Optimal drag-energy entry guidance via pseudospectral convex optimization

Published Version:
Sagliano, M., Mooij, E. (2021). Optimal drag-energy entry guidance via pseudospectral convex optimization. AEROSPACE SCIENCE AND TECHNOLOGY, 117, 1-16 [10.1016/j.ast.2021.106946].

Availability:
This version is available at: https://hdl.handle.net/11585/1041350 since: 2026-02-19

Published:
DOI: http://doi.org/10.1016/j.ast.2021.106946

Optimal Drag-Energy Entry Guidance via Pseudospectral Convex Optimization

Marco Sagliano[1]

German Aerospace Center

Erwin Mooij[2]

Delft University of Technology

Abstract

In this paper a new drag-energy scheme for atmospheric entry guidance, based on the use of pseudospectral methods and convex optimization, is proposed. One of the most successful technologies to deal with atmospheric entry is the class of drag-tracking schemes, a direct heritage of the Space Shuttle program. The method that we propose exploits the drag-dynamics, and allows for an efficient automatic design of an optimal entry profile satisfying all the longitudinal constraints acting on the vehicle. A new representation of the entry-guidance problem, able to loss-less convexify the formulation, is provided. Numerical simulations confirm the validity of the proposed scheme as tool for further improving the autonomy of modern entry guidance systems, with a mean final range-to-go error at the end of entry phase smaller than 2 km, and the capability to re-compute a complete constrained trajectory to meet the mission requirements.

Keywords: Drag-Energy guidance, Pseudospectral Methods, Convex Optimization, Atmospheric Reentry

[1]Visiting GNC Specialist, GNC Department, Institute of Space Systems, Robert Hooke Straße 7, 28359, Bremen, Germany

[2]Associate Professor, Astrodynamics and Space Missions Department, Faculty of Aerospace Engineering, Kluyverweg 1, 2629 HS, Delft, The Netherlands

1. Introduction

Since the beginning of the Apollo program, entry guidance has been widely treated by engineers and researchers. The first successful approach, used for several programs (Apollo, Space Transportation System [1]), was based on the planning of an entry trajectory in terms of the drag-velocity plane. The rationale for this choice resides in the fact that the typical environmental constraints (dynamic pressure, heat flux and load factor), as well as the range-to-go, can be efficiently represented in the drag-velocity plane. The longitudinal guidance can then be derived in several ways. For instance, assuming the equilibrium-glide approximation [2, 3], extracting the longitudinal states (altitude, speed, flight-path angle) from the drag acceleration and its derivatives [4], or implementing constraints-tracking guidance schemes [5]. Similar results can be obtained when the drag-velocity plane is replaced by the drag-energy representation [6, 7]. In any case, approximations, disturbances and modeling errors make use of a feedback controller, necessary to track the scheduled nominal drag profile.

In addition, a bank-reversal logic is usually implemented to keep the heading error within prescribed limits, chosen to steer the vehicle towards the terminal area for energy management (TAEM). In parallel to these approaches, the use of techniques based on optimal control [8, 9, 10, 11] has achieved significant improvements. The increased CPU (Central Processing Unit) capabilities, together with the development of dedicated algorithms, have led to the possibility to transcribe the problem into a discrete, finite-dimension problem, i.e., a non-linear programming (NLP) problem, which can be efficiently solved with one of the available well-known NLP solvers [12, 13]. However, it is in general not possible to define an analytical upper bound for the computational time required to obtain a solution, making the use of these techniques in real-time difficult.

Significant steps to relax the CPU burden for inflight applications can be found in the use of interpolation-based techniques. Saraf et al. [14] used interpolation schemes applied to extremal drag-energy profiles for generating landing footprints for entry missions. Schierman et al. [15] implemented a scheme able to compute a trajectory online by using piecewise-linear functions, and solved the online trajectory-generation problem as a linear programming problem. Sagliano et al. [16] proposed a multivariate pseudospectral approach to provide a real-time capable method able to deal with large off-nominal conditions. Although properly working, the method only provides sub-optimal solutions, as the optimization is replaced by a multivariate interpolation in between optimal solutions.

The development of convex optimization [17] over the last thirty years boosted the research aiming at solving optimization problems in real time, since it is possible to predict an upper bound for the number of operations required to solve the problem. This new technology was successfully applied to the powered descent and landing problem by using the so-called *lossless convexification* introduced by Acikmese et al. [18], and further improved over the years [19]. The underlying guidance scheme was successively merged with a pseudospectral optimal control framework [20, 21] to improve the accuracy of the results. Further applicatons in the frame of entry, descent, and landing include the works of Liu et al. [22], Wang [23], and Liu [24]. All of these methods have the application of convex-optimization methods to the equations of motion formulated with respect to either time or energy in common.

The application of the hybrid pseudospectral convex optimization gave the hint to revisit the drag-energy schemes previously mentioned, rather than the longitudinal motion as alternative scheme to generate constrained feasible solutions to the entry problem. States and controls are then extracted from the drag-energy solution with limited computational effort. In fact, the drag dynamics and the longitudinal constraints can be reformulated as a convex problem. The subsequent discretization according to the pseudospectral convex framework provides a real-time capable drag-energy solution satisfying all requirements for a constrained atmospheric entry. The contributions of this paper are the following: first, a generalized drag-energy dynamics, containing terms usually neglected, is derived. Second, the extended drag-energy scheme is used to formulate the entry-guidance problem in convex form by introducing a new formulation based on the inverse drag acceleration. Third, some of the features, such as the differential and integral operators, coming from pseudospectral methods, and specifically the Lobatto Pseudospectral Method (LPM) [25], are combined to efficiently transcribe and solve the entry problem, aiming at retrieving the robustness of the original Space-Shuttle-class guidance algorithms and combining it with the real-time decision-making capabilities provided by convex optimization and the high-accuracy solutions obtained by using pseudospectral methods. The proposed approach embeds the longitudinal guidance formulation in a single convex-optimization problem: this feature allows to avoid the use of sequential convex programming approaches, leading to faster solutions. Moreover, no initial guess is required at all, with a further benefit for what concerns the robustness of the method. The method here proposed is meant to be applied to high L/D vehicles, having a shallow entry flight-path angle. This assumption is quite common in literature [1, 2, 4], while its application to low L/D vehicles depends on the overall load factor / range requirements, and is kept for future studies.

The remainder of this work is organized as follows. Sections 2 and 3 provide a brief overview on pseudospectral methods and convex optimization, respectively. The atmospheric entry problem is described in Sec. 4, and applied to HORUS, a representative winged entry vehicle [26]. The drag-dynamics is the subject of Sec. 5, whilst Sec. 6 describes the proposed convexification for the drag-energy guidance problem. Section 7 demonstrates the performance of the proposed method, whereas conclusions are drawn in Sec. 8.

2. Pseudospectral Methods

2.1. Optimal Control Problem

There are several approaches for the generation of reference trajectories. Some methods exploit the structure of the specific problem we deal with. Often, they require simplifications to make the problem mathematically tractable, and therefore generate solutions valid under given hypotheses. A different approach, which nowadays has become a standard method and benefits from the development of the computational capabilities of modern CPUs, is the representation of the trajectory-generation problem as an optimal-control problem. This means that we are looking for solutions minimizing (or maximizing) a given criterion, and satisfying at the same time several constraints, which can be differential (i.e., the equations of motion of a spacecraft) and / or algebraic (e.g., the maximum heat-flux that a vehicle can tolerate during entry). The standard form for representing optimal-control problems is the so-called Bolza formulation. Given a state vector $\mathbf{x}(t) \in \mathbb{R}^{n_s}$, a control vector $\mathbf{u}(t) \in \mathbb{R}^{n_c}$, the scalar functions $\Phi(t_F, \mathbf{x}(\mathbf{t_F}))$ and $\Psi(t, \mathbf{x}, \mathbf{u})$, and the constraint vector $\mathbf{g}(t, \mathbf{x}, \mathbf{u}) \in \mathbb{R}^{n_g}$ we can formulate the problem as described in Table 1.

Table 1: Bolza problem

Description	**Formulation**
Cost function	$J = \Phi\left[t_f, \mathbf{x}\left(t_f\right)\right] + \int_{t_0}^{t_f} \Psi\left[\mathbf{x}(t), \mathbf{u}(t)\right] dt$
Dynamics	$\dot{\mathbf{x}} = \mathbf{f}\left(t, \mathbf{x}, \mathbf{u}\right)$
Path constraints	$\mathbf{g}_L \leq \mathbf{g}\left(t, \mathbf{x}, \mathbf{u}\right) \leq \mathbf{g}_U$
State boundaries	$\mathbf{x}_L \leq \mathbf{x}(t) \leq \mathbf{x}_U$
Control boundaries	$\mathbf{u}_L \leq \mathbf{u}(t) \leq \mathbf{u}_U$
Initial conditions	$\mathbf{x}_0 = \mathbf{x}(t_0)$
Final conditions	$\mathbf{x}_f \in \chi(t_f)$

The formulation in Table 1 represents a generic continuous optimal control problem (OCP). In the next section we will see how this type of OCP can be transcribed by using pseudospectral methods.

2.2. Properties of Pseudospectral Methods

Numerical methods for solving OCPs are divided into two major classes, namely, indirect and direct methods. Indirect methods are based on Pontryagin's Maximum Principle, which leads to a multi-point boundary-value problem. Direct methods, instead, consist of the proper discretization of the OCP (or *transcription*), having as a result a finite-dimensional NLP problem. Pseudospectral methods represent a particular area of interest in the frame of the wider class of direct methods. Examples of tools implementing pseudospectral methods include DIDO [27], GPOPS [28] and SPARTAN [29]. Pseudospectral methods feature several interesting properties, such as the so-called *spectral* (i.e., quasi-exponential) convergence of the NLP results to the OCP solution when the number of nodes employed is increased (and the problem is smooth), the avoidance of the Runge phenomenon, as well as the intrinsic sparsity of the structure of the associated transcribed problem.

The transcription process does not only involve the choice of the discrete nodes, but also determines the discrete differential and integral operators needed to solve the associated OCP. Therefore, *transcription* is a more general process than *discretization*. The minimum fundamental steps of a *transcription* are the following: 1) domain discretization; 2) discrete to conversion (and viceversa) of states and / or controls, and 3) characterization of differential and integral operators

Among the families of pseudospectral (PS) methods, a specific one was considered for this work: the Lobatto Pseudospectral Method (or LPM). It is noted saying that this is not the only possible choice, as other sets of nodes, like Gauss [25], Chebyshev [30] or Radau [31, 29] exist. While both the direct and flipped Radau methods exhibit an exact mapping between the costates of the optimal control problem and the Lagrange Multipliers of the corresponding Nonlinear Programming problem this is not the case for the Lobatto pseudospectral method. However, in this context we are not interested in the dual variables of the problem, but want to exploit the fact that LPM allows collocation of both sides of the interval we are considering. This aspect implies that the LPM allows straightforward computation of initial and final controls, since all the nodes are *collocated* while still showing good performance [31]. Therefore, we selected LPM as reference pseudospectral implementation for this work.

2.3. Lobatto Pseudospectral Method

LPM is based on a symmetric set of nodes, associated with the $n+1$ roots of the Legendre-Lobatto polynomial, defined as

$$L_{n+1}(\tau) = (1-\tau^2)\dot{\tilde{L}}_n \quad \tau \in [-1,1] \tag{1}$$

where $\dot{\tilde{L}}_n$ is the derivative of the Legendre polynomial of order n. This discrete representation of the domain can be used to reconstruct continuous approximations of the function $x(t)$ as:

$$x(t) \approx \sum_{i=0}^{n} X_i P(t), \qquad\qquad P(t) = \prod_{\substack{k=0 \\ k\neq i}}^{n} \frac{t-t_k}{t_i-t_k} \tag{2}$$

Once the domain has been discretized, and the discrete-to-continuous conversion of states has been defined, the corresponding differential operator needs to be defined. This is required for the proper representation of the left-hand side of the differential equations describing the motion of the spacecraft. The differential operator will be in the form

$$\dot{\mathbf{X}} \approx \mathbf{D}^1_{Lb} \cdot \mathbf{X} \tag{3}$$

and the dynamics defined in Table 1 will be replaced by

$$\mathbf{D}^1_{Lb} \cdot \mathbf{X} = \frac{t_f - t_0}{2}\mathbf{f}(t, \mathbf{X}, \mathbf{U}) \tag{4}$$

where t_0 and t_f are the initial and final time, and the term $\frac{t_f - t_0}{2}$ is a scale factor related to the transformation between the physical time domain t, and the pseudospectral time domain $\tau \in [-1,1]$, given by the following affine transformations:

$$t = \frac{t_f - t_0}{2}\tau + \frac{t_f + t_0}{2} \tag{5}$$

$$\tau = \frac{2}{t_f - t_0}t - \frac{t_f + t_0}{t_f - t_0} \tag{6}$$

Moreover it is possible to map any domain with respect to the pseudospectral time, by simply changing t_0 and t_f with the initial and final values of a new independent variable, like the energy. The only pre-requisite for such a choice is that the variable shows monotonic behavior. The matrix $\mathbf{D}_{Lb}^1$ has dimensions equal to $[n+1 \times n+1]$. The initial and final states, together with the corresponding controls, can be determined by the optimization process. However, since the initial state is generally known, further constraints need to be imposed to make sure that the solution found by the optimizer satisfies the condition $\mathbf{x}(t_0) = \mathbf{x}_0$. We can also compute a matrix $\mathbf{D}_{Lb}^2$, which represents the discrete linear operator to compute the second derivative, that is

$$\ddot{\mathbf{X}} \approx \mathbf{D}_{Lb}^2 \cdot \mathbf{X} \tag{7}$$

This definition is useful for imposing constraints on the second derivative of a function as a linear operator.

If we look at Eq. (2) and we take the derivative with respect to time, we get

$$\dot{\mathbf{x}}(t) \approx \frac{\mathrm{d}}{\mathrm{dt}} \sum_{i=0}^{n} \mathbf{X_i} P(t) = \sum_{i=0}^{n} \mathbf{X_i} \frac{\mathrm{d}}{\mathrm{dt}} P_i(t) \tag{8}$$

as the nodal points are time-independent. These derivatives can be efficiently computed through Barycentric Lagrange Interpolation [32]. In addition to the differential operator, we need an integral operator. This operator is required, as the cost function in Table 1 may contain the Lagrange term, which needs a proper discretization. In that case Gauss's quadrature formula is used [33]. For the LPM the approach consists of replacing the continuous integral with the discrete sum given by:

$$\int_{t_0}^{t_f} \Psi\left[t, \mathbf{x}(t), \mathbf{u}(t)\right] dt \approx \frac{t_f - t_0}{2} \sum_{i=0}^{n} w_i \Psi\left[\mathbf{X}_i, \mathbf{U}_i\right] \tag{9}$$

It can be shown that Eq. (9) yields exact results for polynomials of order at most equal to $2n - 3$ [31]. Once again, the presence of the term $\frac{t_f - t_0}{2}$ is a consequence of the mapping between pseudospectral and physical time domains given by Eqs. (5) and (6). The weights w_j can be computed as

$$w_j = \begin{cases} \dfrac{2}{(n-1)n}, & j = 0 \\ \dfrac{2}{(n-1)nL_{n-1}(\tau_j)^2}, & j = [1, \ldots, n-1] \end{cases} \tag{10}$$

Once the differential and integral operators have been described, the general NLP transcription, which approximates the original OCP can be summarized in Table 2.

Table 2: Lobatto discrete form of generic Optimal Control Problem

Description	**Formulation**
Cost function	$J = \Phi\left[\mathbf{X}_f\right] + \frac{t_f - t_0}{2} \sum_{i=0}^{n} w_i \Psi\left[\mathbf{X}_i, \mathbf{U}_i\right]$
Dynamics	$\mathbf{F}_i = \mathbf{D}^1_{Lb,i} \cdot \mathbf{X} - \frac{t_f - t_0}{2} \mathbf{f}(t_i, \mathbf{X}_i, \mathbf{U}_i) = \mathbf{0}$
Path constraints	$\mathbf{g}_L \leq \mathbf{G}\left(\mathbf{X}_i, \mathbf{U}_i\right) \leq \mathbf{g}_U$
State boundaries	$\mathbf{x}_L \leq \mathbf{X}_i \leq \mathbf{x}_U$
Control boundaries	$\mathbf{u}_L \leq \mathbf{U}_i \leq \mathbf{u}_U$
Initial state	$\mathbf{x}_0 = \mathbf{x}(t_0)$
Final state	$\mathbf{x}_f \in \chi(t_f)$

More details about LPM can be found in Refs. [31, 20].

3. Convex Optimization

Over the last thirty years several researchers focused on the development of convex optimization theory [17, 34]. They demonstrated that for a large class of problems the key-property is not the linearity of the system, but the convexity. A relevant sub-category of problems enjoying this property can be solved in real-time, and if the problem is feasible, the computed solution is the global optimum.

3.1. Convex programming

In general, a convex optimization problem is defined as follows:

$$\text{minimize } J = f_0(x) \tag{11}$$

subject to

$$f_i(x) \leq a_i, \qquad i = 1, \ldots, m \tag{12}$$

where $x \in \mathbb{R}^n$ represents the vector of variables to be determined. The functions f_i, with $i = 0, \ldots, m$, are convex functions, which means that they satisfy the following relationship.

$$f_i(\alpha x + \beta y) \leq \alpha f_i(x) + (1 - \alpha) f_i(y), \qquad i = 0, \ldots, m, \quad \forall \alpha \in [0,\ 1] \tag{13}$$

The previous expression suggests one of the properties of convex problems, that is, they generalize the notion of linearity of a function, leading to the notion of convexity, which has the equality as special case instead of the inequality in Eq. (13). Further details and exhaustive explanations can be found in the works of Boyd [17], and Ben Tal and Nemirovski [34]. The following properties characterize convex optimization:

1. A large number of problems can be reformulated in convex form
2. There are efficient methods to solve convex problems (e.g., primal-dual interior point methods), such that it can be considered more and more a mature technology
3. This class of methods does not require an initial guess (a problem, which affects many problems when NLP solvers are employed)
4. If a solution to the problem exists, it is the global optimum.

While the category of convex optimization is still quite large, and includes several subfields (e.g., semidefinite programming, quadratically constrained quadratic programming, and so on), we will instead focus on a specific form of convex optimization, that is, the so-called second-order cone programming (or SOCP). We will briefly describe this specific subclass of methods in the next section, whereas more extensive and rigorous descriptions of the theoretical properties and the potential applications of this technology can be found in literature [17, 34, 35].

3.2. Second-Order Cone Programming

An interesting subcategory of convex optimization is represented by SOCP. This definition encloses all the problems which can be formulated as follows:

$$\text{minimize} \quad \mathbf{c}_0^T \mathbf{x} \tag{14}$$

subject to

$$\begin{aligned} \mathbf{G}_i \mathbf{x} &\leq \mathbf{h}_i, \quad i = 1, \dots, l \\ \mathbf{A}_0 \mathbf{x} &= \mathbf{b}_0 \\ \|\mathbf{A}_i \mathbf{x} + \mathbf{b}_i\|_2 &\leq \mathbf{c}_i^T \mathbf{x} + d_i, \quad i = 1, \dots, p \end{aligned} \tag{15}$$

with $\mathbf{x} \in \mathbb{R}^{n\times 1}$ representing the variables to determine, $\mathbf{c}_0 \in \mathbb{R}^{n\times 1}$ is the vector defining the cost function, whereas $\mathbf{G}_i \in \mathbb{R}^{l_i\times n}$ and $\mathbf{h}_l \in \mathbb{R}^{l_i\times 1}$ represent (with l_i that might vary for every i) a set of component-wise inequalities. $\mathbf{A}_0 \in \mathbb{R}^{m\times n}$ and $\mathbf{b}_0 \in \mathbb{R}^{m\times 1}$ describe the linear system of m equations that the solution has to satisfy. The terms $\mathbf{A}_i \in \mathbb{R}^{m_i\times n}$, $\mathbf{b}_i \in \mathbb{R}^{m_i\times 1}$, $\mathbf{c}_i \in \mathbb{R}^{n\times 1}$ and $d_i \in \mathbb{R}$ describe a conic constraint of order m_i+1. These constraints imply that, given the affine transformations

$$\begin{aligned} t &= \mathbf{c}_i^T \mathbf{x} + d_i \\ y &= \mathbf{A}_i \mathbf{x} + \mathbf{b}_i, \ i = 1, \dots, p \end{aligned} \tag{16}$$

the solution will always be contained within the volume of each of the p m_i-dimensional cones. Among others, linear programming or quadratically constrained problems can be reformulated as conic programming problems. Moreover, they can be efficiently solved by using primal-dual interior-point methods, and several solvers, such as the open source ECOS [36], are available. These aspects make the SOCP technology appealing for several applications including formation flying [37], powered descent and landing [38, 39, 20, 40], and ascent scenarios [41].

4. Atmospheric Entry

This section will introduce the reentry vehicle used for the simulations, HORUS, visible in Fig. 1, and will briefly explain its reference mission and aerodynamics, the environment models, controls, as well as the trajectory constraints.

HORUS was designed to be a manned, Space-Shuttle-like re-entry vehicle. The original purpose of HORUS was to serve as a reusable second stage to the Ariane-5 launcher. HORUS was later redesigned as the second stage of Sänger II, a two-stage-to-orbit concept, for which HORUS would be fitted with a rocket engine to bring it up to orbit. Both projects were canceled for budgetary reasons, but a complete aerodynamic database of HORUS (the version without rocket engine) is available [26], which makes it possible to use it as a reference vehicle for reentry simulation.

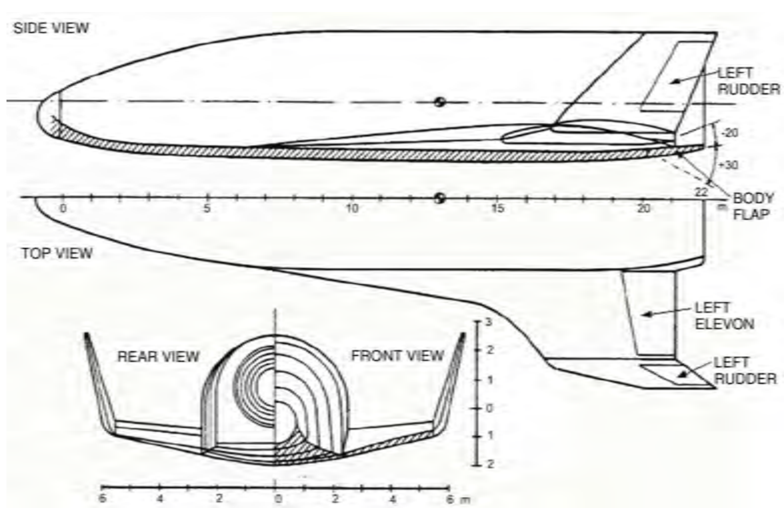

Figure 1: Different views and the dimensions of HORUS.

4.1. Reference Mission

The relevant mission details of HORUS are shown in Table 3. HORUS flight starts above the Pacific Ocean and it ends at the runway at Kourou, French Guiana. The trajectory is split into three parts, and begins with the pre-guidance phase, where the aerodynamic forces are too small, and the following predefined bank-angle law is used,

$$\sigma = \sigma^* \frac{e}{e^*}, \quad e \in [0,\ 0.1] \tag{17}$$

with σ^* equal to 60° and $e^* = 0.1$, selected to ensure that the drag acceleration is still within the prescribed boundaries at the moment of invoking the proposed scheme.

Once that the vehicle dives deep into the atmosphere, and the aerodynamic accelerations become strong enough to counteract the gravity, the hypersonic guided entry phase begins. This phase nominally starts when 10% of the energy is dissipated. The third and last phase before landing is the Terminal Area Energy Management (TAEM) phase. To properly protect HORUS during entry, a predefined angle of attack is commanded (40°) for most of the time to limit the maximum heat flux. After about 960 s the angle of attack is linearly reduced towards a value of 15° (corresponding to the maximum L/D configuration), achieved at t = 1250 s, and kept constant after that, as visible in Fig 2.

HORUS flies an equilibrium-glide trajectory, i.e., its flight-path angle changes only very little over time. Since the chosen angle of attack implies a large lift force, the vehicle requires a large bank angle to prevent it from skipping and to keep it flying the equilibrium-glide trajectory. The bank angle can be modulated between -89 and 89 degrees. Moreover, to remain flying towards the target, HORUS performs a number of bank reversals, during which the sign of the bank angle is changed. In this context it is assumed that this change is instantaneous, even if proper change with realistic angular rate can be modeled in a more advanced modeling phase. This simplification does no affect the conclusions drawn about the proposed method, which is fundamentally conceived for solving the longitudinal entry guidance problem.

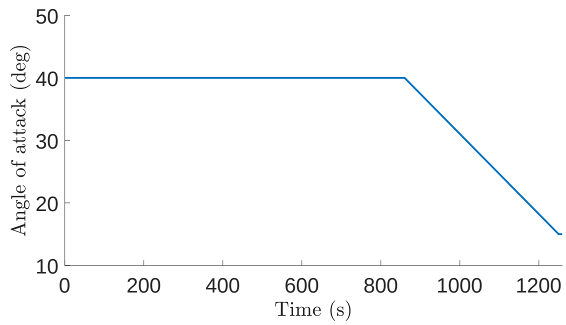

Figure 2: Reference angle-of-attack profile for HORUS.

Table 3: Main characteristics of HORUS' reference mission [42].

Property	Value	Property	Value
Wing area (S_{ref}) [m^2]	110	Maximum heat flux [kW/m^2]	550
Re-entry mass [kg]	26029	Maximum g-load [-]	2.5
Nose Radius [m]	0.8	Maximum dynamic pressure [N/m^2]	$1{\cdot}10^4$
Initial altitude [km]	122	Terminal altitude [km]	26.92 ± 2
Initial longitude [°]	-106.7	Terminal Mach number [-]	2.5 ± 0.5
Initial latitude [°]	-22.3	Terminal distance error [km]	± 2.5
Initial velocity [m/s]	7435.5	Terminal longitude [°]	-53.743
Initial flight-path angle [°]	-1.43	Terminal latitude [°]	4.990
Initial heading [°]	70.75		

4.2. Environmental and Aerodynamic Models

Here, the models for the shape, gravity field and atmosphere of the Earth are summarized.

Gravity model with J_2 term

An acceptable representation of the gravity field can be obtained if the oblateness of mass distribution of Earth is taken into account with the introduction of the zonal harmonic J_2. This correction depends on the radius and the geodetic latitude ϕ, and can be expressed as

$$g(r,\phi) = \frac{\mu_\oplus}{r^2} \cdot \left[1 + \frac{3}{2} J_2 \left(\frac{r}{R_\oplus}\right)^2 \left(1 - 3\sin^2\phi\right)\right] \tag{18}$$

where the gravitational parameter of the Earth $\mu_\oplus$ is equal to $3986004.418 \cdot 10^8$ $\mathrm{km}^3/\mathrm{s}^2$, the zonal coefficient J_2 is $1.0826271 \cdot 10^{-3}$, while the equatorial axis $R_\oplus$ is considered equal to 6378.137 m. Higher-order zonal coefficients can be neglected, as for the mission proposed here, as no significant differences were observed when these terms would be included.

Atmosphere

For the atmosphere modeling we use the combination of an exponential fit for the density and a temperature segmentation as is defined in the United States 1976 atmosphere to facilitate the derivation [43],

$$\rho = \rho_0 e^{-h/h_s} \tag{19}$$

where ρ_0 is equal to 1.225 kg/m^3, and h_s is the scale height, equal to 7200 m, while the temperature is modeled as follows.

$$T(h) = T_{i-1} + T_{h,i-1}(h - h_{i-1}), \quad h \in [h_{i-1},\ h_i] \tag{20}$$

It is stressed that this is not the same as the exponential atmosphere model commonly used, because such a mode would require a constant temperature profile. Given the described model for density and temperature we can simply compute the Mach number as

$$M = \frac{V}{\sqrt{\gamma_{air} R_{air} T}} \tag{21}$$

where $\gamma_{air} = 1.4$ is the specific heat ratio for the air, and $R_{air} = 287.058$ J / (kg K) is the specific air constant.

Aerodynamics

The aerodynamic coefficients of HORUS can be computed as follows.

$$C_{D,trim} = a_{D,0} + a_{D,1}\alpha + a_{D,2}\alpha^2 + a_{D,3}\alpha^3 + a_{D,4}M + a_{D,5}M^2 \\ + a_{D,6}\alpha M + a_{D,7}\frac{\alpha}{M} + a_{D,8}\frac{\alpha^2}{M} + a_{D,9}\frac{\alpha}{M^2} \tag{22}$$

$$C_{L,trim} = a_{L,0} + a_{L,1}M + \frac{a_{L,2}}{M + a_{L,3}} + \left(a_{L,4} + \frac{a_{L,5}}{M + a_{L,6}}\right)\sin\left(a_{L,7}\alpha + \frac{a_{L,8}\alpha}{M + a_{L,9}}\right) \\ + \left(a_{L,10} + a_{L,11}M + \frac{a_{L,12}}{M + a_{L,13}}\right)\cos\left(a_{L,7}\alpha + \frac{a_{L,8}\alpha}{M + a_{L,9}}\right) \tag{23}$$

The coefficients $a_{D,i}$ and $a_{L,i}$ can be found in the work of Bergsma and Mooij [44]. Note that the angle of attack is scheduled, and not used for feedback action. The Mach number always stays within the prescribed boundaries, therefore there is no risk of extrapolation of the aerodynamic coefficients. This occurrence is in any case forbidden in the guidance logic, since a saturation block is applied to the Mach number before entering the aerodynamic function.

4.3. Trajectory Constraints

In this research, three different trajectory constraints are used. The first is the convective heat flux, given by [45]

$$\dot{q}_c = \frac{C_1}{\sqrt{R_N}}\sqrt{\rho}V^m \leq 5.3 \cdot 10^5 \text{ W/m}^2 \tag{24}$$

with R_N, C_1 and m equal to 0.8 m, $5.28137 \cdot 10^{-5}$ Js$^{2.15}$/kg$^{0.5}$/m$^{2.15}$ and 3.15, respectively. The second constraint is the normal load factor,

$$n_z = \frac{|L\cos\alpha + D\sin\alpha|}{g_0} \leq 2.5 \tag{25}$$

while the last constraint is the dynamic pressure.

$$\bar{q} = \frac{1}{2}\rho V^2 \leq 10^4 \text{ N/m}^2 \tag{26}$$

The mission and the scenario are therefore completely characterized, and is possible to analyze the drag-energy dynamics.

5. Drag-Energy Dynamics

In this section the drag-energy dynamics will be derived. The results are not restricted to classical assumptions, such as the use of exponential atmospheric density or constant aerodynamic coefficients. All the contributions coming from the environment and the aerodynamics will be analyzed. The relationships will be used to extract the longitudinal states from the drag-energy solution obtained with the method described in Sec. 6. The motion is represented in terms of altitude h, longitude θ, latitude ϕ, speed V, flight-path angle γ and velocity azimuth angle ψ, equal to zero when the vehicle flies towards the north. The motion of a vehicle in a rotating spherical Earth can be described by the following set of differential equations.

$$
\begin{aligned}
\dot{h} &= V \sin\gamma \\
\dot{\theta} &= \frac{V\cos\gamma\sin\psi}{r\cos\phi} \\
\dot{\phi} &= \frac{V\cos\gamma\cos\psi}{r} \\
\dot{V} &= -D - g\sin\gamma + \omega_{\oplus}^2 r\cos\phi\left(\sin\gamma\cos\phi - \cos\gamma\sin\phi\cos\psi\right) \\
\dot{\gamma} &= \frac{L\cos\sigma}{V} + \left(\frac{V}{r} - \frac{g}{V}\right)\cos\gamma + 2\omega_{\oplus}\cos\phi\sin\psi + \\
&\quad + \frac{\omega_{\oplus}^2 r}{V}\cos\phi\left(\cos\gamma\cos\phi + \sin\gamma\sin\phi\cos\psi\right) \\
\dot{\psi} &= \frac{L\sin\sigma}{V\cos\gamma} + \frac{V}{r}\cos\gamma\sin\psi\tan\phi + 2\omega_{\oplus}\left(\sin\phi - \cos\phi\tan\gamma\cos\psi\right) + \\
&\quad + \frac{\omega_{\oplus}^2 r}{V\cos\gamma}\sin\phi\cos\phi\sin\psi
\end{aligned}
\tag{27}
$$

The model represented by Eq. (27) has as a feedforward control the angle of attack α (if it can be modulated), which affects C_L and C_D, whereas the bank angle σ consists of both a feedforward and a feedback contribution. We are interested to change the independent variable from time t to energy $E =$

$\frac{V^2}{2} - \frac{\mu_\oplus}{r}$, assumed to be 0 at infinity. Note that this choice is harmless, since we work with normalized energy, and therefore a different choice of reference value of energy would only lead to a different scaling constant for the energy normalization. To do it we use the rule of composite function differentiation, that applied to the longitudinal motion, that is, to the equations for h, V, and γ gives [46]

$$\begin{aligned} h' &= -\frac{\sin\gamma}{D} \\ V' &= \frac{1}{V} + \frac{g}{DV}\sin\gamma + C_V \\ \gamma' &= \frac{\cos\gamma}{V^2}\left(g - \frac{V^2}{r}\right)\frac{1}{D} - \frac{1}{V^2}u + C_\gamma \end{aligned} \tag{28}$$

where the control u is defined as

$$u = \frac{L}{D}\cos\sigma \tag{29}$$

and σ is the bank angle. The first derivative with respect the energy has been indicated with the symbol $'$. The terms C_V and C_γ embed the terms which take into account the Earth rotation with angular rate $\omega_\oplus$, and the energy dissipation rate with respect to time can be expressed as

$$\dot{E} = -DV \tag{30}$$

We are interested to develop a guidance system able to generate feasible solutions in the drag - energy domain. In a similar fashion to what has been done in literature [1, 4, 14], we can use a feedback-linearization technique, which allows us to control the second derivative of D as affine function with respect to the control u. Let us now differentiate the drag acceleration with respect to the energy. Starting from the drag acceleration definition, we can differentiate its logarithm, obtaining

$$\begin{aligned} D &= \frac{1}{2}\rho V^2 \frac{S}{m} C_D \\ D' &= D\left(\frac{\rho'}{\rho} + 2\frac{V'}{V} + \frac{{C_D}'}{C_D}\right) \end{aligned} \tag{31}$$

If we evaluate the second derivative, we get

$$\begin{aligned} D'' \quad = \quad & D'\left(\frac{\rho'}{\rho} + 2\frac{V'}{V} + \frac{{C_D}'}{C_D}\right) + \\ & +D\left[\frac{\rho''}{\rho} - \left(\frac{\rho'}{\rho}\right)^2 + 2\frac{V''}{V} - 2\left(\frac{V'}{V}\right)^2 + \frac{C_D''}{C_D} - \left(\frac{C_D'}{C_D}\right)^2\right] \end{aligned} \tag{32}$$

From the analysis of Eq. (32), we can see that we need some other derivatives as well. In the next sections, each of the contributions will be derived. Specifically, these contributions involve the atmospheric density, the aerodynamic coefficients, the Mach number, and the angle of attack.

5.1. Atmospheric density

Equation (32) requires the computation of the derivative of the atmospheric density with respect to the energy, which can be computed as

$$\begin{aligned} \rho' &= \rho_h h' \\ \rho'' &= \rho_{hh} {h'}^2 + \rho_h h'' \end{aligned} \tag{33}$$

where ρ_h and ρ_{hh} are the first and second derivatives of the atmospheric density with respect to the altitude, which can come from either an analytical or a numerical model. Therefore, we can write

$$\frac{\rho''}{\rho} - \frac{{\rho'}^2}{\rho^2} = \frac{\rho_{hh}{h'}^2 + \rho_h h''}{\rho} - \frac{\rho_h^2 {h'}^2}{\rho^2} \tag{34}$$

Clearly, Eq. (34) can be applied to any atmospheric model depending only on the altitude, and reduces to the model used for the Space Shuttle entry guidance [1] when the hypothesis of exponential atmospheric density is assumed.

5.2. Aerodynamics

Equation (32) includes the first and second derivative of the drag coefficient C_D. In general, C_D might depend on the angle of attack α, the Mach number

M, and the altitude h. Therefore, its derivatives are

$$\begin{aligned}
{C_D}' &= C_{D,\alpha}\alpha' + C_{D,M}M' + C_{D,h}h' \\
{C_D}'' &= C_{D,\alpha\alpha}{\alpha'}^2 + C_{D,\alpha M}\alpha' M' + C_{D,\alpha h}\alpha' h' + C_{D,\alpha}\alpha'' + \\
&\quad + C_{D,M\alpha}M'\alpha' + C_{D,MM}{M'}^2 + C_{D,Mh}M'h' + C_{D,M}M'' + \\
&\quad + C_{D,h\alpha}h'\alpha' + C_{D,hM}h'M' + C_{D,hh}{h'}^2 + C_{D,h}h''
\end{aligned} \tag{35}$$

The derivatives $C_{D,i}$ and $C_{D,jk}$ are the first and second derivatives of the drag coefficient with respect to the variables i, and j, k, respectively. These terms are computed analytically through Eq. (22).

Note that we need the derivatives of the Mach number with respect to several variables. We can therefore write

$$\begin{aligned}
M' &= M_h h' + M_V V' \\
M'' &= M_{hh}{h'}^2 + 2M_{Vh}h'V' + M_h h'' + M_{VV}{V'}^2 + M_V V''
\end{aligned} \tag{36}$$

The derivatives of the Mach number with respect to h and V can be obtained analytically from the definition of Mach number, $M = V/\sqrt{\gamma_{air}R_{air}T}$

$$M_h = -\frac{M}{2}\frac{T_h}{T}, \quad M_V = \frac{1}{\sqrt{\gamma_{air}R_{air}T}} \tag{37}$$

T_h is the temperature derivative with respect to the altitude, and can be obtained from Eq. (20).

Remark 1 The derivatives of the angle of attack α' and α'', are known, as the function $\alpha(E)$ is given in numerical form.

5.3. Feedback linearized dynamics

With the previous contributions defined, it is now possible to define the feedback-linearized dynamics. Let us define the second derivatives of altitude h and velocity V with respect to the energy. Starting from Eq. (28), and differentiating twice we get the following expressions.

$$
\begin{aligned}
h'' &= \frac{\sin\gamma}{D^2}D' - \frac{\cos\gamma}{D}\gamma' \\
V'' &= -\frac{V'}{V^2} + \frac{g'\sin\gamma}{DV} + \frac{g\cos\gamma\gamma'}{DV} - \frac{g\sin\gamma}{D^2V^2}\left(D'V + DV'\right)
\end{aligned}
\tag{38}
$$

Since the required derivatives have been defined, we can build the relationship between the second derivative of drag acceleration D'' and the control u. To do so, we will express all variables in affine form, separating the part depending on the control, from the part that does not. For the flight-path angle, by differentiating again Eq. (28) with respect to the energy we have

$$\gamma' = a_\gamma + b_\gamma u \tag{39}$$

with

$$a_\gamma = \frac{\cos\gamma}{D}\left(\frac{g}{V^2} - \frac{1}{r}\right) + C_\gamma, \quad b_\gamma = -\frac{1}{V^2} \tag{40}$$

with C_γ representing a correction term coming from the presence of $\omega_\oplus$ in the differential equation of the flight-path angle in Eq. (27), and equal to

$$C_\gamma = -2\frac{\omega_\oplus\cos\phi\sin\psi}{DV} - \frac{\omega_\oplus^2 r\cos\phi}{DV^2}\left(\cos\gamma\cos\phi + \sin\gamma\sin\phi\cos\psi\right), \tag{41}$$

where the profiles for ϕ and ψ are obtained by linearly interpolating between their corresponding initial and final values along the trajectory. The second derivative of altitude becomes

$$h'' = a_h + b_h u \tag{42}$$

with

$$a_h = \sin\gamma\frac{D'}{D^2} - \frac{\cos\gamma}{D}a_\gamma, \quad b_h = -\frac{\cos\gamma}{D}b_\gamma \tag{43}$$

The second derivative of atmospheric density becomes

$$\rho'' = a_\rho + b_\rho u \tag{44}$$

with

$$a_\rho = \rho_{hh} {h'}^2 + \rho_h a_h, \quad b_\rho = \rho_h b_h \tag{45}$$

For the velocity we have

$$V'' = a_V + b_V u \tag{46}$$

with

$$a_V = -\frac{V'}{V^2} + \frac{g' \sin\gamma}{DV} + \frac{g \cos\gamma a_\gamma}{DV} - \frac{g \sin\gamma}{D^2 V^2} \left(D'V + DV'\right), \quad b_V = \frac{g \cos\gamma b_\gamma}{DV} \tag{47}$$

The Mach number in affine form is

$$M'' = a_M + b_M u \tag{48}$$

with

$$a_M = M_{hh} {h'}^2 + 2M_{Vh} h' v' + M_h a_h + M_{VV} {V'}^2 + M_V a_V, \quad b_M = M_h b_h + M_V b_V \tag{49}$$

and the additional second derivatives are

$$M_{hh} = \frac{3}{4} M \left(\frac{T_h}{T}\right)^2 - \frac{M}{2} \frac{T_{hh}}{T}, \quad M_{hV} = M_{Vh} = -\frac{1}{2} \frac{T_h}{T} \quad M_{VV} = 0 \tag{50}$$

Finally, for the second derivative of C_D, we have

$$C_D'' = a_{C_D} + b_{C_D} u \tag{51}$$

with

$$\begin{aligned} a_{C_D} &= C_{D,\alpha\alpha} {\alpha'}^2 + C_{D,\alpha M} \alpha' M' + C_{D,\alpha h} \alpha' h' + C_{D,\alpha} \alpha'' + \\ &\quad + C_{D,M\alpha} M' \alpha' + C_{D,MM} {M'}^2 + C_{D,Mh} M' h' + C_{D,M} a_M + \\ &\quad + C_{D,h\alpha} h' \alpha' + C_{D,hM} h' M' + C_{D,hh} {h'}^2 + C_{D,h} a_h \\ b_{C_D} &= C_{D,M} b_M + C_{D,h} b_h \end{aligned} \tag{52}$$

The second derivative of drag acceleration can thus be expressed as

$$D'' = a_D + b_D u \tag{53}$$

with

$$\begin{aligned} a_D &= D'\left(\frac{\rho'}{\rho} + 2\frac{V'}{V} + \frac{{C_D}'}{C_D}\right) + D\left(\frac{a_\rho}{\rho} - \frac{\rho'^2}{\rho^2} + 2\frac{a_V}{V} - 2\frac{V'^2}{V^2} + \frac{a_{C_D}}{C_D} - \frac{{C_D}'^2}{C_D^2}\right) \\ b_D &= D\left(\frac{b_\rho}{\rho} + 2\frac{b_V}{V} + \frac{b_{C_D}}{C_D}\right) \end{aligned} \tag{54}$$

We can now design the guidance scheme based on the pseudospectral convex scheme.

6. Convexification of drag-energy guidance problem

In this section the drag-energy problem in pseudospectral convex form will be formulated. The first step is the proper computation of the feasible drag-energy space. Second, it is possible to formulate the problem to be solved in terms of drag-energy. Third, the pseudospectral convex form, based on the inverse of drag acceleration is proposed.

6.1. Drag-energy feasible space

The first operation to perform is the determination of the feasible drag-energy space. By feasible space in this context we mean those points on the drag-energy plane, which do not cause any constraint violation, (e.g., a too large load factor or heat-flux). In fact, one can observe that, in the hypothesis of nominal angle of attack, $\alpha = \alpha(E)$, all the constraints are a function of altitude and velocity only. E is the specific mechanical energy of the system, equal to $\frac{V^2}{2} - \frac{\mu_\oplus}{r}$, and is a function of altitude and velocity, too. Therefore, for each of the constraints and each energy level, we can solve a system of two nonlinear equations, obtaining the altitude and velocity, which generate the maximum value of a specific constraint. Since α is a given function of energy, we can compute the corresponding drag acceleration. If we exceed that specific value at that energy level, the specific constraint will be violated. This process can be done for each of the three constraints considered here, and is performed by using the bisection method to solve the systems of nonlinear equations. Note that other numerical methods could be employed to speed up the convergence of solution, but bisection is here selected for its simplicity.

Dynamic pressure

In this case, for each value of the energy E, the two equations to be solved are

$$\begin{aligned} \frac{V^2}{2} - \frac{\mu_\oplus}{h + r_\oplus} &= E \\ \frac{1}{2}\rho(h)V^2 &= \bar{q}_{max} \end{aligned} \tag{55}$$

where $\bar{q}_{max}$ is the maximum value allowed for the dynamic pressure. The solution to Eq. (55) will give us the altitude $h_{\bar{q}}$ and velocity $V_{\bar{q}}$ profiles, which maximize the dynamic pressure. The corresponding drag acceleration is

$$D_{\bar{q},max} = \frac{1}{2}\rho(h_{\bar{q}}){V_{\bar{q}}}^2\frac{S}{m}C_D(\alpha, h_{\bar{q}}, V_{\bar{q}}) \tag{56}$$

Heat flux

In this case the two equations to be solved are the first of Eq. (55) and

$$\frac{C_1}{\sqrt{R_N}}\sqrt{\rho}V^m = \dot{q}_{c,max} \tag{57}$$

where $\dot{q}_{c,max}$ is the maximum value allowed for the heat-flux. In this case the solution to Eq. (57) will give us the altitude h_Q and velocity V_Q profiles which maximize the heat flux. The corresponding drag acceleration is

$$D_{\dot{q}_c} = \frac{1}{2}\rho(h_{q_c}){V_{q_c}}^2\frac{S}{m}C_D(\alpha, h_{q_c}, V_{q_c}) \tag{58}$$

Note that this equation, based on cold-wall model, is conservative, which implies that the effective heat flux experienced by the vehicle will be lower.

Vertical load factor

The third constraint taken into account is the vertical load factor. The corresponding equations to be solved are again the first of Eq. (55) together with

$$\frac{|L(\alpha,h,V)\cos\alpha + D(\alpha,h,V)\sin\alpha|}{g_0} = n_{z,max} \tag{59}$$

where $n_{z,max}$ is the maximum value allowed for the vertical load factor, and g_0 is the gravity acceleration at sea level. In this case the solution to Eq. (59) will give us the altitude h_{n_z} and velocity V_{n_z} profiles, which maximize the vertical load factor. The corresponding drag acceleration is

$$D_{n_z,max} = \frac{1}{2}\rho(h_{n_z}){V_{n_z}}^2\frac{S}{m}C_D(\alpha,h_{n_z},V_{n_z}) \tag{60}$$

The same procedure employed for the constraints is applied for the equilibrium-glide condition [47], with this lower limit in the drag-energy domain shown in Fig. 3. However, this constraint is not strictly enforced, as, especially during the beginning of the entry violations might occur due to the fact that the dynamic pressure is not large enough to realize the equilibrium condition. It is worth noticing that Fig. 3 shows the envelope in physical units to give an idea of the accelerations and energies involved, while for the rest of the work non-dimensional units are shown, with the drag acceleration expressed in g and the energy domain normalized to vary between 0 and 1.

Finally, note that no restrictions on the atmospheric model have been formulated. Therefore, Eqs. (55), (57), and (59) are completely generic.

$$D,E : D_{Eq_g} \leq D(E) \leq \min\begin{cases} D_{\bar{q},max} \\ D_{\dot{Q},max} \\ D_{n_z,max}\end{cases} \tag{61}$$

The drag reference profile to be generated needs to be defined between the upper and the lower boundaries depicted in Fig. 3. We can observe that in the first phase of the atmospheric entry the limiting factor is the heat flux, which

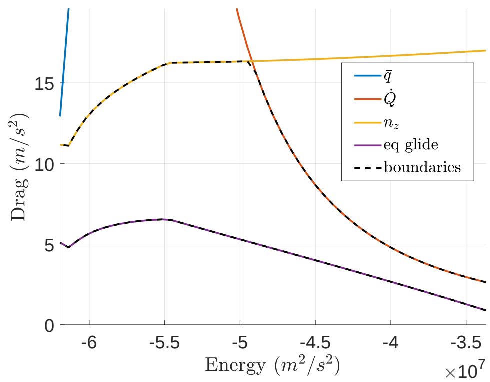

Figure 3: Drag-energy feasible space obtained for HORUS.

grows proportionally to $V^{3.15}$. Once the aerodynamic accelerations become more relevant and the speed decreases as some energy has been dissipated, the load factor becomes the prominent factor. For this specific mission profile the less demanding constraint is the dynamic pressure, as one can see. Finally, the equilibrium-glide condition provides the minimum level of drag to be generated.

6.2. *Drag-energy guidance problem*

We can now state the drag-energy guidance problem as follows: we want to compute a profile $D = D(E)$, such that

$$D_{min} = D_{Eq_g} \leq D(E) \leq D_{max} = \min \begin{cases} D_{\bar{q},max} \\ D_{\dot{q}_c,max} \\ D_{n_z,max} \end{cases} \tag{62}$$

Moreover, another constraint comes from the downrange requirements. In fact, one can show that the range traveled by a vehicle can be approximated by [46]

$$R \approx -\int_{E_0}^{E_f} \frac{dE}{D} \tag{63}$$

Initial conditions for the drag value come from the initial conditions of altitude and velocity.

$$D(E_0) = \frac{1}{2}\rho(h_0)V_0^2\frac{S}{m}C_D(\alpha_0, M_0) \tag{64}$$

Moreover, the initial condition of the drag derivative with respect to the energy is a direct consequence of the initial flight-path angle $\gamma(E_0)$. In fact, one can write

$$D'(E_0) = a_{D,\gamma} + b_{D,\gamma}\sin\gamma \tag{65}$$

where $a_{D,\gamma}$ and $b_{D,\gamma}$ can be computed by combining Eqs. (28), (31) and (35):

$$\begin{aligned} a_{D,\gamma} &= D\left[\frac{2}{V^2} + \frac{1}{C_D}\left(C_{D,\alpha}\alpha' + \frac{C_{D,M}M_V}{V}\right)\right] \\ b_{D,\gamma} &= D\left[-\frac{1}{D}\left(\frac{\rho_h}{\rho} + M_h\frac{C_{D,M}}{C_D}\right) + \frac{g}{DV}\left(\frac{2}{V} + \frac{C_{D,M}M_V}{C_D}\right)\right] \end{aligned} \tag{66}$$

Note that the same conditions can be applied to the final values of D and D' if corresponding values of h, V and γ at the end of the mission are assigned. Finally, a common assumption in drag-energy guidance methods for atmospheric entry is to assume the second derivative of the drag acceleration, given by Eqs. (53) and (54), to be equal to 0. Therefore, it is meaningful to include it in the problem as a variable to be minimized, such that the resulting trajectories will tend to a quasi-glide solution. The problem can be stated as follows. We are interested to compute drag-energy profiles such that

$$J = D'' = a_D + b_D u \tag{67}$$

is minimized while Eqs. (62)-(66) are satisfied. This choice will generate a drag profile which does not evolve too abruptly, leading to smoother longitudinal states. This problem can be discretized by using LPM introduced in Sec. 2. Pseudospectral methods provide the linear operators to approximate first and second derivatives with respect to the energy through Eqs. (3) and (7). This means that we can transcribe the previous problem as follows: let us define the discrete values of drag acceleration as

$$\mathbf{X} = \begin{Bmatrix} X_0 \\ X_1 \\ \vdots \\ X_n \end{Bmatrix} \tag{68}$$

We can now define the problem in discrete Lobatto form as

$$\text{minimize} \;\; D'' = \left\| \mathbf{D}^2_{Lb} \mathbf{X} \right\|_2 \tag{69}$$

subject to

$$D_{min} \leq X_i \leq D_{max} \quad i \in [0, \dots n] \tag{70}$$

and

$$X_0 = D_0, \quad X'_0 = D'_0 \tag{71}$$

To impose the range to be flown, we can use Eq. (9), and write

$$R \approx -\int_{E_0}^{E_f} \frac{dE}{D} = -\frac{E_f - E_0}{2} \sum_{i=0}^{n} \frac{w_i}{X_i} \tag{72}$$

Moreover, further constraints might be added at the end of the mission as

$$X_n = D_f, \quad X'_n = D'_f \tag{73}$$

The problem represented by Eqs. (69)-(73) is completely transcribed in Lobatto pseudospectral form. Unfortunately, it is not a convex problem, as we have a nonlinear equality constraint, given by Eq. (72), and therefore convex programming cannot be applied to the problem in this form. To overcome this difficulty the following observations were made.

- The only non-convex constraint is a linear combination of the inverses of the variables X_i, $i \in [0, \dots n]$

- All other constraints are boundary constraints, except the minimization of the second derivative with respect to energy

Let us introduce a new set of variables I_i, defined as

$$I_i = \frac{1}{X_i} \quad i = [0, \dots n] \tag{74}$$

We can reformulate all the constraints by using this new definition. The first, and most important consequence is that the non-convex equality constraint becomes a linear equality relationship, which, in fact, can be imposed as convex constraint.

$$R = -\frac{E_f - E_0}{2} \sum_{i=0}^{n} w_i I_i \quad i \in [0, \dots n] \tag{75}$$

Concerning the other constraints we can use Eq. (74) and rewrite them taking this transformation into account. Therefore, Eq. (71) becomes

$$I_0 = \frac{1}{D_0}, \quad I_0' = -\frac{2D_0'}{D_0^2} \tag{76}$$

Equivalently, Eq. (73) is expressed as

$$I_n = \frac{1}{D_f}, \quad I_n' = -\frac{2D_f'}{D_f^2} \tag{77}$$

The upper and lower boundaries expressed by Eq. (70) become now

$$D_{max} = I_{min} \leq I_i \leq I_{max} = D_{min}, \quad i \in [0, \dots n] \tag{78}$$

Finally, we need to transform the cost function. If we invert Eq. (74), that is

$$X(E) = \frac{1}{I(E)} \tag{79}$$

and differentiate twice we get

$$X'' = \frac{2(I')^2}{I^3} - \frac{I''}{I^2} \tag{80}$$

which means that

$$\|X''\|_2 = \left\| \frac{2(I')^2}{I^3} - \frac{I''}{I^2} \right\|_2 \tag{81}$$

Since I always takes finite, positive values, we can approximately minimize $\|X''\|_2$ by minimizing the single contributions, that is,

$$\min \|X''\|_2 \Longleftarrow \begin{cases} \min \|I'\|_2 \\ \min \|(I'')\|_2 \end{cases} \tag{82}$$

We can therefore include two slack variables $s' > 0$ and $s'' > 0$ such that

$$\begin{aligned} \|I'\|_2 &= \left\|\mathbf{D}^1_{Lb}\mathbf{I}\right\|_2 \leq s' \\ \|I''\|_2 &= \left\|\mathbf{D}^2_{Lb}\mathbf{I}\right\|_2 \leq s'' \end{aligned} \tag{83}$$

and rewrite the problem as follows: we aim at minimizing the cost function J

$$\text{minimize} \quad J = k_{s'} s' + k_{s''} s'' \tag{84}$$

subject to

$$\begin{aligned} I_L \leq I_i &\leq I_U \\ R = -\frac{E_f - E_0}{2} \sum_{i=0}^{n} &\frac{w_i}{I_i} \\ I_0 &= I(E_0) \\ I_f &\in I(E_f) \\ \|I'_i\|_2 &\leq s' \\ \|I''_i\|_2 &\leq s'' \end{aligned} \tag{85}$$

These $k_{s'}$ and $k_{s''}$ factors measure the relative importance of the first and second derivative of the drag profile with respect to the energy.

Remark 2 Note that the approximation taken in this approach is based on the fact that the values of I^2 and I^3, which appear in the denominator of Eq. (81), are not taken into account. However, we know that I is always positive; therefore, if I' and I'' tend to 0, X'' tends to 0 as well.

Equations (84)-(85) completely define the drag-energy guidance problem, which can be solved as a standard SOCP problem. Throughout this work ECOS [36] has been used. An example of the resulting drag-energy profile is depicted in Fig. 4, and it is clear that all the constraints are satisfied. Note that Fig.

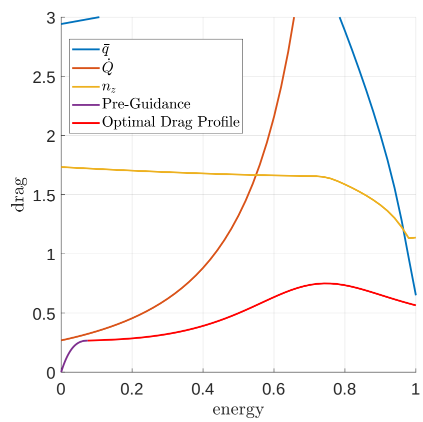

Figure 4: Example of reference drag profile obtained via pseudospectral convex optimization.

4 shows the drag acceleration and the energy in non-dimensional form. This is especially important for the energy, which is scaled such that by definition at the beginning of the trajectory the energy is equal to 0, and at the TAEM interface is equal to 1. Note that this only a convenient choice for the normalization, and has nothing to do with the physical phenomenon of energy dissipation, intrinsically associated with its monotonic decrease. The resulting profile of Fig. (4) is smooth, and defined between D_{min} and D_{max}. Moreover, conditions on the initial drag value and the drag derivative are also satisfied, as the generated profile perfectly matches with the values at the end of the pre-guidance phase, represented in the plot by the green profile. Moreover, as expected, the second derivative of the drag acceleration is minimized, excepted in the initial value that is constrained by the initial bank angle.

6.3. States extraction

Once the drag-energy profile has been computed, we can extract the longitudinal states and the bank angle, which satisfy the associated drag dynamics. For each value on the drag-energy plane belonging to the solution we can extract the corresponding altitude and speed by numerically solving the following set of equations:

$$\begin{aligned} E &= \frac{V^{*2}}{2} - \frac{\mu_\oplus}{h^* + r_\oplus} \\ D &= \frac{1}{2}\rho(h^*)V^{*2}\frac{S}{m}C_D \end{aligned} \tag{86}$$

The flight-path angle can be computed by inverting Eq. (65):

$$\gamma = \sin^{-1}\left(\frac{D' - a_{D,\gamma}}{b_{D,\gamma}}\right) \tag{87}$$

Finally, Eq. (54) can be inverted to extract the bank angle. First, the feedforward vertical control can be computed as

$$u_{FF} = \frac{D'' - a_D}{b_D} \tag{88}$$

and the bank angle is

$$\sigma = \cos^{-1}\left(\frac{u_{FF}}{C_L/C_D}\right) \tag{89}$$

The longitudinal states and the bank angle are depicted in Fig. 5. The extracted states satisfy the constraints defined in Table 3, and are consistent with the solution obtained in previous research for the same scenario [44]. We can observe a very good matching of the propagated states with the one predicted by the dynamics-inversion procedure proposed in this section.

6.4. Feedback Guidance

We can complete the guidance law by integrating a feedback term in the scheme. In a similar fashion to what has been done in literature [48], let us

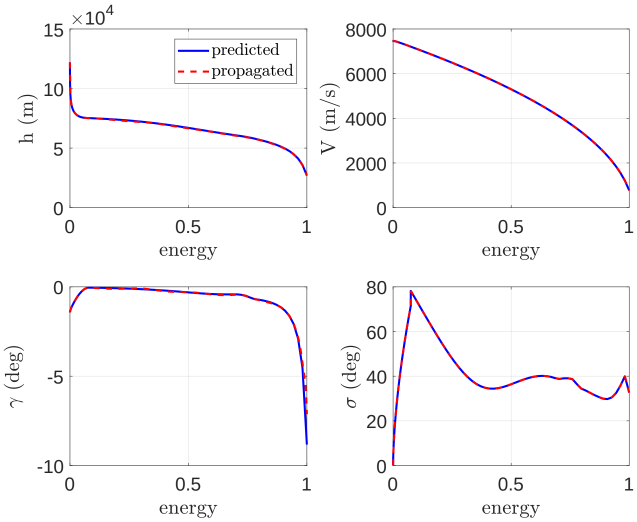

Figure 5: Longitudinal states and bank angle extracted from drag-energy profile, compared with corresponding open-loop propagation.

define the drag error and its derivatives,

$$\begin{aligned} \Delta D &= D - D_{ref} \\ \Delta D' &= D' - D'_{ref} \\ \Delta D'' &= D'' - D''_{ref} \end{aligned} \tag{90}$$

where the subscript $_{ref}$ refers to the reference solution obtained by using the method described in Sec. 6. The objective is to obtain a combined feedforward-feedback control law:

$$u = u_{FF} + u_{FB} \tag{91}$$

where the first term of the right side of Eq. (91) is given by Eq. (88). That leaves us to u_{FB}. Specifically, we are interested to model the error dynamics as a second-order system, that is

$$\Delta D'' + 2\zeta_D \omega_D \Delta D' + \omega_D^2 \Delta D + k_D^i \int_{E_0}^{E_F} \Delta D dE = 0 \tag{92}$$

If we solve for the control u, we will obtain

$$u = \frac{D''_{ref} - a_D - 2\zeta_D \omega_D \Delta D' - \omega_D^2 \Delta D - k_D^i \int_{E_0}^{E_F} \Delta D dE}{b_D} \tag{93}$$

If we combine Eqs. (89), (91), and (93) we can finally complete the optimal guidance scheme for HORUS.

$$u = u_{FF} + \frac{-2\zeta_D \omega_D \Delta D' - \omega_D^2 \Delta D - k_D^i \int_{E_0}^{E_F} \Delta D dE}{b_D} \tag{94}$$

The terms a_D, and b_D can be computed with Eq. (54). The damping ratio ζ_D and the pseudofrequency ω_D are chosen as a compromise between the saturation of the controls and the tolerated errors, while the integral term is included to improve the steady-state response of the system. The term pseudofrequency refers to the fact that we are not really dealing with a frequency (having units in Hz), but with something, which is only a frequency from a mathematical point of view. Physically it is not, as the independent domain is energy and not time. Therefore, the pseudofrequency is a parameter expressed in $\mathrm{s}^2/\mathrm{m}^2$. Moreover, since the energy domain is decreasing, to make the system stable we need to impose eigenvalues with positive real part. This implies that the damping ratio has to assume negative values. Finally, the sign of the bank angle can be computed by using a standard bank-reversal logic [1, 44].

7. Numerical Simulations

7.1. Monte-Carlo analysis

In this section some numerical results are described. The model described by the set of Eq. (28) will be used for validation of the proposed method, performed with a Monte-Carlo campaign of 1000 cases. The initial dispersions included in the analysis are described in Table 4. Values for $k_{s'}$ and $k_{s''}$ are equal to 25 and 1, respectively, while the proportional and derivative gains of the control law with respect to the energy are defined such that the damping ratio and the pseudofrequency are equal to $-7.9.057 \cdot 10^{-9}$ and $3.1623 \cdot 10^{-4}$ s^2/m^2 respectively. Finally, an integral gain equal to $2.8257 \cdot 10^{-13}$ has been included. The unusual nature of some of these values is a consequence of the completely different range of definition of the energy with respect to the more traditional time-based tuning.

Table 4: Dispersions at the entry interface (uniform distribution).

Variable	Dispersion	Units
Δh	$[-500\ 500]$	m
$\Delta\theta$	$[-0.25\ 0.25]$	deg
$\Delta\phi$	$[-0.25\ 0.25]$	deg
ΔV	[-15 15]	m/s
$\Delta\gamma$	[-0.01 0.01]	deg
$\Delta\psi$	[-0.01 0.01]	deg

Note that larger dispersions could be handled by the method itself. However, the dispersion included in the analysis is limited here, as the solution could be outside the vehicle's capabilities in terms of heat-flux and / or load factor. For instance, in case the vehicle has to strongly reduce its range-to-go, the larger drag required could be incompatible with the constraints' limits, leading therefore to an infeasible problem.

In Fig. 6 the longitudinal states obtained as function of the non-dimensional energy $e = \frac{E-E_0}{E_f-E_0}$ are depicted. Altitude, velocity, and flight-path angle are smooth, and continuously differentiable. Instead we can observe small jumps in the bank angle, due to the updates in the drag-energy profiles. This effect is reduced in magnitude (always in the range of 5 degrees), and it can nevertheless be mitigated by using a command shape filter [49].

Figure 7 shows the constraints, which are indirectly depicted in Fig. 8 as well, in the drag-energy space. All the solutions lie within the feasible constraint space. Specifically, for this scenario the heat-flux plays a major role in the generation of the constrained trajectory. Dynamic pressure and load factor are less stringent, as for all cases these constraints are never active. Note that the

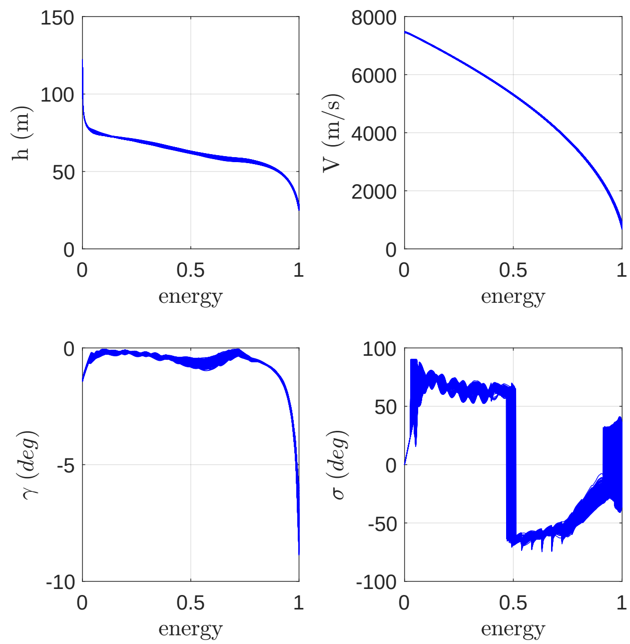

Figure 6: MC campaign: altitude, speed, flight-path angle, and bank angle (N=1000).

constraints are evaluated by using the current states, and not the reference ones obtained by the guidance solution, confirming the validity of the approximations taken in the development of the method proposed here. Finally, Fig. 9 shows the envelope of obtained trajectories, which are all consistent with the scenario. The accuracy of the method in terms of final position is visible in Fig. 10, where one can observe the final errors obtained in terms of longitude and latitude with respect to the reference one. Final dispersions are summarized in Table 6. The mean error values on final altitude (44 m), speed (17 m/s) and range-to-go (1846 m) errors are well within the requirements of the scenario. The standard deviations in terms of speed (about 67 m/s) and range-to-go (1172 m) are within the limits too. The distribution of the errors in terms of altitude, range-to-go and speed are also depicted in the histograms of Figs. 11-13. We can observe a uniform distribution of the final errors on altitude and speed. This in consistent

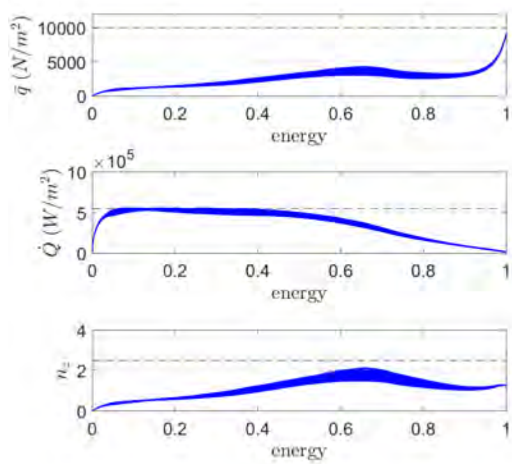

Figure 7: MC campaign: dynamic pressure, heat flux, and normal load factor.

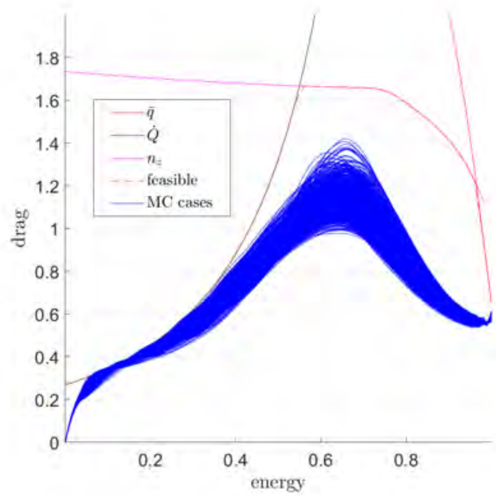

Figure 8: MC campaign: optimal drag-energy profiles.

with the capability of the method to reduce the initial uniformly distributed errors within acceptable boundaries specified for this scenario. However, a price in terms of accuracy of final altitude and speed is paid due to the fact that the current methodology generates minimum-curvature / minimum-slope drag profiles, showing properties similar to the quasi equilibrium-glide condition [50]. Towards the end of the trajectory this condition cannot be accurately satisfied anymore, leading therefore to the errors observed in Table 6. For what regards the error in terms of range-to-go, depicted in Fig. 12 we can see an average error which is 1.8 km. However, a closer inspection at the nature of this error shows that it is largely due to the lateral logic. This is visible in Fig. 10, where we can see that the largest errors are almost orthogonal to the downrange direction. On one side the results confirm the validity of the longitudinal logic proposed for entry guidance. On the other hand, they also show that to obtain superior performance a longitudinal-lateral integrated logic shall be included in the design. This further evolution of the methodology is still unexplored and is kept for future research work. A further contribution to the range error comes from the fact that the approximation in Eq. (63), which neglects the term $\cos\gamma$, is less accurate towards the end of the trajectory, leading to a further increase of the range error. Since the method has been mainly developed for HORUS, (i.e., a high L/D vehicle with a shallow entry flight-path angle), the effects of a larger dispersion error in terms of the initial γ have been investigated too. To assess the influence of this parameter on the range error a series of four subsets of Monte-Carlo campaigns of 25 cases each has been run. The campaigns are identical, except for the error in terms of γ, and the subsequent error on the range has been assessed. Results are summarized in Table 5, showing that the method is generally able to tolerate larger envelopes of γ. In fact, for an increase of a factor 10 in the γ error the corresponding increase in the range mean error is only of about 33% , while the standard deviation increases by a factor of 3.7, showing that the range error grows less than the initial flight-path angle. The range error scales even better when the dispersion is smaller, i.e., for the cases ± 0.025 deg and ± 0.05 deg. Nevertheless, when the initial γ reaches values closer to -0.1 deg it becomes more likely that the upper drag bound might be violated due to the steeper flight-path angle, and the impossibility for the vehicle to change its acceleration fast enough to stay within the drag boundaries. However, these limitations would require a re-design of the scenario, and are therefore more mission / vehicle-based than a limitation of the guidance method we are proposing here.

Table 5: Effects of initial γ error envelope on the range error

γ (deg)	error on final range - mean (km)	error on final range - std (km)
± 0.010	1.7578	0.5991
± 0.025	1.8891	0.6402
± 0.050	1.7809	0.9398
± 0.100	2.3311	2.1895

Finally, note that the terms composing the continuous expressions in Eq. (35) imply that we can compute the aerodynamic derivatives by using analytical expressions. Since this is not true in general, we tested the system with a discrete set of aerodynamic coefficients C_L and C_D sampled with $\Delta\alpha = 1$ deg, and $\Delta M = 0.1$. We built the remaining aerodynamic derivatives numerically from these coefficients by using finite differences, with no visible effects observed in the results. The plots are omitted for brevity, but confirm that the method can be also applied to standard look-up tables typically computed to model the aerodynamic database of hypersonic vehicles.

Table 6: Final dispersion errors at the terminal interface.

Variable	Mean	Standard Deviation	Units
h	44.83	1172.63	m
θ	-0.0052	0.0013	deg
ϕ	-0.0034	0.0117	deg
V	17.55	67.63	m/s
R	1846.64	881.88	m

7.2. Computational performance

In this section we characterize the computational performance of the proposed method, and perform a comparison with NLP-based transcription meth-

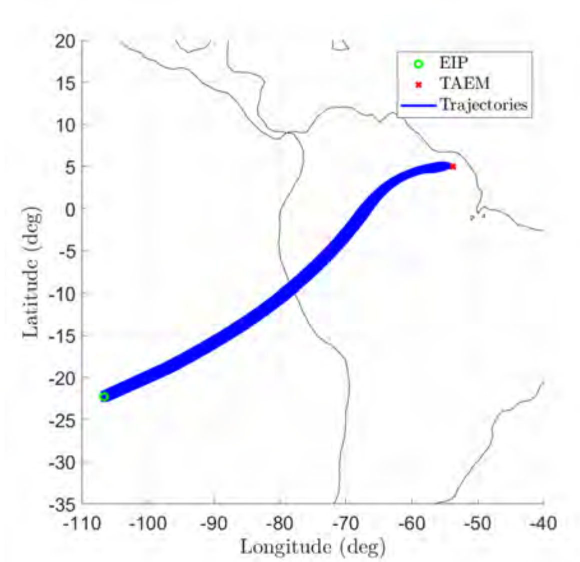

Figure 9: MC campaign: closed-loop trajectories (N=1000).

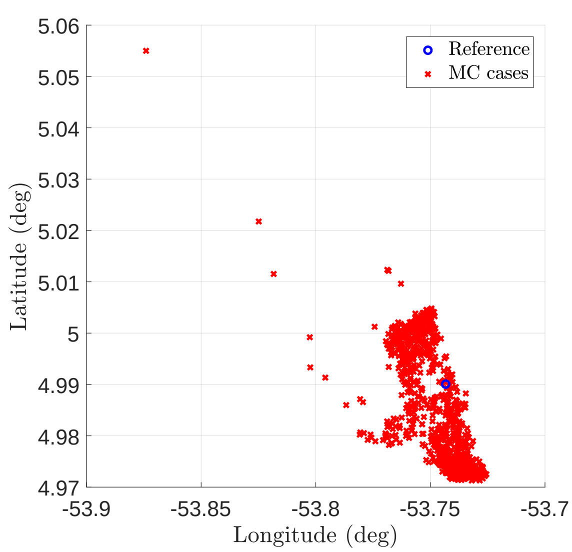

Figure 10: MC campaign: final longitudes and latitudes (N=1000).

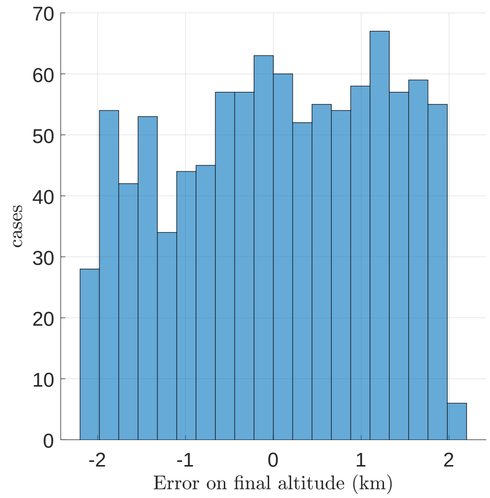

Figure 11: MC campaign: altitude error distribution (N=1000).

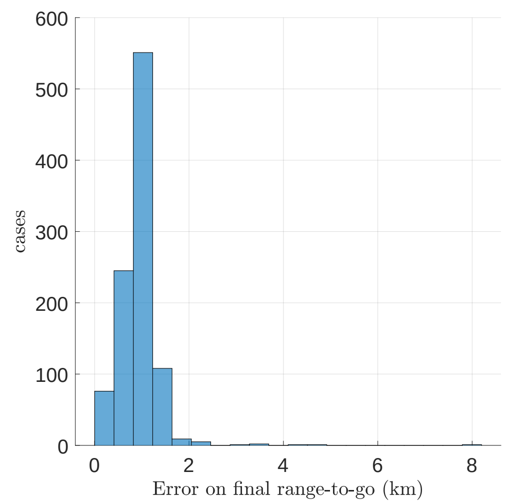

Figure 12: MC campaign: range-to-go error distribution (N=1000).

ods, specifically SPARTAN[29] and GPOPS[28]. In this case the full nonlinear equations of motion were considered, which include longitudinal-lateral coupling. The cost function is the integral of the square of the bank angle accel-

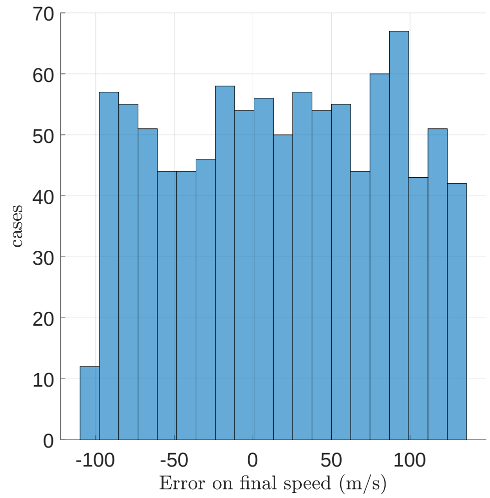

Figure 13: MC campaign: speed error distribution (N=1000).

eration, while the final states are enforced as hard constraints. A comparison of states, trajectories, and constraints is depicted in Figs. 14-16. When we look at the states the first observation that we can make is that the proposed method allows for a smoother trajectory, due to the fact that we are generating minimum-curvature / minimum-slope drag-energy profiles. This formulation implicitly embeds the quasi-equilibrium-glide condition in the trajectory, leading to reduced oscillations in altitude, speed and flight-path angle profiles, while the lack of enforcement of equilibrium-glide in the NLP-based formulation translates into solutions that oscillate more. In terms of bank-angle profiles, since our proposed method is intrinsically a longitudinal strategy, the bank reversal logic is included to satisfy the requirements on final position. This is visible from the sign switches at $t = 670$ s and after $t = 1000$ s. This extra-logic layer is not needed in the full nonlinear programming formulation: the bank angle changes sign along the trajectory generation in a continuous way. However, it is visible that the main sign change occurring at about 670 s is consistent among the three methods. All the trajectories end in the prescribed position. However, it should be emphasized that in the NLP approaches the final states are enforced as hard constraints. Therefore, the errors associated with them in the reference solution is theoretically equal to 0, although there will be a discretization mismatch, especially for SPARTAN, which does not use any mesh refinement scheme. The proposed scheme will instead show a final error, that needs to be compensated by the feedback law, and whose statistical characterization is summarized in Figs. 12 and 13. Moreover, note that SPARTAN exhibits a larger overshoot, which means that while GPOPS finds a solution that oscillates more longitudinally, SPARTAN exhibits a larger lateral motion to dissipate the energy along the reentry. Finally, all the methods satisfy the constraints requirements: however the drag-energy method exhibits a larger load-factor peak compared to the NLP methods, which instead, due to the higher oscillations aforementioned, especially towards the end of the trajectory, show higher peaks in the dynamic pressure. With reference to Table 7 there is a huge difference in the computational times observed: the Drag-Energy guidance method computes a solution in less than 50 ms on average, with a standard deviation of 16 ms. This is the benefit coming from solving the overall longitudinal guidance problem with one single convex optimization problem. The individual runs to get NLP solutions exhibit much larger computational time, with SPARTAN finding a solution in 55 s, and GPOPS in slightly more than 79 s. Moreover, SPARTAN does not include at the moment any mesh refinement, that would be necessary to improve the accuracy of the solution, and that would further increase the computational time needed to obtain a valid solution. This comparison suggests that the proposed methodology could effectively be employed for real-time computations.

Table 7: Computational performance comparison (Intel i5-6200U, 2.3 GHz)

Method	CPU Time, μ (m)	CPU Time, σ (ms)
Drag-Energy Guidance	0.048	0.016
SPARTAN	55.1	-
GPOPS	79.6	-

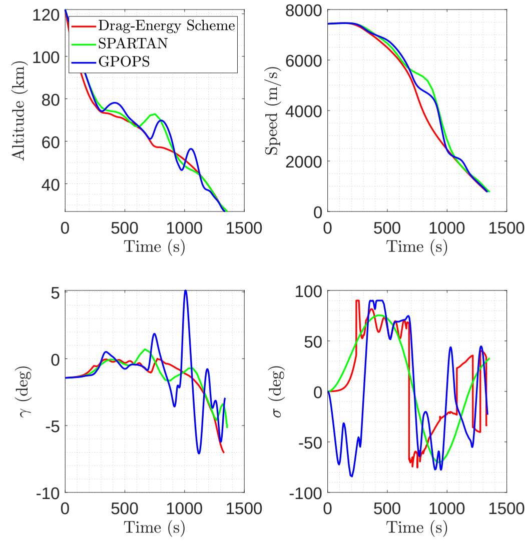

Figure 14: Comparison with NLP-based solutions - longitudinal states.

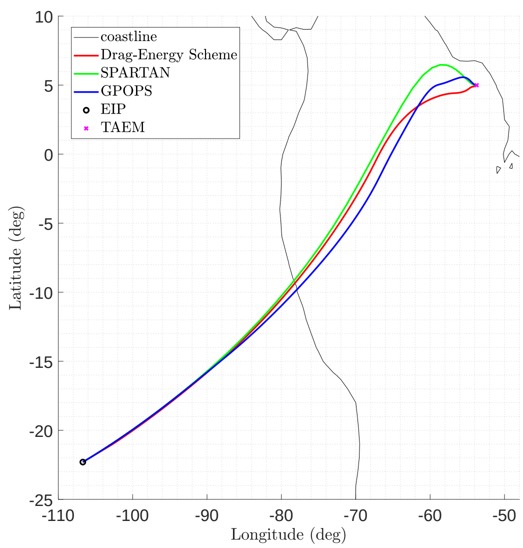

Figure 15: Comparison with NLP-based solutions - trajectories.

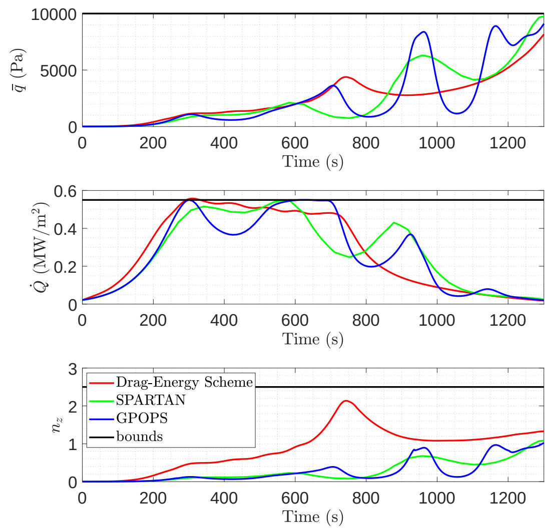

Figure 16: Comparison with NLP-based solutions - constraints.

8. Conclusions

This paper proposes a new strategy for the generation of atmospheric entry guidance solution via convex optimization. The problem is described in terms of drag-energy dynamics, and is not convex because of the nonlinear equality constraint representing the range-to-go condition. The introduction of a new set of variables corresponding to the inverse of drag acceleration allows for a lossless convexification of the problem, and therefore the possibility to solve it in real-time.

The main difference with respect to the pre-existing literature concerning convex optimization applied to entry guidance problems resides in the fact that in this case only one convex problem is required to be solved to generate a feasible drag-energy profile. The drag profile is later converted into altitude, velocity, flight-path angle and bank angle by exploiting drag-dynamics equations. Moreover, pseudospectral methods are integrated in the proposed strategy to have more accurate differential and integral operators.

The analysis of these preliminary simulations show that the method provides very good results, leading to a mean error in terms of range to-go of about 1.8 km at the Terminal area for energy management over a total flight distance of more than 6500 km, and an initial dispersion up to about 27.5×25.8 km^2. In all the cases the constraints acting on the system were satisfied, confirming the validity of the drag-energy representation to deal with nonlinear constraints. Most importantly, given its reduced computational burden the technique enables the capability to quickly recompute the trajectories in-flight, and therefore it is suitable for a potentially high technology-readiness level.

Acknowledgments

The authors thank the DLR technology marketing department, the TU Delft intellectual property office, the patent attorneys Rehberg Hueppe + Partner and the European Patent Office for the strong support in filing our related patent application EP17211042.1, as well as the anonymous reviewers for the useful comments.

References

[1] J. D. Harpold, C. A. Graves, Shuttle entry guidance, Journal of Astronautical Sciences, Vol.27 No.3, May-June 1979, pp.239-268 (1979).

[2] P. Lu, Entry guidance: A unified method, Journal of Guidance, Control, and Dynamics, Vol.37 No.3, May-June (2014) 713–728doi:10.2514/1.62605.

[3] X. Wang, J. Guo, S. Tang, S. Qi, Z. Wang, Entry trajectory planning with terminal full states constraints and multiple geographic constraints, Aerospace Science and Technology 84 (2019) 620–631. doi:10.1016/j.ast.2018.10.035.

[4] K. D. Mease, D. T. Chen, P. Teufel, H. Schöneberger, Reduced-order entry trajectory planning for acceleration guidance, Journal of Guidance,

Control, and Dynamics, Vol.25 No.2, March-April (2002) 257–266doi:10.2514/2.4906.

[5] E. Mooij, Adaptive heat-flux tracking for re-entry guidance, in: AIAA/AAS Astrodynamics Specialist Conference, San Diego, USA, 2014. doi:doi:10.2514/6.2014-4142.

[6] P. Lu, Entry guidance and control for reusable launch vehicles, in: AIAA Guidance, Navigation, and Control Conference, San Diego, USA, 1996. doi:10.2514/6.1996-3700.

[7] K. Y. Tu, M. S. Munir, K. D. Mease, Drag-based predictive tracking guidance for mars precision landing, Journal of Guidance, Control and Dynamics, Vol.23 No. 4, July-August 2000, pp.620-628 (2000) 620–628doi:10.2514/2.4607.

[8] J. T. Betts, Practical Methods for Optimal Control and Estimation Using Nonlinear Programming, 2nd ed., SIAM, Philadelphia, 2010.

[9] K. P. Bollino, I. M. Ross, D. Doman, Optimal nonlinear feedback guidance for reentry vehicles, in: AIAA Guidance Navigation, and Control Conference and Exhibit, Keystone, USA, 2006. doi:10.2514/6.2006-6074.

[10] B. Singh, R. Bhattacharya, Optimal guidance of hypersonic vehicles using b-splines and galerkin projection, in: AIAA Guidance, Navigation and Control Conference and Exhibit, 2008. doi:10.2514/6.2008-7263.

[11] M. Marwaha, B. Singh, J. Valasek, R. Bhattacharya, Integrated guidance and fault tolerant adaptive control for mars entry vehicle, in: AIAA (Ed.), AIAA Guidance, Navigation, and Control Conference, Vol. 2009, 2009. doi:10.2514/6.2009-5668.

[12] P. E. Gill, W. Murray, M. A. Saunders, User's Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming, University of California, San Diego, USA (2008).

[13] A. Wächter, L. T. Biegler, On the implementation of an interior-point filter linesearch algorithm for large-scale nonlinear programming, Math. Program. 106(1) , Springer-Verlag, New York, 2006 (2006) 25–57.

[14] A. Saraf, J. A. Leavitt, D. T. Chen, K. D. Mease, Desing and evaluation of an acceleration guidance algorithm for entry, Journal of Spacecraft and Rockets, Vol.41 No.6, November-December (2004) 986–996`doi:10.2514/1.1015`.

[15] J. Schierman, J. Hull, In-flight entry trajectory optimization for reusable launch vehicles, Guidance, Navigation, and Control and Co-located Conferences, American Institute of Aeronautics and Astronautics, 2005. `doi:10.2514/6.2005-6434`.
URL `https://doi.org/10.2514/6.2005-6434`

[16] M. Sagliano, E. Mooij, S. Theil, Onboard trajectory generation for entry vehicles via adaptive multivariate pseudospectral interpolation, Journal of Guidance, Control, and Dynamics 40 (2) (2016) 466–476. `doi:10.2514/1.G001817`.

[17] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.

[18] B. Acikmese, S. R. Ploen, Convex programming approach to powered descent guidance for mars landing, Journal of Guidance, Control, and Dynamics 30 (5) (2007) 1353–1366. `doi:10.2514/1.27553`.

[19] L. Blackmore, B. Acikmese, D. P. Scharf, Minimum-landing-error powered-descent guidance for mars landing using convex optimization, Journal of Guidance, Control, and Dynamics 33 (4) (2010) 1161–1171. `doi:10.2514/1.47202`.
URL `http://dx.doi.org/10.2514/1.47202`

[20] M. Sagliano, Pseudospectral convex optimization for powered descent and landing, Journal of Guidance, Control and Dynamics (2018) 320–334`doi:`

10.2514/1.G002818.
URL http://arc.aiaa.org/doi/abs/10.2514/1.G002818

[21] Y. Li, W. Chen, H. Zhou, L. Yang, Conjugate gradient method with pseudospectral collocation scheme for optimal rocket landing guidance, Aerospace Science and Technology 104 (2020) 105999. doi:10.1016/j.ast.2020.105999.

[22] X. Liu, Z. Shen, P. Lu, Entry trajectory optimization by second-order cone programming, Journal of Guidance, Control, and Dynamics 39 (2) (2015) 227–241. doi:10.2514/1.G001210.

[23] Z. Wang, Optimal trajectories and normal load analysis of hypersonic glide vehicles via convex optimization, Aerospace Science and Technology 87 (2019) 357–368. doi:10.1016/j.ast.2019.03.002.

[24] X. Liu, Fuel-optimal rocket landing with aerodynamic controls, Journal of Guidance, Control, and Dynamics (2018) 65–77doi:10.2514/1.g003537.
URL https://doi.org/10.2514/1.G003537

[25] G. T. Huntington, Advancement and analysis of a gauss pseudospectral transcription for optimal control problems, Ph.D. thesis, Citeseer (2007).

[26] Mooij, E., The HORUS-2B Reference Vehicle, Delft University of Technology. Memorandum M-682, 1995.

[27] Elissar, Description of dido optimal control software (June 2015).
URL http://www.elissarglobal.com/

[28] A. V. Rao, D. A. Benson, C. L. Darby, M. A. Patterson, C. Francolin, I. Sanders, G. T. Huntington, Gpops: A matlab software for solving multiple-phase optimal control problems using the gauss pseudospectral method, ACM Transactions on Mathematical Software 37 (2) (2010) 1–39. doi:10.1145/1731022.1731032.

[29] M. Sagliano, S. Theil, M. Bergsma, V. D'Onofrio, L. Whittle, V. G., On the radau pseudospectral method: theoretical and implementation advances, CEAS SPACE Journal (2017) 313–331doi:10.1007/s12567-017-0165-5.

[30] F. Fahroo, I. M. Ross, Direct trajectory optimization by a chebyshev pseudospectral method, Journal of Guidance, Control, and Dynamics 25 (1) (2002) 160–166. doi:10.2514/2.4862.
URL http://dx.doi.org/10.2514/2.4862

[31] D. Garg, Advances in global pseudospectral methods for optimal control, Ph.D. thesis, University of Florida, Gainesville (2011).

[32] J. R. R. Martins, P. Sturdza, J. J. Alonso, The complex-step derivative approximation, ACM Transactions on Mathematical Software 29 (3) (2003) 245–262. doi:10.1145/838250.838251.

[33] M. Abramovitz, I. A. Stegun, Handbook of Mathematical Functions, Dover Publications, 1695.

[34] A. Ben Tal, A. Nemirovski, Modern Lectures on Convex Optimization, Society for Industrial and Applied Mathematics, 2001.

[35] M. S. Lobo, L. Vandenberghe, S. Boyd, H. Lebret, Applications of second-order conic programming, Linear Algebra and Its Applications (1998) 193–228doi:10.1016/S0024-3795(98)10032-0.

[36] A. Domahidi, E. Chu, S. Boyd, Ecos: An socp solver for embedded systems, in: European Control Conference, 2013. doi:10.23919/ECC.2013.6669541.

[37] B. Acikmese, D. Scharf, F. Hadaegh, E. Murray, A convex guidance algorithm for formation reconfiguration, Guidance, Navigation, and Control and Co-located Conferences, American Institute of Aeronautics and Astronautics, 2006. doi:10.2514/6.2006-6070.
URL http://dx.doi.org/10.2514/6.2006-6070

[38] M. Szmuk, B. Acikmese, A. W. Berning, Successive convexification for fuel-optimal powered landing with aerodynamic drag and non-convex constraints, AIAA SciTech Forum, American Institute of Aeronautics and Astronautics, 2016. doi:10.2514/6.2016-0378.
URL http://dx.doi.org/10.2514/6.2016-0378

[39] M. Szmuk, U. Eren, B. Acikmese, Successive convexification for mars 6-dof powered descent landing guidance, AIAA SciTech Forum, American Institute of Aeronautics and Astronautics, 2017. doi:10.2514/6.2017-1500.
URL http://dx.doi.org/10.2514/6.2017-1500

[40] M. Sagliano, Generalized hp pseudospectral-convex programming for powered descent and landing, Journal of Guidance, Control, and Dynamics 42 (7) (2019) 1562–1570. doi:10.2514/1.g003731.

[41] Y. Li, Y. Guan, C. Wei, R. Hu, Optimal control of ascent trajectory for launch vehicles: A convex approach, IEEE Access 7 (2019) 186491–186498. doi:10.1109/access.2019.2960864.

[42] MBB, Study on re-entry guidance and control, Tech. Rep. ESA CR (P) 2652, Space Communication and Propulsion Systems Division (1988).

[43] NASA, U.s. standard atmosphere 1976, Tech. Rep. NASA Technical Memorandum X-74335 (1976).

[44] M. C. W. Bergsma, E. Mooij, Application of taylor-series integration to reentry problems with wind, Journal of Guidance, Control, and Dynamics 39 (10) (2016) 2324–2335. doi:10.2514/1.g000378.
URL https://doi.org/10.2514/1.G000378

[45] Mooij, E., Aerospace-Plane Flight Dynamics: Analysis of Guidance and Control Concepts, Ph.D. thesis, Delft University of Technology (1998).

[46] K. D. Mease, D. T. Chen, S. Tandon, D. H. Young, S. Kim, A three-dimensional predictive entry guidance approach, in: AIAA Guidance, Nav-

igation, and Control Conference and Exhibit, 2000. doi:10.2514/6.2000-3959.

[47] P. Lu, Gliding guidance of high l/d hypersonic vehicles, AIAA Guidance, Navigation, and Control Conference, Boston, USA, (2013). doi:10.2514/6.2013-4648.

[48] S. Bharadwaj, A. V. Rao, K. D. Mease, Entry trajectory tracking law via feedback linearization, Journal of Guidance, Control, and Dynamics, Vol.21 No.5, September-October 1998 (1998) 726–732doi:10.2514/2.4318.

[49] G. F. Trigo, Robust and adaptive nonlinear attitude control of a spacecraft (2011).

[50] P. Lu, Predictor-corrector entry guidance for low lifting vehicles, Journal of Guidance, Control, and Dynamics, Vol.31 No.4, July-August (2008). doi:10.2514/1.32055.