



HAL
open science

Global optimization approach for the ascent problem of multi-stage launchers

Olivier Bokanowski, Eric Bourgeois, A Désilles, Hasnaa Zidani

► **To cite this version:**

Olivier Bokanowski, Eric Bourgeois, A Désilles, Hasnaa Zidani. Global optimization approach for the ascent problem of multi-stage launchers. In: Bock H.G., Jäger W., Kostina E., Phu H.X. (eds). Modeling, Simulation and Optimization of Complex Processes HPSC 2018. Springer, Cham., pp.1–42, 2021, 978-3-030-55239-8. 10.1007/978-3-030-55240-4_1 . hal-01113819

HAL Id: hal-01113819

<https://ensta-paris.hal.science/hal-01113819>

Submitted on 6 Feb 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Global optimization approach for the climbing problem of multi-stage launchers*

O. Bokanowski[†] E. Bourgeois[‡] A. Désilles[§] H. Zidani[¶]

Abstract

This paper deals with a problem of trajectory optimization of the flight phases of a three-stage launcher. The aim of this optimization problem is to minimize the consumption of ergols that is need to steer the launcher from the Earth to the GEO. Here we use a global optimization procedure based on Hamilton-Jacobi-Bellman approach and consider a complete model including the transfer from the GTO to the GEO orbit. The Hamilton-Jacobi approach proposed here takes also into account parametric optimisation that appears in the flight phases. The work presented in this paper has been performed in the frame of CNES Launchers' Research and Technology program

AMS Classification: 34H05, 49J15, 49L20

Keywords: multi-stage launchers, trajectory optimization, minimum time problem, Hamilton-Jacobi-Bellman approach

1 Introduction - A revoir

Trajectory optimization for space launchers is a classical problem in optimal control. The pioneering Goddard [1] problem is perhaps the simplest model. It consists in maximizing the final altitude of the rocket, for a vertical flight, with a given initial propellant allocation. In one dimension this model is described by three state variables: the altitude r of the launcher, its velocity v and its mass m . The system is submitted to the aerodynamic force (the drag \vec{F}_D) and is controlled via the thrust force \vec{F}_T . Since this work, several studies were made on theoretical properties of the optimal trajectories [2, 3, 4] and numerical methods allowing to calculate these trajectories [2, 5, 4, 6, 7, 8, 9, 10], and in particular [11, 12, 13, 14] for the ascent problem.

Both classes of methods mentioned above present several difficulties when the optimization problem is non-convex. Indeed in this case the above methods are not able to avoid local minima, especially for shooting methods that have the reputation to have a small convergence radius. The initialization of these methods can also be very hard.

In this work, we investigate the resolution of the climbing problem by the so-called Hamilton-Jacobi-Bellman (HJB) approach. This approach is based on the Dynamic Programming Principle (DPP) studied by R. Bellman [15]. It leads to a characterization of the value function as a solution of an HJB equation

*This work is partially supported by Centre National d'Études Spatiales (CNES), under the grant RT-CR-430-1304-CNES.

[†]Laboratoire J.-L. Lions, Université Pierre et Marie Curie 75252 Paris Cedex 05, France, and UFR de Mathématiques, Site Chevaleret, Université Paris-Diderot, 75205 Paris Cedex, France. boka@math.jussieu.fr

[‡]CNES Launcher Directorate, 52 rue Jacques Hillairet, 75012 Paris, France. eric.bourgeois@cnes.fr

[§]Unité des Mathématiques Appliquées (UMA), Ensta ParisTech, 828 Bd des Maréchaux, 91762 Palaiseau Cedex, Anna.Desilles@ensta-paristech.fr

[¶]Unité des Mathématiques Appliquées (UMA), Ensta ParisTech, 828 Bd des Maréchaux, 91762 Palaiseau Cedex, Hasnaa.Zidani@ensta-paristech.fr

which is a first order nonlinear Partial Differential Equation (PDE) in dimension d , where d is the number of variables involved in the problem. The HJB equation may be viewed as a differential form of the DPP. An important breakthrough for this approach occurred in the 80's, when the notion of *viscosity solutions* of nonlinear PDEs was introduced by Crandall and Lions [16, 17, 18]. This viscosity theory allows to establish a rigorous framework for the theoretical and numerical study the HJB equations arising in optimal control theory. The contributions in this direction do not cease growing, see the book of Bardi and Capuzzo-Dolcetta [19], and the Appendix A by Falcone in the same reference.

An interesting by-product of the HJB approach is the synthesis of the optimal control in feedback form. Once the HJB equation is solved, for any starting point, the reconstruction of the optimal trajectory can be performed in real time. Also the method gives a global optimum and do not need any initialization procedure, see [20]. Another advantage of the HJB approach is that state or mixed state-control constraints can be taken into account. Of course, we should be careful when handling viscosity solutions for the HJB equation associated with such problems, see [21, 22].

Although the theoretical framework of the HJB approach is well known, and despite its advantages, this approach is seldom used in real control problem, because of the difficulties of computing the solution of the HJB equation in higher dimensions [23].

In this work we aim at showing that combining several new techniques for the HJB approach we can obtain efficient solutions to a fully nonlinear control problem.

More precisely, we investigate the potential of the HJB approach for the climbing problem in the case a launcher of the Ariane5 class (the simulations performed in this paper will consider a launcher with capacity of injection of roughly 10T on a GTO orbit). For a given payload (fixed final mass), we aim at steering the launcher to the Geostationary Orbit (GEO) with *minimal* propellant consumption.

The paper is organized as follows. Section 2 is devoted to the presentation of the physical model and the related optimal control problem. In Section 3 we present the optimization approach that will be used in this paper. The results of numerical simulations are presented in section 4. Some technical definitions are given in appendix A.

2 Problem statement. Mathematical formulation.

This section is aimed at describing the physical problem and its mathematical formulation. Several frames will be defined to describe the motion of the launcher in the most suitable way. Then the list of forces involved during the flight sequence will be defined. The differential system that governs the trajectory will be obtained by Newton's law.

2.1 Physical model

Let O denotes the center of the Earth. We define a first frame $\mathcal{R}_I = (O, \vec{i}_I, \vec{j}_I, \vec{k}_I)$ to be considered as inertial. The vector \vec{k}_I is co-linear with the North-South axis of rotation, the vector \vec{i}_I is located in the equatorial plane of the Earth and points to the Greenwich meridian at an elected date set here as $t = 0$. The vector \vec{j}_I completes the orthonormal frame (see figure 1(a)). Consider also the frame $\mathcal{R}_R = (O, \vec{i}_R, \vec{j}_R, \vec{k}_R)$ (see figure 1(b)) that coincides with \mathcal{R}_I at time $t = 0$ and that is rotating with the Earth around the axis $\vec{k}_I = \vec{k}_R$ with the angular velocity Ω .

In all the sequel, we denote by r_T the Earth's mean radius and G the mass center of the vehicle. The spherical coordinates of G are (r, L, ℓ) , where r is the distance between G and O , L is the longitude and ℓ is the latitude (voir figure 2(a)). The vehicle's position at time $t = 0$ will be denoted G_0 and its spherical candidates are (r_0, L_0, ℓ_0) .

Two other local frames will be also used: A vertical local frame $\mathcal{R}_V = (G, \vec{i}_V, \vec{j}_V, \vec{k}_V)$ centred at G and defined such that \vec{k}_V is colinear with \vec{r}_G and pointing in the same direction. The vector \vec{j}_V is in the orthogonal plane to \vec{k}_V and pointing to the local North. The third vector $\vec{i}_V := \vec{j}_V \wedge \vec{k}_V$ is defined in such way to complete the orthonormal frame (see figure 2).

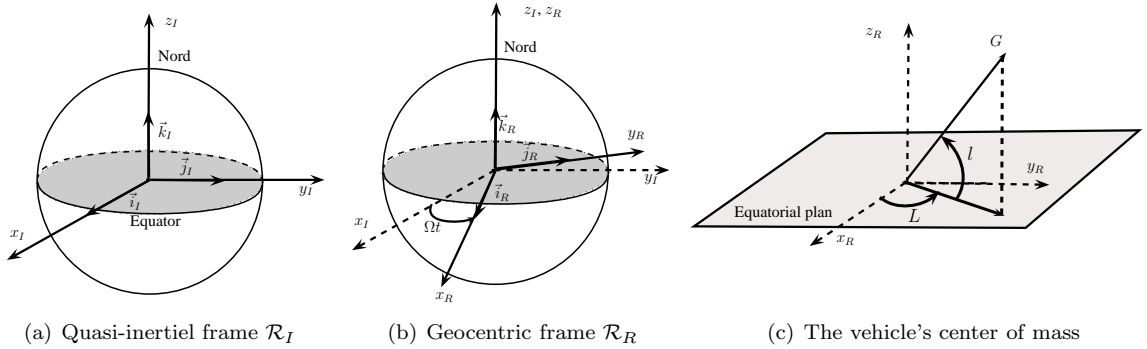


Figure 1: Quasi inertial and geocentric frames

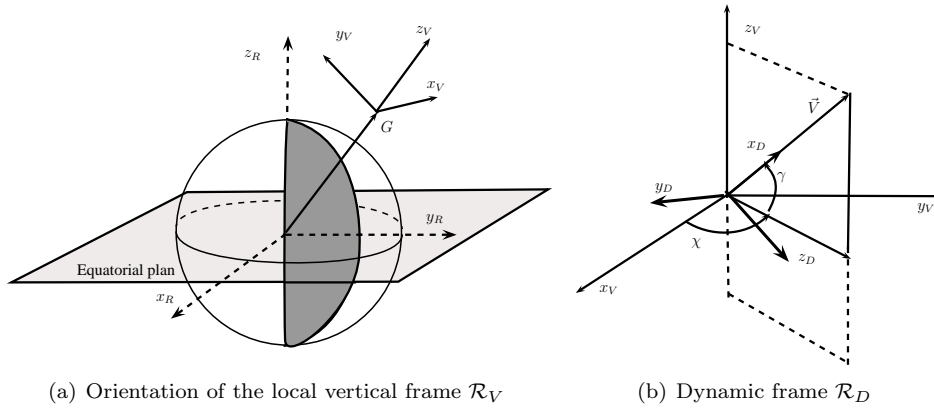


Figure 2: Local vertical and dynamic frames

The frame \mathcal{R}_V will be linked to the center of mass of the launcher and should therefore evolve over time along the trajectory of the launcher. We will also need to consider the local inertial frame \mathcal{R}_{IL} defined as the vertical local coordinate $\mathcal{R}_R = (G_0, \vec{i}_V, \vec{j}_V, \vec{k}_V)$ at time $t = 0$.

Let $X : t \mapsto (x(t), y(t), z(t))$ be the trajectory of G in the quasi-inertial frame \mathcal{R}_R , and let $\vec{V} := \dot{x}\vec{i}_R + \dot{y}\vec{j}_R + \dot{z}\vec{k}_R$ be the relative velocity. We define \vec{V} in the local frame \mathcal{R}_V by its polar coordinate: the modulus v , the azimuth χ which is the angle entre \vec{i}_V et la projection de \vec{V} sur (\vec{i}_V, \vec{j}_V) , the path inclinaison (flight angle) γ which is the angle between the projection of \vec{V} on (\vec{i}_V, \vec{j}_V) and \vec{i}_V (see Figure 2).

Likewise the local coordinate system related to the trajectory center of mass, we define also a local tangential frame related to the velocity of the launcher. For this we introduce the "dynamical" orthonormal frame $\mathcal{R}_D = (G, \vec{i}_D, \vec{j}_D, \vec{k}_D)$ defined such that \vec{i}_D has the same direction as the velocity \vec{V} (i.e., $\vec{i}_D = \frac{\vec{V}}{\|\vec{V}\|}$), \vec{k}_D is the unitary vector in the plane (\vec{i}_D, \vec{k}_V) perpendicular to \vec{i}_D and satisfying $\vec{k}_V \cdot \vec{k}_D < 0$ and $\vec{j}_D = \vec{k}_D \wedge \vec{i}_D$ (see figure 2). The frame \mathcal{R}_D will be useful to express some forces that act on the launcher.

According to the flight phase of the launcher, it may be appropriate to consider the Cartesian coor-

coordinates of the position of G and its velocity (\vec{X}, \vec{V}) in the frame \mathcal{R}_R with

$$\vec{X} = \begin{pmatrix} x \\ y \\ y \end{pmatrix}, \quad \vec{V} = \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix}.$$

The launcher may be also represented by the spherical coordinates of the position of G in \mathcal{R}_R and those of the velocity in the frame \mathcal{R}_V . These coordinates will be denoted by $(r, L, l, v, \chi, \gamma)$. The formulas to pass from one coordinate system into another are classical and can be found in [?, Reference de Anya].

2.2 Axis, angles, and forces

During the phases of flight, the launcher is subject to different forces. We will express each of these forces in the most appropriate frame. For this, we will need to introduce some additional notations for the angles between the velocity and the axis of the launcher:

- angle of incidence α : is the angle between the velocity vector and the axis of the launcher in the plane (\vec{i}_V, \vec{k}_V) ;
- sideslip angle δ : is the angle measured in the plane (\vec{i}_V, \vec{j}_V) ;
- and the bank or heeling angle μ : is the angle between the axis of the shuttle and the axis \vec{k}_V .

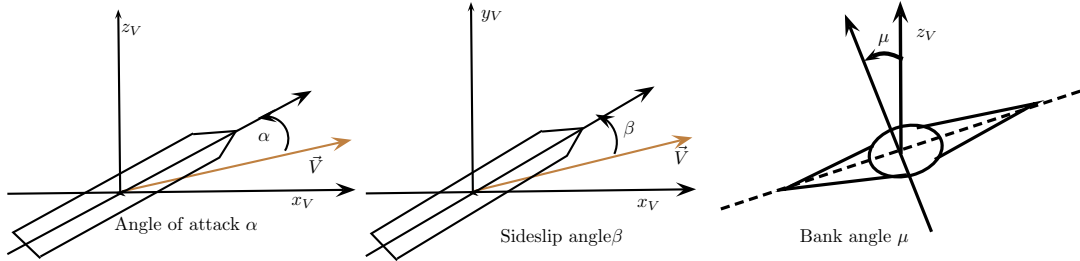


Figure 3: Angles of the launcher

The launcher is subject to different forces that we describe here below:

- **Gravitational force:** $\vec{F}_g = m\vec{g}$, where m is the mass of the vehicle and $\vec{g} = -g(r)\vec{e}_r = -g(r)(\cos\gamma\vec{i}_R + \sin\gamma\vec{j}_R)$ is the gravitational field. The term $g(r)$ is given by

$$g(r) := \frac{\mu}{r^2},$$

where μ is Earth's gravitational constant (we neglect high order term in the harmonic expansion of the gravitational field).

- **Aerodynamic forces:** The best frame suited to express the aerodynamic forces here is \mathcal{R}_D . We will assume that the plane of symmetry of the vehicle coincides with the plane (x_V, z_V) of the dynamical reference frame (i.e., the sideslip and the heeling angles are zero during atmospheric flight, see figure 3). Under this assumption, the aerodynamic forces are:

- Drag force: $\vec{F}_D = -F_D(r, v, \alpha)\vec{i}_D$ opposite to the velocity \vec{V} . In this paper, we consider that F_D is given by

$$F_D(r, v, \alpha) = S_r Q(r, v)(C_x(r, v) + C_{xx}|\alpha|)$$

where α is the incidence angle and $Q(r, v)$ is the dynamic pressure defined by $Q(r, v) = \frac{1}{2}\rho(r)v^2$, with $\rho(r)$ the atmospheric density, S_r is the reference surface, C_x and C_{xx} are aerodynamic coefficients.

– Lift function: $\vec{F}_L = -F_L(r, v, \alpha)\vec{k}_D$ orthogonal to the velocity vector, and with modulus

$$F_L(r, v, \alpha) = S_r Q(r, v)(C_L(r, v) + C_{LL}|\alpha|).$$

Where $C_L(r, v)$, C_{LL} are aerodynamic coefficients.

The resultant of the aerodynamic forces is $\vec{F}_A = \vec{F}_D + \vec{F}_L$.

- **Thrust force:** It is assumed that the direction of the thrust force coincides with the axis of the launcher. Assume ($\mu = 0$), the orientation of the thrust in the dynamical reference frame is defined by the incidence α and the slip δ . Then the thrust force is given by:

$$\vec{F}_T = F_T(r)(\cos\alpha \cos\delta \vec{i}_D - \cos\alpha \sin\delta \vec{j}_D + \sin\alpha \vec{k}_D),$$

with $F_T(r) = \beta g_0 I_{sp} - SP(r)$ where $g_0 = 9.81 \text{ ms}^{-2}$, $P(r)$ is the atmospheric pressure, and β (flow rate), I_{sp} (specific impulse) and S (surface) depend on the flight phases (see section 2.4).

- **Coriolis force F_C et centripetal force F_{CP} .** These functions are defined by:

$$\vec{F}_C = 2m\vec{\Omega} \wedge \vec{V} \quad \text{et} \quad \vec{F}_{CP} = m\vec{\Omega} \wedge (\vec{\Omega} \wedge \vec{OG}),$$

where $\vec{\Omega}$ is the Earth's angular velocity. These two forces are important to be taken into account as far as the launchers's trajectory is represented in a relative reference frame and not in the inertial one.

Remarque 2.1. *The aerodynamic forces are generally defined with the relative launcher's speed with respect to the air (or equivalently, relative to the ground when the atmosphere is considered static and therefore rigidly driven by the Earth's rotation).*

Remarque 2.2. *The sideslip angle is assumed to be zero during the atmospheric flight (i.e., the launcher flies with zero incidence angle). To be more rigorous, if we consider a static atmosphere, so rigidly driven by the Earth's rotation, the launcher has a non-zero velocity component out of plane relative to the air or on the ground (except in the limiting case of a launch from zero latitude). This assumption seems reasonable for a launch close to Ecuador, since the sideslip angle generally remains very low in the atmospheric phase .*

2.3 Motion's equations

Taking into account all these forces, and using Newton's laws of motion, we get:

$$\frac{d\vec{X}}{dt} = \vec{V}, \tag{1a}$$

$$m \frac{d\vec{V}}{dt} = \vec{F}_g + \vec{F}_A + \vec{F}_T - 2m\vec{\Omega} \wedge \vec{V} - m\vec{\Omega} \wedge (\vec{\Omega} \wedge \vec{OG}). \tag{1b}$$

Straightforward calculations yield to the motion's equation in the spherical coordinates:

$$\frac{dr}{dt} = v \sin \gamma \quad (2a)$$

$$\frac{dL}{dt} = \frac{v \cos \gamma \sin \chi}{r \cos \ell} \quad (2b)$$

$$\frac{d\ell}{dt} = \frac{v}{r} \cos \gamma \cos \chi \quad (2c)$$

$$\begin{aligned} \frac{dv}{dt} = & -g(r) \sin \gamma - \frac{F_D(r, v, \alpha)}{m} + \frac{F_T(r)}{m} \cos \alpha \cos(\delta) \\ & + \Omega^2 r \cos \ell (\sin \gamma \cos \ell - \cos \gamma \sin \ell \cos \chi) \end{aligned} \quad (2d)$$

$$\begin{aligned} \frac{d\gamma}{dt} = & -\cos \gamma \left(\frac{g(r)}{v} - \frac{v}{r} \right) + \frac{F_L(r, v, \alpha)}{vm} + \frac{F_T(r)}{vm} \sin \alpha \\ & + 2\Omega \cos \ell \sin \chi + \Omega^2 \frac{r}{v} \cos \ell (\cos \gamma \cos \ell + \sin \gamma \sin \ell \cos \chi) \end{aligned} \quad (2e)$$

$$\begin{aligned} \frac{d\chi}{dt} = & -\frac{v}{r} \cos \gamma \tan \ell \sin \chi - 2\Omega (\sin \ell - \tan \gamma \cos \ell \cos \chi) \\ & + \Omega^2 \frac{r}{v} \frac{\sin \ell \cos \ell \sin \chi}{\cos \gamma} - \frac{F_T(r)}{m v \cos(\gamma)} \cos(\alpha) \sin(\delta) \end{aligned} \quad (2f)$$

Of course, the evolution of the mass is also very important to get a complete model. This evolution is given by the ODE:

$$\dot{m}(t) = \beta(t). \quad (3)$$

where the function β is known and represents the consumption flow rate and depends on the launcher's parameters. The mass of the launcher includes the structure's mass, the payload, and propellant's mass:

$$m(t) := \underbrace{M_{EAP} + M_{E1} + M_{E2} + m_F}_{Structure} + \underbrace{m_{CU}}_{Payload} + \underbrace{M_{P,EAP}(t) + M_{P,E1}(t) + M_{P,E2}(t)}_{Propellant},$$

where $M_{EAP}, M_{E1}, M_{E2}, m_F$ denote respectively the mass of the boosters, the first and second stages, and the firin; m_{CU} denotes the mass of the payload; and $M_{P,EAP}(t), M_{P,E1}(t), M_{P,E2}(t)$ denote the mass, at time $t \geq 0$ of propellant respectively in the boosters, in the first and second stages.

Equations (1) and (2) are two different formulations for the same motion. In the sequel, we will see that the first formulation is convenient to describe the atmospheric phase, while the second is more suitable for the flight outside the atmosphere (more details will be given in the next sections).

2.4 The flight's phases

The launcher that is considered in this paper is of Ariane-5 type with three-stages. The latter are parts of the launcher that contain propellant and provide propulsion for the launcher. We denote by β_{EAP}, β_{E1} and β_{E2} the mass flow rates for the boosters, the first and the second stage respectively. These rates are functions depending in time, see figure ??.

Our aim is to minimize the ergol's consumption while steering the vehicle from a given initial position on the Earth to the GEO. The launcher evolves according to the following phases.

Phase 0: This phase starts when the vehicle leaves the launch base. Both boosters along with the stage E_1 are ignited and consume propellant with flow rates β_{EAP} and β_{E1} respectively .

- Phase 0-1: vertical flight for a fixed time τ_0 . The flight is vertical relative to the ground in the approximate initial \mathcal{R}_{IL} . Thus, during this phase, the orientations of the thrust are constant :

$$\theta(t) := \theta_0 = \frac{\pi}{2}, \quad \psi(t) := \psi_0 \quad \text{for } t \in [0, \tau_0],$$

where ψ_0 is the shooting azimuth.

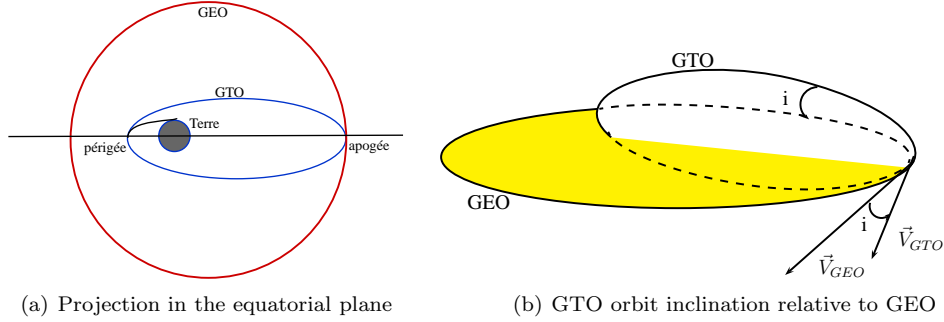


Figure 4: Orbits GEO et GTO

- Phase 0-2: the launcher rotates with constant speed $\omega_{\text{bas}c}$ changing its orientation during a time interval $\tau_{\text{bas}c}$:

$$\theta(t) = \frac{\pi}{2} + \omega_{\text{bas}c}t, \quad \psi(t) = \psi_0 \quad \text{for } t \in [\tau_0, \tau_0 + \tau_{\text{bas}c}].$$

- Phase 0-3: The direction of the thrust is then fixed at the final values of the previous sub-phase until the angle of incidence is zero; i.e. when $\alpha = 0$ or equivalently

$$\cos(\alpha) = \frac{\vec{F}_T \cdot \vec{V}}{\|\vec{F}_T\| \|\vec{V}\|} \equiv 1.$$

- Phase 0-4: Zero incidence flight until rejection of the boosters. In this paper, we consider that the boosters are ejected after the exhaustion of the propellants, and since the propellant flow β_{EAP} is known the rejection's time is known and will be denoted t_1 . Note that this assumption is not restrictive and the whole study can be extended to other situations where the boosters are ejected after reaching an accelerometric threshold or a threshold ratio of propulsive forces (even if it remains some propellant).

The durations τ_0 , $\tau_{\text{bas}c}$ and t_1 are fixed, while the value of the parameters ψ_0 and $\omega_{\text{bas}c}$ are unknowns that must be determined in such a way to optimize the launcher's consumption. The set of possible positions corresponding to a large sample of these parameters can be obtained by a simple integration of the motion's equations. The computation of the trajectories can be performed in parallel and with a high accuracy.

Phase 1. during this phase the propulsion is assured only by first stage E_1 . The engine is on and consume with the propellant flow β_{E1} until exhaustion. The firing is rejected after reaching a threshold heat flux. The ejection of the stage E_1 may be followed by a ballistic short flight during a fixed duration τ_1 that corresponds to the end of the propellant contained in the stage E_1 .

Phase 2. during this phase, the second stage is ignited. It ends at the time of injection of the launcher on a GTO orbit near its perigee. The choice of the transfer orbit GTO is considered as an optimization parameter. It requires the GTO having its ascending node (one of the two intersection points of the orbit with the plane Equatorial) located on GEO orbit.

Phase 3. Once the launcher has reached a GTO, the engine of the second floor is off. Then follows a ballistic flight phase until it reaches the apogee of the GTO. The duration of this flight is not fixed and depends on the GTO parameters. In our study, we assume that the GTO- GEO orbital transfer is performed through an impulsional boost (to change the velocity's modulus and direction). The amount of propellant required for the orbital transfer depends on the the GTO parameters and the launcher's mass. It can be determined via Tsiolkovski formula, see appendix A. In particular, we note that this amount of propellant depends only on the orbital inclination i of the GTO relative to the GEO (see figure 4(b)).

2.5 Optimal control problem

The optimization problem that we aim to study consists of:

Minimizing the launcher's consumption for steering a given payload m_{CU} until the GEO

Control laws: During the Phase 0, the launcher's trajectory is defined by the choice of the parameters $\psi_0, \omega_{\text{bas}c}$. Then the launcher will be controlled by measurable functions representing the evolution of the incidence $\alpha(\cdot)$ and the sideslip angle $\delta(\cdot)$. In the following, we will use the notation $u = (\alpha, \delta)$ for the control law. The admissible controls are measurable functions that take their values in the compact $U = [-\pi, \pi]^2$. Note also \mathcal{U}_{ad} the set all those admissible control inputs.

Parameters. We distinguish the control variables, which are time dependent functions, from the parameters that do not evolve in time but are still important for the path optimization. The parameters are:

- $p_1 := (\psi_0, \omega_{\text{bas}c}) \in \mathcal{P}_{\text{Ini}}$, where ψ_0 is the shooting azimuth, $\omega_{\text{bas}c}$ is the angular velocity.
- the inclination angle \mathbf{i} of the GTO from the GEO. This parameter lies in an interval \mathcal{I} .

Objective function. Let $t_{f,GEO}$ be the time of injection on the GEO and t_f be the time of injection on the GTO orbit. The objective is to maximize the mass of remaining propellant after the injection on the GEO orbit: $M_{P,E2}(t_{f,GEO})$. This remaining mass can be expressed as

$$M_{P,E2}(t_{f,GEO}) = M_{P,E2}(t_f) - \Delta M(t_f, \mathbf{i})$$

where \mathbf{i} is the inclination of the GTO orbit, and $\Delta M(t_f, \mathbf{i})$ is the amount of propellant for orbit transfer between GTO and GEO. The later is explicitly given using the Tseolkovski formula (28) such that

$$\Delta V(\mathbf{i}) = g \cdot \text{Isp} \cdot \ln \left(\frac{m(t_f)}{t_f - \Delta M(t_f, \mathbf{i})} \right)$$

where $\Delta V(\mathbf{i})$ is defined by (27) and $m(t_f)$ is the global mass at the time t_f defined by

$$m(t_f) = m_{CU} + M_{E2} + M_{P,E2}(t_f).$$

One can easily deduce the mass of remaining propellant is equal to

$$M_{P,E2}(t_{f,GEO}) = M_{P,E2}(t_f) \cdot e^{-\frac{\Delta V(\mathbf{i})}{g \cdot \text{Isp}}} - (m_{CU} + M_{E2}) \cdot \left(1 - e^{-\frac{\Delta V(\mathbf{i})}{g \cdot \text{Isp}}} \right) \quad (4)$$

Mathematical formulation of the control problem. Suppose the state of the launcher at any time is described by a vector $y \in \mathbb{R}^d$ (representing the position, velocity and mass).

$$\begin{cases} \dot{y}(t) = f(t, y(t), u(t), p), & t \geq 0, \\ y(0) = y_0, \end{cases} \quad (5)$$

with y_0 is the initial condition corresponding to position 0, velocity 0, and the initial mass of the launcher (total mass including the structure's mass, the payload and the ergol). The optimization problem is then:

$$\left\{ \begin{array}{l} \text{Maximize } M_{P,E2}(t_f) - \Delta M(t_f, \mathbf{i}) \\ \text{subject to:} \\ \dot{y}(t) = f(t, y(t), u(t), p) \quad t \in (0, t_f), \\ y(0) = 0, \\ u(t) \in U, \text{ p.p. } t \in [0, t_f], \quad y(t) \in \mathcal{K}, \quad \forall t \in [0, t_f], \\ y(t_f) \in \mathcal{C}_i, \quad \mathbf{i} \in \mathcal{I}, \\ p \in \mathcal{P}_{\text{Ini}}, \quad t_f > 0. \end{array} \right. \quad (6)$$

The state-constraints set \mathcal{K} is a closed subset of \mathbb{R}^d that represents the set where the feasible trajectories lie (this set will be defined in the section devoted to numerical simulations). The target $\mathcal{C}_i \subset \mathbb{R}^d$ represents the GTO associated to the inclination angle $i \in \mathcal{I}$.

3 Global optimization approach

As described in the previous sections, the launcher's motion follows a dynamical system that we can write in a general form as:

$$\dot{y}(t) = f(t, y(t), u(t), p) \quad \text{for } t \geq 0, \quad y(0) = y_0, \quad (7)$$

where $u \in \mathcal{U}_{ad}$.

Let \mathcal{C} be a nonempty closed set of \mathbb{R}^d that will represent the target. Let \mathcal{K} be a nonempty closed set (of "state constraints"). These constraints are obtained via physical considerations and provide restrictions on the domain where the trajectory can lie (the set \mathcal{K} will be defined clearly in the section devoted to numerical simulations). For t a fixed positive time, we will say that a solution of (7) $y_x^{u,p}$ is admissible on $[0, t]$ if it is associated to an admissible control $u \in \mathcal{U}_{ad}$ and a feasible parameter $p \in P_{\text{Ini}}$, and $y_x^{u,p}(s)$ belongs to \mathcal{K} for every $s \in [0, t]$.

The control problem is defined by:

$$\left\{ \begin{array}{l} \inf \phi(t_f, y_x^{u,p}(t_f), i) \\ y_x^{u,p} \text{ is the solution of (7) associated to } (u(\cdot), p) \\ u(\cdot) \in \mathcal{U}_{ad}, p \in P_{\text{Ini}}, t_f > 0 \\ y_x^{u,p}(\tau) \in \mathcal{K} \quad \forall \tau \in [0, t_f], \\ \text{and } y_x^{u,p}(t_f) \in \mathcal{C}_i \quad \text{for } i \in \mathcal{I}. \end{array} \right. \quad (\mathcal{P})$$

The above optimization problem depends on several parameters (namely, final time t_f , $p = (\psi_0, \omega_{\text{bas}c})$, and \mathbf{i}) that have to be chosen to get the optimal cost. In general, to solve this problem with the HJB problem, the vector of parameters Π should be considered as an additional state vector evolving under the simple dynamics:

$$\dot{\Pi}(t) = 0.$$

This simple remark sets back the problem in the general framework where the state is $(y, \Pi)^\top$. However, in this formulation the dimension of the state will increase and will become $d + 4$ which automatically increases the dimension of the HJB equation associated to the control problem.

For the launcher's path optimisation problem, we are concerned by a control problem where the dynamics depends on the parameters in a very specific way. More precisely, there exist a fixed (known) time $t_1 > 0$ such that on the interval $[0, t_1]$, the function f does not depend on the control and depends only on 2 parameters in P_{Ini} . On $[t_1, t_2]$, the function depends only on the control variable and has no explicit dependency on parameters. In other terms, for every $(x, u) \in \mathbb{R}^d \times U$ and every $p \in P_{\text{Ini}}$, we have:

$$f(t, x, u, p) = \begin{cases} f_1(t, x, p) & \text{for } t \in (0, t_1) \\ f_2(t, x, u) & \text{for } t \in (t_1, t_f) \end{cases}$$

We introduce X_0 the set of all positions that can be reached from y_0 with parameters $p \in P_{\text{Ini}}$:

$$X_0 = \{y_x^p(t_1) \mid p \in P_{\text{Ini}}, \dot{y}_x^p(t) = f_1(t, y_x^p(t), p), y_x^p(0) = y_0\}.$$

Therefore, the problem (\mathcal{P}) is equivalent to:

$$\left\{ \begin{array}{l} \inf_{\mathbf{i} \in \mathcal{I}} \inf \phi(t_f, y_x^u(t_f), \mathbf{i}) \\ \dot{y}_x^u(t) = f(t, y_x^u(t), u(t)) \quad t \in (t_1, t_f), \\ y_x^u(t_1) \in X_0, \\ u \in \mathcal{U}_{ad}, \quad t_f > 0 \\ y_x^u(\tau) \in \mathcal{K} \quad \forall \tau \in [t_1, t_f], \\ \text{and } y_x^u(t_f) \in \mathcal{C}_i. \end{array} \right. \quad (8)$$

In order to perform a global optimization of the above problem, we will first discretize the set \mathcal{I} and consider a set of finite values $\mathbf{i} \in \mathcal{I}^\Delta$. For every value $\mathbf{i} \in \mathcal{I}^\Delta$, we first solve the sub-problem (\mathcal{P}_i) :

$$\left\{ \begin{array}{l} \inf \phi(t_f, y_x^u(t_f), \mathbf{i}) \\ \dot{y}_x^u(t) = f(t, y_x^u(t), \alpha(t)) \quad t \in (t_1, t_f), \\ y_x^u(t_1) \in X_0, \quad u \in \mathcal{U}_{ad}, \quad t_f > 0 \\ y_x^u(\theta) \in \mathcal{K} \quad \forall \theta \in [0, t_f], \\ \text{and } y_x^u(t_f) \in \mathcal{C}_i. \end{array} \right. \quad (9)$$

Then a comparison of the optimal performances of the (\mathcal{P}_i) 's can be carried out for finite number of parameters $\mathbf{i} \in \mathcal{I}$.

Now, the problem (\mathcal{P}_i) can be solved by using HJB theory. We will describe this part in the next subsection.

3.1 HJB approach for (\mathcal{P}_i)

Now, we consider the optimal control problem for steering the launcher until the GTO (with a given inclination angle \mathbf{i}). Keeping in mind the expression of the cost function (4), it turns out that the inner control problem (\mathcal{P}_i) (for a fixed \mathbf{i} is equivalent to the optimal control problem:

$$(\mathcal{P}) \quad \left\{ \begin{array}{l} \text{Minimize } t_f, \\ \text{with } \left\{ \begin{array}{l} \dot{y}_x(t) = f(t, y_x(t), u(t)), \quad t \in [t_1, t_f], \\ y_x(t_1) \in X_0, \\ t_f \geq 0, \quad u(t) \in U \quad \text{for a.e. } t \in [0, t_f], \\ y_x(t_f) \in \mathcal{C}_i, \end{array} \right. \end{array} \right.$$

where $f : \mathbb{R}_+ \times \mathbb{R}^d \times \mathcal{A} \rightarrow \mathbb{R}^d$ is the dynamics, $\mathcal{C}_i \subset \mathbb{R}^d$ is the target.

Let us recall some theoretical results concerning the HJB approach for problem (\mathcal{P}) . In this section, we assume that f and Ψ satisfy some classical assumptions:

- (A1) f is a continuous function, and for every $(t, x) \in \mathbb{R}_+ \times \mathbb{R}^d$ the set $f(t, x, \mathcal{A})$ is closed and convex. There exists $c_0 \geq 0$ s.t. $\sup_{a \in \mathcal{A}} |f(t, \xi, a)| \leq c_0(1 + |\xi|)$. Moreover, for every $R > 0$, there exists $L_R > 0$, such that

$$\forall t \in \mathbb{R}_+, \quad \forall \xi, z \in B(0, R), \quad \sup_{a \in \mathcal{A}} |f(t, \xi, a) - f(t, z, a)| \leq L_R |\xi - z|.$$

Consider the minimal time function, which associates to any point $x \in \mathbb{R}^d$ the minimal time needed to reach the target with an admissible trajectory y_x^u solution of (??) and satisfying $y_x^u(\theta) \in \mathcal{K}$:

$$\mathcal{T}(x) := \inf \{ t \geq 0, \exists \alpha \in L^\infty((0, t); \mathcal{A}), y_x^u(t) \in \mathcal{C}, \text{ and } y_x^u(\theta) \in \mathcal{K} \quad \forall \theta \in [0, t] \}. \quad (10)$$

Many works have been devoted to the regularity of the minimum time function \mathcal{T} . When $\mathcal{K} \equiv \mathbb{R}^d$, and under some local metric properties around the target, the function \mathcal{T} is the unique continuous viscosity solution of an HJB equation [?].

Here, without assuming any controllability assumption at the boundary of the target, and neither at the boundary of \mathcal{K} , the function \mathcal{T} may be discontinuous. Indeed, if, for $x \in \mathbb{R}^d$, no trajectory y_x^u reaches the target \mathcal{C} , or if any trajectory leaves \mathcal{K} before reaching the target, we set $\mathcal{T}(x) = +\infty$. Nevertheless, the next proposition states that \mathcal{T} is lsc and characterizes it by using the knowledge of the function ϑ .

To do so, we first consider a Lipschitz continuous function $\vartheta_0 : \mathbb{R}^d \rightarrow \mathbb{R}$ such that

$$\vartheta_0(x) \leq 0 \Leftrightarrow x \in \mathcal{C}. \quad (11)$$

For instance, we may choose $\vartheta_0(x) := d_{\mathcal{C}}(x)$; then ϑ_0 is Lipschitz continuous (see, for instance, [?]). In particular, we have $\text{capt}(0) = \mathcal{C} = \{x, \vartheta_0(x) \leq 0\}$.

Consider the value function u associated to the *Mayer problem* with final cost ϑ_0 :

$$u(x, t) := \inf\{\vartheta_0(y_x^\alpha(t)), \alpha \in \mathcal{A}_{ad}, y_x^\alpha(\theta) \in \mathcal{K} \forall \theta \in [0, t]\}. \quad (12)$$

It is well known that the capture basin is characterized by

$$\text{capt}(t) = \{x, u(x, t) \leq 0\}.$$

However, function u is a value function of a state constrained problem, and we are still faced with the problem of characterizing this value function if no controllability assumption is made. To overcome this difficulty, we consider another Lipschitz continuous function $g : \mathbb{R}^d \rightarrow \mathbb{R}$ such that

$$g(x) \leq 0 \Leftrightarrow x \in \mathcal{K}. \quad (13)$$

Note that such a function always exists since we can choose $g(x) := d_{\mathcal{K}}(x)$.

We then consider the control problem

$$\vartheta(x, t) := \inf \left\{ \max \left(\vartheta_0(y_x^\alpha(t)), \max_{\theta \in [0, t]} g(y_x^\alpha(\theta)) \right), \alpha \in \mathcal{A}_{ad} \right\}. \quad (14)$$

Problem (14) has no “explicit” state constraint. In fact, in this new setting, the term $\max_{\theta \in [0, t]} g(y_x^\alpha(\theta))$ plays the role of a penalization that a trajectory y_x^u would pay if it violates the state constraints. We will see in Theorem 3.2 that the advantage of considering (14) is that ϑ can now be characterized as the unique continuous solution of an HJB equation.

Theorem 3.1 (characterization of the capture basin). *Assume (H1)–(H2). Let ϑ_0 and g be Lipschitz continuous functions defined, respectively, by (11) and (13). Let u and ϑ be the value functions defined, respectively, by (12) and (14). Then, for every $t \geq 0$, we have the following:*

(i) *The capture basin is given by*

$$\text{capt}(t) = \{x, u(x, t) \leq 0\} = \{x, \vartheta(x, t) \leq 0\}.$$

(ii) *If $\vartheta(x, t) < 0$ and $\overset{\circ}{\mathcal{K}} = \{x, g(x) < 0\}$, then $u(x, t) < 0$, and there exists, on $[0, t]$, an admissible trajectory y^α that never touches the boundary ∂K .*

Proposition 3.1. *Assume (H1)–(H2). The minimal time function $\mathcal{T} : \mathbb{R}^d \mapsto \mathbb{R}^+ \cup \{+\infty\}$ is lsc. Moreover, we have*

$$\mathcal{T}(x) == \inf\{t \geq 0, \vartheta(x, t) \leq 0\},$$

with ϑ the value function defined in (14), where ϑ_0 and g are any Lipschitz functions satisfying, respectively, (11) and (13).

Remarque 3.1. It is known that when (H2) does not hold, the lower semicontinuity of \mathcal{T} is no longer true. In this case, it is possible to prove that $\mathcal{T}_*(x) = \inf\{t \geq 0, \vartheta(x, t) \leq 0\}$, where \mathcal{T}_* is the lsc envelope of \mathcal{T} .

Remarque 3.2. The use of a level-set approach is a standard way to determine the minimal time function of unconstrained control problems [?].

In our work, we generalize this point of view to the case when the time control problem is in the presence of state constraints. Our formulation also allows us to obtain the capture basins.

As mentioned before, the function ϑ can be characterized as the unique solution of a Hamilton–Jacobi equation. More precisely, considering the Hamiltonian

$$H(x, p) := \max_{\alpha \in \mathcal{A}} (-f(x, \alpha) \cdot p), \quad (15)$$

we have the following.

Theorem 3.2. *Assume (H1) and that ϑ_0 and g are Lipschitz continuous. Then ϑ is the unique continuous viscosity solution of the variational inequality (obstacle problem)*

$$\min(\partial_t \vartheta + H(x, \nabla \vartheta), \vartheta - g(x)) = 0, \quad t > 0, \quad x \in \mathbb{R}^d, \quad (16a)$$

$$\vartheta(x, 0) = \max(\vartheta_0(x), g(x)), \quad x \in \mathbb{R}^d. \quad (16b)$$

Inequalities such as (16) appear also in the framework of exit time problems, where the obstacle g represents the exit cost that should be *paid*. Here, g is a “fictitious cost” that a trajectory would pay if it leaves \mathcal{K} .

Let us also point out that the obstacle term in (16) comes from the presence of the sup-norm $\max_{\theta \in [0, t]} g(y_x^u(\theta))$ in the cost function which defined ϑ (see (14)). We refer to the works of Barron and Ishii [?] and the references therein for optimal control problems with sup-norm cost functions.

3.2 Numerical approximation of ϑ

Although equation (16) is set in the whole space $\mathbb{R}^+ \times \mathbb{R}^d$, with $d = 6$, in order to perform computations a finite domain $[0, T] \times \mathcal{B}$ is used, where T is an upper bound of the needed time to reach the target (which can be increased if needed) and \mathcal{B} is chosen as a d -dimensional box containing the target and the initial point of the optimal trajectory. Hence the HJB equation (16) is discretized on $[0, T] \times \mathcal{B}$. For this we choose a uniform space grid $\mathcal{G} = \{x_i\}$ on \mathcal{B} , with mesh steps $(\Delta x_j)_{j=1, \dots, 6}$. For each time t_n and at each node point $x_i \in \mathcal{G}$, the value V_i^n denotes an approximation of $\vartheta(t_n, x_i)$. Then the discrete approximation of (16) is given by:

$$\min \left(\frac{V_i^{n+1} - V_i^n}{\Delta t} + \mathcal{H}^{num}(t_n, x_i, p_{i,-}^n, p_{i,+}^n), V_i^{n+1} - g(x_i) \right) \quad x_i \in \mathcal{G}, \quad n \geq 0, \quad (17a)$$

$$V_i^0 = \max(\Phi(x_i), g(x_i)), \quad x_i \in \mathcal{G}, \quad (17b)$$

where \mathcal{H}^{num} remains to be defined and $p_{i,\pm}^n$ are gradient approximations that will depend of the values (V_k^n) and that will be also defined below. Because it will be chosen $\Delta t > 0$, it is easy to see that (17a) is equivalent to

$$V_i^{n+1} = \max \left(V_i^n - \Delta t \mathcal{H}^{num}(t_n, x_i, p_{i,-}^n, p_{i,+}^n), g(x_i) \right), \quad x_i \in \mathcal{G}, \quad n \geq 0.$$

(therefore V_i^{n+1} is an explicit expression in terms of the values (V_k^n) .)

In order to define \mathcal{H}^{num} we follow the ENO approach for Hamilton–Jacobi equations as in [?] in order to obtain a consistent approximation of (16), first order in time (Euler forward) and second order in space. First a monotone Lax–Friedrich hamiltonian is considered:

$$\mathcal{H}^{num}(t, x, p_-, p_+) := H(t, x, \frac{p_- + p_+}{2}) - \sum_{j=1}^d \frac{C_j}{2} (p_{+j} - p_{-j}).$$

with $p_{\pm} := (p_{\pm 1}, \dots, p_{\pm d})$. Furthermore, an analytic expression for the Hamiltonian H can be performed (see below).

For each $j = 1, \dots, d$, the constant $C_j > 0$ is chosen to be an upper bound of $\max_{t \in [0, T], x \in \mathcal{B}, p} |\frac{\partial H}{\partial p_j}(t, x, p)|$, which, in our case, turns out to be an upper bound for $\max_{t \in [0, T], x \in \mathcal{B}, u \in U} |F_j(t, x, u)|$.

For a given $i = (i_1, \dots, i_d)$, let $e_j \in \mathbb{R}^d$ be the unit vector such that $(e_j)_k = \delta_{jk}$ ($\forall k$), and therefore

$$i \pm e_j := (i_1, \dots, i_{j-1}, i_j \pm 1, i_{j+1}, \dots, i_d).$$

At given time step t_n , we define

$$DV_{i, \pm j}^n := \pm \frac{V_{i \pm e_j}^n - V_i^n}{\Delta x_j}, \quad D^2V_{i, j}^n := \frac{V_{i+e_j}^n - 2V_i^n + V_{i-e_j}^n}{\Delta x_j^2}.$$

and

$$p_{i, \pm j}^n := DV_{i, \pm j}^n \mp \frac{1}{2} \Delta x_j \minmod \left(D^2V_{i \mp e_j}^n, D^2V_i^n \right)$$

where the minmod function is defined here by $\minmod(a, b) := a$ if $(ab > 0$ and $|a| \leq |b|)$, $\minmod(a, b) := b$ if $(ab > 0$ and $|b| \leq |a|)$, and $\minmod(a, b) = 0$ otherwise.

The time step $\Delta t > 0$ is chosen constant for simplicity, and for stability reasons it is assumed to satisfy the following CFL condition:

$$\Delta t \sum_{j=1}^d \frac{C_j}{\Delta x_j} \leq \frac{1}{2}.$$

Optimal trajectory reconstruction procedure. Once the function ϑ is computed everywhere, the minimal time function is computed as

$$\mathcal{T}(x_i) := \inf \left\{ t_n, t_n \geq 0, \vartheta(t_n, x_i) \right\}.$$

It is then possible to reconstruct the optimal feedback control law and the corresponding optimal trajectory (see for instance [19, Appendix A]). In all the tests performed in the next section, the reconstruction of the optimal trajectory is done by using a classical second-order Runge-Kutta scheme.

3.3 Analytic expression of the hamiltonian

The hamiltonian $H(x, p)$ calculation leads to the evaluation for a given $x \in \mathbb{R}^6$ and $p \in \mathbb{R}^6$ of the following maximum:

$$M = \max_{\alpha, \delta} \left(q_1 \cos(\alpha) \cos(\delta) + q_2 \sin(\alpha) + q_3 \cos(\alpha) \sin(\delta) \right)$$

where

$$q_1 = -p_2, \quad q_2 = -p_3/v, \quad q_3 = p_4/(v \cos \gamma)$$

and $\alpha \in [\alpha_{min}, \alpha_{max}]$, $\delta \in [\delta_{min}, \delta_{max}]$.

Let denote

$$q_4 = \sqrt{q_1^2 + q_3^2}, \quad \zeta = \arctan(q_3/q_1) \in [-\pi, \pi]$$

Let us introduce the functions

$$A(\alpha, \delta) = q_4 \cos(\alpha) \cos(\delta - \zeta) + q_2 \sin \alpha$$

and

$$B(\alpha) = \max_{\delta \in [\delta_{min}, \delta_{max}]} A(\alpha, \delta) = \max_{\delta} (q_4 \cos(\alpha) \cos(\delta - \zeta) + q_2 \sin \alpha)$$

One can remark that

$$B(\alpha) = q_4 \cos(\alpha) \cdot \left[\max_{\delta \in [\delta_{min}, \delta_{max}]} \cos(\delta - \zeta) \right] + q_2 \sin \alpha$$

Denoting

$$\delta^* = \arg \max_{\delta \in [\delta_{min}, \delta_{max}]} \cos(\delta - \zeta) \quad (18)$$

one can see that δ^* does not depend on α and that

$$B(\alpha) = q_4 \cos(\alpha) \cos(\delta^* - \zeta) + q_2 \sin \alpha$$

Let denote

$$\psi = \arctan \left(\frac{q_2}{q_4 \cos(\delta^* - \zeta)} \right) \in [-\pi, \pi]$$

and $q_5 = \sqrt{q_2^2 + (q_4 \cos(\delta^* - \zeta))^2}$. Then one has

$$B(\alpha) = q_5 \cos(\alpha - \psi)$$

The maximum value M is then equal to

$$M = \max_{\alpha, \delta} \left(q_1 \cos(\alpha) \cos(\delta) + q_2 \sin(\alpha) + q_3 \cos(\alpha) \sin(\delta) \right) = \max_{\alpha \in [\alpha_{min}, \alpha_{max}]} B(\alpha) = \max_{\alpha \in [\alpha_{min}, \alpha_{max}]} \cos(\alpha - \psi).$$

One can write it as follows

$$M = A(\alpha^*, \delta^*)$$

where

$$\alpha^* = \arg \max_{\alpha \in [\alpha_{min}, \alpha_{max}]} B(\alpha) \quad (19)$$

Let us remark that the computation of δ^* dans (18) and α^* dans (19) leads to the same problem: find the maximum of the function $\cos(\phi)$ on an interval $[\phi_1, \phi_2]$. In the case of δ^* one has

$$[\phi_1, \phi_2] = [\delta_{min} - \zeta, \delta_{max} - \zeta], \quad -\frac{\pi}{4} < \delta_{min} < 0 < \delta_{max} < \frac{\pi}{4}, \quad -\pi \leq \zeta \leq \pi$$

and for the computation of α^* :

$$[\phi_1, \phi_2] = [\alpha_{min} - \psi, \alpha_{max} - \psi], \quad -\frac{\pi}{2} < \alpha_{min} < 0 < \alpha_{max} < \frac{\pi}{2}, \quad -\pi \leq \psi \leq \pi$$

One can see then that if $[\phi_1, \phi_2] \subset [-\pi, \pi]$

$$\phi^* = \arg \max_{\phi \in [\phi_1, \phi_2]} \cos(\phi) = \text{Proj}(0, [\phi_1, \phi_2])$$

where

$$\text{Proj}(0, [\phi_1, \phi_2]) = \min(\phi_2, \max(\phi_1, 0))$$

is the projection of 0 onto the interval $[\phi_1, \phi_2]$. Indeed, if $0 \in [\phi_1, \phi_2]$ then $\phi^* = 0$, and if $0 < \phi_1 < \phi_2$ then $\cos(\phi)$ is decreasing and $\phi^* = \phi_1 = \text{Proj}(0, [\phi_1, \phi_2])$. Finally if $\phi_1 < \phi_2 < 0$ then $\cos(\phi)$ is increasing and then $\phi^* = \phi_2 = \text{Proj}(0, [\phi_1, \phi_2])$. The last case we have to consider is when $\pi \in [\phi_1, \phi_2]$ or $-\pi \in [\phi_1, \phi_2]$. In both cases the maximum point is one of two endpoints of the interval.

Finally one can resume the optimal point as follows

$$\phi^* = \begin{cases} \text{Proj}(0, [\phi_1, \phi_2]), & \text{si } \frac{\phi_1 + \phi_2}{2} \in [-\pi, \pi] \\ \phi_2, & \text{si } \frac{\phi_1 + \phi_2}{2} > \pi \\ \phi_1, & \text{si } \frac{\phi_1 + \phi_2}{2} < -\pi \end{cases} \quad (20)$$

4 Numerical simulations

4.1 Numerical data used for the simulations

We give in this section the list of principal numerical data describing the dynamical properties of the launcher and the atmospheric model used for all presented computations. These data were provided by CNES.

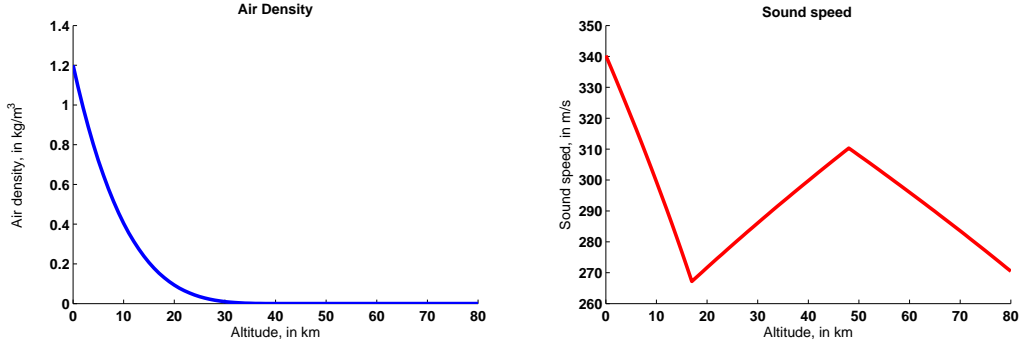


Figure 5: Speed velocity (in red) and air density

| Notation | Value | Units | Comments |
|---------------|---------------------------|--------------------------------|--|
| C_x | | | aerodynamic coefficients |
| F_T | see figures 5 and 6 | | thrust force |
| M_{EAP} | $2 \times 37 \times 10^3$ | kg | mass of boosters without propellant () |
| M_{E1} | 18×10^3 | kg | mass of first stage without propellant () |
| M_{E2} | 7×10^3 | kg | mass of second stage without propellant |
| M_C | 2500 | kg | mass of cap |
| $M_{P,EAP}$ | 2×237000.0 | kg | mass of propellant in boosters |
| $M_{P,E1}$ | 170 000.0 | kg | mass of propellant in first stage |
| $M_{P,E2}$ | 24 500.0 | kg | mass of propellant in second stage |
| M_{PL} | = | mass of the payload | |
| r_T | 6378×10^3 | m | Earth's mean radius |
| S_r | 23 | m ² | reference surface |
| S_{EAP} | 2×7.0 | m ² | exit nozzle area of boosters |
| S_{E1} | 4.0 | m ² | exit nozzle area of first stage |
| S_{E2} | 0 | m ² | exit nozzle area of second stage |
| β_{EAP} | see figure ?? | kg s ⁻¹ | flow rate for boosters |
| β_{E1} | see figure ?? | kg s ⁻¹ | flow rate for first stage |
| ISP_{E2} | 465 | | flow rate for second stage |
| β_{E2} | 40 | kg s ⁻¹ | flow rate for second stage |
| μ | 0.3986020E+15 | m ³ s ⁻² | Earth's gravitational constant |
| ρ | see figure 5 | | atmospheric density |
| Ω | 0.7292115E-04 | rad/s | Earth's angular velocity |

The figures 5 and 6 represent the air density model and the thrust and drag data for the launcher.

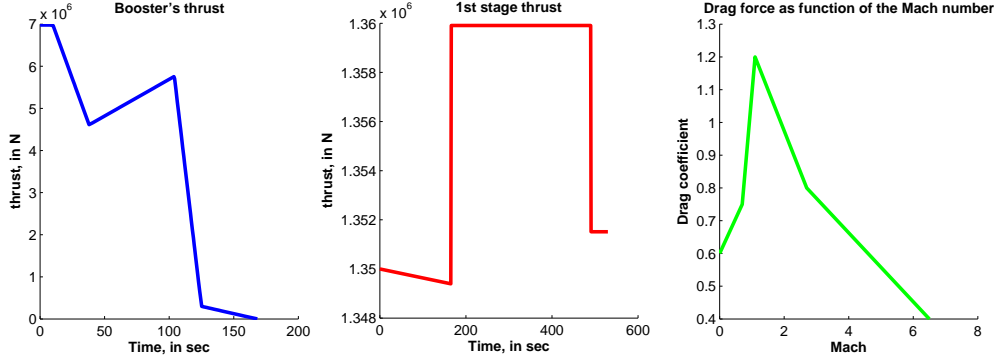


Figure 6: Trust data for boosters (red) and the first stage (blue) and the drag coefficient (green)

4.2 Computational domain and adaptive grid refinement

Let us point out that physical reasons allow us to restrict the computational domain, thus decreasing the CPU time. For instance, the region of small altitude with high velocity is clearly not interesting. Here, we shall consider local computation on the subdomain K_1 of points (r, l, v, γ, χ) such that

$$(r, l, v, \gamma, \chi) \in \mathcal{B}, \quad f_{min}(r) < v < f_{max}(r), \quad (21)$$

with

$$\begin{aligned} f_{min}(r) &:= \max(\max((r - b_1)/a_1, (r - b_2)/a_2), \min(500.0, (r - b_3)/a_3)), \\ f_{max}(r) &:= (r - b_4)/a_4, \end{aligned}$$

and with constants given by:

$$\begin{aligned} a_1 = 33.330, \quad b_1 = 6544700; \quad a_2 = 95.000, \quad b_2 = 6388000; \\ a_3 = 52.000, \quad b_3 = 6379000; \quad a_4 = 14.285, \quad b_4 = 6370900. \end{aligned}$$

By restricting the effective calculation in the domain K we reduce the CPU time. To reduce the computations on a given domain K we introduce an artificial obstacle :the complementary set of K . See [24, 25] for more details about HJB approach to constraint optimal control problems.

In this study we will introduce also an adaptive refining technic. First, we solve the HJB equations on a coarse grid with the domain restriction for the state variables r and v as described below. After computing all the trajectories we determine numerically a subdomain K_2 of points (r, l, v, γ, χ) such that

$$(r, l, v, \gamma, \chi) \in K_1, \quad G_{min}(r, v) < \gamma < G_{max}(r, v) \quad (22)$$

where $G_{min}(r, v)$ and $G_{max}(r, v)$ are empirically computed affine functions of variables (r, v) defining a large enough domain around all computed trajectories. Then we can refine the state space grid and solve the HJB equation by reducing the computation domain on K_2 . The test 2 described later shows that this approach allows us to get more computation precision without increasing too much the CPU time. For this study we have used the following expressions for the functions G_{min} and G_{max} :

$$G_{min}(r, v) = \max(\gamma_{min}, (r - a_{6,1} \cdot v - a_{6,2})/b_6); \quad G_{max}(r, v) = \max(0.15, (r - a_{7,1} \cdot v - a_{7,2})/b_7);$$

with the following coefficients:

$$\begin{aligned} a_{6,1} = -26.6666667 \quad a_{6,2} = -266666.6666667 \quad b_6 = 6538135.0 \\ a_{7,1} = -29.999750 \quad a_{7,2} = -399996.6666667 \quad b_7 = 6718132.833333 \end{aligned}$$

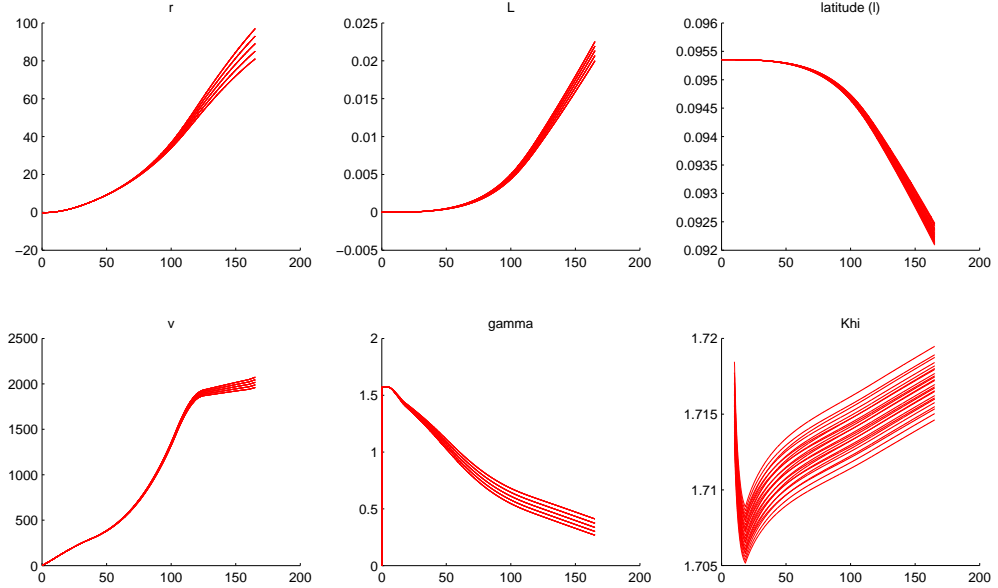


Figure 7: The set of all atmospheric trajectories corresponding to the discret set of initial parameters

4.3 Computation of the set X_0 of initial positions for the optimal control problem

As explained in the section 3 to solve the global optimization problem (8) we discretize the set \mathcal{I} and solve for each $\mathbf{i} \in \mathcal{I}^\Delta$ the sub-problem $(\mathcal{P}_\mathbf{i})$ defined by (9). We apply this procedure in the Test 3 below.

For a fixed value of the parameter $\mathbf{i} \in \mathcal{I}$ the optimisation problem (9) is defined for a set of initial positions X_0 that are all positions that can be reached from y_0 with parameters $p \in P_{\text{Ini}}$. For numerical solution of this problem we discretize also the set of initial parameters P_{Ini} . In all computations presented below this set is fixed to be equal to

$$P_{\text{Ini}} = [\psi_{\min}, \psi_{\max}] \times [\theta_{\min}, \theta_{\max}] = [-0.1326, -0.1292] \times [0.0159, 0.0162]$$

Taking a uniform grid $P_{\text{Ini}}^\Delta = \{(\psi_i, \theta_j), i = 0, \dots, N_I, j = 0, \dots, N_I\}$ of $N_I \times N_I$ points on this set we define the discret set

$$X_0^\Delta = \{y_x^p(t_1) \mid p \in P_{\text{Ini}}^\Delta, \dot{y}_x^p(t) = f_1(t, y_x^p(t), p), y_x^p(0) = y_0\}.$$

The figure 7 shows the set of all governed by the dynamics $f_1(t, x, p)$ for $t \in (0, t_1)$ corresponding to all values of the discret parameters set P_{Ini}^Δ .

4.4 Description of tests

4.4.1 Computation models comparison

In this study the motion equations used for the HJB equation and for the trajectory reconstruction are modeled in spherical coordinates (see section 2). This approach allows us to reduce the dimension of the system to 5 by isolating the equation for the longitude L (the second one in the system (2)). However, the reference trajectory to which we will compare some of our results, was calculated using the motion

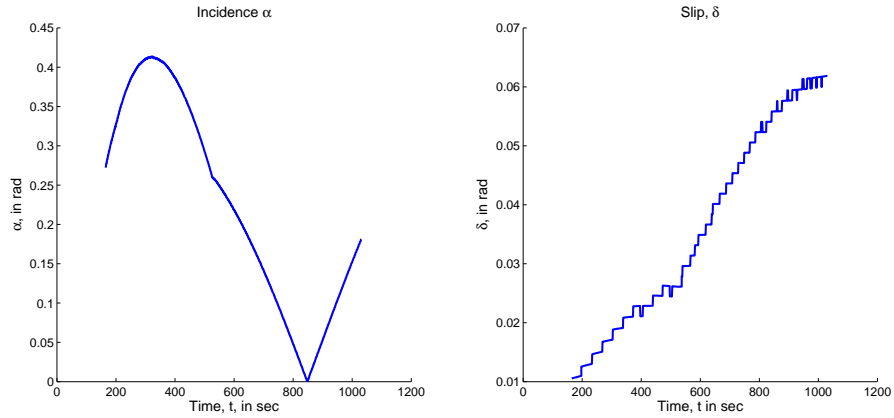


Figure 8: Control laws $\alpha(t)$ and $\delta(t)$ corresponding to the reference trajectory

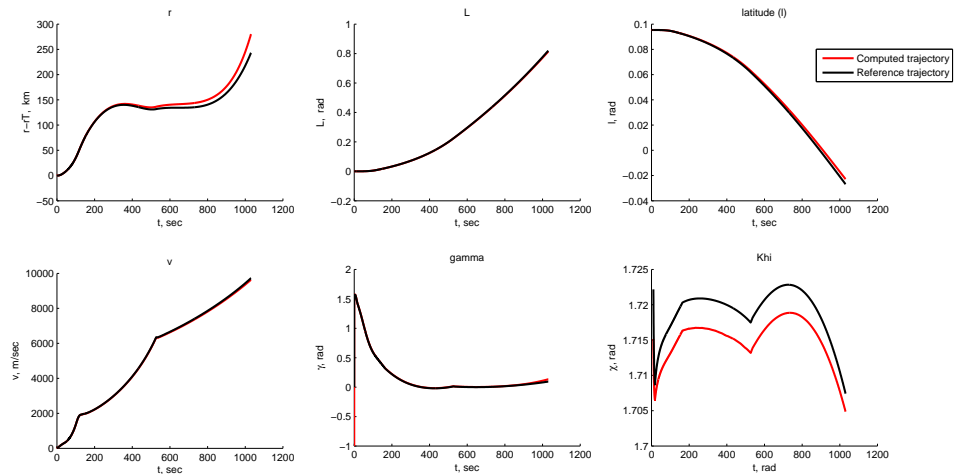


Figure 9: Reference trajectory (in black) and computed trajectory (red)

equations in cartesian coordinate system, corresponding to the quasi inertial equatorial frame \mathcal{R}_I . Let us first compare the computational model used in this study with the reference one. In particular, let fix the initial condition parameters and the control laws $\alpha(t)$ and $\delta(t)$ to those used by CNES to compute the reference trajectory.

The figure 8 shows the controls laws $\alpha(t)$ and $\delta(t)$ obtained by a change of variables from the reference ones θ and ψ defining the thrust force vector direction in the local inertial frame. The angles α and δ are the angles of the same thrust force vector in the dynamic frame \mathcal{R}_D . The figure 9 shows the trajectory obtained by solving the motion equations (2) in spherical coordinates with the control laws α and δ as in figure 8.

One can observe some difference in the obtained trajectory with respect to the reference one (in black on the figure 9, in spherical coordinates). This can be explained by several minor approximations were admitted to the study. In particular, the gravitation model used for the computations is reduced to a first order approximation. The atmospheric data were also simplified for the purposes of this study.

4.4.2 Tests for the HJB approach

Test 1. Our goal for this test is to validate the quality of the trajectories obtained with the HJB approach. We compare our trajectories with a reference data, obtained by CNES. We observe also l'evolution of the computation performances when the number of grid points grows.

This test is realized under the following hypothesis:

- The set of injection points is reduced to a segment of a single GTO orbit, corresponding to the orbit parameters of the reference trajectory. The only orbit parameter is the true anomaly θ .
- The initial condition parameters, azimuth ψ and the angular speed $\omega_{\text{bas}c}$ are varying in some neighborhoods of the values fixed for the reference trajectory.

The initial parameters are discretized, as explained in the previous section. We test here two different grids of initial parameters: P_{Ini}^1 with $N_I = 5$ and P_{Ini}^2 with $N_I = 9$ such that $P_{\text{Ini}}^1 \subset P_{\text{Ini}}^2$. For each grid of initial parameters we select two particular trajectories. The first one, $y^*(t)$ is the optimal trajectory which is the approximation of the solution of the problem (9). The second one $y^r(t)$ is the trajectory obtained as solution of the particular optimal control problem 6 with the same values of initial parameters as for the CNES reference trajectory.

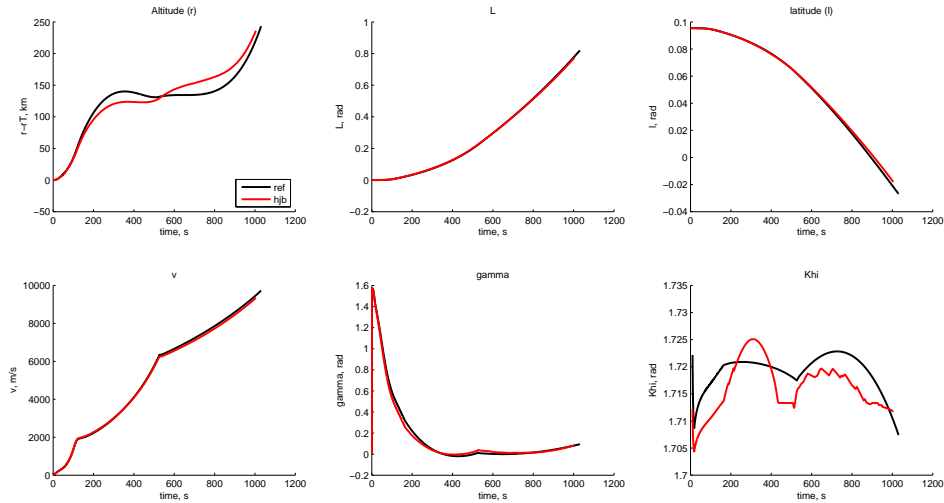


Figure 10: Test1. Optimal trajectory (red) and trajectory with reference initial parameters (blue)

The figure 10 shows the time optimal trajectory : $y^*(t)$ in red obtained using the finest grid $60 \times 15 \times 60 \times 45 \times 15 \times 5$. It is compared to the reference trajectory computed by CNES (black line). The figure ?? shows the optimal control laws.

The table 1 resumes the optimal shooting parameters and the CPU time as functions of the size of the state space grid and the initial parameters grid.

| grid | Azimut ψ in deg. | | Angular speed w , deg./s | | CPU in s |
|---|--------------------------|-------------|-------------------------------|-------------|-------------|
| | P_{Ini}^2 | P_{Ini}^1 | P_{Ini}^2 | P_{Ini}^1 | |
| $20 \times 5 \times 20 \times 15 \times 5 \times 5$ | -0.129 | -0.129 | 0.01518 | 0.01518 | 43 |
| $40 \times 10 \times 40 \times 30 \times 10 \times 5$ | -0.130 | -0.130 | 0.01693 | 0.01606 | 1889 |
| $60 \times 15 \times 60 \times 45 \times 15 \times 5$ | -0.130 | -0.130 | 0.01693 | 0.01780 | 36239 |
| Reference trajectory of CNES | -0.1309 | | 0.01606 | | NA |

Table 1: (Test 1) Comparison of optimal initial parameters

The table 2 resumes the properties of the optimal trajectories obtained with different state space grid and the initial parameters grids.

| grid | t_f on GTO | | Distance to GTO (m) | | $M_{P,E2}(t_f, GEO)$ kg | |
|---|--------------|-------------|---------------------|-------------|----------------------------|-------------|
| | P_{Ini}^2 | P_{Ini}^1 | P_{Ini}^2 | P_{Ini}^1 | P_{Ini}^2 | P_{Ini}^1 |
| $20 \times 5 \times 20 \times 15 \times 5 \times 5$ | 966.913 | 966.913 | 2570.452 | 504.843 | 1780.911 | 1780.911 |
| $40 \times 10 \times 40 \times 30 \times 10 \times 5$ | 993.753 | 994.203 | 510.122 | 509.372 | 1009.587 | 996.655 |
| $60 \times 15 \times 60 \times 45 \times 15 \times 5$ | 1004.563 | 1005.263 | 390.026 | 389.935 | 698.930 | 678.814 |
| Reference trajectory of CNES | 19746 | | 4754 | | 0 | |

Table 2: (Test 1) Comparison of optimal trajectories parameters for different grids

The next table compares the performances of ergol consumptions.

| grid | $M_{P,E2}(t_f)$ (on GTO) (kg) | | $\Delta M_{P,E2}(t_f, \mathbf{i})$ kg | | $M_{P,E2}(t_f, GEO)$ kg | |
|---|----------------------------------|-----------|--|----------|----------------------------|----------|
| | $y^*(t)$ | $y^r(t)$ | $y^*(t)$ | $y^r(t)$ | $y^*(t)$ | $y^r(t)$ |
| $20 \times 5 \times 20 \times 15 \times 5 \times 5$ | 17198.160 | 17231.360 | 5520.929 | 5511.581 | 1780.911 | 1757.059 |
| $40 \times 10 \times 40 \times 30 \times 10 \times 5$ | 18271.760 | 18411.360 | 5218.653 | 5179.349 | 1009.587 | 909.291 |
| $60 \times 15 \times 60 \times 45 \times 15 \times 5$ | 18704.160 | 18794.960 | 5096.910 | 5071.345 | 698.930 | 633.695 |
| Reference trajectory of CNES | 19746 | | 4754 | | 0 | |

Table 3: (Test 1) Comparison of boost consumption for optimal and reference initial parameters

Test 2. In this test we introduce a new domain reduction method for tree state variables (r, v, γ) , as described below. We use the same hypothesis as in the test 1. The only difference is the domain of computation. Then we measure the gain of the CPU time obtained with the new domain as function of the size of the grid. The table 4 shows the evolution of the CPU time for two reduction methods when the size of the grid grows. The last column of this table shows the distance between the sets of trajectories computed with two different computational domains.

| grid | CPU time | CPU time | Mass gain | Mass gain |
|---|-----------------|------------------------|-----------------|------------------------|
| | $r-v$ reduction | $r-v-\gamma$ reduction | $r-v$ reduction | $r-v-\gamma$ reduction |
| $20 \times 5 \times 20 \times 15 \times 5 \times 5$ | 43 | 32 | 1967.133 | 1853.62 |
| $40 \times 10 \times 40 \times 30 \times 10 \times 5$ | 1889 | 1432 | 1038.324 | 1005.28 |
| $60 \times 15 \times 60 \times 45 \times 15 \times 5$ | 18631 | 14645 | 727.955 | 755.256 |
| $75 \times 20 \times 75 \times 60 \times 20 \times 5$ | - | 79725 | - | 464.716 |

Table 4: (Test 2) comparison of two different computation domains

Test 3. In this test we use a set of injection points defined by tree varying orbit parameters: ω , \mathbf{i} and θ . The goal of this test is to check if this more large injection set allows to find better solutions in terms of mass partition. All computations for this tests was realized with the grid $60 \times 15 \times 60 \times 50 \times 15 \times 5$. The next table shows the optimal solution parameters as function of the inclination i of the GTO orbit.

| Inclination, i , in deg | Azimet ψ in deg. | Angular speed w , deg./s | Final time t_f , en s | mass gain en s |
|---------------------------|--------------------------|-------------------------------|----------------------------|-------------------|
| 4 | -0.1292 | 0.0161 | 1018.1 | 370.63 |
| 5 | -0.1292 | 0.0152 | 1008.9 | 624.6455 |
| 6 | -0.1309 | 0.0169 | 1015.7 | 413.973 |
| 7 | -0.1309 | 0.0143 | 1015.2 | 410.316 |
| 8 | -0.1309 | 0.0161 | 1015.0 | 393.062 |
| 7 | -0.1309 | 0.0143 | 1015.7 | 350.277 |
| Traj. de référence CNES | -0.131 | 0.0160 | 1031 | NA |

Table 5: (Test 3) Optimal parameters as functions of the GTO inclination

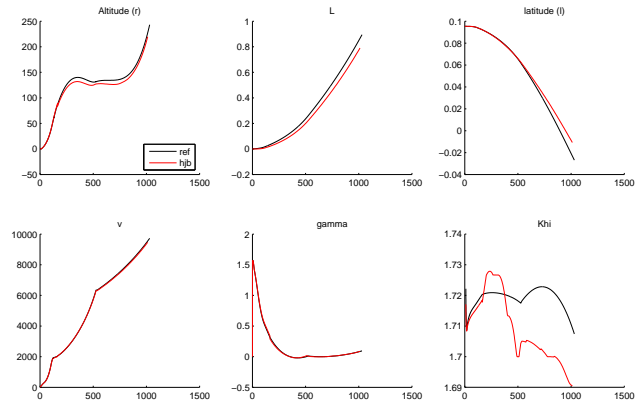


Figure 11: Optimal trajectory

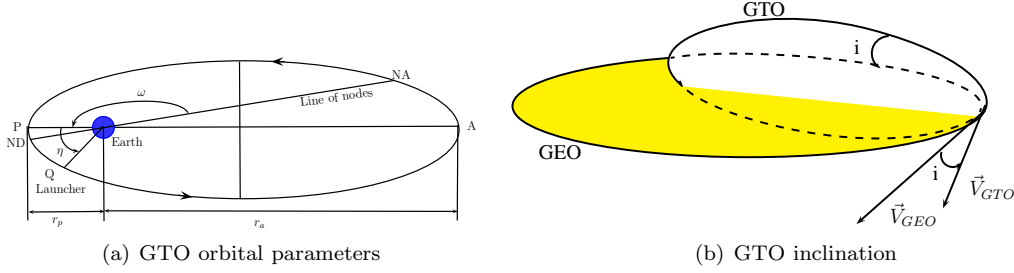


Figure 12: GEO and GTO orbits

A Definition of orbital parameters

For the phase 3 of the flight sequence there are two kinds of orbital maneuvers:

- Injection on a GTO orbit
- Orbital transfert from GTO to the GEO.

We define here all necessary parameters to describe these maneuvers.

A.1 GTO injection parameters

A GTO orbit is represented on the figure A.1. We denote A and P the perigee and apogee centers of the orbit respectively. The line of nodes is the line of intersection of the orbital plane with the equatorial plane of the Earth. It's extremal points are respectively ascending (NA) and descending (ND) nodes of the orbit. A GTO orbit can be fully defined with the following set of parameters $(r_a, r_p, \Omega_{GTO}, \omega, \mathbf{i})$, where:

- r_a and r_p are respectively the distances from the center of the Earth of the perigee and the apogee of the orbit
- Ω_{GTO} is the right ascension of the ascending node.
- ω is the polar position of the perigee P , measured positively from the ascending node NA .
- \mathbf{i} is the inclination of the orbital plane with respect to the equatorial plane of the Earth.

On can deduce from these parameters the semimajor axis

$$a = \frac{r_a + r_p}{2} \quad (23)$$

and the eccentricity

$$e = \frac{r_a - r_p}{r_a + r_p} \quad (24)$$

A point Q on the orbit is defined by it's angular position η , called true anomaly (see figure A.1). Let $\mu = 3.986013 \cdot 10^{14} \text{ m}^3/\text{s}^2$ be the central attraction term of the Earth. Given orbital parameters defined below and the true anomaly of a point Q on the orbit, one can deduce the corresponding orbit radius $r_Q(\eta)$ and orbit velocity $v_Q(\eta)$ as follows:

$$r_Q(\eta) = \frac{r_a r_p}{r_a(1 + \cos(\eta)) + r_p(1 - \cos(\eta))} \quad (25)$$

$$v_Q(\eta) = \sqrt{\frac{\mu}{a} \frac{1 + e^2 + 2e \cos(\eta)}{1 - e^2}} \quad (26)$$

In this study we consider a family of GTOs with a fixed perigee altitude r_p^0 and that intersect a given GEO. Recall that a GEO is a circular orbit in the equatorial plane of the Earth. It is fully defined by the data of its radius r_{GEO} . If a GTO intersects a GEO of a give radius r_{GEO} this fixes the orbit radius of it's ascending node NA . Note that by definition (see figure A.1)) the true anomaly of the point NA is $-\omega$. Then we have the following condition:

$$r_{NA} = r_Q(-\omega) = r_{GEO}$$

from which we deduce the apogee altitude:

$$r_a = \frac{r_{GEO} \cdot r_p (1 - \cos(\omega))}{2r_p - r_{GEO}(1 + \cos(\omega))}$$

So, if we fix the perigee altitude r_p and the GEO radius r_{GEO} the parameter r_a is also fixed. Then the set of considered GTO orbits is defined by two varying parameters : inclination $\mathbf{i} \in [i_{min}, i_{max}]$ and the perigee orientation $\omega \in [\omega_{min}, \omega_{max}]$. For each GTO orbit we define a segment of injection as a neighborhood of the perigee : $\eta \in [\omega - \Delta\eta, \omega + \Delta\eta]$ with a given $\Delta\eta > 0$.

A.2 Orbital transfer parameters

In our study, we assume that the GTO- GEO orbital transfer is performed through an impulse boost (to change the velocity's modulus and direction). The amount of propellant required for the orbital transfer is determined by the choice of the GTO via Tsiolkovski formula. Let \vec{V}_{GTO} be the speed at the ascending node of the GTO, and \vec{V}_{GEO} be the desired GEO speed. Then the differential gear to provide the vehicle is the vector difference:

$$\Delta\vec{V} = \vec{V}_{GEO} - \vec{V}_{GTO}$$

with the modulus:

$$\Delta V = \sqrt{V_{GEO}^2 + V_{GTO}^2 - 2V_{GTO}V_{GEO} \cos(\mathbf{i})} \quad (27)$$

Note that the speed V_{GTO} can be expressed using (26) for the ascending node:

$$V_{GTO} = v_{NA} = v_Q(-\omega)$$

The formula Tsiolkovski connects the differential speed to provide the required fuel mass

$$\Delta V = g \cdot I_{sp} \cdot \ln \left(\frac{m_{init}}{m_{final}} \right) \quad (28)$$

where m_{init} and m_{final} are respectively the launcher's masses before and after the operation. Thus one can calculate the weight of propellant required to the GTO-GEO orbital transfer in a single impulse, knowing the speed at the height of the GTO and its inclination with respect to the GEO.

References

- [1] R.H. Goddard. A method of reaching extreme altitudes. *Smithsonian Miscellaneous Collection*, 71(4), 1919.
- [2] F. Bonnans, P. Martinon, and E. Trélat. Singular arcs in the generalized Goddard's problem. *J. Optimization Theory Applications*, 139 (2):439–461, 2008.

- [3] B. Bonnard, L. Faubourg, and E. Trélat. Optimal control of the atmospheric arc of a space shuttle and numerical simulations with multiple shooting method. *Mathematical Models and Methods in Applied Sciences*, 15(1):109–140, 2005.
- [4] J. Laurent-Varin. *Optimal ascent and reentry of reusable rockets*. PhD thesis, Ecole Polytechnique, 2005. Supported by CNES and ONERA.
- [5] K.P. Zondervan, T.P. Bauer, J.T. Betts, and W.P. Huffman. Solving the optimal control problem using a nonlinear programming technique. part 3: Optimal shuttle reentry trajectories. *Proceedings of the AIAA/AAS Astrodynamics conference, Seattle.*, 1984.
- [6] J.T. Betts. *Practical methods for optimal control using nonlinear programming*. Society for Industrial and Applied Mathematics, Philadelphia, 2001.
- [7] H.J. Oberle. Numerical computation of singular control functions in trajectory optimization problems. *Journal of Guidance, Control and Dynamics*, 13:153–159, 1990.
- [8] H.J. Pesch. A practical guide to the solution of real-life optimal control problems. *Control and Cybernetics*, 23(1-2):7–60, 1994.
- [9] H.J. Pesch. Real-time computation of feedback controls for constrained optimal control problems, Part 2: a correction method based on multiple shooting. *Optimal Control, Applications and Methods*, 10(2):147–171, 1989.
- [10] J.T. Betts. Survey of numerical methods for trajectory optimization. *Journal of Guidance Control and Dynamics*, 21(2):193–207, 1998.
- [11] P. Martinon, J. F. Bonnans, J. Laurent-Varin, and E. Trélat. Numerical study of optimal trajectories with singular arcs for an Ariane 5 launcher. *Journal of Guidance, Control, and Dynamics*, 32(1):51–55, 2009.
- [12] A.J. Calise and P.F. Gath. Optimization of launch vehicle ascent trajectories with path constraints and coast arcs. *J. Guidance, Control, and Dynamics*, 24(2):2001, 296-304.
- [13] P. Lu, H. Sun, and B. Tsai. Closed-loop endoatmospheric ascent guidance. *J. of Guidance, Control, and Dynamics*, 26(2):283–294, 2003.
- [14] L. Zhang and P. Lu. Fixed-point algorithms for optimal ascent trajectories of launch vehicles. *Engineering optimization*, 40(4):361–381, 2008.
- [15] R. Bellman. *Dynamic programming*. Princeton University Press, Princeton, 1961.
- [16] M.G. Crandall and P.-L. Lions. Viscosity solutions of Hamilton Jacobi equations. *Bull. American Mathematical Society*, 27(1):1–42, 1983.
- [17] M.G. Crandall and P.-L. Lions. Two approximations of solutions of Hamilton-Jacobi equations. *Mathematics of Computation*, 43(167):1–19, 1984.
- [18] M.G. Crandall, L.C. Evans, and P.-L. Lions. Some properties of viscosity solutions of Hamilton-Jacobi equations. *Trans. Amer. Math. Soc.*, 282(2):487–502, 1984.
- [19] M. Bardi and I. Capuzzo-Dolcetta. *Optimal Control and viscosity solutions of Hamilton-Jacobi-Bellman equations*. Birkhäuser Boston, 1997.
- [20] O. Bokanowski, E. Cristiani, and H. Zidani. An efficient data structure and accurate scheme to solve front propagation problems. *J. Sci. Comput.*, 42(10):251–273, 2010.
- [21] H. Ishii and S. Koike. A new formulation of state constraint problems for first order PDEs. *SIAM J. Control and Optimization*, 34(2):554–571, 1996.

- [22] H. M. Soner. Optimal control with state space constraint. *SIAM Journal of Control and Optimization*, 24(3):552–561, 1986.
- [23] D. Drake, Ming Xin, and S. N. Balakrishnan. Reusable launch vehicle guidance and control : New nonlinear control technique for ascent phase of reusable launch vehicles. *Journal of guidance, control, and dynamics*, 27(6):938–948, 2004.
- [24] O. Bokanowski, N. Forcadel, and H. Zidani. L^1 -error estimate for numerical approximations of Hamilton-Jacobi-Bellman equations in dimension 1. *Mathematics of Computations*, 79:1395–1426, 2010.
- [25] O. Bokanowski, N. Forcadel, and H. Zidani. Reachability and minimal times for state constrained nonlinear problems without any controllability assumption. *Preprint Hal*, 2009.