73rd International Astronautical Congress (IAC), Paris, France, 18-22 September 2022.

IAC-22-C1.5.2

Assessment of Reusable Launch Vehicles Re-entry Dynamics Control Effectiveness with Enhanced Aerodynamics Modelling

Alice De Oliveira[a*] **and Michèle Lavagna**[a]

[a]*Aerospace Science and Technology Department, Politecnico di Milano, via Giuseppe La Masa 34, 20156 Milano, Italy,* alice.deoliveira@polimi.it, michelle.lavagna@polimi.it
[*]Corresponding author

Abstract

The controlled atmospheric re-entry associated with the precision soft-landing of Reusable Launch Vehicles (RLV) on Earth is very challenging as it depends on multiple parameters. Over the last decade, the cost-effectiveness of such a technology has been finally demonstrated with the successful landing and reuse of SpaceX's Falcon 9 first-stage rocket. This breakthrough has been made possible by the development of advanced computational methods able to compute in real-time the flight conditions and to command the optimal vehicle's deflections accordingly to achieve a safe pinpoint landing on Earth. During an atmospheric re-entry, the vehicle is subjected to fast system dynamics changes partly induced by external loads associated with the terrestrial environment, such as aerodynamics, wind, and gust. In that purpose, the Guidance, Navigation & Control system must be advanced and robust enough to counteract them. This paper studies the development of a Six-Degree-of-Freedom (6-DoF) RLV re-entry dynamics simulator with a particular focus on the modelling of the aerodynamics and the uncertainties involved. It covers the atmospheric re-entry and vertical landing of a first-stage rocket equipped with a thrust vector control system. Then, advanced guidance and control methods are embedded into the simulator to assess their performance, particularly towards aerodynamics consideration. An aerodynamic model for reusable rockets with the quantification of the uncertainties involved has been developed and is presented in this study. This model, incorporated into the simulator, allows to obtain the aerodynamic coefficients and the center of pressure as function of the flight parameters (Mach number, aerodynamic angles), needed to calculate the corresponding forces and moments acting on the vehicle. Then, guidance and control methods are integrated in a closed-loop fashion into the simulator, and tested to assess their performance, as well as their robustness towards aerodynamic uncertainties through Monte-Carlo analyses. The results show that successive convex optimisation guidance method, which considers the aerodynamic drag, is particularly efficient to generate an optimal reference trajectory, then updated and followed by the vehicle until pinpoint landing. Furthermore, outputs of the Monte-Carlo analyses show the relevance of adding control authority (e.g. planar fins) and developing a robust control system able to withstand aerodynamic uncertainties. Finally, this study paves the way towards the design of a 6-DoF RLV re-entry dynamics simulator as a tool to develop and assess advanced guidance and control techniques. This study lies within the ASCenSIon (Advancing Space Access Capabilities - Reusability and Multiple Satellite Injection) project, an innovative training network funded within H2020.
Keywords: RLV, GNC, Aerodynamics, Re-entry dynamics simulator, Successive convex optimisation, PID control

Nomenclature

In this paper, the mathematical typesetting adopts the following notation: (i) scalars with non-bold, italic, expressed as x_{id} where 'id' is a descriptive identifier; (ii) vectors with bold, non-italic, expressed as $\mathbf{x}_{id,A}$ where A is the reference frame; (iii) $\mathbf{R}_C^B$ is a rotation matrix illustrating the rotation from reference frame B to C.

Acronyms/Abbreviations

6-DoF	Six-Degree-of-Freedom
ASCenSIon	Advancing Space Access Capabilities - Reusability and Multiple Satellite Injection
CG	Center of gravity
CP	Center of pressure
G&C	Guidance & Control
GNC	Guidance, Navigation & Control
LaWGS	Langley Wireframe Geometry Standard
NASA	National Aeronautics and Space Administration

PID	Proportional-Integral-Derivative
RLV	Reusable Launch Vehicle
S/HABP	Supersonic/Hypersonic Arbitrary-Body Program
SOCP	Second Order Cone Programming
VTVL	Vertical Take-off Vertical Landing

1. Introduction

Launch vehicle design experienced a significant paradigm shift over the last decade. Private companies like SpaceX and Blue Origin have indeed demonstrated the competitiveness and feasibility of launcher reusability for reducing the cost of access to space, therefore paving the way towards future manned missions such as a return to the Moon, or even more ambitiously, the first step on Mars. Consequently, national agencies and intergovernmental institutions are following the same path, increasing research and development on launcher reusability.

The atmospheric re-entry and pinpoint landing of the reuilsable stages on Earth is particularly challenging due to the atmosphere presence, the fast dynamics involved and existing uncertainties and perturbations [1]. Therefore, the Guidance, Navigation & Control (GNC) system must be advanced and robust enough to steer the vehicle towards the landing site while counteracting these effects and ensuring stability. In what regards Vertical Take-off Vertical Landing (VTVL) vehicles, this demanding problem became feasible in the past decade with the development of convex optimisation guidance [2]: a particular class of methods which allows to compute in real-time and based on the current flight conditions, optimal trajectories to be followed by the vehicle satisfying desired constraints (which must be convex).

More particularly, convex optimisation guidance consists in transforming the fuel-optimal re-entry trajectory problem into a convex one, more precisely into a Second Order Cone Programming (SOCP) problem, which can be solved with efficient solvers in polynomial time. This technique requires relatively high computational power as well as mathematical developments to convert the non-convex state and control constraints into convex forms. Whereas in the past decade, computational power available on-board have significantly increased, the non-convexity remains a challenging task. That conversion has been made possible in 2007 with the so-called *lossless convexification* method, first applied to Mars landing [3]. From there, further research on this method have been carried out with multiple applications [4], as well as the development of advanced related techniques such as successive convex optimisation [5], and pseudospectral convex optimisation [6].

Successive convex optimisation have enabled the consideration of the aerodynamics in the trajectory problem, previously ignored. Indeed, for Earth re-entry and landing trajectory problem, these forces must be considered for increasing performance and accuracy and therefore for enabling a pinpoint landing. In Ref. [7], Liu extended that method by combining aerodynamic forces and propulsion as control inputs to gain optimality. Then, in Ref. [8], Simplício *et al.* have achieved a trade-off between efficiency and optimality according to the study of the coupled flight mechanics and considered a larger flight envelope encountered by the RLV while using the aerodynamic model developed in Ref. [5]. Finally, in Ref. [9], Sagliano *et al.* combined successive convex optimisation and pseudospectral convex optimisation methods to develop a *sequential pseudospectral convex optimisation* method. Then, they proposed to separate the aerodynamic descent and the powered landing into two different optimal control problems, using the aerodynamic forces as control inputs for the first phase, and a combination of aerodynamic and propulsive control for the second phase.

The work in this paper is part of the project ASCenSIon (Advancing Space Access Capabilities - Reusability and Multiple Satellite Injection) which is born as an innovative training network with fifteen Early-Stage Researchers, ten beneficiaries, and fourteen partner organisations across Europe, to study the critical technologies needed to develop a Reusable Launch Vehicle (RLV) capable of injecting multiple payloads into multiple orbits. Therefore, one of the activities of this consortium is focused on the development of optimal trajectories for reusable launchers re-entry. Under that activity, a Six-Degree-of-Freedom (6-DoF) RLV re-entry dynamics simulator has been developed in Ref. [10] to assess Guidance & Control (G&C) techniques. It simulates the atmospheric re-entry and vertical landing of a first-stage rocket equipped with a Thrust Vector Control (TVC) system. Also in that reference, successive convex optimisation guidance and feedback control through PID controllers, both adapted from Ref. [11], is proposed and the preliminary assessment can be found in Ref. [12].

This paper aims to improve the previously defined re-entry dynamics simulator by developing and implementing an advanced aerodynamic model of the RLV with the quantification of the uncertainties involved. This model allows to obtain the aerodynamic coefficients and the center of pressure as function of the flight parameters (Mach number, aerodynamic angles), whose the values are needed to calculate the forces and moments acting on the vehicle during the flight. Then, the G&C effectiveness can be assessed towards this enhanced aerodynamic modelling of the launcher, particularly the successive convex optimisation guidance algorithm, and the robustness to uncertainties.

The layout of the paper is as follows. Sec. 2 introduces the reusable launchers re-entry dynamics simulator with the description of the building blocks: from flight mechanics models and vehicle's properties to closed-loop guidance and control methods implementation. Then, in Sec. 3, the aerodynamic model and the uncertainties involved are described and validated against models found in literature. Subsequently, several simulations and Monte-Carlo analyses are performed and compared in Sec. 4. Finally, conclusions and future works are provided in Sec. 5.

2. Reusable launchers re-entry dynamics simulator

This paper relies on the nonlinear 6-DoF RLV re-entry dynamics simulator developed in Ref. [10, 12]. It simulates the Earth atmospheric re-entry and landing of a VTVL vehicle first-stage booster modelled as a rigid body with varying mass, subjected to external forces induced by the terrestrial atmosphere and controlled through embedded closed-loop guidance and control strategies. Therefore, it is composed of several building blocks with interconnections between them. The elements have been implemented through MATLAB/Simulink.

At this stage of the RLV re-entry simulator design, the vehicle is only steered via TVC, while the outputs of a study aiming to add control authority with planar fins can be found in Ref. [10]. The simulator also includes a closed-loop G&C implementation with a successive convex optimisation guidance law for retro-propulsive entry, descent and pinpoint landing and feedback attitude control through gain-scheduled PID controllers. A description of the model developed with the existing interactions between the building blocks is depicted in Fig. 1 and presented in the following subsections.

2.1. RLV flight mechanics modelling

The equations of motion are written in the landing site-centered inertial and the vehicle's body-fixed reference frames. They are based on the initial state vector $\mathbf{x}_I(0) = \begin{bmatrix} m(0) & \mathbf{r}_I^T(0) & \mathbf{v}_I^T(0) & \mathbf{q}_B^I(0)^T & \boldsymbol{\omega}_B^T(0) \end{bmatrix}$ and on the assumptions that the vehicle is a rigid body with no effect induced by the varying mass (e.g. propellant sloshing) and structural flexibilities.

The mass depletion dynamics are modelled by an affine function of the thrust magnitude as follows:

$$\dot{m}(t) = -\frac{||\mathbf{F}_{TVC,I}(t)||_2}{I_{sp}g_0} - \frac{A_{nozzle}P_{amb}(t)}{I_{sp}g_0} \qquad (1)$$

where I_{sp} is the vacuum specific impulse of the engine, g_0 is the standard Earth gravity, A_{nozzle} is the nozzle exit area of the engine and $P_{amb}(t)$ is the ambient atmospheric pressure. $\mathbf{F}_{TVC,I}(t) \in \mathbb{R}^3$ is the thrust vector coming from the TVC system, represented in the inertial reference frame. The second term is related to the reduction of the specific impulse due to the atmospheric back-pressure [5].

The translational states, position and velocity of the vehicle in the inertial reference frame, $\mathbf{r}_I(t) \in \mathbb{R}^3$ and $\mathbf{v}_I(t) \in \mathbb{R}^3$, are governed by the following dynamics:

$$\begin{aligned} \dot{\mathbf{r}}_I(t) &= \mathbf{v}_I(t) \\ \dot{\mathbf{v}}_I(t) &= \frac{1}{m(t)}\left[\mathbf{F}_{TVC,I}(t) + \mathbf{F}_{aero,I}(t)\right] + \mathbf{g}_I(t) \end{aligned} \qquad (2)$$

where $\mathbf{F}_{aero,I}(t) \in \mathbb{R}^3$ and $\mathbf{g}_I(t) \in \mathbb{R}^3$ described, respectively, the aerodynamic and gravitational forces acting on the vehicle in the inertial reference frame.

Then, the attitude states are governed by the following rotational dynamics, using the quaternion-based kinematics equation:

$$\begin{aligned} \dot{\mathbf{q}}_B^I(t) &= \frac{1}{2}\begin{bmatrix} q_4(t) & -q_3(t) & q_2(t) \\ q_3(t) & q_4(t) & -q_1(t) \\ -q_2(t) & q_1(t) & q_4(t) \\ -q_1(t) & -q_2(t) & -q_3(t) \end{bmatrix} \boldsymbol{\omega}_B(t) \\ \dot{\boldsymbol{\omega}}_B(t) &= J^{-1}(t)\left[\mathbf{M}_{TVC,B}(t) + \mathbf{M}_{aero,B}(t)\right. \\ &\quad \left. -\boldsymbol{\omega}_B(t) \times J\boldsymbol{\omega}_B\right] \end{aligned} \qquad (3)$$

where $J(t)$ is the inertia matrix of the vehicle, and $\mathbf{M}_{TVC,B}(t) \in \mathbb{R}^3$ and $\mathbf{M}_{aero,B}(t) \in \mathbb{R}^3$ represent, respectively, the thrust and aerodynamic torques acting on the vehicle. In Eq. (3), the coupling between angular velocity and inertia along the three axes, and the effect of centroid movement on the inertia caused by mass consumption, are ignored.

The trajectory of the vehicle during the descent is controlled by adjusting the magnitude and the direction of the thrust vector generated by the main engine. It is achieved with the TVC actuator by deflecting the engine's nozzle by $\beta_{T,y}(t)$ and $\beta_{T,z}(t)$, respectively in the $\mathbf{y}_B$-axis and the $\mathbf{z}_B$-axis. The required thrust magnitude $T(t)$ and deflection angles $\{\beta_{T,y}(t), \beta_{T,z}(t)\}$ are obtained, from the guidance algorithm and the control method used, respectively. The decoupling between translational and rotational dynamics is common for TVC control due to the fact that the attitude of the vehicle can be changed faster than its trajectory [13]. Thus, the TVC-generated force and moment can be expressed in the body-fixed frame by:

$$\begin{aligned} \mathbf{F}_{TVC,B}(t) &= T(t)\begin{bmatrix} \cos\left(\beta_{T,y}(t)\right)\cos\left(\beta_{T,z}(t)\right) \\ \cos\left(\beta_{T,y}(t)\right)\sin\left(\beta_{T,z}(t)\right) \\ -\sin\left(\beta_{T,y}(t)\right) \end{bmatrix} \\ \mathbf{M}_{TVC,B}(t) &= \left[\mathbf{x}_{PVP} - \mathbf{x}_{CG}(t)\right] \times \mathbf{F}_{TVC,B}(t) \end{aligned} \qquad (4)$$

with $\mathbf{x}_{PVP} = \begin{bmatrix} x_{PVP} & 0 & 0 \end{bmatrix}^T$ the TVC pivot position and $\mathbf{x}_{CG}(t) = \begin{bmatrix} x_{CG}(t) & 0 & 0 \end{bmatrix}^T$ the center of gravity (CG).

2.2. Recovery G&C techniques

Considering the recovery of the vehicle, a thrust vector is commanded by the guidance subsystem and then

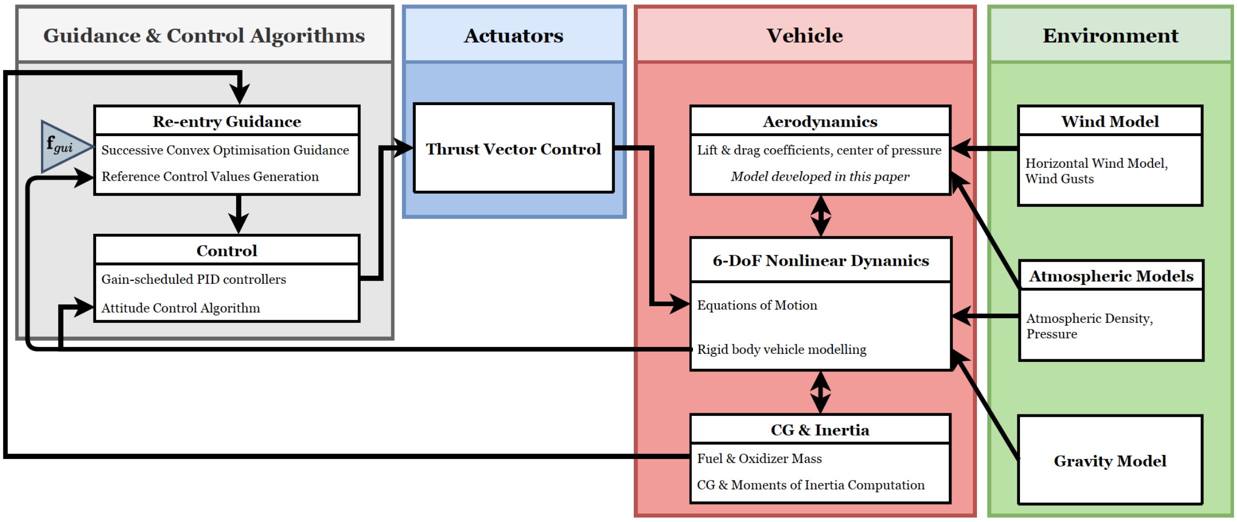

Guidance & Control Algorithms
$\mathbf{f}_{gui}$
Re-entry Guidance
Successive Convex Optimisation Guidance
Reference Control Values Generation
Control
Gain-scheduled PID controllers
Attitude Control Algorithm
Actuators
Thrust Vector Control
Vehicle
Aerodynamics
Lift & drag coefficients, center of pressure
Model developed in this paper
6-DoF Nonlinear Dynamics
Equations of Motion
Rigid body vehicle modelling
CG & Inertia
Fuel & Oxidizer Mass
CG & Moments of Inertia Computation
Environment
Wind Model
Horizontal Wind Model,
Wind Gusts
Atmospheric Models
Atmospheric Density,
Pressure
Gravity Model

Fig. 1: 6-DoF RLV re-entry dynamics simulator description. The development of the Aerodynamics block will be described in this paper

converted to reference pitch $\theta_{ref}(t)$ and yaw $\psi_{ref}(t)$ angles and thrust magnitude $T(t)$. Then, the control subsystem generates the necessary commands to correct the deviations between the reference and the actual attitude angles in terms of gimbal deflection angles $\beta_{T,y}(t)$ and $\beta_{T,z}(t)$ to be used by the TVC system.

The guidance subsystem relies on a successive convex optimisation algorithm which consists in iteratively solving convex optimisation SOCP subproblems in which the non-convex dynamics and constraints are repeatedly linearised using information coming from the previous iteration solution. This algorithm has been implemented in MATLAB using the CVX library [14] to formulate the convex problems and the ECOS routine [15] to solve them. At each simulation instance defined by the simulation rate, the reference thrust profile $\mathbf{T}(t)$ and the reference attitude angles $\{\theta_{ref}(t), \psi_{ref}(t)\}$ are computed from the most recent guidance solution via linear interpolation. Indeed, that solution is stored as an online lookup table which is updated at each guidance step, with the guidance update frequency $f_{gui} = 0.1$ Hz, so every 10 seconds.

The strength of this technique lies in the ability to define state and control constraints in addition to the re-entry dynamics, as well as to approximate the remaining nonlinearities, such as the aerodynamic effects, thanks to the solving of successive sub-problems using the information coming from the previous iteration solution. Therefore, a *spherical* aerodynamic model is used to describe the aerodynamics of the vehicle. This model, introduced by Szmuk *et al.* in Ref. [5], approximates the relationship between the aerodynamic force and the velocity vector and has the advantage to be easily implementable in the successive convex optimisation guidance method. The aerodynamic force $\mathbf{F}_{aero,I}(t)$ is considered always anti-parallel with respect to the velocity $\mathbf{v}_I(t)$ as if the vehicle was subjected to a pure drag force. Assuming that the rocket is axisymmetric, the aerodynamic forces and moments in the vehicle's body-fixed reference frame are expressed by:

$$\begin{aligned} \mathbf{F}_{aero,B}(t) &= -\frac{1}{2}\rho(t)||\mathbf{v}_I(t)||_2 S_a C_{aero} \mathbf{R}^I_B(t)\mathbf{v}_I(t) \\ \mathbf{M}_{aero,B}(t) &= [\mathbf{x}_{CP} - \mathbf{x}_{CG}(t)] \times \mathbf{F}_{aero,B}(t) \end{aligned} \qquad (5)$$

where S_a is the reference aerodynamic area of the vehicle, and $\mathbf{x}_{CP} = \begin{bmatrix} x_{CP} & 0 & 0 \end{bmatrix}^T$ the center of pressure (CP). C_{aero} is the aerodynamic coefficient matrix defined by $C_{aero} = \text{diag}\left(\begin{bmatrix} c_{a,x} & c_{a,x} & c_{a,x} \end{bmatrix}\right)$ where $c_{a,x}$ is a positive scalar. Finally, $\rho(t)$ is the air atmospheric density.

Then, the nonlinear equations of motion, define in Eqs. (1-3), are discretised and linearised about the solution of the previous iteration, through a first-order Taylor approximation and the non-convex constraints are convexified using the same linearisation method. Finally, the objective of this optimal control problem is to find the optimal trajectory, subjected to the defined re-entry dynamics and state and control constraints, minimising the vehicle's fuel consumption. Due to the monotonic behaviour of the propellant consumption with respect to time, the time of flight is selected as the value to be minimised.

The control subsystem relies on the use of feedback control through gain-scheduled PID controllers. In order to tune the gains, the problem is separated into two 3-DoF subproblems. One is characterised by the motion in the $\mathbf{x}_B\mathbf{z}_B$-plane with the controller on the pitch angle $\theta(t)$ through the deflection angle $\beta_{T,y}(t)$. The second problem is characterised by the motion in the $\mathbf{x}_B\mathbf{y}_B$-plane with the controller on the yaw angle $\psi(t)$ through the deflection angle $\beta_{T,z}(t)$. Then, using a reference trajectory pre-computed offline corresponding to the conditions of the problem, the system of equations is linearised and evaluated at the points of the reference trajectory.

Finally, due to the time-varying nature of the problem, a single PID controller maybe unable to stabilise the system for the whole trajectory. Therefore, the time of flight has been discretised into 25 slots where the linearisation has been performed. In this way, the problem is divided into regions where it is possible to analyse if the controller is able to stabilise the system. Thanks to that, the controllers can be considered as gain-scheduled PID as the gains can be changed in order to have the desired levels of performance in all the regions.

More details about the G&C algorithms definition are given in Ref. [10].

3. Developing the aerodynamic model for RLV re-entry

The aim of this aerodynamic model is to generate the aerodynamic coefficients and the center of pressure of an user-defined rocket as function of the flight parameters such as the aerodynamic angles, angle of attack $\alpha(t)$ and sideslip angle $\beta(t)$, and the Mach number. These parameters depend on the structure of the rocket and on the flow characteristics. They are required in order to compute the aerodynamic forces and moments generated by the vehicle.

The Supersonic/Hypersonic Arbitrary-Body Program (S/HABP) developed in 1973 by the United States Air Force Flight Dynamics Laboratory [16], has been adapted and used to build the aerodynamic database. This program, which has been also used by the National Aeronautics and Space Administration (NASA), is written historically in FORTRAN but then adapted in modern Fortran in Ref. [17].

The aerodynamic characteristics of a reusable rockets are given in Sec. 3.1. Then, the development of the aerodynamic database is explained in Sec. 3.2 before to be verified and validated through available literature with the quantification of the uncertainties in Sec. 3.3.

3.1. Aerodynamic characteristics

Aerodynamic forces and moments generated by the vehicle depend on its structure, as well as on the instantaneous dynamic pressure. This atmospheric parameter is usually given by:

$$Q(t) = \frac{1}{2}\rho(t)V^2(t) \tag{6}$$

where $V(t) = ||\mathbf{v}_{air,I}(t)||$ and $\mathbf{v}_{air,I}(t) = \mathbf{v}_I(t) - \mathbf{w}_I(t)$ is the air-relative velocity vector written in the inertial reference frame which account for the wind $\mathbf{w}_I(t)$.

Then, for the computation of the aerodynamic loads, it is common to define a velocity reference frame which is fixed to the vehicle's CG but directed along the air-relative velocity written in the body-fixed reference frame $\mathbf{v}_{air,B}(t)$. This reference frame enables the definition of the two aerodynamic angles, the angle of attack $\alpha(t)$ and the sideslip angle $\beta(t)$, in order to illustrate the rotation from the body-fixed to the velocity reference frame $\mathbf{R}_V^B(t)$, as follows:

$$\mathbf{R}_V^B = \begin{bmatrix} \cos\alpha\cos\beta & \sin\beta & \sin\alpha\cos\beta \\ -\cos\alpha\sin\beta & \cos\beta & -\sin\alpha\sin\beta \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix} \tag{7}$$

where the aerodynamic angles are given by:

$$\begin{aligned} \alpha(t) &= \text{atan2}(v_{air,B,z}(t), v_{air,B,x}(t)) \\ \beta(t) &= \arcsin\left(\frac{v_{air,B,y}(t)}{||\mathbf{v}_{air,B}(t)||}\right) \end{aligned} \tag{8}$$

With these definitions and assuming that the vehicle has an axisymmetric shape, the aerodynamic forces and moments generated by the vehicle are expressed in the body-fixed reference frame as:

$$\begin{aligned} \mathbf{F}_{aero,B}(t) &= -Q(t)S_a\mathbf{R}_B^V(t)\begin{bmatrix} C_D(\alpha_{eff}(t), M(t)) \\ 0 \\ C_L(\alpha_{eff}(t), M(t)) \end{bmatrix} \\ \mathbf{M}_{aero,B}(t) &= [\mathbf{x}_{CP} - \mathbf{x}_{CG}(t)] \times \mathbf{F}_{aero,B}(t) \end{aligned} \tag{9}$$

where $\{C_D, C_L\}$ are the drag and lift coefficients, respectively. Coming from the aerodynamic database, these coefficients are estimated as functions of the effective angle of attack $\alpha_{eff}(t) = \sqrt{\alpha^2(t) + \beta^2(t)}$ and the Mach number $M(t) = V(t)/a(t)$ where $a(t)$ is the speed of sound.

3.2. Aerodynamic database

As mentioned previously, S/HABP is used to estimate the aerodynamic coefficients as function of the aerodynamic angles and the Mach number. This program is designed to operate from about Mach 2 on up into the hypersonic range [18]. However, for RLV re-entry and particularly for this study, the range of Mach number starts around Mach 5 and then goes down below Mach 1 until zero velocity at landing. Moreover, the aerodynamic coefficients are assumed to be independent of the level of the thrust. This approximation is very rough for retro-propulsive flight where there are significant interactions between the exhaust plume of the engine and the oncoming flow which really impact the drag coefficient and the heat loads [19]. Therefore, the approximations obtained for the aerodynamic coefficients might diverge from true values. However, the goal of this work is not to provide with high-fidelity models but to study the interactions and challenges that exist in the design of a RLV re-entry dynamics simulator and to assess the advanced and robust G&C methods that must be developed accordingly.

S/HABP uses a specific geometry method to achieve the three-dimensional meshing of the user-defined vehicle named the surface element or quadrilateral method [18]. In order to use it, the update version of S/HABP written in modern Fortran and available in Ref. [17] needs the geometry of the studied vehicle to be written in the format Langley Wireframe Geometry Standard (LaWGS) [20]. This format, developed by the NASA in 1983, uses points and lines to define the wireframe geometry of the vehicle. These elements are then used by the S/HABP routine by sets of four points which form a plane-quadrilateral element in which the approximations are carried out. An illustration of this method is shown in Fig. 2.

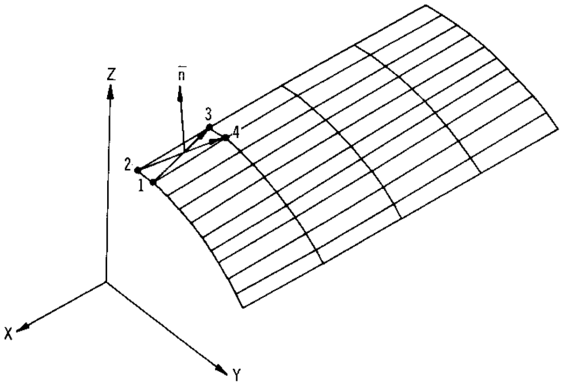


Fig. 2: Surface element geometry method description [18]

For this study, a VTVL first-stage RLV configuration, representative of the first stage of the Europe's lightweight VEGA launcher, was considered. Basically, without accounting for the main engine, it considers a 12-meter height and 3-meter diameter cylinder [13]. The geometry of this vehicle was generated with the software GMSH [21] which allows to export the relevant points in a `.msh` file which can be translated in a LaWGS `.wgs` file thanks to a Python script developed by the author. Fig. 3 shows the geometry representation of the first-stage rocket in GMSH before to be translated in LaWGS format. Note that to be used in S/HABP, the vehicle must be positioned with its nose at the coordinate system origin and with the length of the body stretching in the negative **X**-direction.

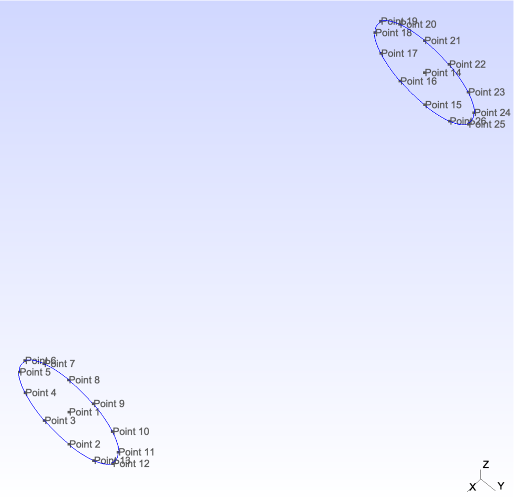


Fig. 3: First-stage rocket geometry representation in GMSH

Once the geometry has been defined and meshed by the S/HABP routine, the program offers several methods to calculate the pressure coefficient which is a crucial parameter to compute the aerodynamic forces and moments generated by the vehicle. For each method, the only one geometric parameter required is the so-called *impact angle* of the element and is written δ. This parameter enables to understand if the element is facing the flow, therefore in an impact region, or facing away from the flow, which means being in a shadow region. It is defined as the angle that the element makes with the freestream velocity vector U_∞. Therefore, this angle varies as functions of the vehicle aerodynamic angles. The impact angle δ is obtained from the following relationship:

$$\begin{aligned} \delta &= \frac{\pi}{2} - \phi \\ \cos\phi &= \frac{\mathbf{n}\cdot\mathbf{V}}{||\mathbf{n}||\ ||\mathbf{V}||} \end{aligned} \tag{10}$$

where **n** is the unit normal outward from the surface of the element and **V** is the local velocity vector, whose the explanations on how to obtain them are written in [18].

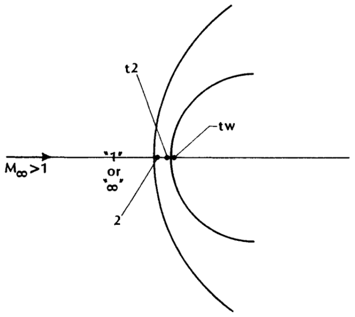
From the pressure calculation methods available in S/HABP, the modified Newtonian method is used. As defined in Ref. [18], the modified Newtonian method is widely used for blunt shapes at high hypersonic speed. The usual form of the Newtonian pressure coefficient is:

$$C_p = K \sin^2 \delta \tag{11}$$

where, usually, the parameter K is taken equal to 2 in impact regions and equal to zero in shadow regions.

However, in the modified Newtonian theory, K is taken as being equal to the stagnation pressure coefficient $c_{p,stag}$. This coefficient is obtained using shock wave theory [22]. Indeed, near the stagnation point, the flow passes through the normal portion of the shock wave reaching state 2 and then decelerates isentropically to state t_2, which constitutes the outer edge condition for the thermal boundary layer at the stagnation point. This is illustrated in Figure 4.

Considering a thermally and calorically perfect gas, the ratio of the values of flow properties across the shock



1 or ∞	denotes the free-stream conditions.
2	denotes the conditions immediately downstream of the shock wave.
t2	denotes the conditions at the stagnation point (downstream of the normal portion of the shock wave) but outside of the boundary layer.
tw	denotes conditions at the wall(the inner edge of the boundary layer at the downstream stagnation point).

Fig. 4: Nomenclature for the stagnation region [22]

wave can be written as a unique function of the freestream Mach number M_∞, and the ratio of specific heat γ, equal to 1.4 for a perfect gas. These are the Rankine-Hugoniot equations given by:

$$\begin{aligned}\frac{p_{t_2}}{p_\infty} &= \frac{2\gamma M_\infty^2 - (\gamma - 1)}{\gamma + 1}\\ \frac{\rho_2}{\rho_\infty} &= \frac{U_\infty}{U_2} = \frac{(\gamma + 1)M_\infty^2}{(\gamma - 1)M_\infty^2 + 2}\\ \frac{T_2}{T_\infty} &= \frac{[2\gamma M_\infty^2 - (\gamma - 1][(\gamma - 1)M_\infty^2 + 2]}{(\gamma + 1)^2 M_\infty^2}\end{aligned} \tag{12}$$

Then, with the assumption that the flow decelerates isentropically from the conditions at point 2 (immediately downstream of the normal portion of the shock wave) to the stagnation point outside of the thermal boundary layer, at point t_2, the pression at the stagnation point is obtained as follows:

$$p_{t_2} = p_\infty \left[\frac{(\gamma + 1)M_\infty^2}{2}\right]^{\left(\frac{\gamma}{\gamma - 1}\right)} \left[\frac{\gamma + 1}{2\gamma M_\infty^2 - (\gamma - 1)}\right]^{\left(\frac{1}{\gamma - 1}\right)} \tag{13}$$

and the stagnation pressure coefficient is therefore equal to:

$$c_{p,stag} = \frac{p_{t_2} - p_\infty}{\frac{1}{2}\rho_\infty U_\infty^2} \tag{14}$$

Finally, Eq. (11) is updated to:

$$C_p = c_{p,stag} \sin^2 \delta \tag{15}$$

Note that the calculation of the pressure stagnation coefficient, previously defined, has been added in the S/HABP routine to be automatically achieve at the beginning of the simulation according to the Mach number defined as input by the user. Moreover, as mentioned previously, when the impact angle is negative, which means that the element is in a shadow region, the pressure stagnation coefficient is set equal to zero.

With the pressure force coefficient, S/HABP is then able to calculate the three force coefficients, axial ΔC_A, side ΔC_Y and normal ΔC_N, and the three moment coefficients, rolling ΔC_l, pitching ΔC_m and yawing ΔC_n, for each quadrilateral as follows:

$$\begin{aligned}\Delta C_A &= C_p n_x \frac{\Delta A}{S_a}\\ \Delta C_Y &= C_p n_y \frac{\Delta A}{S_a}\\ \Delta C_N &= -C_p n_z \frac{\Delta A}{S_a}\\ \Delta C_l &= \Delta C_Y \frac{z}{b} + \Delta C_N \frac{y}{b}\\ \Delta C_m &= \Delta C_N \frac{x}{c} + \Delta C_A \frac{z}{c}\\ \Delta C_n &= \Delta C_Y \frac{x}{b} - \Delta C_A \frac{y}{b}\end{aligned} \tag{16}$$

where x, y and z are the distances from the CG, n_x, n_y and n_z are the components of the surface normal vector of the element **n** and ΔA is the element area, all calculated by the program, while S_a, the reference area of the vehicle, b the reference span and c the mean aerodynamic chord are inputs given by the user. Note that the minus signs are required because of the sign convention used in the program [18].

The total aerodynamic coefficients are then obtained by summing the contributions of all the elements:

$$C_\# = \sum \Delta C_\#, \quad \# = \{A, Y, N, l, m, n\} \tag{17}$$

Finally, these coefficients are converted to drag C_D and lift C_L coefficients using the rotation matrix defined in Eq. (7), as follows:

$$\begin{aligned}C_D &= C_A \cos\alpha \cos\beta - C_Y \sin\beta + C_N \sin\alpha \cos\beta\\ C_L &= -C_A \sin\alpha + C_N \cos\alpha\end{aligned} \tag{18}$$

From the coefficients obtained, it is then possible to approximate the center of pressure position x_{CP} by:

$$x_{CP} = -\frac{c \cdot C_m}{C_N} + x_{CG} \tag{19}$$

Note that x_{CP} is not defined for α equal to 0 and 180 deg where C_N is equal to 0; hence its value must be linearly extrapolated from the adjacent interval.

Therefore, to obtain the aerodynamic database, several simulations have been carried out using the first-stage vehicle configuration described previously for a Mach number going from Mach 0.8 to 5 and then for angles of attack ranging from 0 to 180 deg. Tab. 1 summarised the constant values used as inputs for the simulations. Note that for simplicity, only two values of

Tab. 1: Input values for the aerodynamic database generation

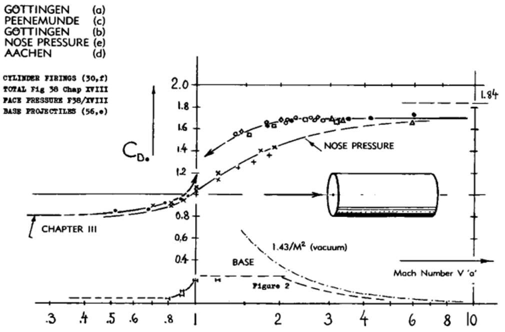
Parameter	Value
M	$\{0.8, 1, 1.2, 1.6, 2, 2.5, 3, 3.5, 4, 4.5, 5\}$
α	$\{0, ..., 180\}$ deg
S_a	7.0685 m^2
c	3 m
b	3 m
x_{CG}	$\{4.5990, 6.38697\}$ m

x_{CG} have been considered for the calculation of the x_{CP} which corresponds to the maximum and the minimum ones obtained during the nominal simulation of the RLV re-entry. Finally, the variation of C_D, C_L and x_{CP} with respect to α and M is illustrated in Fig. 5.

3.3. Verification & Validation and quantification of the uncertainties

The modified Newtonian method was first validated in comparing the obtained drag coefficients in transonic and supersonic axial flow, meaning at 0 deg angle of attack. For that, the geometry of a 1-meter height, 1-meter diameter cylinder was uploaded in S/HABP and calculations were performed at 0 deg angle of attack and at Mach 2 and Mach 4. The results obtained for the drag coefficient are displayed in Tab. 2. These values are similar to the drag characteristic of blunt cylinders in axial flow depicted in Ref. [23] and reproduced in the table.

Tab. 2: Drag coefficient in axial flow for $M = \{2, 4\}$ and comparison with literature

	Mach 2	Mach 4
$C_D(\alpha = 0)$ from S/HABP:	1.61957	1.75100
$C_D(\alpha = 0)$ from Ref. [23]:		
		

Then, the aerodynamic database is compared with another database used in Ref. [13] for a similar first-stage vehicle configuration. Fig. 6 shows the differences in the variation of the aerodynamic coefficients with respect to the angle of attack and the Mach number for both aerodynamic databases.

From this figure, it is possible to notice that the drag and lift aerodynamic coefficients computed by S/HABP for the first-stage vehicle configuration are slightly overestimated compared to the ones obtained in Ref. [13]. Moreover, the obtained values show a linear relationship with respect to the Mach number: the higher the Mach number, the higher the drag coefficient. The same trend exits in absolute value for the lift coefficient. However, that relationship is not verified in the aerodynamic database found in the literature. These deviations are probably due to the pressure calculation method used in S/HABP as well as the geometric representation of the first-stage vehicle, approximated as a cylinder. In what concerns the center of pressure, the computed model does not show significant variations with respect to the Mach number and the angle of attack which is not representative of the reality but manages to give a mean value close to what is obtained in the literature. For more accurate values, computational fluid dynamics analyses should be carried out. It is important to recall that S/HABP is designed to operate from about Mach 2, which can explain the increase of deviations when the Mach number is decreasing.

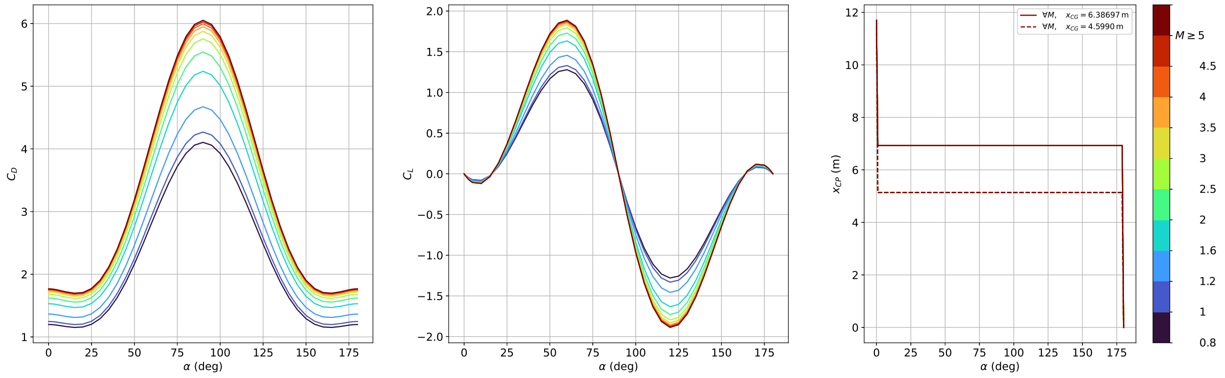
Finally, the calculation methods used in S/HABP to build the aerodynamic database have been validated and verified through a full vehicle configuration. Therefore, a geometry of the full-stage VEGA launcher has been achieved with GMSH and converted in LaWGS format. This analysis considers the aerodynamic coefficients at 2 deg and 5 deg angle of attack for different Mach number. The results have been compared with data found in literature in Refs. [13, 24]. The outcomes of this comparison are illustrated in Fig. 7.

From that analysis, it is again possible to notice that the coefficients computed by S/HABP present a slight deviation compared to the values available in the literature, and more particularly at low Mach number. Indeed, the increase in the drag coefficient around Mach 1-2 is not shown by the outputs of the program. Finally, the center of pressure still does not vary with respect to the Mach number, however, the references found in the literature show different results for this parameter which does not allow to draw relevant conclusions.

From these analyses, some uncertainties have been considered for the designed aerodynamic database and are summarized in Tab. 3.

Tab. 3: Aerodynamic model uncertainties

Aerodynamic parameter	Uncertainties (%)
$C_D(\alpha, M)$	20 %
$C_L(\alpha, M)$	20 %
$x_{CP}(\alpha, M)$	10 %

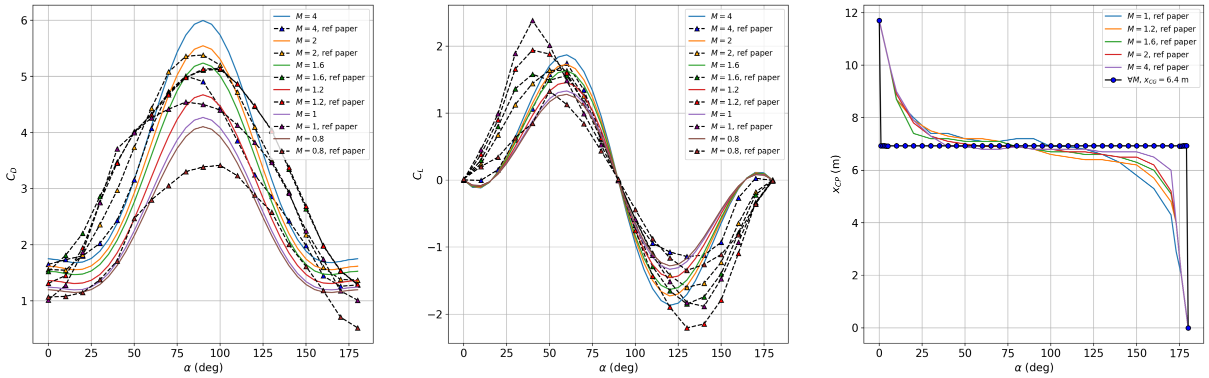


(a) C_D

(b) C_L

(c) x_{CP}

Fig. 5: Aerodynamic coefficients database. Note that the values of x_{CP} are found independent from the Mach number M.

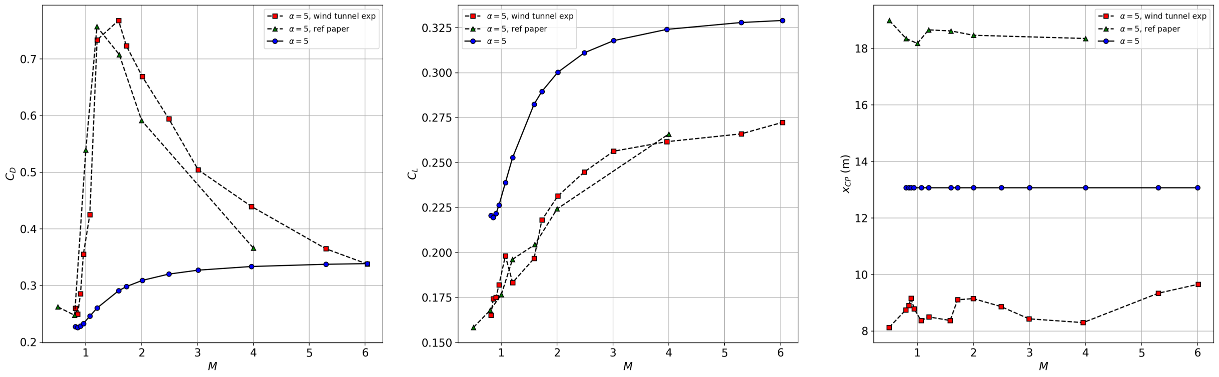


(a) C_D

(b) C_L

(c) x_{CP}

Fig. 6: Aerodynamic coefficients database comparison for a first-stage vehicle configuration. The reference paper is to Ref. [13]



(a) C_D

(b) C_L

(c) x_{CP}

Fig. 7: Aerodynamic coefficients database comparison for a full vehicle configuration. The reference paper corresponds to Ref. [13] while the results of the wind tunnel experiments are taken from Ref. [24]

4. Simulation Results

This section illustrates the results obtained with the 6-DoF RLV re-entry dynamics simulator in which the aerodynamic database developed in the previous section has been added as a look-up table, from where the aerodynamic coefficients are interpolated as functions of the current Mach number $M(t)$ and the effective angle of attack $\alpha_{eff}(t)$.

The initial and final conditions adopted in the simulator are fixed for all the simulations to maintain consistency between them. The starting altitude is at 25 km with a downrange in the east position of 20 km. The first-stage rocket is considered with an initial mass of 12,000 kg and a dry mass m_{dry} of 2,750 kg, uploaded as a minimum mass constraint for the guidance algorithm. Note that the final time of flight t_f is considered as a free parameter to be optimized. Tab. 4 summarised the initial and final conditions considered in the simulator.

Tab. 4: Initial and final conditions

(a) Initial conditions

Parameter	Value	Unit
$\mathbf{r}_I(0)$	$\begin{bmatrix}25,000 & -20,000 & 0\end{bmatrix}^T$	m
$\mathbf{v}_I(0)$	$\begin{bmatrix}-850 & 1,200 & 0\end{bmatrix}^T$	m/s
$\omega_B(0)$	$\begin{bmatrix}0 & 0 & 0\end{bmatrix}^T$	rad/s
$m(0)$	$12,000$	kg

(b) Final conditions

Parameter	Value	Unit
$\mathbf{r}_I(t_f)$	$\begin{bmatrix}0 & 0 & 0\end{bmatrix}^T$	m
$\mathbf{v}_I(t_f)$	$\begin{bmatrix}-5 & 0 & 0\end{bmatrix}^T$	m/s
$\omega_B(t_f)$	$\begin{bmatrix}0 & 0 & 0\end{bmatrix}^T$	rad/s
$\mathbf{q}_B^I(t_f)$	$\begin{bmatrix}0 & 0 & 0 & 1\end{bmatrix}^T$	-

Then, Tab. 5 depicted the physical parameters of the RLV first-stage configuration studied.

Tab. 5: Physical parameters of the RLV

Parameter	Value	Unit
I_{sp}	282	s
g_0	9.81	m/s^2
x_{PVP}	0.96	m
m_{dry}	2750	kg
S_a	7.14	m^2
A_{nozzle}	3.1416	m^2

For the first analysis, three cases are compared. The first case is considered as the *perfect aerodynamics* case; the guidance algorithm inputs the constant aerodynamic coefficient matrix $C_{aero} = \text{diag}\left(\begin{bmatrix}0.82 & 0.82 & 0.82\end{bmatrix}\right)$ during the flight while the aerodynamic coefficients are not taken from any aerodynamic database but equal during all the flight to $C_D = 0.82$ for the drag coefficient, and $C_L = 0$ for the lift coefficient. It is equivalent to consider an atmospheric re-entry with only drag (no lift) where the guidance algorithm predicts exactly the actual aerodynamic conditions.

The second case, called *reference aerodynamics*, uses the aerodynamic database available in Ref. [13]. In what concerns the guidance algorithm, the aerodynamic coefficient matrix is updated at different step of the atmospheric re-entry simulation. When the Mach number is greater than 1, the following matrix is considered $C_{aero} = \text{diag}\left(\begin{bmatrix}1 & 1 & 1\end{bmatrix}\right)$ while otherwise, the matrix is taken equal to $C_{aero} = \text{diag}\left(\begin{bmatrix}0.82 & 0.82 & 0.82\end{bmatrix}\right)$. Note that this choice has been made after comparing different configurations, as well as the one chosen by Ref. [8] where the aerodynamic coefficient matrix was taken with $c_{a,x} = C_D|_{\alpha=\pi}$, and that the case presented is the one presenting the best results.

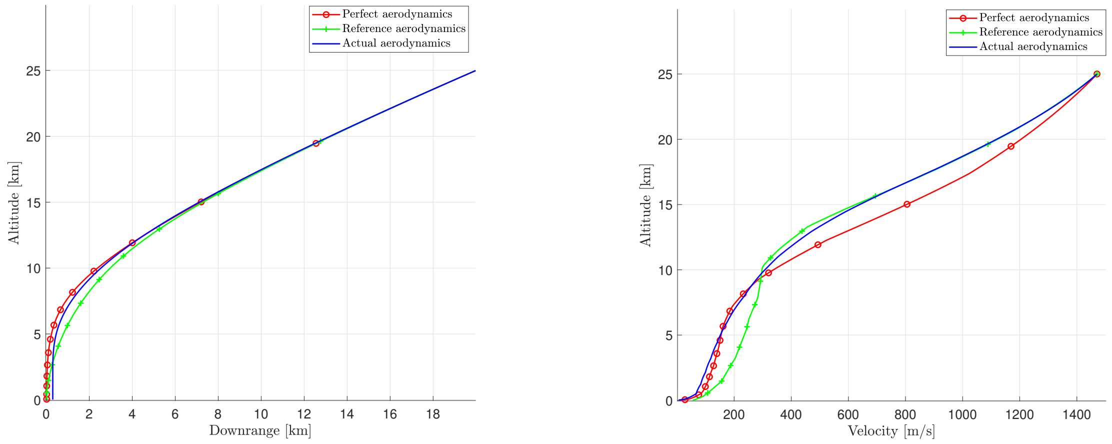
Finally, the third case uses the aerodynamic database described in this paper. Following the same model than the previous case, the guidance algorithm uses as input an aerodynamic coefficient matrix equal to $C_{aero} = \text{diag}\left(\begin{bmatrix}2 & 2 & 2\end{bmatrix}\right)$ when the Mach number is greater than 1, and equal to $C_{aero} = \text{diag}\left(\begin{bmatrix}1.7 & 1.7 & 1.7\end{bmatrix}\right)$ otherwise. Again, several aerodynamic coefficient matrices have been tested and the one presented here shows the best results.

The three cases studied are summarised in Tab. 6. For all of them, the center of pressure is taken as constant and equal to $x_{CP} = 6.9$, which is a relevant value regarding both aerodynamic database. Morevover, it is important to recall that the successive convex optimisation algorithm is solved with the ECOS routine. It has been noticed that the SDPT3 solver, also available in CVX, showed better performance than ECOS, when converging to a solution; indeed, the downrange error of the *perfect aerodynamic* case was smaller with SDPT3 than ECOS. However, it has also been noticed that for complex cases, as for example with varying aerodynamic coefficients, ECOS was more often managing to converge towards a feasible solution and was therefore chosen as the solver. A thorough assessment of the difference between these two solvers leading to these deviations, and the understanding on the way to obtain the best results with each of them would constitute another study, not addressed in this paper.

Fig. 8 shows the performance of each case for what regards downrange error (Fig. 8a) and final velocity error (Fig. 8b). For all cases, a successful recovery is achieved. It is possible to notice that changes in the aerodynamic model really affect the trajectory of the vehicle; therefore, it confirms that they cannot be neglected in

Tab. 6: Simulation cases description

	Perfect aerodynamics	*Reference aerodynamics*	*Actual aerodynamics*
Aerodynamic model	$C_D = 0.82, \quad C_L = 0$	from Ref. [13]	from Sec. 3.2
$c_{a,x}$	0.82	$\begin{cases} 1, & \text{if } M(t) > 1 \\ 0.82, & \text{otherwise} \end{cases}$	$\begin{cases} 2, & \text{if } M(t) > 1 \\ 1.7, & \text{otherwise} \end{cases}$
x_{CP}	6.9 m	6.9 m	6.9 m



(a) Altitude vs. downrange

(b) Altitude vs. velocity

Fig. 8: Comparison of the performance for the three cases

the G&C algorithms. With both aerodynamic databases used in cases *reference aerodynamics* and *actual aerodynamics*, the guidance algorithm managed to compute accordingly the reference trajectory to be followed. However, even if the recovery seems successful, the simulation for these both cases leads to downrange and final velocity errors. Indeed, while the downrange error is negligible, the *reference aerodynamics* case leads to a relatively high final velocity, around 50 m/s, which can be sufficient to damage the vehicle at landing in real scenario. For the *actual aerodynamics* case, the final velocity corresponds to the desired one, however, the downrange error is about 300 m. It is more likely due to the control method used which is not robust enough as the PID gains have been tuned using a reference trajectory computed offline. But it can also be because of a lack of control authority which could be corrected in adding another actuation system in the simulator as planar fins, particularly efficient for counteracting aerodynamics. Finally, it can also be due to the aerodynamic model used in the successive convex optimisation algorithm, expressed by C_{aero}, which may not model accurately the actual aerodynamic loads.

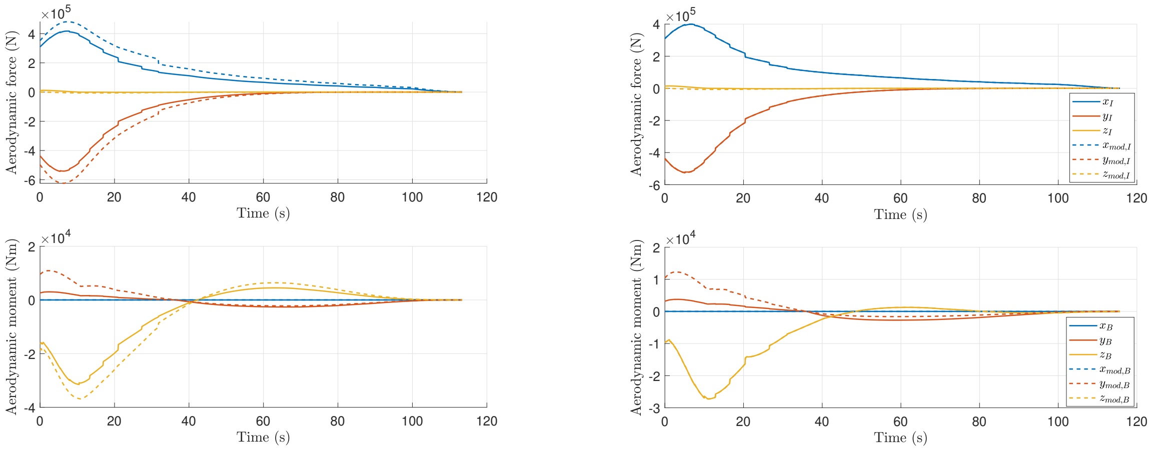
Therefore, the second analysis consists in looking for the best aerodynamic model in the guidance, meaning the best aerodynamic coefficient matrix C_{aero}. In that purpose, another simulation case has been run, called *adapted aerodynamics* and summarised in Tab. 7. In that case, C_{aero} is updated each time a new guidance is executed, so each time a new reference trajectory is computed. At each guidance step, $c_{a,x}$ is taken equal to the drag coefficient at 180 deg angle of attack for the actual Mach number, $C_D(M(t))|_{\alpha=\pi}$. This simulation has been compared to the *actual aerodynamics* case.

Tab. 7: *Adapted aerodynamics* case description

	Adapted aerodynamics
Aerodynamic model	from Sec. 3.2
$c_{a,x}$	$C_D(M(t))\|_{\alpha=\pi}$
x_{CP}	6.9 m

Fig. 9 shows the actual aerodynamic loads generated by the vehicle in both cases with also those modelled by the aerodynamic model implemented in the successive convex optimisation guidance algorithm. Fig. 10 presents the performance of both cases in terms of downrange error (see Fig. 10a) and final mass (see Fig. 10).

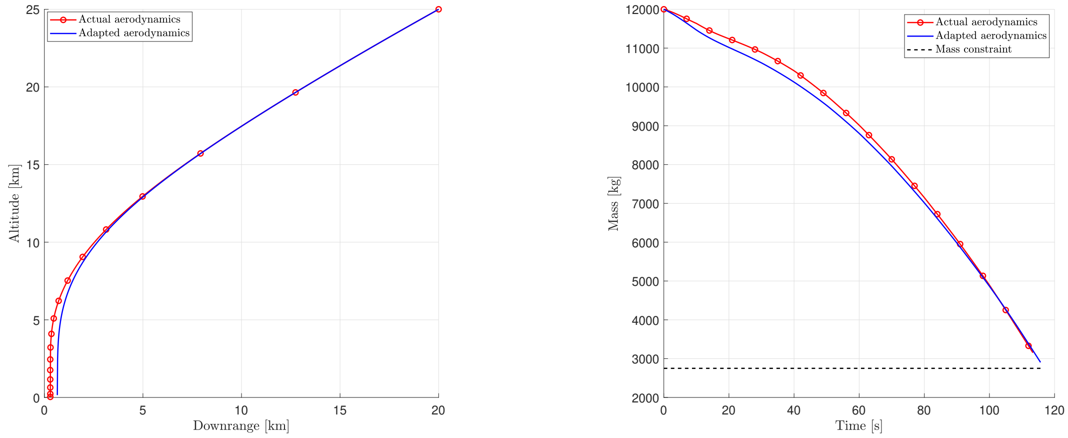
It is possible to notice that the aerodynamic forces modelled by the guidance in the *adapted aerodynamics* case, displayed in Fig. 9b, follow exactly the actual ones whereas they are slightly overestimated at the



(a) Actual aerodynamics

(b) Adapted aerodynamics

Fig. 9: Comparison of the aerodynamic loads generated by the vehicle. Dashed line corresponded to the aerodynamic loads modelled by the guidance



(a) Altitude vs. downrange

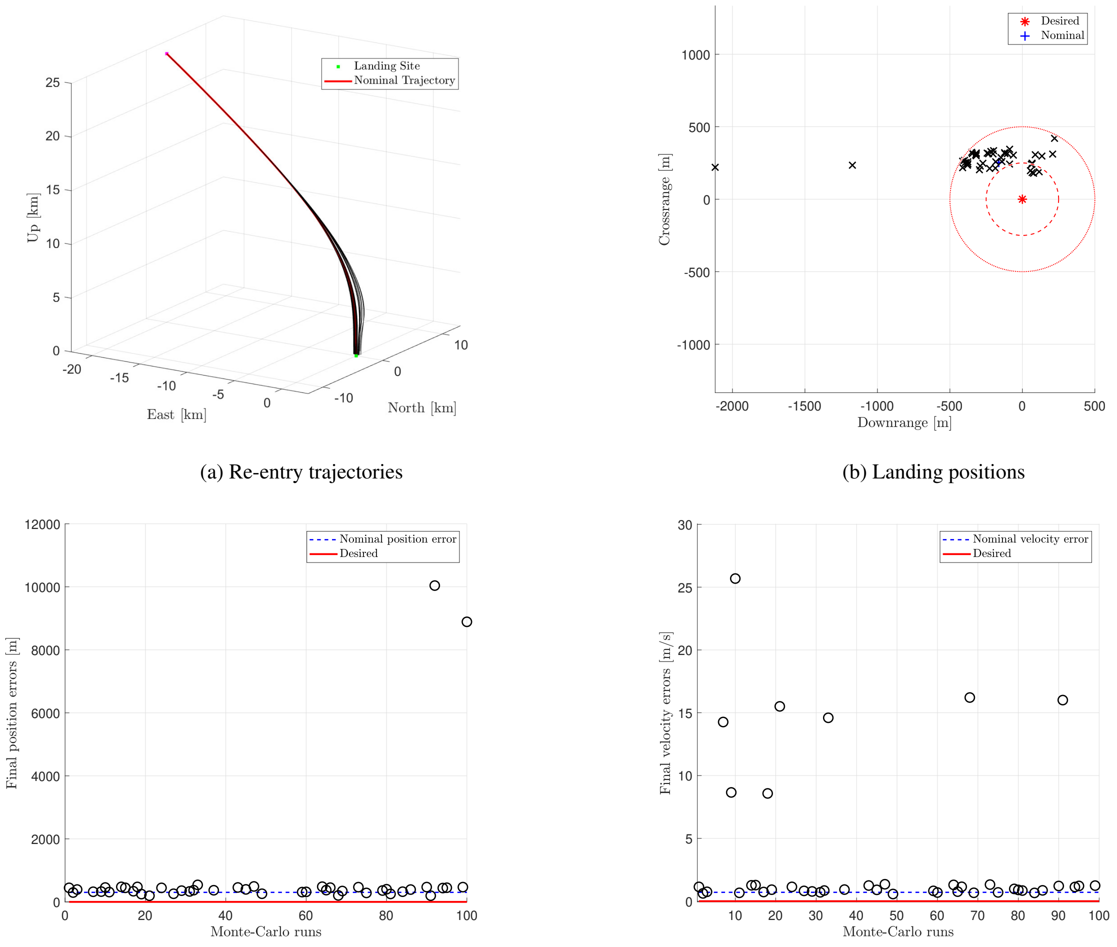
(b) Mass vs. time

Fig. 10: Comparison of the performance for the *actual aerodynamics* (red) and the *adapted aerodynamics* (blue) cases

end of the flight by the *actual aerodynamics* case, displayed in Fig. 9a. However, in terms of performance, the *actual aerodynamic* case leads to less downrange error and consumes less propellant. Even if the aerodynamic model implemented in the guidance is more accurate, the performance of the algorithm is not increased. Therefore, the computation of the aerodynamic coefficient matrix C_{aero} is not straightforward and would deserve a deeper analysis in which the sensitive parameters are highlighted and a trade-off between them is carried out in order to find the best solution to be implemented in the guidance system.

Finally, two 100-run Monte-Carlo analyses have been achieved in order to assess the robustness of the G&C algorithms towards possible uncertainties in the aerodynamic model. Following the conclusions of Sec. 3.3 with the uncertainties given for each parameter in Tab. 3, the first Monte-Carlo analysis considers dispersion up to 20% for the drag and lift coefficients $\{C_D, C_L\}$, interpolated in the aerodynamic database developed in this paper, while the second one considers dispersion up to 10% for the CP position, x_{CP}. Fig. 11 and Fig. 12 displayed the results of the 100-run Monte-Carlo analyses for the aerodynamic coefficients and the CP position, respectively.

These figures represent the different re-entry trajectories (Figs. 11a-12a), landing positions from the desired landing site with respect to a 500-m diameter outer circle and a 250-m inner circle (Figs. 11b-12b), final position errors (Figs. 11c-12c), and final velocity errors



(a) Re-entry trajectories

(b) Landing positions

(c) Final position errors

(d) Final velocity errors

Fig. 11: 100-run Monte-Carlo analysis results with aerodynamic coefficients dispersion. Nominal results are in blue

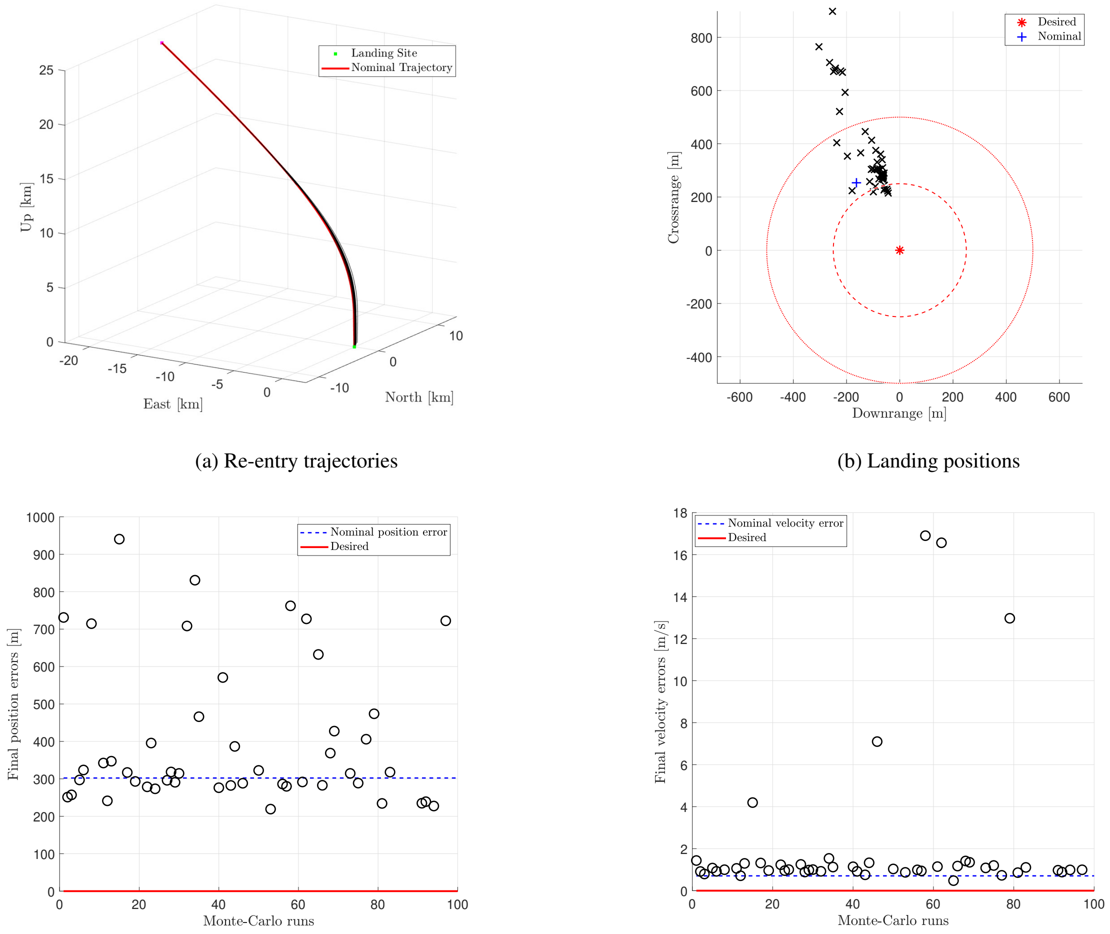
(Figs. 11d-12d). First, in what concerns the 100-run Monte-Carlo analysis with dispersion in the drag and lift coefficients, it is important to mention that only 45% of the simulations ended up in a successful recovery, indeed the others encountered a convergence issue during the simulation. This result highlights the difficulty to withstand aerodynamic loads during an Earth atmospheric re-entry and therefore the importance for a robust G%C system. At this stage of the study, the developed 6-DoF re-entry dynamics simulator is consequently not robust enough. However, for the cases which did not show a convergence issue, the obtained trajectories are almost similar and the landing conditions (position, velocity) do not show a large dispersion.

The same conclusions can be drawn for the second Monte-Carlo analysis which considers dispersion in the CP position. In that case, 48% of the simulations were successful and the final position and velocity dispersion is similar to what is obtained with the previous analysis.

5. Conclusions

This paper presents a RLV re-entry dynamics simulator with closed-loop G&C integration. It covers a VTVL first-stage booster atmospheric re-entry and soft pinpoint landing on Earth. In that purpose, the vehicle is equipped with a TVC system controlled by gain-scheduled PID controllers, which correct the trajectory deviations with respect to the reference profile generated by a successive convex optimisation guidance algorithm. To better assess the performance of the developed G&C algorithms for real-case scenario, an aerodynamic database with uncertainties has been developed and implemented in the simulator and the outputs are studied in this paper; from design, to validation and critical discussion on some representative simulation cases.

Numerical results show that the G&C methods implemented well manage to consider the aerodynamic effects and therefore to enable an atmospheric re-entry and



(a) Re-entry trajectories

(b) Landing positions

(c) Final position errors

(d) Final velocity errors

Fig. 12: 100-run Monte-Carlo analysis results with CP dispersion. Nominal results are in blue

pinpoint landing. However, adding uncertainties in the aerodynamic parameters points out that the robustness of the simulator must still be improved. Future extensions of this work will add the consideration of planar fins in the simulator and in the aerodynamic database to use them as an additional control authority to steer the spacecraft to the landing site with more robust control methods.

6. Acknowledgments

The author would like to acknowledge and give thanks for the support and help of Ysolde Prévereaud to develop the aerodynamic database during a 3-month visiting period at ONERA, the French Aerospace Lab in Toulouse, France. The project leading to this application has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 860956.

References

[1] Lars Blackmore. Autonomous precision landing of space rockets. *The Bridge on Frontiers of Engineering*, 4(46):15–20, 2016.

[2] Daniel P. Scharf, Behçet Açıkmeşe, Daniel Dueri, Joel Benito, and Jordi Casoliva. Implementation and experimental demonstration of onboard powered-descent guidance. *Journal of Guidance, Control, and Dynamics*, 40(2):213–229, feb 2017. doi:10.2514/1.g000399.

[3] Behçet Açıkmeşe and Scott R. Ploen. Convex programming approach to powered descent guidance for mars landing. *Journal of Guidance, Control, and Dynamics*, 30(5):1353–1366, sep 2007. doi:10.2514/1.27553.

[4] Behçet Açıkmeşe, J. Casoliva, J. M. Carson, and L. Blackmore. G-FOLD: A Real-Time Implementable Fuel Optimal Large Divert Guidance Algorithm for Planetary Pinpoint Landing. In *Con-*

cepts and Approaches for Mars Exploration, volume 1679 of *LPI Contributions*, page 4193, jun 2012.

[5] Michael Szmuk, Taylor P. Reynolds, and Behçet Açıkmeşe. Successive convexification for real-time six-degree-of-freedom powered descent guidance with state-triggered constraints. *Journal of Guidance, Control, and Dynamics*, 43(8):1399–1413, aug 2020. doi:10.2514/1.g004549.

[6] Marco Sagliano. Pseudospectral convex optimization for powered descent and landing. *Journal of Guidance, Control, and Dynamics*, 41(2):320–334, feb 2018. doi:10.2514/1.g002818.

[7] Xinfu Liu. Fuel-optimal rocket landing with aerodynamic controls. *Journal of Guidance, Control, and Dynamics*, 42(1):65–77, jan 2019. doi:10.2514/1.g003537.

[8] Pedro Simplício, Andrés Marcos, and Samir Bennani. Guidance of reusable launchers: Improving descent and landing performance. *Journal of Guidance, Control, and Dynamics*, 42(10):2206–2219, oct 2019. doi:10.2514/1.g004155.

[9] Marco Sagliano, Ansgar Heidecker, José Macés Hernández, Stefano Farì, Markus Schlotterer, Svenja Woicke, David Seelbinder, and Etienne Dumont. Onboard guidance for reusable rockets: Aerodynamic descent and powered landing. In *AIAA Scitech 2021 Forum*. American Institute of Aeronautics and Astronautics, jan 2021. doi:10.2514/6.2021-0862.

[10] Alice De Oliveira and Michèle Lavagna. Reusable launchers re-entry controlled dynamics simulator. Proceedings of the 9th European Conference for Aeronautics and Aerospace Sciences. Lille, France, 27th June - 1st July 2022, 2022.

[11] Jacopo Guadagnini, Michèle Lavagna, and Paulo Rosa. Model predictive control for reusable space launcher guidance improvement. *Acta Astronautica*, oct 2021. doi:10.1016/j.actaastro.2021.10.014.

[12] Alice De Oliveira and Michèle Lavagna. Reusable launch vehicles re-entry: Preliminary architecture towards optimal guidance and robust control. Proceedings of the XXVI International Congress of the Italian Association of Aeronautics and Astronautics (AIDAA). On-line event hosted by the Tuscany AIDAA Section in Pisa, 31 August - 3 September 2021, 2021.

[13] Pedro Simplício, Andrés Marcos, and Samir Bennani. Reusable launchers: Development of a coupled flight mechanics, guidance, and control benchmark. *Journal of Spacecraft and Rockets*, 57(1):74–89, jan 2020. doi:10.2514/1.a34429.

[14] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx, March 2014.

[15] Alexander Domahidi, Eric Chu, and Stephen Boyd. ECOS: An SOCP solver for embedded systems. In *2013 European Control Conference (ECC)*. IEEE, jul 2013. doi:10.23919/ecc.2013.6669541.

[16] Arvel E. Gentry, Douglas N. Smyth, and Wayne R. Oliver. The mark iv supersonic-hypersonic arbitrary-body program. volume i. user's manual. Technical Report AFFDL-TR-73-159, USAF Flight Dynamics Laboratory, November 1973.

[17] Ralph L. Carmichael. Public domain aeronautical software, 2021. URL: https://www.pdas.com/hyperdownload.html.

[18] Arvel E. Gentry, Douglas N. Smyth, and Wayne R. Oliver. The mark iv supersonic-hypersonic arbitrary-body program. volume ii - program formulation. Technical Report AFFDL-TR-73-159, USAF Flight Dynamics Laboratory, November 1973.

[19] Tobias Ecker, Sebastian Karl, Etienne Dumont, Sven Stappert, and Daniel Krause. A numerical study on the thermal loads during a supersonic rocket retro-propulsion maneuver. In *53rd AIAA/SAE/ASEE Joint Propulsion Conference*. American Institute of Aeronautics and Astronautics, jul 2017. doi:10.2514/6.2017-4878.

[20] C. B. Craidon. A description of the langley wireframe geometry standard (lawgs) format. Technical Memorandum NASA-TM-85767, NASA, February 1985.

[21] Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, may 2009. doi:10.1002/nme.2579.

[22] John J. Bertin. *Hypersonic aerothermodynamics*. AIAA Education Series. American Institute of Aeronautics and Astronautics, Washington, DC, 1994.

[23] Sighard F. Hoerner. *Fluid-Dynamic Drag. Practical Information on Aerodynamic Drag and Hydrodynamic Resistance*. 1965.

[24] Alessio Nicolì, Biagio Imperatore, Roberto Fauci, and Antonio Pizzicaroli. Wind tunnel test campaigns of the VEGA launcher. In *44th AIAA Aerospace Sciences Meeting and Exhibit*. American Institute of Aeronautics and Astronautics, jan 2006. doi:10.2514/6.2006-257.