

# OPTIMAL LOW THRUST ASSISTED NATURAL REENTRY FOR LOW EARTH ORBITS

Damien Courteville<sup>(1)</sup>, Joan Pau Sánchez<sup>(1)</sup>, Bertrand Raffier<sup>(2)</sup>, and Quentin Harbulot<sup>(3)</sup>

<sup>(1)</sup>ISAE-SUPAERO, 31400 Toulouse, France, Email: {damien.courteville, joan-pau.sanchez}@isae-supero.fr

<sup>(2)</sup>CNES, 31400 Toulouse, France, Email: bertrand.raffier@cnes.fr

<sup>(3)</sup>Thales Alenia Space, 06150 Cannes, France, Email: quentin.harbulot@thalesaleniaspace.com

## ABSTRACT

Assisted Natural Reentry (ANR) is a deorbitation strategy conceived for low-thrust LEO satellites, aiming to minimize the duration of the uncontrolled reentry phase. Such an approach allows to limit uncertainties on the debris impact area, which ensures compliance with human casualty risk requirement. However, due to the limiting low-thrust capability, an ANR operation can extend over several months, leading to significant operational cost as a ground team must continuously monitor and adjust the trajectory. As the first step toward an autonomous ANR guidance solution, this paper presents a new method for computing an optimal open-loop ANR guidance using an indirect optimization approach. This approach uses Pontryagin's Maximum Principle to derive the necessary conditions of optimality and rewrites the ANR optimization problem into a two-point boundary value problem. This problem is then solved using a multiple shooting method to obtain the optimal solution. An application of this method is shown for the initial computation and subsequent updates of a fuel-optimal ANR trajectory. The ability to compute and update objective-optimal strategies makes it an interesting alternative to the existing ANR strategy computation tools for ground teams, but the main interest of this approach is its use as a basis for the development of an autonomous closed-loop guidance.

Keywords: Deorbitation; Indirect Optimization; Fuel-Optimal Many-Revolution Trajectory.

## 1. INTRODUCTION

Satellites in low Earth orbit must be deorbited after their mission lifetime to limit the proliferation of space debris. Controlled reentry is a common deorbiting strategy, in which the satellite is deorbited with a relatively large velocity decrement. This allows to keep the human casualty risk under the regulatory threshold by precisely controlling the impact area. However, the propulsion system must be dimensioned accordingly and it is, therefore, unfeasible for low-thrust propulsion systems. A common alternative is natural reentry, where the orbit is left to

decay under the sole effect of atmospheric drag. However, due to uncertainties in the atmospheric drag force, the impact point cannot be predicted, which implies that the spacecraft must be designed such that it satisfies casualty risks assuming a random impact point.

Assisted Natural Reentry (ANR) is an intermediate strategy in which the satellite is progressively brought in a controlled manner to an interface orbit with a very low perigee (between 130km and 170km), from which it quickly re-enters the atmosphere after a few uncontrolled revolutions. The short final uncontrolled reentry limits the uncertainty introduced by the drag force, and allows to target an unpopulated debris impact area to reduce the ground casualty risk compared to a random natural reentry. And the progressive controlled descent is feasible with lower thrust chemical or electrical propulsion systems that would not permit a controlled reentry.

The computation of a maneuver strategy for the ANR's controlled phase poses several challenges. Firstly, as the altitude decreases the increasing atmospheric drag, and its associated uncertainty, become predominant over the spacecraft thrust toward the end of the descent. Secondly, the increasing aerodynamic torque will put a strain on the attitude control system, which will not be able to point the thrusters or solar panels below a certain altitude (typically around 250km). Thus Borobia et al. (2017) determined that the controlled descent and the interface orbit should be elliptical to allow to maintain control around the apogee while the perigee decreases. Tools were developed by Airbus (Lagadec and Gegout, 2023) and CNES (Goester et al., 2022) to enable ground operation teams to compute and readjust maneuvers for the ANR. However, with low-thrust propulsion the ANR may take several months, which come at a high operational cost since ground teams must remain mobilized to adjust the frequent maneuvers. Thus the next step to facilitate the adoption of ANR for low-thrust missions is to embed an autonomous ANR guidance on the satellite.

The first step toward autonomous guidance is to have a method for computing an open-loop guidance, which gives a nominal trajectory, and which can then be converted into a closed-loop guidance and embedded into the spacecraft software. Airbus's tool is not accessible and was thus not considered. And while CNES's tool,

SIRENA, is available to the author, it relies on shape-based methods that are sub-optimal, especially for very low-thrust propulsion, and not easily translatable to a robust closed-loop guidance.

This paper thus presents a new method for computing an open-loop ANR guidance using an indirect optimization approach. This approach uses the Pontryagin's Maximum Principle to derive the necessary conditions of optimality and rewrite the ANR optimization problem into a two-point boundary value problem (TPBVP). This problem is then solved using a multiple shooting method to obtain the optimal solution. The indirect optimization is applied in this paper for the computation of a fuel-optimal guidance, but the approach can be applied to other optimization objectives.

An advantage of the indirect approach is its low dimensionality, since the resulting optimal guidance is characterized by a costate vector of same dimension than the state. Furthermore, Sidhoum and Oguri, 2024 shows that such open-loop guidance can be turned into a robust closed-loop control by mapping deviations of the state to adjustment of the costate vector. This method is also a potential alternative to existing ANR maneuvers computation tools for ground operations since it allows to efficiently update a previously computed maneuvers strategy to take into account deviations from the nominal trajectory. However, more work is still needed on the enforcement of maneuver constraints for the computed guidance to be a suitable candidate.

The rest of this article is organized as follows. Section 2 defines the ANR guidance optimization problem. Section 3 presents the indirect optimization approach for a fuel-optimal objective. Section 4 shows a numerical application of this method for the initial computation and subsequent updates of an ANR trajectory.

## 2. PROBLEM STATEMENT

### 2.1. System dynamics

The Modified Circular Elements (MCE) along with the satellite mass  $m$  are used to represent the satellite state. The relation between MCE and keplerian elements is

$$\mathbf{x}_{orb} = \begin{cases} p & = a(1 - e^2) \\ e_x & = e \cos \omega \\ e_y & = e \sin \omega \\ i & \\ \Omega & \\ \alpha & = \omega + \nu \end{cases} \quad (1)$$

Where  $a$  is the semi-major axis,  $e$  the eccentricity,  $i$  the inclination,  $\Omega$  the Right Ascension of Ascending Node (RAAN),  $\omega$  the argument of perigee,  $\nu$  the true anomaly and  $\alpha$  the Argument of Latitude (AoL). The eccentricity vector  $(e_x, e_y)$  avoids the singularity for circular orbits, and the orbit is always assumed to be non-equatorial

( $i \neq 0$ ). MCE are preferred over the more commonly used Modified Equinoctial Elements since having the RAAN directly in the state allows for a more practical representation of the ANR interface orbit constraints.

The equation of the state dynamics is

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \sigma) = \begin{bmatrix} \dot{\mathbf{x}}_{orb} \\ \dot{m} \end{bmatrix} = \begin{bmatrix} \mathbf{B} \left( \sigma \frac{T}{m} \mathbf{u} + \mathbf{a}_{drag} \right) + \mathbf{b} \\ -\sigma \frac{T}{c} \end{bmatrix} \quad (2)$$

Where  $T$  is the maximum thrust,  $c$  is the exhaust velocity,  $\sigma$  is the commanded throttle in  $[0, 1]$ ,  $\mathbf{u}$  is the thrust direction unit vector. The thrust and acceleration vectors are expressed in the QSW local orbital frame where Q is the radial direction along the position vector, W is the off-plane direction along the orbital angular momentum, and S is the orthoradial direction that completes the direct frame.  $\mathbf{B}$  is the matrix of Gauss variational equations mapping the satellite acceleration to its effect on orbital elements,  $\mathbf{b}$  is the secular variation of orbital elements under Earth potential. The drag acceleration  $\mathbf{a}_{drag}$  is expressed

$$\mathbf{a}_{drag} = -\frac{1}{2} \frac{C_D S}{m} \rho v \mathbf{v} \quad (3)$$

Where  $C_D$  is the drag coefficient,  $S$  is the satellite cross-section,  $\rho$  is the atmosphere density computed with the NRLMSISE-00 model,  $\mathbf{v}$  is the velocity vector and  $v$  is its magnitude.

### 2.2. Maneuver constraints

Maneuvers are subject to two types of constraints. The first are the forced coast arcs, which are portions of the orbit where the satellite is not allowed to perform thrusts. Such constraints can be modelled as an inequality constraint

$$\sigma g(t, \mathbf{x}) \geq 0 \quad (4)$$

where  $g(t, \mathbf{x})$  is a coast function which is positive when thrust is allowed and strictly negative otherwise. A typical example of forced coasting in ANR scenarios is passages under a certain altitude  $h_{min}$  below which the aerodynamic torque becomes too strong for the reaction wheels to maintain the satellite pointing. The coast function associated with this constraint, assuming a spherical Earth of radius  $R_e$  is

$$g = \frac{p}{1 + e_x \cos \alpha + e_y \sin \alpha} - (R_e + h_{min}) \quad (5)$$

Another common forced coast arc is eclipse passages, which are not considered here but their implementation on an indirect optimization problem is shown in Pontani (2021).

The second type of constraint for low-thrust propulsion is battery charge, which limits the duration of thrust arcs

and enforces coasting arcs for battery charging between each thrust. However, such constraints cannot be directly modeled as an inequality constraint like Eq. 4, so the thrust duration is left unconstrained in this paper.

### 2.3. Objectives

The initial conditions are the predicted date  $t_0$ , satellite mass  $m_0$ , and orbit  $\mathbf{x}_{orb,0}$  at the beginning of end-of-life operations, which are all fixed. The target orbit is the interface orbit of the ANR between the controlled and uncontrolled phase. Its defining feature is that its ascending node is fixed at a terrestrial longitude, so the final RAAN  $\Omega_f$  is tied to the final date  $t_f$  by the relation:

$$\Omega_f = \Lambda_f - \theta_0 - \dot{\theta}(t_f - t_0) \quad (6)$$

Where  $\Lambda_f$  is the target Longitude of Ascending Node (LAN),  $\theta_0$  is the sidereal time at  $t_0$  and  $\dot{\theta}$  is the Earth angular rate. The other orbital elements  $a_f, e_{xf}, e_{yf}, i_f, \alpha_f$  are fixed, and the final mass  $m_f$  is free. The rotating angular elements  $\alpha, \Omega$  as well as  $\Lambda_f$  are not bounded in  $[-\pi, \pi[$  to keep track of the number of turns. In particular,  $\Lambda_f - \Omega_f$  is directly related to the duration of the ANR trajectory.

The choice of the interface orbit is discussed in Borobia et al. (2017) and Lagadec and Gegout (2023). To begin with, the final total AoL  $\alpha_f$  is defined modulo  $2\pi$ .

$$\alpha_f = \bar{\alpha}_f + 2\pi N_\alpha \quad (7)$$

Where  $\bar{\alpha}_f$  is the target final AoL in  $] -\pi, \pi]$ , and  $N_\alpha$  is the integer number of AoL turns which is a free variable. A practical choice is to set the interface orbit at perigee ( $\alpha_f = \omega_f$ ) since it is by design in a forced coasting arc imposed by Eq. 5, which tends to facilitate optimization. The target apogee altitude  $h_{af}$  and perigee altitude  $h_{pf}$  are chosen high enough for the interface orbit to be reached in a controlled manner, and low enough to reduce the duration of the subsequent uncontrolled phase, the uncertainty on the debris impact area, and thus the risk of casualty below the target threshold. The target LAN  $\Lambda_f$  is chosen to align the impact area of debris in unpopulated areas. The target argument of perigee is chosen to either adjust the uncontrolled phase duration or the orbit orientation relative to the Earth shadow. The target inclination  $i_f$  is usually close to the mission inclination to avoid the need for out-of-plane orbit adjustments.

Finally, to complete the optimization problem statement an objective function to minimize must be defined in the following form

$$J = \Phi(t_0, \mathbf{x}_0, t_f, \mathbf{x}_f) + \int_{t_0}^{t_f} L(t, \mathbf{x}, \mathbf{u}, \sigma) \quad (8)$$

The objective chosen in this paper is to find the guidance that minimises fuel consumption for a given value of  $\Lambda_f$ . The corresponding fuel-optimal objective function is

$$J = \int_{t_0}^{t_f} L dt = - \int_{t_0}^{t_f} \dot{m} dt = \frac{T}{c} \int_{t_0}^{t_f} \sigma dt \quad (9)$$

## 3. INDIRECT OPTIMIZATION

### 3.1. Indirect formulation

The indirect approach is formulated for the fuel-optimal problem, where the control inputs to optimize are the thrust direction  $\mathbf{u}$  and throttle  $\sigma$ .

The Hamiltonian associated with this problem is formed by adjoining the system dynamics in Eq. 2 to the cost function  $L$  in Eq. 9.

$$\begin{aligned} H &= \boldsymbol{\lambda}^T \mathbf{f} + L \\ &= \boldsymbol{\lambda}_x^T \left[ \mathbf{B} \left( \sigma \frac{T}{m} \mathbf{u} + \mathbf{a}_{drag} \right) + \mathbf{b} \right] \\ &\quad + \sigma \frac{T}{c} (1 - \lambda_m) \end{aligned} \quad (10)$$

Where  $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_x^T \lambda_m]^T$  is a set of time-varying Lagrangian multipliers called costates,  $\boldsymbol{\lambda}_x$  is the costate vector associated with the orbital elements  $\mathbf{x}_{orb}$ , and  $\lambda_m$  is the costate associated with the mass  $m$ .

The Pontryagin's Maximum Principle (PMP) states that the optimal control, states and costates must minimize  $H$  at all time. For the control input, this translates to

$$\mathbf{u}^* = \arg \min_{\|\mathbf{u}\|=1} H \quad (11)$$

$$\sigma^* = \arg \min_{\sigma \in [0,1]} H \quad (12)$$

The optimal thrust direction is expressed with Lawden's primer vector (Rutherford, 1964)

$$\mathbf{u}^* = - \frac{\mathbf{B}^T \boldsymbol{\lambda}_x}{\|\mathbf{B}^T \boldsymbol{\lambda}_x\|} \quad (13)$$

Since  $\sigma$  must also satisfy the forced coasting inequality in Eq. 4, we define

$$\sigma = \sigma_m \sigma_g \quad (14)$$

Where  $\sigma_m$  is the fuel-optimal throttle and  $\sigma_g$  is the constraint throttle, both in  $[0, 1]$ . Since  $H$  is linear in  $\sigma_m$ , the optimal throttle  $\sigma_m^*$  is either 0 (no thrust) or 1 (full thrust) depending on the sign of  $\frac{\partial H}{\partial \sigma}$ , leading to a bang-bang thrust profile. We derive the switching function from  $\frac{\partial H}{\partial \sigma}$

$$S_f = \frac{c}{m} \|\mathbf{B}^T \boldsymbol{\lambda}_x\| + \lambda_m - 1 \quad (15)$$

And the optimal throttle is then given by

$$\sigma_m^* = \begin{cases} 1, & \text{if } S_f > 0 \\ 0, & \text{if } S_f < 0 \end{cases} \quad (16)$$

We assume that the case  $S_f = 0$  where  $\sigma_m^*$  is undefined only happens on singular time intervals and is thus not considered. The constraint throttle is defined as

$$\sigma_g = \begin{cases} 1, & \text{if } g(t, \mathbf{x}) \geq 0 \\ 0, & \text{if } g(t, \mathbf{x}) < 0 \end{cases} \quad (17)$$

Table 1: Summary of boundary constraints.

Component	Initial bound		Final bound	
	State	Costate	State	Costate
$p$	$p_0$	free	$p_f$	free
$e_x$	$e_{x0}$	free	$e_{xf}$	free
$e_y$	$e_{y0}$	free	$e_{yf}$	free
$i$	$i_0$	free	$i_f$	free
$\Omega$	$\Omega_0$	free	Eq. 6	Eq. 19
$\alpha$	$\alpha_0$	free	$\alpha_f$	free
$m$	$m_0$	free	free	0
$t$	$t_0$			Eq. 6
$H$		free		Eq. 19

The PMP also allows to derive the costates differential equations

$$\dot{\lambda} = - \left[ \frac{\partial H}{\partial \mathbf{x}} \right]^T \quad (18)$$

as well as the costates boundary conditions on  $\lambda$  and  $H$ , summarized in Table 1. If a state component is fixed at a boundary, the associated costate is free. If a state component is free at a boundary, the associated costate is fixed at 0. Finally, the constraint in Eq. 6 translates to the constraint Eq. 19 on the final RAAN costate  $\lambda_\Omega$  and Hamiltonian  $H$ .

$$\dot{\theta} \lambda_\Omega(t_f) + H(t_f) = 0 \quad (19)$$

Finally, we define the augmented state vector  $\mathbf{y} = [\mathbf{x}^T \ \boldsymbol{\lambda}^T]^T$ . The optimal command  $\mathbf{u}^*, \sigma^*$  is a function of  $\mathbf{y}$ , meaning that the augmented state dynamics on the optimal trajectory is a function of itself.

$$\dot{\mathbf{y}} = \begin{bmatrix} \dot{\mathbf{x}}(\mathbf{x}, \mathbf{u}^*, \sigma^*) \\ \dot{\boldsymbol{\lambda}}(\mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}^*, \sigma^*) \end{bmatrix} = \begin{bmatrix} \dot{\mathbf{x}}(\mathbf{y}) \\ \dot{\boldsymbol{\lambda}}(\mathbf{y}) \end{bmatrix} = f_{\mathbf{y}}(\mathbf{y}) \quad (20)$$

The fuel-optimal optimization problem thus becomes a Two Points Boundary Value Problem (TPBVP).

$$\begin{cases} \text{Find } \mathbf{y}(t) \\ \text{s.t.} \\ \dot{\mathbf{y}} = f_{\mathbf{y}}(\mathbf{y}) \\ t \in [t_0, t_f] \\ t_0, \mathbf{y}(t_0), H(t_0) \rightarrow \text{Table 1} \\ t_f, \mathbf{y}(t_f), H(t_f) \rightarrow \text{Table 1} \end{cases} \quad (21)$$

### 3.2. Forced coasting arcs

As demonstrated in Pontani (2021), transition between a thrust arc and a forced coasting arc introduces a discontinuity on the costate vector  $\boldsymbol{\lambda}$  and on Hamiltonian  $H$ .

$$\boldsymbol{\lambda}^+ = \boldsymbol{\lambda}^- - \xi \left[ \frac{\partial g}{\partial \mathbf{x}} \right]^T \quad (22)$$

$$H^+ = H^- + \xi \frac{\partial g}{\partial t} \quad (23)$$

Where the superscripts  $-$  and  $+$  denotes the value of the variables respectively before and after the discontinuity, and  $\xi$  is an unknown parameter. By expanding  $H$  with Eq. 10 in Eq. 23 we obtain

$$\begin{aligned} 0 = & -\frac{T}{c} [\lambda_m^+ \sigma^+ - \lambda_m^- \sigma^-] \\ & - \frac{T}{m} [\sigma^+ \|\mathbf{B}^T \boldsymbol{\lambda}_x^+\| - \sigma^- \|\mathbf{B}^T \boldsymbol{\lambda}_x^-\|] \\ & - \xi \left[ \left( \frac{\partial g}{\partial \mathbf{x}} \right)^T (\mathbf{B} \mathbf{a}_{drag} + \mathbf{b}) + \frac{\partial g}{\partial t} \right] \end{aligned} \quad (24)$$

And after expanding  $\boldsymbol{\lambda}^+$  with Eq. 22 in Eq. 24 we obtain the equation to solve for the unknown parameter  $\xi$ .

The process of integrating the augmented state vector through these discontinuities is as follows. When a coasting event  $g(t, \mathbf{x}) = 0$  is detected, the integration is stopped and the throttles  $\sigma^-, \sigma^+$  before and after the event are determined. Then Eq. 24 is solved with the MATLAB<sup>®</sup> solver *fzero* to obtain the parameter  $\xi$ . Finally, the costate vector is updated with Eq. 22 and the integration is resumed.

Forced coasting arcs can be an issue when searching for the solution of the TPBVP, as they can cause a discontinuous variation of the trajectory when a new discontinuity is introduced or removed between two guesses for  $\mathbf{y}(t)$ . As a workaround, the discontinuous coasting constraint in Eq. 17 can be replaced with a smoothed coasting constraint

$$\sigma_g = \frac{1}{2} \left( 1 + \tanh \frac{g(t, \mathbf{x})}{2\epsilon} \right) \quad (25)$$

Where  $\tanh$  is the hyperbolic tangent and  $\epsilon$  is the smoothing parameter. Using this smoothed constraint also smooths out the costate discontinuity when transitioning between a thrust arc and a forced coasting arc.

### 3.3. Solving methodology

Since the solution to the TPBVP must satisfy the boundary conditions on both ends, an iterative shooting method is used to find it. The most simple method, single shooting, consists in guessing a value for  $\mathbf{y}(t_0)$ , integrating the differential equation to the final bound, computing the error on  $\mathbf{y}(t_f)$  and adjusting the guess. However, small variations in the candidate  $\mathbf{y}(t_0)$  lead to considerable and non-linear variations in the resulting trajectory and final augmented state. This is especially true in the case of the ANR where small variations of the trajectory are amplified by the atmospheric drag, which scales exponentially with altitude. The consequence is that the convergence domain of the shooting method is very small, and thus finding a good enough initial guess is difficult.

To increase the convergence domain, the problem is cut into two segments date at an intermediate date  $t_i$  to perform a multiple shooting search. The date  $t_i$  is chosen around the point where the drag force becomes the

main perturbing acceleration, which roughly corresponds to the transition between the first phase and the second phase of the ANR in Lagadec and Gegout (2023). In the first segment, the candidate  $\mathbf{y}(t_0)$  is propagated from  $t_0$  to  $t_i$  to obtain the intermediate  $\mathbf{y}^-(t_i)$ . In the second segment, the candidate  $\mathbf{y}(t_f)$  is back-propagated from  $t_f$  to  $t_i$  to obtain the intermediate  $\mathbf{y}^+(t_i)$ . The shooting method is then used to search the combination of  $\mathbf{y}(t_0)$  and  $\mathbf{y}(t_f)$  that satisfies both boundaries constraints and the junction constraint

$$\mathbf{y}^+(t_i) - \mathbf{y}^-(t_i) = 0 \quad (26)$$

This approach has several advantages that extend the convergence domain of the multiple shooting compared to single shooting. Firstly, the boundary constraints are easily enforced on both ends. Secondly, propagating the second segment backward avoids the non-linear amplification effect of the atmospheric drag. Finally, having two smaller segments means that variations of  $\mathbf{y}(t_0)$  and  $\mathbf{y}(t_f)$  have a smaller effect on their respective segments than if they were propagated on the entire interval at once.

The multiple shooting can be represented as a function  $\mathbf{E} = F(\mathbf{Y})$  that maps the vector of unknown parameters  $\mathbf{Y}$  to the shooting error  $\mathbf{E}$ . The 14 unknown parameters are  $\Omega_f$  and the 13 free states and costates in Table 1. The final time  $t_f$  and  $\lambda_\Omega(t_f)$  can be computed from the other parameters with Eq. 6 and Eq. 19 respectively. The shooting error is composed of the 14 residuals from Eq. 26. The TPBVP is solved by finding the root of  $F$  using the MATLAB<sup>®</sup> solver *fsolve* with the Levenberg-Marquardt algorithm. At each iteration, the Jacobian matrix of  $F$  is computed using finite difference. The last unknown variable is the number of AoL turns  $N_\alpha$ . Since it is an integer, it is optimized separately by solving the TPBVP for different values of  $N_\alpha$  to find the one that minimizes mass consumption.

Finding a suitable first guess for  $\mathbf{Y}$  is often the most difficult step of the indirect approach. A common approach is homotopy where a series of similar TPBVP are solved successively and the solution of a TPBPV is used as first guess for the next one. This is applied by first solving a TPBVP with a smoothed forced coasting throttle from Eq. 25, which has a greater convergence domain, and then using the solution as the first guess for the TPBVP with discontinuous forced coasting throttle from Eq. 17. Homotopy is also used to search for the optimal  $N_\alpha$ , where solution for one value is reused as first guess for the next value.

However, finding an initial guess for the first TPBVP is still difficult. Since the shape of an ANR trajectory is known from previous studies and existing tools like SIRENA, the initial guess for  $\mathbf{Y}$  can be found by hand through trial-and-error by adjusting the unknown parameters to roughly reproduce the expected trajectory shape. However, this method of initializing the initial guess requires a good knowledge of the behaviour and effect of the costates, and is thus a barrier to the use of the indirect approach. A priority in future works will be to provide a

Table 2: Satellite properties.

Property	Value
$m_0$	90 kg
$T$	0.002 N
ISP	2400 s
$c$	23 544 m s <sup>-1</sup>
$S$	1.1 m <sup>2</sup>
$C_D$	2.25

Table 3: Initial and target orbit.  $h_a$  and  $h_p$  are the apogee and perigee altitudes respectively.

Initial orbit		Final orbit	
$h_{a0}$	580 km	$h_{af}$	300 km
$h_{p0}$	580 km	$h_{pf}$	160 km
$\omega_0$	N/A	$\omega_f$	0°
$i_0$	98.2°	$i_f$	98.2°
$\Omega_0$	0°	$\Lambda_f$	-300 × 360°
$\alpha_0$	90°	$\alpha_f$	0°

method for consistently initializing the first guess without such knowledge.

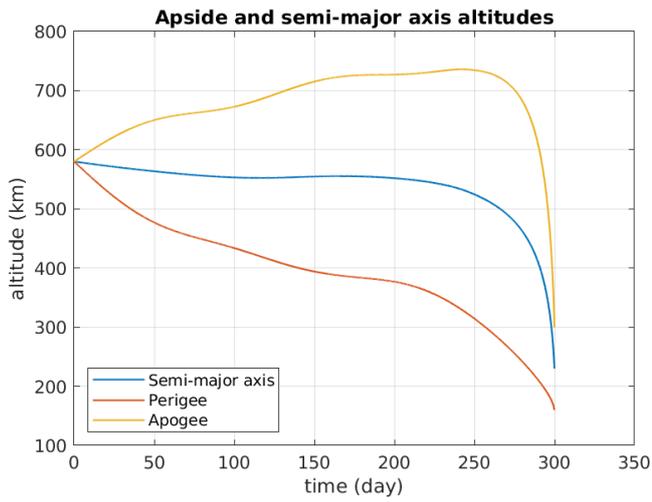
## 4. APPLICATION

In this section, an ANR open-loop guidance is computed using indirect optimization on a deorbitation scenario adapted from CNES's mission PlayerOne. After optimizing the initial guidance, a deviation from the nominal trajectory is simulated to demonstrate an update of the optimal guidance. The satellite's parameters are shown Table 2, and the initial and final orbital elements are shown Table 3. It is assumed that the solar panels are oriented face to the wind for passage at low-altitude perigee since this is the most stable orientation, so the given cross-section corresponds to this configuration. A constant solar activity is used for the NRLMSISE-00 atmosphere model with an  $F_{10.7}$  index of 100 and a geomagnetic index of 4. The satellite is not allowed to perform thrust below  $h_{min} = 250$  km, which is represented by the forced coasting constraint in Eq. 5.

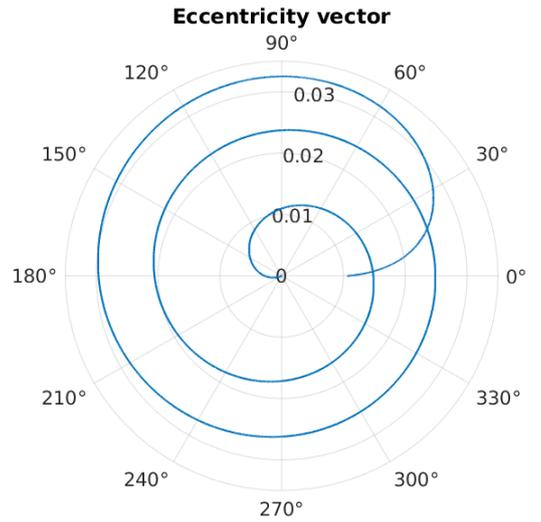
### 4.1. Initial trajectory computation

The fuel-optimal optimal guidance is first optimized using the smoothed forced coasting throttle in Eq. 25, with a smoothing factor  $\epsilon = 0.1$ . This optimization is repeated with successive values for the number of AoL turns to find the one that minimizes the consumed mass  $\Delta_m$ . The search start at  $N_\alpha = 4505$ , and since the variation of  $\Delta_m$  is very small between two consecutive  $N_\alpha$ , the number of turns is increased with a step of 5. Once the optimal  $N_\alpha^*$  is found, the guidance is optimized again with the

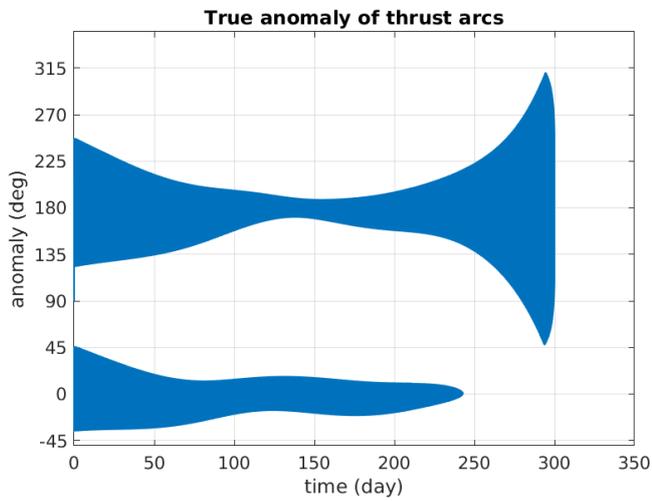
Figure 1: Fuel-optimal ANR trajectory



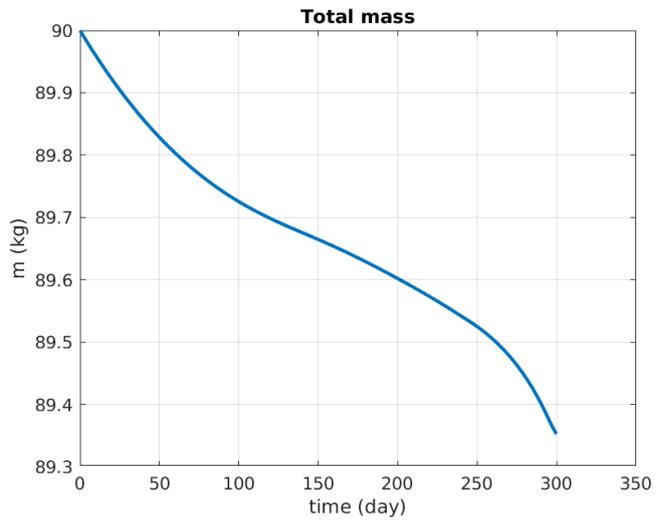
(a) Mean apside altitudes versus time



(b) Mean eccentricity vector, starting at 0 at initial date and rotating under the secular effect of Earth potential zonal harmonics



(c) True anomaly of thrust arcs versus time



(d) Satellite mass versus time

Table 4: Summary of solved TPBVP, with the number of AoL turns  $N_\alpha$ , the smoothing factor  $\epsilon$  and the fuel consumption of the optimal trajectory  $\Delta m$

$N_\alpha$	$\epsilon$	$\Delta m$ (kg)	Duration (min)
4505	0,1	0,64883	25
4510	0,1	0,64844	21
4515	0,1	0,64803	31
4520	0,1	0,64762	76
4525	0,1	0,64743	41
4530	0,1	0,64765	34
4525	0	0,64743	80

discontinuous forced coasting throttle in Eq. 17. This homotopic search is summarized in Table 4. Solving a TPBVP takes between 20 to 80 minutes with 14 threads on a 4.8 GHz processor. The optimal solution is found for  $N_\alpha^* = 4525$  for a total computation time of 5 hours.

The trajectory resulting from the optimized guidance is shown Figure 1a and Figure 1b, the mass in Figure 1d, and the location and width of thrust arcs are shown Figure 1c. The most surprising features, for a fuel-optimal trajectory, are the apogee altitude increase at the beginning and the increasingly large thrust arcs around apogee toward the end. This is because the total loss of apogee altitude from drag  $\Delta h_a^{drag}$  is greater than the apogee difference between initial and target orbit  $\Delta h_a^*$ , forcing the guidance to spend fuel to increase the apogee using thrust.

$$\Delta h_a^{thrust} = \Delta h_a^* - \Delta h_a^{drag} \quad (27)$$

To reduce  $\Delta h_a^{drag}$  and thus  $\Delta h_a^{thrust}$ , bigger perigee decrements can be made during the final descent to reduce the number of passes at low altitude perigee, but this also cost fuel since it require longer and less efficient thrust arcs around apogee. Thus, there is a trade-off between spending fuel to reduce  $\Delta h_a^{drag}$  or spending fuel on  $\Delta h_a^{thrust}$ . As the perigee altitude decreases and the drag force increases, this trade-off shifts in favor of accelerating the descent. As a consequence, the thrust arc length progressively increases at the end of the trajectory, until the thrust altitude constraint in Eq. 5 limits the thrust arc length at the very end. Such long thrust arcs are not feasible for most satellites, so a priority in further works will be to enforce a limitation on thrust duration.

## 4.2. Trajectory update

During end-of-life operations, the satellite will deviate from the predicted trajectory due to modeling uncertainties or contingencies in the execution of the maneuver plan. In addition, the short-term solar activity and atmospheric density predictions may change, which affects the spacecraft dynamics. As a consequence, the open-loop guidance must be updated regularly to account for those deviations. Consider a previously computed nominal state  $\bar{x}(t)$  and costate  $\bar{\lambda}(t)$  trajectory, and a new deviated state  $x(t_u) \neq \bar{x}(t_u)$  at date  $t_u$ . An updated TPBVP

Table 5: Relative difference between the costate components at  $t_u$  before and after the update.

Component	Relative difference
$\lambda_p$	1.35 %
$\lambda_{ex}$	0.02 %
$\lambda_{ey}$	-0.01 %
$\lambda_i$	1.02 %
$\lambda_\Omega$	0.07 %
$\lambda_\alpha$	1.75 %
$\lambda_p$	0.03 %

can be defined with the same final boundary conditions but with the new initial date  $t_u$  and state  $x(t_u)$ . The updated open-loop guidance is then obtained by computing the new costate vector  $\lambda(t_u)$  that solves this updated TPBVP. We make the assumption that if the state  $x(t_u) - \bar{x}(t_u)$  is small enough, the costate difference  $\lambda(t_u) - \bar{\lambda}(t_u)$  is also small, and the previous trajectory costate  $\bar{\lambda}(t_u)$  is thus a suitable initial guess for the shooting method.

To demonstrate the update process, a contingency is simulated on the previously computed trajectory. At time  $t_{cut} = 60$  day, the nominal state  $\bar{x}(t_{cut})$  is propagated for 10 orbital periods without any thrust to obtain a deviated state  $x(t_u)$  at time  $t_u$ . The new TPBVP is solved directly with the discontinuous forced coasting throttle in Eq. 17, and the initial guess for  $\mathbf{Y}$  is constructed by taking  $\bar{\lambda}(t_u)$ ,  $\bar{\lambda}(t_f)$ ,  $\bar{\Omega}(t_f)$  and  $\bar{m}(t_f)$  from the previous trajectory. The optimization converges successfully in 18 minutes. The update has no significant impact on the shape of the trajectory or the mass consumption, so the trajectory is not shown again.

The relative difference  $\lambda(t_u) - \bar{\lambda}(t_u)$  is small as expected, as shown in Table 5. This suggests that a closed-loop guidance could be constructed by mapping deviations of the state to costate adjustments. For example, Sidhoum and Oguri (2024) proposes a robust closed-loop solution where a series of state feedbacks on the costate are optimized to minimize the state covariance at critical dates.

## CONCLUSION

In this paper a method for computing a low-thrust optimal Assisted Natural Reentry open-loop guidance using indirect optimization is presented. The minimum fuel ANR optimization problem is defined, with an interface orbit defined relatively to the rotating Earth and an altitude constraint on the thrust. This optimization problem is then converted into a Two Point Boundary Value Problem using the Pontryagin's Maximum Principle to derive the necessary conditions of optimality. To reduce the difficulty in finding an initial guess for the TPBVP's solution, a multiple shooting method and an intermediate TPBVP with smoothed throttle constraint are used to improve the convergence domain. Finally, the initial guess

is found through trial-and-error and the TPBVP is solved to obtain the optimal guidance.

To demonstrate that indirect optimization is a potential alternative to existing ANR strategies computation tools, this method is successfully applied on a low-thrust ANR scenario to compute and then update a fuel-optimal trajectory. However, further work is needed to enforce maneuver duration constraints and to provide a practical method for finding an initial guess. Ultimately, the main goal behind the development of this open-loop guidance solution is to convert it into a robust and autonomous closed-loop guidance, which would reduce the operational cost of Assisted Natural Reentry.

## REFERENCES

- Borobia, Elisabet Cid, Claire Frémeaux, and Jean-François Goester (2017). “Fast Re-Entry Deorbitation with Acceptable Risk Level”. In: *7th European Conference on Space Debris*, pp. 18–21.
- Goester, Jean-François, Aurélie Bellucci, and Claire Fremeaux (2022). “CNES tools and methods for de-orbit and re-entry trajectories evaluation”. In: Artwork Size: 14 pages Medium: PDF Publisher: Proceedings of the 9th European Conference for Aerospace Sciences. Lille, France, 27 June - 1 July, 2022, 14 pages. DOI: 10.13009/EUCASS2022-7020.
- Lagadec, Kristen and Dorian Gegout (2023). “Assisted Reentry for All-Electric LEO Spacecraft”. In: *Papers of ESA GNC-ICATT 2023*. ESA 12th International Conference on Guidance Navigation and Control and 9th International Conference on Astrodynamics Tools and Techniques. ESA. DOI: 10.5270/esa-gnc-icatt-2023-096.
- Pontani, Mauro (2021). “Optimal Space Trajectories with Multiple Coast Arcs Using Modified Equinoctial Elements”. In: *Journal of Optimization Theory and Applications* 191.2, pp. 545–574. ISSN: 1573-2878. DOI: 10.1007/s10957-021-01867-2.
- Rutherford, D. E. (1964). “Optimal Trajectories For Space Navigation. By D. F. Lawden. Pp. viii, 126. 21s. net. 1963. (Butterworth and Co.)” In: *The Mathematical Gazette* 48.366, pp. 478–479. ISSN: 0025-5572, 2056-6328. DOI: 10.2307/3611765.
- Sidhoum, Yanis and Kenshiro Oguri (2024). “Robust Low-thrust Trajectory Correction Planning under Uncertainty: Primer Vector Theory Approach”. In.