for both position and velocity. Figure 8 illustrates the aforementioned optimal transfer.

Algorithm 2 Procedure to calculate the orbit solution.

Inside `Orbitsol` % Find the optimal solution for Earth-Venus transfer.
$guess = [\lambda_0; t_f]$;
$[opt_{sol}] = fsolve(@(vars)obj(vars(1:8), admen, LWO), guess)$; % Where `admen` is the struct that contains the initial state at `LWO`.
Inside `obj`
$[\mathrm{X}_f; \boldsymbol{\lambda}_f] \leftarrow$ ODE propagation % SC final state.
$[\mathbf{X}_v] \leftarrow$ `cspice_spkezr` (t_f) % Venus final state.
$\mathcal{H} \leftarrow 1 + \boldsymbol{\lambda}_f \cdot \mathbf{f}(tf)$ % Hamiltonian calculation.
$\mathcal{H}(t_f) - \boldsymbol{\lambda}_r(t_f) \cdot \mathbf{v}(t_f) - \boldsymbol{\lambda}_v(t_f) \cdot \mathbf{a}(t_f)$ % Transversality condition.
Output `obj`: residual $= [\mathrm{X}_f - \mathrm{X}_v; \lambda_m; t_{\text{cond}}]$;
Output `Orbitsol`: Optimal solution $[\boldsymbol{\lambda}_f; t_f]$

$\boldsymbol{\lambda}_{0,r}$	0.4979	−13.8150	0.1082
$\boldsymbol{\lambda}_{0,v}$	5.1823	−10.3975	1.4848
$\lambda_{0,m}$	1.7511		
t_f	`2023 OCT 17 03:57:58.043 UTC`		
TOF [days]	141.5728		

Table 5: Time-optimal Earth-Venus transfer solution ($T_{\max} = 800$ mN).

$\lvert\lvert\mathbf{r}_f(t_f) - \mathbf{r}_V(t_f)\rvert\rvert$	[km]	0.00664
$\lvert\lvert\mathbf{v}_f(t_f) - \mathbf{v}_V(t_f)\rvert\rvert$	[m/s]	1.4192×10^{-6}

Table 6: Final state error with respect to Venus' center ($T_{\max} = 800$ mN).

To verify the consistency of the results, both the Hamiltonian and the switching function are computed. For a non-time-dependent problem with an optimal-time solution, it is ensured that the Hamiltonian remains constant. Figure 9 illustrates the evolution of the Hamiltonian over time, with the error observed on the order of 10^{-12} due to numerical approximations. Additionally, the switching function is plotted to demonstrate that the thrust throttling policy is achieved with $u = 1$, as S consistently remains less than zero, aligning with the expected behavior defined by the time-optimal problem.

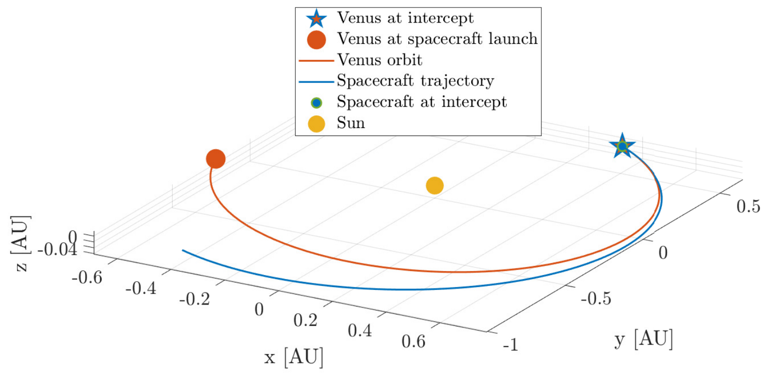

Figure 8: Venus and spacecraft trajectory for 800 mN case.

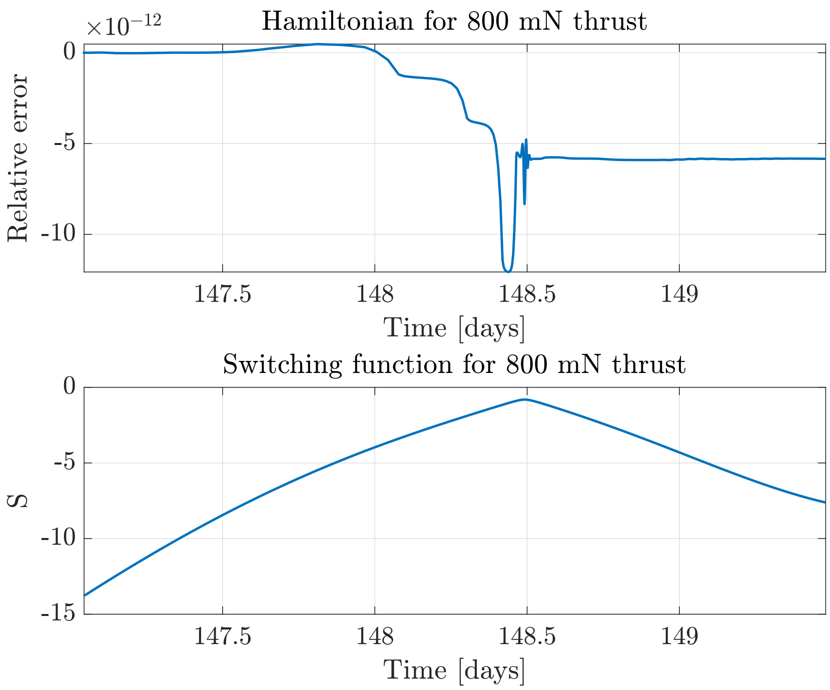

Figure 9: Hamiltonian and Switching function evolution.

4. **Resolution for numerical continuation problem**:

   For the numerical continuation, a thrust variation is defined as `T = linspace(0.8, 0.5, 50)`, and Algorithm 2 is iterated for the total length of the thrust vector `T` with a `for` cycle. In detail, for this resolution, the optimal solution found at Point 3 is used as the initial guess of the problem, and then each optimal solution is used as the initial guess for the subsequent iteration. Table 7 shows the result obtained, and Table 8 shows the errors for the final state. Figure 10 shows the trajectory of the aforementioned problem.

$\boldsymbol{\lambda}_{0,r}$	-0.5634	-24.5354	-0.4750
$\boldsymbol{\lambda}_{0,v}$	14.1805	-18.9082	1.9967
$\lambda_{0,m}$	2.7282		
t_f	`2024 JAN 01 04:10:55.650 UTC`		
TOF [days]	217.5818		

Table 7: Time-optimal Earth-Venus transfer solution ($T_{\max} = 500$ mN).

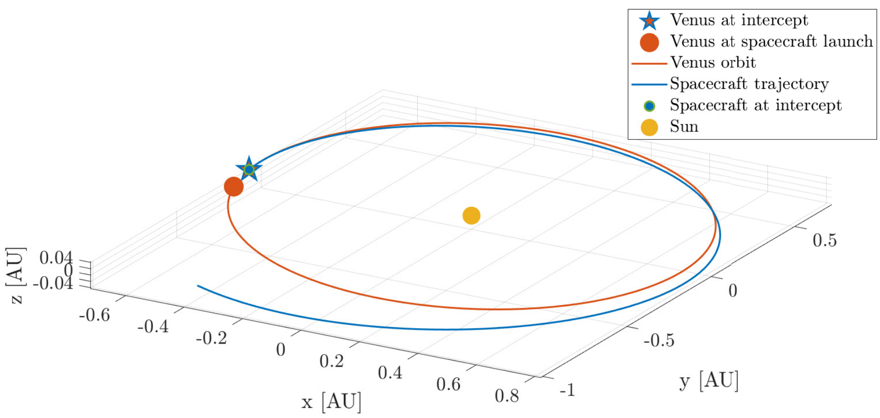

Figure 10: Venus and spacecraft trajectory for numerical continuation case.

$\|\|\mathbf{r}_f(t_f) - \mathbf{r}_V(t_f)\|\|$	[km]	0.00480
$\|\|\mathbf{v}_f(t_f) - \mathbf{v}_V(t_f)\|\|$	[m/s]	2.0785×10^{-6}

Table 8: Final state error with respect to Venus' center ($T_{\max} = 500$ mN).