Reachability analysis to design zero-wait entry guidance

Gonzalez-Puerta, Alejandro; Mooij, Erwin; Valles, Celia Yabar

DOI
10.2514/6.2018-1316

Publication date
2018

Document Version
Accepted author manuscript

Published in
AIAA Guidance, Navigation, and Control

Citation (APA)

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.
Reachability Analysis to Design Zero-Wait Entry Guidance
A. Gonzalez-Puerta and Mooij, E.
Delft University of Technology, Delft, The Netherlands
C. Yabar Valles
European Space Agency, ESTEC, Noordwijk, The Netherlands
Reachability Analysis to Design Zero-Wait Entry Guidance

A. Gonzalez-Puerta* and E. Mooij†

Delft University of Technology, Faculty of Aerospace Engineering,
Kluyverweg 1, 2629 HS Delft, The Netherlands

C. Yabar Valles‡

European Space Agency, ESTEC,
Keplerlaan 1, 2201 AZ Noordwijk, The Netherlands

This paper presents a novel reentry guidance architecture that aims to improve current mission safety by enabling zero-wait orbital aborts. To do so, an on-board trajectory planner based on Adaptive Multivariate Pseudospectral Interpolation (AMPI) is developed. This planner generates mathematical approximations of the reference trajectory to be tracked by comparing the vehicle-state information at the entry point against a pre-computed trajectory database that is stored on-board. Such a trajectory database is populated by systematically solving a number of optimal-control problems, which ultimately generate a set of reference angle-of-attack and bank-angle profiles that feed a Linear Quadratic Regulator trajectory tracker. The architecture proposed in this paper is tested using a scaled-down version of the X-38, SPHYNX, a small lifting-body reentry vehicle developed by the European Space Agency. The proposed on-board trajectory planner is able to generate reference commands in the order of a few milliseconds, thus proving the ability to run the system in current flight-hardware. Furthermore, a Monte Carlo test campaign showed that the trajectory database can be compressed with a loss-less process without having a detrimental impact on the landing dispersions and constraint compliance, achieving database sizes as low as 18 MB. Finally, the presented architecture allows for easy targeting of any landing site, as long as reference guidance commands are stored in the database.

I. Introduction

To ensure the survivability of a reentry vehicle and its payload, the changes in the vehicle’s altitude and velocity are carefully controlled during the descent using a guidance system. Such a system ensures that the planned nominal trajectory is followed closely and thus removes any deviations caused by external disturbances. Although reentry vehicles mostly fly a nominal trajectory, one should account for those trajectories that are required to be flown in case an unplanned event occurs. Such trajectories are referred to as abort trajectories and they must be part of the guidance design process.

Abort trajectories differ significantly from each other, where such differences are driven by the abort scenario that triggered them, the mission phase when the abort occurs and the capabilities of the vehicle flying the abort trajectory.1 According to Ref. 1, such abort scenarios include launcher structural failure, degraded launcher propulsion performance, emergency crew recovery from the International Space Station (ISS), or aborts due to primary landing site unavailability, amongst others. The guidance system discussed in this paper mainly considers orbital aborts, which are those scenarios that originate once the reentry vehicle

* MSc Graduate Student, Section Astrodynamics and Space Missions, agzpuerta@gmail.com.
† Assistant Professor, Section Astrodynamics and Space Missions, e.mooij@tudelft.nl, Associate Fellow AIAA.
‡ GNC Systems Engineer, Section GNC, AOCS and Pointing, celia.yabar.valles@esa.int.
is on a stable operational orbit. Orbital aborts begin with a de-orbit burn that puts the reentry vehicle in a
descent path towards the Entry Interface Point (EIP). Once atmospheric flight begins, the vehicle is guided
towards the neighborhood of an emergency landing site.

In the past, guidance systems have been designed in such a way that they accept limited vehicle state
errors at a few nominal EIPs. In such systems, on-board switching from one nominal EIP to another
would be limited by any preliminary mission analysis and would likely require significant in-orbit wait time
to schedule a de-orbit burn that brings the vehicle to one of the limited EIPs available. For these reasons,
the available guidance systems are not fully suitable for abort scenarios, where a long planning and execution
time may compromise the mission and crew safety. Consequently, an adaptive abort guidance system must
accept a wide range of EIPs, only limited by the geometry of the operational orbit at which the abort is
called. Furthermore, such a system could be designed in such a way that global coverage is guaranteed,
which would ensure that a landing site can be safely reached by immediately executing a de-orbit burn right
after the abort calling.

To ensure such capabilities, the system’s trajectory-planner architecture must be revisited. The architec-
ture presented in this paper is largely based on Adaptive Multivariate Pseudospectral Interpolation (AMPI),
which generates mathematical approximations of the reference trajectory to be tracked by comparing the
vehicle-state information at the EIP against a pre-computed trajectory database that is stored on-board. Such an architecture results in a robust guidance system that is capable of running on current flight hardware
and uses proven software building blocks, such as off-board trajectory optimization and interpolation algo-
rithms. In terms of abort trajectory-planning, AMPI requires the generation of a sufficiently large trajectory
database that accounts for all possible abort scenarios. The flexibility of this architecture is only limited by
the information stored in the database, allowing for multiple landing sites to be targeted and a wide range
of entry conditions. Once the planned trajectories are generated on-board, these are fed to the trajectory
tracker, which safely steers the vehicle during descent.

The architecture proposed in this paper is tested using the Subscale Precursor Hypersonic X Vehicle
(SPHYNX), a small lifting-body reentry vehicle developed by the European Space Agency (ESA). It is
assumed that separate propulsion and deorbitation guidance systems are able to place the SPHYNX vehicle
at the given EIP, provided that it is capable of delivering a $\Delta V$ of 125 m/s from a circular operational orbit
at 52 deg inclination at 500 km altitude. Past the EIP, the abort reentry guidance issues angle-of-attack and
bank-angle commands that bring the vehicle from the EIP to the neighborhood of an emergency landing
site, where appropriate altitude and speed conditions must be met for a parafoil to be deployed. The reader
is referred to Fig. 1 for a schematic of the mission profile.

Figure 1. Orbital abort mission profile as discussed in Sec. I.
Section II presents the mathematical models that govern the reentry mechanics. In Sec. III, the theory behind the algorithms that build the abort guidance system is presented. Section IV presents the method used to generate the optimal trajectory database that feeds the trajectory-planner together with the main features of such database. Next, the guidance system is tested with the goal of assessing its performance, where the results of such tests are discussed in Sec. V. Finally, Sec. VI summarizes all the findings of this paper and presents a number of recommendations for future research.

II. The Reentry Problem

This section presents the mathematical description of the reentry problem, including a description of the reference vehicle, the reentry dynamics, the trajectory constraints and the vehicle controls.

II.A. Reference Vehicle

The selected reference vehicle for the orbital abort mission, SPHYNX, was conceived by the ESA as a 1:3 scaled version of the X-38. The goal of SPHYNX was to demonstrate European reentry capabilities as well as validating the multiple technologies developed for the X-38. The development of SPHYNX concluded in 2003 once a feasibility consolidation study was completed at system level. Table 1 indicates the most relevant vehicle dimensions. Note that the nose radius of the vehicle has been artificially increased to 0.5 m from the nominal value of 0.12 m. The reason behind such increase arises from the fact that the nominal nose radius is simply too small to comply with the heat-flux constraint established in Sec. II.C. Finally, the SPHYNX vehicle is intended to perform the final leg of the descent via the use of a steerable parafoil.

Table 1. Dimensions of the SPHYNX reference vehicle

<table>
<thead>
<tr>
<th>m</th>
<th>kg</th>
<th>R_N [m]</th>
<th>L_ref [m]</th>
<th>W_ref [m]</th>
<th>S_ref [m^2]</th>
<th>X_m [m]</th>
<th>Y_m [m]</th>
<th>Z_m [m]</th>
<th>L_hf [m]</th>
<th>S_hf [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>550</td>
<td>0.5*</td>
<td>2.8042</td>
<td>1.6570</td>
<td>2.4080</td>
<td>1.5476</td>
<td>0</td>
<td>0.2682</td>
<td>0.4577</td>
<td>0.2080</td>
<td></td>
</tr>
</tbody>
</table>

Figure 2 shows the schematics of the SPHYNX vehicle, whereas Table 1 indicates the most relevant vehicle dimensions. Figure 2 shows the schematics of the SPHYNX vehicle, whereas Table 1 indicates the most relevant vehicle dimensions. Note that the nose radius of the vehicle has been artificially increased to 0.5 m from the nominal value of 0.12 m. The reason behind such increase arises from the fact that the nominal nose radius is simply too small to comply with the heat-flux constraint established in Sec. II.C. Finally, the SPHYNX vehicle is intended to perform the final leg of the descent via the use of a steerable parafoil.

II.B. Dynamics

The entry dynamics that govern the vehicle motion throughout the descent are described by the following three equations, which relate the external forces to variations in the vehicle’s velocity:

\[
\dot{V} = \frac{F_V}{m} + \omega_E^2 R \cos \delta (\sin \gamma \cos \delta - \cos \gamma \sin \delta \cos \chi)
\]

\[
V \dot{\gamma} = \frac{F_\gamma}{m} + 2\omega_E V \cos \delta \sin \chi + \frac{V^2}{H} \cos \gamma + \omega_E^2 R \cos \delta (\cos \gamma \cos \delta - \sin \gamma \sin \delta \cos \chi)
\]

\[
V \cos \gamma \dot{\chi} = \frac{F_\chi}{m} + 2\omega_E V (\sin \delta \cos \gamma - \cos \delta \sin \gamma \cos \chi)
\]

where $V$ is the vehicle’s groundspeed, $m$ is the vehicle’s mass, $\omega_E$ is the rotational velocity of Earth, $R$ is the vehicle’s radial position, $\delta$ is the geocentric latitude, $\gamma$ is the flight-path-angle and $\chi$ is the heading-angle. The external forces along each axis of the vertical frame, $F_V, F_\gamma, F_\chi$, are obtained by summing the aerodynamic and gravitational forces:

\begin{align*}
F_V &= -D - mg_R \sin \gamma - m g_0 \cos \gamma \cos \chi \\
F_\gamma &= L \cos \sigma - mg_R \cos \gamma + m g_0 \sin \gamma \cos \chi \\
F_\chi &= -L \sin \sigma + m g_0 \sin \chi
\end{align*}

where $D$ is the aerodynamic drag force, $L$ is the aerodynamic lift force, $\sigma$ is the vehicle’s bank angle and $g_R$ is the component of the gravitational acceleration pointing radially towards the center of the central body and $g_0$ is the component of the gravitational acceleration that lies in the local horizontal plane and points positive in a northern direction. The equations of motion are completed with the three kinematic relations shown in Eqs. (7) to (9). Such equations relate the vehicle’s velocity to variations in the vehicle’s position.

\begin{align*}
\dot{R} &= V \sin \gamma \\
\dot{\tau} &= \frac{V \cos \gamma \sin \chi}{R \cos \delta} \\
\dot{\delta} &= \frac{V \cos \gamma \cos \chi}{R}
\end{align*}

where $\tau$ is the geocentric longitude.

II.C. Trajectory Constraints

Throughout this paper a number of trajectory constraints is enforced that ensure the safety of the vehicle and its payload during descent.

Heat Flux

The heat-flux constraint defines the maximum heating rate per unit area that the vehicle can sustain as a consequence of aerothermodynamic heating. The heat-flux is computed using the hot-wall approximation, which computes a wall temperature $T_w$ by equating the incoming convective flux with the outgoing radiative flux $q_{rad}$ for an assumed adiabatic wall temperature $T_{aw}$.

\begin{align*}
q_c &= (5.28137 \cdot 10^{-5}) \sqrt{\frac{\rho}{R_N}} V^{3.15} \left[ 1 - \frac{T_w}{T_{aw}} \right] \leq 670 \text{ kW/m}^2 \\
q_{rad} &= \epsilon \sigma T_w^4
\end{align*}

where $q_c$ is the convective heat-flux at the vehicle’s stagnation point, $R_N$ is vehicle’s nose radius, $\rho$ is the free-stream air density, $\epsilon$ is the emissivity of the applied material and $\sigma$ is the Stefan-Bolzmann constant. The above equations can be combined together into the relation given below. A root-finding method is used to compute the wall temperature, which is then used to obtain the convective heat-flux using Eq. (10).

\begin{align*}
(5.28137 \cdot 10^{-5}) \sqrt{\frac{\rho}{R_N}} V^{3.15} \left[ 1 - \frac{T_w}{T_{aw}} \right] - \epsilon \sigma T_w^4 &= 0
\end{align*}

Aerodynamic Load

The aerodynamic load constraint defines the maximum aerodynamic acceleration experienced during descent:

\begin{align*}
n_g &= \frac{\rho V^2 S_{ref}}{2m g_0} \sqrt{C_D^2 + C_L^2} \leq 3 \text{ g}
\end{align*}

where $n_g$ is the aerodynamic load, $S_{ref}$ is the vehicle reference area, $g_0$ is the Earth’s gravitational acceleration at sea level, $C_D$ is the drag coefficient and $C_L$ is the lift coefficient.
II.D. Boundary Constraints

The vehicle state at the EIP and the guidance termination point are given in Table 2. Since the abort mission could begin anywhere throughout the operational orbit, the entry latitude and longitude are left free, where the entry latitude is bounded by $\pm 52\,\text{deg}$ as derived from the inclination of the operational orbit. As for the entry heading-angle, this is derived analytically from the entry latitude and longitude using Eqs. (14) to (16).

$$\chi_{EIP} = \pi - \cos^{-1} \left( -\cos(\tau_{EIP}) \cos \left( \frac{\pi}{2} - i \right) \right)$$ \hspace{1cm} (14)

$$\delta_{EIP} = f(\chi_{EIP}) = H \sin^{-1} \left( \frac{\pi}{2} - \frac{\sin \left( \frac{\pi}{2} - i \right)}{\sin \left( \pi - \chi_{EIP} \right)} \right)$$ \hspace{1cm} (15)

$$H = \begin{cases} 
1 & \text{if } \tau_{EIP} < \pi \\
-1 & \text{elsewhere} 
\end{cases}$$ \hspace{1cm} (16)

where $\chi_{EIP}$ is the heading angle of the EIP, $i$ is the orbit inclination and $\delta_{EIP}$ and $\tau_{EIP}$ are the latitude and longitude angles of the EIP, respectively. As for the terminal conditions, these are mostly defined by the parafoil-deployment requirements provided by ESA. These require the parafoil opening to occur at an altitude between 10 and 13 km in the subsonic regime with a maximum vertical speed of 140 m/s. Note that the terminal latitude and longitude are given by the location of the targeted emergency landing site as established in Table 3. Finally, the vehicle must be brought within a 10 km radius from the landing site.

Table 2. State variable values at the EIP and the guidance termination point.

<table>
<thead>
<tr>
<th>State Variable</th>
<th>Initial Value</th>
<th>Terminal Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude [km]</td>
<td>120</td>
<td>10 - 13</td>
</tr>
<tr>
<td>Latitude [deg]</td>
<td>Free</td>
<td>Landing Site Dependent</td>
</tr>
<tr>
<td>Longitude [deg]</td>
<td>Free</td>
<td>Landing Site Dependent</td>
</tr>
<tr>
<td>Groundspeed [m/s]</td>
<td>7640</td>
<td>Mach Number Limit Dependent</td>
</tr>
<tr>
<td>Flight-Path-Angle [deg]</td>
<td>-1.28</td>
<td>Max. Vertical Speed Dependent</td>
</tr>
<tr>
<td>Heading-Angle [deg]</td>
<td>Latitude Dependent</td>
<td>Free</td>
</tr>
</tbody>
</table>

Table 3. Emergency landing sites for SPHYNX.

<table>
<thead>
<tr>
<th>ID</th>
<th>Landing Site</th>
<th>Geodetic Latitude [deg]</th>
<th>Longitude [deg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Azores Airport</td>
<td>36.974</td>
<td>-25.165</td>
</tr>
<tr>
<td>2</td>
<td>Gran Canaria Airport</td>
<td>27.933</td>
<td>-15.388</td>
</tr>
<tr>
<td>3</td>
<td>Ben Guerir Air Force Base</td>
<td>32.128</td>
<td>-7.878</td>
</tr>
<tr>
<td>4</td>
<td>Perth Airport</td>
<td>-31.938</td>
<td>115.967</td>
</tr>
<tr>
<td>5</td>
<td>Darwin Airport</td>
<td>-12.411</td>
<td>130.878</td>
</tr>
<tr>
<td>6</td>
<td>Brisbane Airport</td>
<td>-27.394</td>
<td>153.121</td>
</tr>
<tr>
<td>7</td>
<td>Sidney Airport</td>
<td>-33.940</td>
<td>151.176</td>
</tr>
<tr>
<td>8</td>
<td>Woomera Airfield</td>
<td>-31.145</td>
<td>136.825</td>
</tr>
</tbody>
</table>

II.E. Controls & Aerodynamics

SPHYNX is modeled as a point-mass, meaning that rotational dynamics are not considered. Thus, the vehicle is controlled by means of the bank angle, $\sigma$, and the angle of attack, $\alpha$. It is assumed that the reentry is performed in coordinated flight, consequently the angle of sideslip, $\beta$, is zero and no side forces are exerted on the vehicle. Note that the bank angle is limited to $\pm 85\,\text{deg}$ to avoid lift-down entries, whereas the angle of attack is limited by the regions of $\alpha-M$, where vehicle trimming is possible.

The aerodynamic database of SPHYNX provides the clean form of such coefficients, as well as the contributions of the body-flap as a function of body-flap deflection. In summary, the following aerodynamic...
coefficients are used:

\[
C_L = C_{L,0} (M, \alpha) + \Delta C_{L,bf} (M, \alpha, \delta_{bf}) \tag{17}
\]

\[
C_D = C_{D,0} (M, \alpha) + \Delta C_{D,bf} (M, \alpha, \delta_{bf}) \tag{18}
\]

\[
C_m = C_{m,0} (M, \alpha) + \Delta C_{m,bf} (M, \alpha, \delta_{bf}) \tag{19}
\]

where \(C_L\) is the total lift coefficient, \(C_{L,0}\) is the clean lift coefficient and \(\Delta C_{L,bf}\) is the lift coefficient contribution of the body-flap deflection. Similarly, \(C_D\) is the total drag coefficient, \(C_{D,0}\) is the clean drag coefficient and \(\Delta C_{D,bf}\) is the drag coefficient contribution of the body-flap deflection. Finally, \(C_m\) is the total pitching moment coefficient, \(C_{m,0}\) is the clean pitching moment coefficient and \(\Delta C_{m,bf}\) is the pitching moment coefficient contribution of the body-flap deflection.

Vehicle trimming is achieved by setting a body-flap deflection that counteracts the clean pitching moment coefficient. Such deflection is found by finding the zero of Eq. (19) for every angle of attack and Mach number of the \(\alpha - M\) space using a root-finding method. Once the body-flap deflection required for trim is found, the total lift and drag coefficients are found as described by Eqs. (17) and (18), respectively. Close examination of Fig. 3 reveals areas of the \(\alpha-M\) space where the body-flap deflection is undefined. There are two reasons why this occurs:

1. **Unavailable Data:** The aerodynamic database of SPHYNX is not fully defined for the entire \(\alpha-M\) space. For instance, the subsonic-supersonic regime is only defined for Mach numbers in the \([0.0, 4.6]\) range and angle-of-attack values in the \([-10.0, 40.0]\) deg range. Similarly, the hypersonic regime is defined for Mach numbers in the \([4.6, 30.0]\) range and angle-of-attack values in the \([20.0, 50.0]\) deg range. Extrapolation of coefficient data in the non-defined areas is avoided, due to the questionable validity of the produced coefficients.

2. **Non-trimmable Area:** In the remaining areas of the \(\alpha-M\) space, the body-flap deflection required to trim the vehicle exceeds the maximum deflections allowed for by the actuator mechanism. Although the body-flap mechanism allows deflections in the \([5.0, 35.0]\) degree range, the maximum deflections are set to \([10.0, 30.0]\) to enable aileron usage of the body-flaps.

In essence, the aerodynamic database constraints mentioned above delimit the angle-of-attack values that can be commanded as a function of Mach number. A key feature to highlight is the bottleneck created by such constraints in the supersonic regime for angle-of-attack values near 20 deg. As will be discussed in Section IV.C, this bottleneck strongly constrains the solution space of the reachability study.

It is important to emphasize that the database described in this section uses the nominal vehicle geometry, in other words, it employs a nose radius of 0.12 m. As mentioned in Section II.A, this paper assumes a nose radius of 0.5 m, but does not modify the nominal database. Increasing the nose radius leads to a stronger wave drag and shifts downstream the transition point from laminar to turbulent flow. Consequently, the flow velocity gradients over the vehicle change, leading to changes in the distribution of aerodynamic heating and skin friction. This is a source of error that should be accounted for in future studies, where the aerodynamic database should be re-assessed by performing new Computational Fluid Dynamics (CFD) analysis.

![Figure 3. Elevator deflection required to trim the SPHYNX vehicle.](image-url)
III. Adaptive Abort Guidance

The architecture of a guidance system can be classified under two different blocks: the trajectory planner and the trajectory tracker. On the one hand, the trajectory planner is in charge of determining the magnitude of the attitude commands and schedule these as a function of a particular independent variable. Such planning takes into account the vehicle state at the EIP, the trajectory constraints and the desired vehicle state at the terminal point. On the other hand, the trajectory tracker ensures that the reference attitude profiles produced by the planner are carefully followed during the flight, despite any disturbances or errors on the vehicle state at the EIP. Note that the guidance commands are scheduled in terms of energy instead of time, since it is a better descriptor of the entry dynamics while being a monotonic variable.

III.A. On-board Trajectory Planner

The on-board trajectory planner discussed in this paper is based on the AMPI algorithm presented in Refs. 5 to 7. As shown in Fig. 4, the AMPI algorithm receives two inputs: a set of "physics-following" trajectories and the on-board vehicle state the EIP as provided by the navigation subsystem. The set of "physics-following" trajectories refers to those trajectories that have been produced by an off-board trajectory optimization software. They are referred as "physics-following", because they have been computed by solving a set of equations of motion, and thus represent trajectories that a reentry vehicle would encounter in a potential flight scenario. The role of AMPI is to produce a "mathematical" reference trajectory via interpolation, where the "physics-following" trajectories are used as the supporting nodes of such interpolation and the on-board vehicle state at the EIP provides the interpolation evaluation point. Note that the output of the AMPI algorithm does not represent a formal solution of the equations of motion, but does provide a good real-time approximation.

Definition and Discretization of the Parameter Space

The AMPI algorithm begins with the discretization of all the parameters that are expected to be off-nominal and that can be provided during the flight by the vehicle’s navigation system. In the framework of reentry guidance, these parameters are essentially the vehicle state variables at the EIP, which is later used as the interpolation evaluation point.

By encapsulating the full entry state in the database, Sagliano ensures that the interpolated trajectories resulting from such database are of high-fidelity, resembling those that would be obtained when solving the equations of motion.5 This is possible due to the fact that the parameter range that he uses is limited, compared to the range that would be required in an adaptive abort guidance system. If full global coverage is desired, the resulting database size may become unfeasible if a full six-parameter space is selected. For this reason, the trajectory planner researched in this paper will feed off a database featuring only two parameters: geodetic latitude and longitude. Consequently, the multivariate interpolation featured in Ref. 5 reduces to a simple bivariate interpolation. In mathematical terms, the parameter space employed in this paper is formed by the following two parameters:

\[ p^1 = \tau \subset [p^1_1, p^1_2] = [-180.0, 180.0] \]  \[ p^2 = \delta^* \subset [p^2_1, p^2_2] = [-52.0, 52.0] \]  

where \( p^1 \) is the first database parameter, \( p^2 \) is the second database parameter and \( \delta^* \) is the geodetic latitude. Note that the superscript indicates the referred database parameter and the subscript indicates the particular bound of such parameter. In this paper, the entry longitude is discretized from -180 deg to 180 deg, where the entry geodetic latitude is discretized -52 deg from to 52 deg, thus encapsulating all the EIPs originating
from the operational orbit. Note that the two database parameters form a 2-dimensional rectangular grid \( \mathcal{P} \), as shown in the following equation:

\[
\mathcal{P} = p^1 \times p^2
\]  
(22)

To illustrate the concept of the parameter space and its relation to the trajectory database, the reader is asked to examine Fig. 5. Here, a \([4 \times 7]\) grid represents the trajectory database. Every point in this grid is equivalent to an item in the trajectory database and thus contains trajectory information that complies with the equations of motion. In fact, each of these "points" is essentially an EIP from which a set of optimal attitude control profiles originate that in turn bring the vehicle from that particular EIP to one of the listed landing sites.

![Figure 5. Sketch showing the methodology employed during bivariate interpolation.](image)

**Selection of the Reference Subspace (Offboard)**

Once the parameter space has been established and the trajectory database has been generated, the parameter subspace \( \mathcal{P}^s \), enclosing the interpolation evaluation point \( x_0 \) needs to be found. Such subspace is depicted in Fig. 5 and defined by the following three equations:

\[
\mathcal{P}^s = s^1 \times s^2 \quad (23)
\]

\[
s^i = [s^i_1, s^i_2] \quad (24)
\]

\[
s^i \subset [p^i_1, p^i_2] \quad (25)
\]

As mentioned above, \( x_0 \) is constructed on-board, and it contains the vehicle’s longitude and geodetic latitude at the expected EIP, as indicated in the following equation:

\[
x_0 = [x^1_0, x^2_0] = [\tau_{flt}, \delta^*_{flt}] \quad (26)
\]

\[
x^i_0 \subset [s^i_1, s^i_2] \quad (27)
\]

The subspace \( \mathcal{P}^s \) delimits the area of supporting information that is closest to the interpolation evaluation point, highlighted in Fig. 5. The trajectories that delimit the parameter subspace enclosing the interpolation evaluation point are referred to as "extremals". The naming convention follows the relative grid positioning of such extremals, as shown in the following equation:

\[
e_{TL} = [s^1_1, s^2_2] \quad (28)
\]

\[
e_{TR} = [s^2_1, s^2_2] \quad (29)
\]

\[
e_{BL} = [s^1_1, s^1_2] \quad (30)
\]

\[
e_{BR} = [s^2_1, s^1_2] \quad (31)
\]

where \( e_{TL} \) is the top-left extremal, \( e_{TR} \) is the top-right extremal, \( e_{BL} \) is the bottom-left extremal and \( e_{BR} \) is the top-right extremal. Note that each extremal contains time, energy, the vehicle state, the guidance commands, the guidance gains as well as supporting trajectory information.
The variables stored in each extremal need to be sampled along an independent variable in such a way that they produce a low-density (LD) grid where interpolation is to be performed. It is important to emphasize that the choice for the independent variable is not enforced, thus another variable such as energy, can be selected instead of time. Irrespective of the choice for the independent variable, the four extremals will differ in trajectory length, which is obvious when considering that they represent entirely different solutions to the equation of motion. This is a key aspect to understand, since to make interpolation possible, the independent variables of each extremal need to be converted to a "pseudo" independent variable with common bounds. In the remainder of this paper, such common independent variable is referred to as pseudo-time \( \tau \), which is defined from -1 to 1 as shown in the following equation:

\[
\tau \subset [-1, 1]
\]  

(32)

where the conversion between the actual time and the pseudo-time is given by Eqs. (33) and (34). Once again, note that this conversion is valid for any other independent variable since the ”pseudo” independent variable always ranges from -1 to 1.

\[
t = \frac{t_f - t_0}{2} \tau + \frac{t_f - t_0}{2}
\]

(33)

\[
\tau = \frac{2}{t_f - t_0} t - \frac{t_f + t_0}{t_f - t_0}
\]

(34)

where \( t \) is the independent variable, and \( t_0 \) and \( t_f \) represent the corresponding initial and final values. Similarly to other trajectory variables, the pseudo-time is sampled in a low-density grid, where each sampling point is referred to as a "node". The pseudo-time is sampled into a vector of \( N_{LD} \) elements, as shown in the next equation. Note that the number of low-density nodes \( N_{LD} \) is a key parameter that determines the quality of the interpolated reference trajectory.

\[
\tau = [\tau_0, \tau_1, \ldots, \tau_{N_{LD}}]
\]

(35)

A known curse of polynomial interpolation is the so-called Runge phenomenon, which leads to artificial oscillation towards the ends of the interval when the degree of the interpolating polynomial becomes too large and the nodes are equally spaced. The presence of such oscillations would result in an interpolated reference trajectory that artificially deviates from the extremals shortly after the EIP and just before the terminal point, leading to unwanted dispersions and possible the violation of the trajectory constraints. Ref. 5 tackles this issue by distributing the low-density nodes according to the roots of the Flipped Radau-Legendre Polynomials (FRPs), which are defined according to:

\[
L_N (\tau) = \frac{1}{2^N N!} \frac{d^N}{d\tau^N} \left( (\tau^2 - 1)^N \right)
\]

(36)

\[
R_N (\tau) = L_N (\tau) - L_{N-1} (\tau)
\]

(37)

where \( L_N \) is the Legendre polynomial of \( N^{th} \) degree and \( R_N \) is the FRP polynomial of \( N^{th} \) degree. In Fig. 6, an FRP of 45\(^{th} \) degree is shown. This figure clearly shows that the method proposed by Ref. 5 increases the number of low-density nodes towards the end of the interval, effectively tackling the discussed Runge phenomenon.

**Low-Density Multivariate Interpolation (Onboard)**

The trajectory interpolation discussed in this paper is based on the bilinear interpolation scheme due to its simplicity. Furthermore, using a scheme that uses higher order basis functions, such as cubic spline interpolation, would require additional extremals to support the interpolation process. However, as Fig. 5 reveals, these additional extremals would not necessarily improve the fidelity of the interpolated solution, since they would be located further away from the interpolation evaluation point as compared to the extremals in Eqs. (28) to (31).

The first step in the process of bilinear interpolation is to assemble the extremal functions to be interpolated, \( f \), into matrix form, as shown in the following equation:

\[
C_k = \begin{bmatrix} f (\tau_k, e_{BL}) & f (\tau_k, e_{TL}) \\ f (\tau_k, e_{BR}) & f (\tau_k, e_{TR}) \end{bmatrix}
\]

(38)

where \( C_k \) is the matrix containing the database information evaluated at \( \tau_k \) and \( f (\tau_k, e_i) \) is the function to be interpolated evaluated at \( \tau_k \) on extremal \( e_i \). Note that \( C_k \) is recomputed for every low-density node.
specified in pseudo-time vector. The next step is to assemble the projection matrices \( \phi \) and \( \psi \), which are given in the next equation:

\[
\begin{align*}
\phi(x_0^1, s^1, s^2) &= \frac{1}{(s_2^1 - s_1^1)(s_2^2 - s_1^2)} \begin{bmatrix} s_2^1 - x_0^1 \\ x_0^1 - s_1^1 \end{bmatrix} \\
\psi(x_0^2, s^2) &= \begin{bmatrix} s_2^2 - x_0^2 \\ x_0^2 - s_1^1 \end{bmatrix}
\end{align*}
\] (39)

Finally, the bilinear interpolation at is computed through a matrix multiplication, as indicated in the following equation:

\[
f_{\text{int}}(\tau_k, x_0) = \phi^T C \psi
\] (41)

where \( f_{\text{int}}(\tau_k, x_0) \) is the interpolant of the extremal function \( f \), evaluated at \( \tau_k \) and \( x_0 \). As a final note, it is important to emphasize that Eqs. (38) to (41) assume that the interpolation of any database variable \( f \) throughout the pseudo-time vector \( \tau \) only depends on the relative position of \( x_0 \) with respect to the extremals introduced in Eqs. (28) to (31). The interpolated function can thus be assembled into a vector as shown in the following equation:

\[
f_{\text{int}}(\tau, x_0) = [f_{\text{int}}(\tau_0, x_0), f_{\text{int}}(\tau_1, x_0), \ldots, f_{\text{int}}(\tau_{N_{LD}}, x_0)]
\] (42)

**Low-Density to High-Density Pseudospectral Conversion (Onboard)**

The last stage of the AMPI algorithm is to convert the low-density interpolated solution obtained in the previous section into a high-density (HD) solution that can be used by the trajectory tracker. To do so, a high-density pseudo-time vector is first constructed, as given in the following equation:

\[
\tilde{\tau} = [\tilde{\tau}_0, \tilde{\tau}_1, \ldots, \tilde{\tau}_{N_{HD}}]
\] (43)

where \( \tilde{\tau} \) is the high-density pseudo-time vector and \( N_{HD} \) is the number of high-density nodes. Similarly to the low-density pseudo-time vector, \( \tilde{\tau} \) spans from -1 to 1. The spacing of the nodes is not critical in this case, since the Runge phenomenon is no longer present due to the fact that no interpolation is made at this stage. The LD-HD conversion is made effective by constructing an approximating function that uses the low-density nodes as support points, and the separation of high-density nodes and low-density nodes as basis functions. This is shown by:

\[
\tilde{f}_{\text{int}}(\tilde{\tau}_m) = \sum_{i=0}^{N_{LD}} f_{\text{int}}(\tau_i) \prod_{\substack{k=0 \\ k \neq i}}^{N_{LD}} \frac{\tilde{\tau}_m - \tau_k}{\tau_i - \tau_k}, \quad m = 0, 1, \ldots, N_{HD}
\] (44)
where \( \tilde{f}_{\text{int}}(\tilde{\tau}_m) \) is the high-density interpolated solution at node \( \tilde{\tau}_m \). The above operation can be assembled into matrix form as shown by the following two equations:

\[
\tilde{f}_{\text{int}} = f_{\text{int}} P_{\text{FRP}} \tag{45}
\]

\[
P_{\text{FRP}} = \begin{bmatrix}
\prod_{k=1}^{N_{LD}} \frac{\tilde{\tau}_0 - \tilde{\tau}_k}{\tau_0 - \tau_k} & \cdots & \prod_{k=1}^{N_{HD}} \frac{\tilde{\tau}_{N_{HD}} - \tilde{\tau}_k}{\tau_0 - \tau_k} \\
\cdots & \cdots & \cdots \\
\prod_{k=0}^{N_{LD}-1} \frac{\tilde{\tau}_0 - \tilde{\tau}_k}{\tau_{N_{LD}} - \tau_k} & \cdots & \prod_{k=0}^{N_{LD}-1} \frac{\tilde{\tau}_{N_{HD}} - \tilde{\tau}_k}{\tau_{N_{LD}} - \tau_k}
\end{bmatrix} \tag{46}
\]

where \( \tilde{f}_{\text{int}} \) is the high-density interpolated solution and \( P_{\text{FRP}} \) is the LD-HD conversion matrix. Note that \( P_{\text{FRP}} \) only contains information about the relative spacing of the high-density nodes with respect to the low-density nodes.

In this paper, 190 LD nodes and 4000 HD nodes are used as a baseline. As will be shown in Sec. IV, database entries range from 2000 to 5000 seconds, thus 4000 HD nodes lead to an adequate spacing of 0.5 to 1.25 seconds. It is important to emphasize that the transformation discussed in this section is a loss-less process, enabling the on-board storage of the database in its LD form. The compression factor \( \kappa \) of the database can be simply computed with:

\[
\kappa = \frac{N_{LD}}{N_{HD}} \tag{47}
\]

Finally, the size of the database is estimated according to:

\[
S_{DB} = N_t S_{F,HD} \frac{N_{LD}}{N_{HD}} \tag{48}
\]

where \( S_{DB} \) is the compressed size of the database, \( N_t \) is the number of trajectories in the database and \( S_{F,HD} \) is the average file size for a single trajectory. Note that using 4000 HD nodes, the average file size becomes approximately 300 kB.

### III.B. Trajectory Tracker

The trajectory tracker implemented in this paper is based on the Linear Quadratic Regulator (LQR) theory, which is well known and has been applied in the past to reentry guidance.\(^{11, 12, 13}\) The LQR tracker presented here uses linearized longitudinal dynamics with respect to time to generate the required angle-of-attack and bank-angle commands. Furthermore, the state deviation vector \( \delta x \) proposed here consists of the groundspeed \( \Delta V \), flight-path angle \( \Delta \gamma \), radius \( \Delta R \), and traveled range \( \Delta s \), as shown by the following three equations.

Additionally, the control deviation vector \( \Delta u \) is formed by the angle of attack \( \Delta \alpha \) and the bank angle \( \Delta \sigma \). These deviations are computed with respect to the nominal state \( x_{\text{nom}} \) and the nominal controls \( u_{\text{nom}} \), which are provided by the trajectory planner discussed in Sec. III.A.

\[
\Delta \dot{x} = A\Delta x + B\Delta u \tag{49}
\]

where \( A \) is the system matrix which determines how the rate of change of state deviation is affected by the current state deviations. Furthermore, \( B \) is the input matrix, which determines how \( \Delta \dot{x} \) is affected by the current control corrections. The full mathematical expression of Eq. (49) is given by the following equation:

\[
\begin{bmatrix}
\Delta \dot{V} \\
\Delta \dot{\gamma} \\
\Delta \dot{R} \\
\Delta \dot{s}
\end{bmatrix} =
\begin{bmatrix}
a_{VV} & a_{V\gamma} & a_{VR} & 0 \\
a_{\gamma V} & a_{\gamma \gamma} & a_{\gamma R} & 0 \\
a_{RV} & a_{R\gamma} & a_{R R} & 0 \\
a_{sV} & a_{s\gamma} & a_{sR} & 0
\end{bmatrix}
\begin{bmatrix}
\Delta V \\
\Delta \gamma \\
\Delta R \\
\Delta s
\end{bmatrix}
+ \begin{bmatrix}
b_{V\alpha} & 0 \\
b_{\gamma \alpha} & b_{\gamma \sigma} \\
0 & 0 \\
0 & 0
\end{bmatrix}
\begin{bmatrix}
\Delta \alpha \\
\Delta \sigma
\end{bmatrix} \tag{50}
\]

where the traveled horizontal range is defined by:

\[
s = V_g \cos \gamma \tag{51}
\]
As mentioned above, the goal of the LQR theory is to define a feedback control-law. This law computes the required control deviations as a function of a feedback matrix $K$ and the state deviations:

$$\Delta u = -K \Delta x$$

where the elements of the feedback matrix are given by

$$K = \begin{bmatrix} k_{\alpha,V} & k_{\alpha,\gamma} & k_{\alpha,R} & k_{\alpha,s} \\ k_{\sigma,V} & k_{\sigma,\gamma} & k_{\sigma,R} & k_{\sigma,s} \end{bmatrix}$$

with $k_{i,j}$ quantifying the required deviations in control $u_i$ given the errors in the state variable $x_j$. The feedback matrix is computed by minimizing the performance criterion $J_{LQR}$ presented in the following equation:

$$J_{LQR} = \int_{t_0}^{\infty} (Q_V \Delta V^2 + Q_s \Delta s^2 + R_{\alpha} \Delta \alpha^2 + R_{\sigma} \Delta \sigma^2) \, dt$$

where $Q_i$ refers to the weight associated to the deviation in the state variable $i$ and $R_i$ refers to the weight associated to the deviation in the control variable $i$. A unique solution to this minimization is found by solving the Riccati equation.\(^{14}\) The elements of the weight matrices are given by:

$$Q = \text{diag}(Q_V, 0, 0, Q_s) \quad R = \text{diag}(R_{\alpha}, R_{\sigma})$$

Note that the diagonal elements in the matrices $Q$ and $R$ can be found using Bryson’s rule, which is stated in the following equation:\(^{12}\)

$$Q_V \Delta V_{\text{max}}^2 = Q_s \Delta s_{\text{max}}^2 = R_{\alpha} \Delta \alpha_{\text{max}}^2 = R_{\sigma} \Delta \sigma_{\text{max}}^2 = 1$$

where the variables denoted with subscript $\text{max}$ refer to the maximum allowable deviations in such variable. Note that these maximum allowable deviations are simply control-law design parameters and do not necessarily impose hard constraints.\(^{12}\) Finally, $\Delta V_{\text{max}} = 50.0 \, \text{m/s}$, $\Delta s_{\text{max}} = 10.0 \, \text{m}$, $\Delta \alpha_{\text{max}} = 0.5 \, \text{deg}$, $\Delta \sigma_{\text{max}} = 5.0 \, \text{deg}$ are established as the baseline maximum allowable deviation for the state variables. Note that these values merely establish a baseline and thus may need to be tuned as discussed in Sec. V. Once the feedback matrix is found, the commanded angle of attack and the commanded bank-angle magnitude are given by:

$$\alpha_c = \alpha_{\text{nom}} + \Delta \alpha \quad |\sigma_c| = |\sigma_{\text{nom}}| + \Delta \sigma$$

It is important to emphasize that controlling the four state variables with two control variables has proven to be unfeasible. For this matter, the weights of the flight-path angle and radius are set to zero, in a way that infinite errors are allowed for such variables, but are still part of the linearized dynamic equations. Despite this, non-zero angle-of-attack and bank-angle gains can be obtained. This indicates that errors in the flight-path angle and the radial distance are still accounted for when computing the necessary control deviations. In addition, note that the elements of the input matrix that relate the control inputs to the rate of change of the traveled range error are zero, indicating that the traveled range cannot be controlled directly. Instead, the traveled range is indirectly controlled by means of the groundspeed and the flight-path angle, as shown in Eq. (51). Please note that each trajectory stored in the database has its own set of gains and are thus part of the trajectory interpolation scheme discussed in Sec. III.A. Finally, the interpolation uses the vehicle total energy as the independent variable, instead of time, because it allows to schedule the gains according to a monotonic variable that represents the entry dynamics well.

**Lateral Guidance**

The role of the lateral guidance is to keep the vehicle from deviating too far in the cross-track direction from its nominal path. Such guidance is essential to ensure that the vehicle flies towards the targeted landing site. The lateral guidance implemented in this paper tracks the heading angle error and ensures that it is kept within some predefined bounds. The heading angle to the target landing site $\chi_T$ is computed using:\(^{11}\)

$$\chi_T = \tan^{-1}\left(\frac{\sin (\tau_T - \tau) \sin \left(\frac{\pi}{2} - \delta_T\right)}{\cos \left(\frac{\pi}{2} - \delta_T\right) \sin \left(\frac{\pi}{2} - \delta\right) - \cos (\tau_T - \tau) \cos \left(\frac{\pi}{2} - \delta\right) \sin \left(\frac{\pi}{2} - \delta_T\right)}\right)$$

$$= \tan^{-1}\left(\frac{\sin (\tau_T - \tau) \sin \left(\frac{\pi}{2} - \delta_T\right)}{\cos \left(\frac{\pi}{2} - \delta_T\right) \sin \left(\frac{\pi}{2} - \delta\right) - \cos (\tau_T - \tau) \cos \left(\frac{\pi}{2} - \delta\right) \sin \left(\frac{\pi}{2} - \delta_T\right)}\right)$$

$\chi_T$ is computed using:\(^{11}\)
where $\tau_T$ is the longitude of the landing site and $\delta_T$ is the latitude of the landing site. The heading angle error $\chi_e$ at any point throughout the trajectory is simply the difference between the vehicle’s current heading angle $\chi$ and the heading angle to the target $\chi_T$:

$$\chi_e = \chi - \chi_T$$

(60)

By enforcing the signs of the vehicle’s heading error and commanded bank angle to be opposite, the vehicle is steered in such a way that it flies towards the landing site. In addition, the magnitude of the heading error is controlled to avoid an excessive number of bank reversals. This is done by establishing a constant heading error dead band (HEDB) of $\pm 25$ deg, which proved to be effective in steering the vehicle towards the landing site.

IV. Landing Site Reachability Study

In the framework of entry mission planning, a reachability study can be defined as the process of determining the landing footprint of a vehicle. Performing a thorough reachability study is essential to determine the possible abort opportunities of an entry vehicle. Most importantly, the results of such a study explicitly confirm the trajectory database required by the AMPI trajectory planner discussed in Sec. III.

IV.A. Optimal Control Problem

The reachability study proposed in this paper is completed by solving the following optimal control problem:

*What are the attitude controls required to steer the re-entry vehicle from an arbitrary entry point to one of the designated emergency landing sites, while ensuring that the trajectory constraints are satisfied?*

The optimal control problem is solved using the ASTOS commercial software package\textsuperscript{b}. Note that such problem is governed by the reentry dynamics and thus the software settings are adjusted such that the environment models, vehicle models and trajectory constraints match those described in Sec. II.

A summary of all the trajectory constraints applied in the reachability study is given in Table 4. In particular, the following two constraints are added:

- **Max. flight-path angle:** Limits the phugoid motion of the vehicle in the lower atmosphere. A positive value is allowed to enable solutions that use a skip entry approach to achieve more downrange.
- **Max. / min. bank angle:** Avoids entry trajectories in a lift-down configuration.

The performance index is selected in such a way that the angle-of-attack and bank-angle controls are as smooth as possible. This was found necessary for the trajectory interpolation to work, since doing otherwise would result in discontinuous control profiles at the extremals that are too dissimilar from each other. Furthermore, discontinuous control profiles at the extremals would introduce artificial oscillations at such discontinuities. To achieve such smoothing, the performance index $J_{sm}$ is defined as the integral of control derivatives over time:

$$J_{sm} = \int_{t_0}^{t_f} w_{sm} q(u(t)) \, dt$$

(61)

where $q$ is the quadrature function vector and $u$ is the control vector. The trajectory smoothing factor $w_{sm}$ is set to $10^{-4}$ and was found sufficient to produce smooth trajectories for the interpolation to succeed.

IV.B. Trajectory Database Generation Strategy

The optimal-control problem described in Sec. IV.A generates the attitude commands that steer the vehicle from a given entry point to a chosen landing site, in other words, a single trajectory. To complete the reachability analysis, such an optimal-control problem must be solved for every possible EIP and for all the landing sites listed in Table 3. The reachability study is completed by performing the following approach for every landing site:

\textsuperscript{b}ASTOS AeroSpace Trajectory Optimization Software, Astos Solutions, \url{https://www.astos.de}
1. **Discretize entry window:** The global entry window groups all the possible EIPs that the vehicle can encounter following the deorbitation burn, where the characteristics of the resulting grid must agree with the choice of the parameter space discussed in Sec. III.A.

2. **Probe entry window:** A script loops through all the EIPs obtained from the discretization of the entry window and probes for a possible solution to the optimal control problem.

   2.1 Compute range-to-go given the probed EIP and the targeted landing site.
   
   2.2 Set a constant bank-angle initial guess based on an approximation of downrange capability of the SPHYNX vehicle and the current range-to-go.
   
   2.3 Simulate the initial guess trajectory and evaluate the proximity of the terminal point to the targeted landing site. This is established by an area of allowable terminal points delimited by a [-100, 50] deg longitude boundary and a [-30, 30] deg latitude boundary with respect to the landing site.
      
      (A) **Unfeasible Distance to Landing Site:** Solution of the optimal control problem is avoided and the loop continues with the next EIP.
      
      (B) **Feasible Distance to Landing Site:** An attempt to solve the optimal control problem is made.
         
         (a) **Optimal Solution Found:** The guidance commands and supporting trajectory information is saved to a text file.
         
         (b) **No Solution Found:** If the maximum number of iterations is exceeded or divergence occurs, the solution is discarded and the loop continues with the next EIP.

As shown in Fig. 7, the discussed approach begins by probing all the possible EIP latitudes at the west-most possible EIP longitude (Step 1). This process is repeated for all the discretized longitude values (Step 2), ultimately generating worldwide entry window. In Fig. 7, probe EIPs are distributed with a spacing of 10 deg, running from -180 deg to -100 deg longitude and -52 deg to 52 deg in latitude. In this example, only 18 solution EIPs are found, which together delimit the entry window area. Note that by enlarging the area of allowable end-points, the probability of neglecting potential solution points lowers at the expense of computational time. Since all the entry-window maps are visually inspected after completion, any missing solution points can be quickly identified and executed. In summary, the feasible trajectories to

<table>
<thead>
<tr>
<th>ID</th>
<th>Constraint</th>
<th>Value</th>
<th>Type</th>
<th>Sub-Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Entry Altitude</td>
<td>120 km</td>
<td>EQ</td>
<td>IB</td>
</tr>
<tr>
<td>2</td>
<td>Entry Latitude</td>
<td>Variable</td>
<td>EQ</td>
<td>IB</td>
</tr>
<tr>
<td>3</td>
<td>Entry Longitude</td>
<td>Variable</td>
<td>EQ</td>
<td>IB</td>
</tr>
<tr>
<td>4</td>
<td>Entry Groundspeed</td>
<td>7640 m/s</td>
<td>EQ</td>
<td>IB</td>
</tr>
<tr>
<td>5</td>
<td>Entry Flight-Path Angle</td>
<td>-1.28 deg</td>
<td>EQ</td>
<td>IB</td>
</tr>
<tr>
<td>6</td>
<td>Entry Heading Angle</td>
<td>Variable</td>
<td>EQ</td>
<td>IB</td>
</tr>
<tr>
<td>7</td>
<td>Max. Distance To Target</td>
<td>10 km</td>
<td>IQ</td>
<td>FB</td>
</tr>
<tr>
<td>8</td>
<td>Terminal Altitude Range</td>
<td>10 - 13 km</td>
<td>IQ</td>
<td>FB</td>
</tr>
<tr>
<td>9</td>
<td>Terminal Mach Number Range</td>
<td>0.0 - 1.0</td>
<td>IQ</td>
<td>FB</td>
</tr>
<tr>
<td>10</td>
<td>Max. Terminal Vertical Speed</td>
<td>-140 m/s</td>
<td>IQ</td>
<td>FB</td>
</tr>
<tr>
<td>11</td>
<td>Max. Heat Flux Density</td>
<td>670 kW/m²</td>
<td>IQ</td>
<td>Path</td>
</tr>
<tr>
<td>12</td>
<td>Max. Aerodynamic Load</td>
<td>3.0</td>
<td>IQ</td>
<td>Path</td>
</tr>
<tr>
<td>13</td>
<td>Max. Flight Path Angle</td>
<td>1.0 deg</td>
<td>IQ</td>
<td>Path</td>
</tr>
<tr>
<td>14</td>
<td>Top $\alpha - M$ Boundary</td>
<td>Variable</td>
<td>IQ</td>
<td>Path</td>
</tr>
<tr>
<td>15</td>
<td>Bottom $\alpha - M$ Boundary</td>
<td>Variable</td>
<td>IQ</td>
<td>Path</td>
</tr>
<tr>
<td>16</td>
<td>Max. Bank Angle</td>
<td>85 deg</td>
<td>IQ</td>
<td>Path</td>
</tr>
<tr>
<td>17</td>
<td>Min. Bank Angle</td>
<td>-85 deg</td>
<td>IQ</td>
<td>Path</td>
</tr>
</tbody>
</table>
a particular landing site are determined by systematically probing all the possible EIPs, while keeping the terminal conditions the same.

The discussed approach is implemented using the ASTOS batch-loop functionality, allowing for unattended optimization tasks and thus limiting time wasted as a consequence of user interaction with the software. All the optimization tasks were run on a high-performance Xeon® E5-2687W processor clocked at 3.10 GHz, taking approximately 2 minutes to complete a single optimization task. Note that parallelizing the above strategy for multiple landing sites achieves savings in the total computational time.

IV.C. Trajectory Database Characteristics

In this section, the coverage of the trajectory database is assessed, which refers to the entry-window area from which a particular landing site can be reached at nominal entry conditions and taking into account all of the path constraints. The entry-windows are presented in the form of “coverage maps”, which display a point cloud with all the solution EIPs found in the reachability study. The entry-window is formed by the “area” which such point cloud draws on the map.

For instance, the coverage maps for the Ben Guerir and Perth sites are shown in Figs. 8 and 10. The associated trajectory groundtracks are shown in Figs. 9 and 11, where such groundtracks originate from the EIPs shown in the point cloud. Furthermore, close examination of the entry-windows shows two EIP clusters. On the one hand, the cluster marked by orange crosses represents those EIPs that lay on an ascending leg, in other words, trajectories whose entry heading angle spans the \([0, 90]\) deg range. On the other hand, the cluster marked with blue circles represents those EIPs laying on a descending leg. Note that for all landing sites, one of the clusters is located significantly further west from the landing site, meaning that the trajectories originating from such cluster travel notably further downrange. Figures 8 and 10 also show areas where no EIPs are found, indicating that trajectories originating from such areas would violate at least one of the path constraints or would exceed SPHYNX’s downrange/crossrange capabilities.
As shown in Figs. 12 and 13, SPHYNX displays two types of entries depending on how far the EIP is from the targeted landing site. Note that these trajectories are representative of the whole database, since they exhibit entry profiles that are common to all the targeted sites. If the EIP is close to the landing site, the vehicle performs a regular gliding entry, characterized by a direct shallow descent towards the targeted final altitude. On the contrary, if the EIP is far from the landing site, the vehicle performs a skipping entry characterized by a rebound in the upper atmosphere that leads to a sub-orbital flight followed by a regular gliding entry. By performing a skipping entry, the distance traveled by SPHYNX is significantly increased as shown by Fig. 13, ultimately allowing to reach the targeted landing site. Figure 14 shows that the type of entry is primarily controlled by the bank angle, where a near full-lift-up configuration during the first 400 seconds puts the vehicle in a skipping entry. If a gliding entry is to be performed, a large bank angle magnitude is commanded shortly after the EIP, effectively reducing the vertical component of the lift vector and thus allowing the vehicle to sink into the atmosphere. As discussed in Sec. III.B, the bank-angle direction controls the crossrange error which becomes evident when comparing Figs. 14 and 19. Here, the gliding entries are shown to have opposite bank-angle signs and thus show the vehicle to move in opposite crossrange directions. If the bank angle is negative, the trajectory is shown to curve in the “port” direction, whereas if the bank angle is positive, the trajectory curves in the “starboard” direction. On the other hand, the angle of attack is primarily used to modulate the lift and drag coefficients and thus the dissipation of energy.

Given the feasible entry trajectories found for the available landing sites, full global time-coverage is not guaranteed. This means that a safe landing site cannot be guaranteed irrespective of the epoch at which the de-orbit burn is initiated. With the current sites, approximately 50% global time-coverage is provided, meaning that landing sites can be targeted in 50% of the time. This translates to a maximum in-orbit standby time of approximately 10 hours. In addition, shared entry windows are found, since all sites share over 50% of its entry window with at least one alternative site. Although the ability to reroute to an alternative site largely depends on the total energy available, the results shown in this paper show important re-routing capabilities prior reaching the EIP. Note that any re-routing operation can be achieved by simply selecting a different trajectory subspace within the database. Further details regarding the safety of the current landing-site selection can be found in Ref. 15.

Comparing Figs. 15 and 17 shows that the angle of attack is increased to track the heat-flux constraint, which is particularly evident for the skipping entry. On the other hand, Fig. 16 shows that the aerodynamic load in these four example trajectories is nowhere near the 3.0 g limit. Still, it is relevant to highlight that this load becomes more sensitive to changes in the angle of attack once the vehicle is in the high density lower atmosphere. Finally, Fig. 18 shows that skipping entries lead to an increase in the total heat load due the longer flight-time. Still, such increase is moderate since a significant fraction of the flight occurs outside the atmosphere where the vehicle experiences a heat-flux plateau as a consequence of the low air density.

V. Adaptive Guidance Performance

The goal of this section is to relate the performance results of the abort guidance system discussed in Sec. III to the various trajectory database design parameters.
Figure 12. Altitude profiles for four example trajectories of the database that target the Azores site.

Figure 13. Traveled distance profiles for four example trajectories of the database that target the Azores site.

Figure 14. Bank-angle profiles for four example trajectories of the database that target the Azores site.

Figure 15. Angle-of-attack profiles for four example trajectories of the database that target the Azores site.

Figure 16. Aerodynamic load profiles for four example trajectories of the database that target the Azores site.

Figure 17. Heat-flux profiles for four example trajectories of the database that target the Azores site.

V.A. Performance Impact of the Database Parameter Spacing

By bringing the extremals delimiting the interpolation cell closer to a test EIP, the cell area is reduced and the fidelity of the interpolated solutions is improved, as sketched in Fig. 20. Due to the large coverage area that the database spans, it is unfeasible to recompute the entire database for smaller grid spacings due to the limited computational time. Consequently, the tests performed in this section are restricted to a single arbitrary landing site and EIP, thus serving as a proof-of-concept. The selected EIP is placed at -125 deg longitude and 13 deg geodetic latitude, where the targeted landing site is the Ben Guerir Air Force Base.

The extremals used in this first test and the corresponding database parameter spacings are given in Table 5, where the coarsest spacing is 10 deg and the finest spacing is 0.1 deg.

Figures 21 to 26 show two guided entries, which originate at the same EIP, where the only difference is the
size of grid cell on which the interpolation is performed. The tested entries feed off 10 deg and 1-deg spaced grid cells. Here, the flown trajectory as guided by the abort guidance system is shown together with the reference interpolated trajectory and the supporting extremals. By comparing Figs. 21 and 22, it becomes clear that both the flown trajectory and the interpolated trajectory are more tightly "enclosed" when a 1-deg spacing grid cell is used. As discussed in Sec. III.A, this behavior provides a direct indication that the reference interpolated trajectory obtained using the finer grid spacing will provide a better approximation to the solution of the equations of motion. To support this, Figs. 23 and 24 show the altitude profiles of the two entries discussed here. It becomes clear that the entry, which feeds off a 10 deg cell shows extremals that differ significantly in shape and trajectory length, contrary to the case where a 1-deg spacing is used, where the extremals show similar profiles. Consequently, the quality of the approximation to the solution of the equations of motion will be poor, potentially leading to a trajectory that is not flyable in the first place. This is confirmed by Figs. 25 and 26, which display the angle-of-attack profiles of the two tested entries. It becomes evident that the trajectory feeding of the 10 deg grid cell struggles to approximate the interpolated reference profile, simply because such reference is not compliant with the physics of the problem. On the contrary, the trajectory that feeds off a 1-deg cell is able to approximate the reference angle-of-attack profile from the start of the entry. In summary, Figs. 21 to 26 demonstrate that decreasing the parameter spacing improves the quality of the approximated reference command profiles, potentially leading to a better guidance performance.

Now that the expected trend in guidance performance as a function of the grid spacing has been justified, six 1000-case Monte-Carlo simulation campaigns are performed with the goal of establishing actual performance values. Note that dispersion characteristics used in the Monte-Carlo campaigns are specified in Table 6. Note that the entry geodetic latitude and the entry longitude are dispersed according to a uniform
distribution whose maximum and minimum values are defined by the particular parameter spacing tested. This ensures that EIPs spanning the whole interpolation grid cell are probed and supports the fact that the entry window coverage of a single interpolation cell is proportional to the parameter spacing. Finally, note that the six Monte-Carlo campaigns are fed with the same seed to ensure a fair comparison between the parameter spacings tested.

Table 7 shows significant dispersions in the “landing” accuracy. A trend is revealed, which shows how decreasing the parameter spacing leads to a wider spread of the terminal position dispersions and an increased

<table>
<thead>
<tr>
<th>Parameter Spacing:</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0 deg</td>
</tr>
<tr>
<td>( \tau_{EIP} ) [deg]</td>
</tr>
<tr>
<td>TL Extremal</td>
</tr>
<tr>
<td>-130.0</td>
</tr>
<tr>
<td>-127.5</td>
</tr>
<tr>
<td>-126.0</td>
</tr>
<tr>
<td>-125.5</td>
</tr>
<tr>
<td>-125.05</td>
</tr>
<tr>
<td>-125.05</td>
</tr>
<tr>
<td>( \delta_{EIP} ) [deg]</td>
</tr>
<tr>
<td>18.0</td>
</tr>
<tr>
<td>15.5</td>
</tr>
<tr>
<td>14.0</td>
</tr>
<tr>
<td>13.5</td>
</tr>
<tr>
<td>13.25</td>
</tr>
<tr>
<td>13.05</td>
</tr>
<tr>
<td>( \chi_{EIP} ) [deg]</td>
</tr>
<tr>
<td>40.3</td>
</tr>
<tr>
<td>39.7</td>
</tr>
<tr>
<td>39.4</td>
</tr>
<tr>
<td>39.3</td>
</tr>
<tr>
<td>39.23</td>
</tr>
<tr>
<td>39.2</td>
</tr>
<tr>
<td>TR Extremal</td>
</tr>
<tr>
<td>-120.0</td>
</tr>
<tr>
<td>-122.5</td>
</tr>
<tr>
<td>-124.0</td>
</tr>
<tr>
<td>-124.5</td>
</tr>
<tr>
<td>-124.75</td>
</tr>
<tr>
<td>-124.95</td>
</tr>
<tr>
<td>( \delta_{EIP} ) [deg]</td>
</tr>
<tr>
<td>18.0</td>
</tr>
<tr>
<td>15.5</td>
</tr>
<tr>
<td>14.0</td>
</tr>
<tr>
<td>13.5</td>
</tr>
<tr>
<td>13.25</td>
</tr>
<tr>
<td>13.05</td>
</tr>
<tr>
<td>( \chi_{EIP} ) [deg]</td>
</tr>
<tr>
<td>40.3</td>
</tr>
<tr>
<td>39.7</td>
</tr>
<tr>
<td>39.4</td>
</tr>
<tr>
<td>39.3</td>
</tr>
<tr>
<td>39.23</td>
</tr>
<tr>
<td>39.2</td>
</tr>
<tr>
<td>BL Extremal</td>
</tr>
<tr>
<td>-130.0</td>
</tr>
<tr>
<td>-127.5</td>
</tr>
<tr>
<td>-126.0</td>
</tr>
<tr>
<td>-125.5</td>
</tr>
<tr>
<td>-125.05</td>
</tr>
<tr>
<td>-125.05</td>
</tr>
<tr>
<td>( \delta_{EIP} ) [deg]</td>
</tr>
<tr>
<td>8.0</td>
</tr>
<tr>
<td>10.5</td>
</tr>
<tr>
<td>12.0</td>
</tr>
<tr>
<td>12.5</td>
</tr>
<tr>
<td>12.75</td>
</tr>
<tr>
<td>12.95</td>
</tr>
<tr>
<td>( \chi_{EIP} ) [deg]</td>
</tr>
<tr>
<td>38.4</td>
</tr>
<tr>
<td>38.8</td>
</tr>
<tr>
<td>39.0</td>
</tr>
<tr>
<td>39.1</td>
</tr>
<tr>
<td>39.14</td>
</tr>
<tr>
<td>39.18</td>
</tr>
<tr>
<td>BR Extremal</td>
</tr>
<tr>
<td>-120.0</td>
</tr>
<tr>
<td>-122.5</td>
</tr>
<tr>
<td>-124.0</td>
</tr>
<tr>
<td>-124.5</td>
</tr>
<tr>
<td>-124.75</td>
</tr>
<tr>
<td>-124.95</td>
</tr>
<tr>
<td>( \delta_{EIP} ) [deg]</td>
</tr>
<tr>
<td>8.0</td>
</tr>
<tr>
<td>10.5</td>
</tr>
<tr>
<td>12.0</td>
</tr>
<tr>
<td>12.5</td>
</tr>
<tr>
<td>12.75</td>
</tr>
<tr>
<td>12.95</td>
</tr>
<tr>
<td>( \chi_{EIP} ) [deg]</td>
</tr>
<tr>
<td>38.4</td>
</tr>
<tr>
<td>38.8</td>
</tr>
<tr>
<td>39.0</td>
</tr>
<tr>
<td>39.1</td>
</tr>
<tr>
<td>39.14</td>
</tr>
<tr>
<td>39.18</td>
</tr>
</tbody>
</table>
Figure 25. Angle-of-attack profiles for the test entry with extremals separated 10 deg apart from each other.

Figure 26. Angle-of-attack profiles for the test entry with extremals separated 1 deg apart from each other.

Table 6. Monte-Carlo dispersion settings used to characterize the guidance performance as a function of the parameter spacing.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Dispersion Type</th>
<th>Mean</th>
<th>Min/Max</th>
<th>3σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>EIP Altitude</td>
<td>Gaussian</td>
<td>0 m</td>
<td>150 m</td>
<td></td>
</tr>
<tr>
<td>EIP Longitude</td>
<td>Uniform</td>
<td>±Δp₂/2 deg</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EIP Latitude</td>
<td>Uniform</td>
<td>±Δp₂/2 deg</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EIP Groundspeed</td>
<td>Gaussian</td>
<td>0 m/s</td>
<td>20 m/s</td>
<td></td>
</tr>
<tr>
<td>EIP Flight Path Angle</td>
<td>Gaussian</td>
<td>0 deg</td>
<td>0.1 deg</td>
<td></td>
</tr>
<tr>
<td>EIP Heading Angle</td>
<td>Gaussian</td>
<td>0 deg</td>
<td>0.01 deg</td>
<td></td>
</tr>
<tr>
<td>Aerodynamic Coefficients</td>
<td>Gaussian</td>
<td>0</td>
<td>10 %</td>
<td></td>
</tr>
<tr>
<td>Vehicle Mass</td>
<td>Uniform</td>
<td>±1 %</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Air Density</td>
<td>Gaussian</td>
<td>0</td>
<td>10 %</td>
<td></td>
</tr>
</tbody>
</table>

Table 7. Dispersion properties of the terminal position as a function of parameter spacing, using 190 LD nodes.

<table>
<thead>
<tr>
<th>Parameter Spacing:</th>
<th>10.0 deg</th>
<th>5.0 deg</th>
<th>2.0 deg</th>
<th>1.0 deg</th>
<th>0.5 deg</th>
<th>0.1 deg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean [km]</td>
<td>208.43</td>
<td>82.42</td>
<td>37.38</td>
<td>34.49</td>
<td>19.18</td>
<td>14.58</td>
</tr>
<tr>
<td>Standard Deviation [km]</td>
<td>114.69</td>
<td>45.08</td>
<td>27.66</td>
<td>22.17</td>
<td>20.35</td>
<td>17.50</td>
</tr>
</tbody>
</table>

mean value of such dispersions. Note that a limited number of cases falls significantly further from the landing site as compared to all the other cases of the same Monte-Carlo campaign. It is important to emphasize that these trajectories are removed from the computation of the statistical values given in Table 7, but will be considered in the conclusions drawn in Sec. VI. Either way, the distance-to-target dispersions exceed the 10 km radius established in Sec. II.D, all in all suggesting that a revision of the LQR tracker could lead to performance improvements in this aspect.

Contrary to the distance-to-target dispersions, the guidance algorithm effectively tracks the altitude and groundspeed in almost every case tested, as shown by Fig. 27. Here, three trajectories that differ significantly in altitude-airspeed profile from those in the main group are highlighted in red, which coincide with those trajectories that fall far excessively far from the landing site as discussed above. Such an issue suggests that the gains of the LQR tracker could require some further tuning and possibly a re-design, where direct tracking of the traveled range could provide a better performance. Despite this, the guidance algorithm is capable of bringing the vehicle to the required altitude and Mach number, as summarized in Table 8. Here, a summary of the statistical characteristics of final Mach number $M_f$ and the final vertical speed $V_{v,f}$ is given. It becomes evident that the parameter spacing does not make a significant impact, neither on the final Mach number nor in the final vertical speed, since the mean and the standard deviation of both quantities remain practically unchanged. In fact, the standard deviation of both quantities is relatively small, suggesting robustness to dispersions.

Table 9 summarizes the overshoot characteristics of the aerodynamic-load and heat-flux dispersions for
all the parameter spacings tested, using 190 LD nodes. It is shown that the 10-deg spacing leads to the worst performance in terms of aerodynamic-load, which is violated in a third of the cases tested with a mean overshoot of 3.42 g and a maximum overshoot of 4.11 g. Note that, although the spacings of 5 deg to 1 deg result in a large number of overshoots, the mean and maximum values of these overshoots are significantly lower that for 10 deg and within acceptable bounds. Despite this, it becomes evident that the finest spacings of 0.5 deg and 0.1 deg lead to just three overshoots and are very close to the maximum of allowable value of 3.0 g.

Contrary to the aerodynamic-load overshoots, Table 9 reveals that the heat-flux constraint of 670 kW/m² is complied with in most of the cases. Despite this, some of the tested cases show significantly high maximum heat-flux values, in particular the 10 deg and the 0.5 deg cases. These very high constraint violations correspond to those trajectories that deviate too far from the altitude-groundspeed profile as highlighted above. Finally, it is important to emphasize the large number of heat-flux overshoots that appear for the finest spacing. This is related to the fact that extremals that support the interpolation closely track the heat-flux constraint and consequently any errors in the reference profile could easily lead to the violation

<table>
<thead>
<tr>
<th>$N_{LD} = 190$</th>
<th>10.0 deg</th>
<th>5.0 deg</th>
<th>2.0 deg</th>
<th>1.0 deg</th>
<th>0.5 deg</th>
<th>0.1 deg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean $M_f$ [-]</td>
<td>0.669</td>
<td>0.647</td>
<td>0.664</td>
<td>0.663</td>
<td>0.665</td>
<td>0.666</td>
</tr>
<tr>
<td>Standard Deviation $M_f$ [-]</td>
<td>0.010</td>
<td>0.006</td>
<td>0.006</td>
<td>0.007</td>
<td>0.011</td>
<td>0.014</td>
</tr>
<tr>
<td>Mean $V_{v,f}$ [m/s]</td>
<td>118</td>
<td>106</td>
<td>117</td>
<td>116</td>
<td>117</td>
<td>117</td>
</tr>
<tr>
<td>Standard Deviation $V_{v,f}$ [m/s]</td>
<td>3</td>
<td>7</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$N_{LD} = 190$</th>
<th>10.0 deg</th>
<th>5.0 deg</th>
<th>2.0 deg</th>
<th>1.0 deg</th>
<th>0.5 deg</th>
<th>0.1 deg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nr. $n_{L,max}$</td>
<td>330</td>
<td>236</td>
<td>506</td>
<td>839</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Mean $n_{L,max}$ [-]</td>
<td>3.42</td>
<td>3.03</td>
<td>3.06</td>
<td>3.09</td>
<td>3.00</td>
<td>3.02</td>
</tr>
<tr>
<td>Max. $n_{L,max}$ [-]</td>
<td>4.11</td>
<td>3.12</td>
<td>3.13</td>
<td>3.29</td>
<td>3.01</td>
<td>3.05</td>
</tr>
<tr>
<td>Nr. $q_{max}$</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>56</td>
<td>371</td>
</tr>
<tr>
<td>Mean $q_{max}$ [kW/m²]</td>
<td>754</td>
<td>673</td>
<td>736</td>
<td>693</td>
<td>681</td>
<td>678</td>
</tr>
<tr>
<td>Max. $q_{max}$ [kW/m²]</td>
<td>810</td>
<td>673</td>
<td>753</td>
<td>693</td>
<td>938</td>
<td>749</td>
</tr>
</tbody>
</table>

Figure 27. Altitude-Groundspeed dispersions for a parameter spacing of 2 deg and 190 LD nodes. Marked in red are example trajectories that differ significantly from the main group.
of the constraint. Despite this, the overshoot values are still within acceptable limits and are expected to improve with a revision of the LQR tracker.

In summary, the constraint violations are in line with the expected trend for the performance of the system as a function of the parameter spacing. Coarser spacings lead to reference trajectories that do not approximate the solutions of the equations of motion sufficiently well, thus forcing the guidance system to track a trajectory that is not flyable in the first place.

V.B. Performance Impact of the LD Node Number

All the dispersions discussed so far employed a database sampled on 190 LD nodes. As discussed in Sec. III.A, decreasing the number of nodes can effectively reduce the size of the database even further. The goal of this section is to characterize the impact of the number of LD nodes used on the final dispersions and the compliance of the path constraints. To do so, identical Monte-Carlo campaigns to those presented in the previous section are performed.

To begin with, Tables 10 and 11 show the statistical parameters of the distance-to-target dispersions for 140 LD nodes and 90 LD nodes, respectively. When compared with Table 7, it becomes evident that reducing the number of LD nodes down to 90 does not have a significant impact in the dispersion characteristics of the distance-to-target, where discrepancies of less than 6% are found.

<table>
<thead>
<tr>
<th>Parameter Spacing:</th>
<th>10.0 deg</th>
<th>5.0 deg</th>
<th>2.0 deg</th>
<th>1.0 deg</th>
<th>0.5 deg</th>
<th>0.1 deg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean [km]</td>
<td>222.47</td>
<td>85.51</td>
<td>36.82</td>
<td>34.85</td>
<td>19.43</td>
<td>12.71</td>
</tr>
<tr>
<td>Standard Deviation [km]</td>
<td>129.22</td>
<td>47.28</td>
<td>26.00</td>
<td>22.38</td>
<td>20.52</td>
<td>13.60</td>
</tr>
<tr>
<td>Parameter Spacing:</td>
<td>10.0 deg</td>
<td>5.0 deg</td>
<td>2.0 deg</td>
<td>1.0 deg</td>
<td>0.5 deg</td>
<td>0.1 deg</td>
</tr>
<tr>
<td>Mean [km]</td>
<td>209.91</td>
<td>82.71</td>
<td>35.69</td>
<td>34.47</td>
<td>19.53</td>
<td>13.15</td>
</tr>
<tr>
<td>Standard Deviation [km]</td>
<td>118.15</td>
<td>45.03</td>
<td>23.70</td>
<td>21.26</td>
<td>21.90</td>
<td>13.59</td>
</tr>
</tbody>
</table>

Similarly to the final distance-to-target dispersions, decreasing the number of LD nodes does not have a significant impact in the aerodynamic-load and heat flux overshoot characteristics. In fact, similar trends to those shown for 190 LD nodes are seen. To begin with, a parameter spacing of 10 deg proves again to be too coarse, since the maximum aerodynamic-load of 3.0 g is largely violated. Using 140 LD nodes leads to a maximum aerodynamic-load of 4.59 g, while using 90 LD nodes results in a maximum load of 4.27 g. Furthermore, spacings of 5 deg, 2 deg and 1 deg lead to a large reduction in the excess aerodynamic-load, although still marginally above the limit, taking mean values that range from 3.04 g to 3.10 g. Once again, only grid spacing values of 0.5 deg and 0.1 deg provide a sufficient reduction in the number of trajectories that violate the aerodynamic-load constraint.

In the case of the heat-flux constraint, reducing the number of LD nodes does not change the trends seen for 190 LD nodes. To begin with, the number of heat-flux overshoots is very small for spacing values of 10 deg, 5 deg, 2 deg and 1 deg. In addition, the corresponding mean and maximum values show that the magnitude of the overshoots are within acceptable bounds. Once again, spacing values of 0.5 deg and 0.1 deg lead to a large number of heat-flux constraint overshoots characterized with an unacceptable maximum values. This is again linked to the performance of the LQR tracker, which produces a limited number of trajectories that deviate too far from altitude-grounds speed corridor early in the trajectory. Finally, reducing the parameter spacing leads to trajectories that still comply with the terminal velocity requirements established, where the final vertical speed mean values are well below the maximum allowable speed of 140 m/s and the mean final Mach number is within the subsonic range at 10 km altitude.

Compression of the Trajectory Database

In summary, it becomes clear that reducing the number of LD nodes down to 90 results in no significant change in the terminal point dispersions with respect to 190 LD nodes. Furthermore, no relevant differences
are found in the aerodynamic-load and heat-flux constraint overshoots. In addition, the final Mach number and the final vertical speed are within the requirements established.

All in all, it is evident that 90 LD nodes are sufficient to capture the reentry dynamics. Consequently, the database size could be successfully reduced without posing a significant impact on the performance of the guidance system. A summary of the database sizes for global coverage according to the parameter spacing and LD node number is given in Table 12. Such table shows that the database size ranges from 40.0 GB for a fine spacing of 0.1 deg and no compression, down to 9 MB for a spacing of 10 deg with the maximum compression. In addition, the guidance system developed in this paper was tested for speed and although the LD-to-HD decompression of the database proved to be the most expensive guidance task, it is still within a few milliseconds. All in all, it becomes clear that 90 LD nodes should be used to achieve the largest compression factor, since it poses no significant drawbacks on the guidance performance. Finally, according to Ref. 16 the flash memory of commercially available radiation-hardened on-board computers can amount up to 16 GB, thus proving that the trajectory database can be stored on-board even if the finest grid spacing is used.

Table 12. Estimated database size values for 190, 140 and 90 LD nodes.

<table>
<thead>
<tr>
<th>Parameter Spacing</th>
<th>10.0 deg</th>
<th>5.0 deg</th>
<th>2.0 deg</th>
<th>1.0 deg</th>
<th>0.5 deg</th>
<th>0.1 deg</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Compression</td>
<td>400 MB</td>
<td>800 MB</td>
<td>2.0 GB</td>
<td>4.0 GB</td>
<td>8.0 GB</td>
<td>40.0 GB</td>
</tr>
<tr>
<td>$N_{LD} = 190$</td>
<td>19 MB</td>
<td>38 MB</td>
<td>95 MB</td>
<td>190 MB</td>
<td>380 MB</td>
<td>1.9 GB</td>
</tr>
<tr>
<td>$N_{LD} = 140$</td>
<td>14 MB</td>
<td>28 MB</td>
<td>70 MB</td>
<td>140 MB</td>
<td>280 MB</td>
<td>1.4 GB</td>
</tr>
<tr>
<td>$N_{LD} = 90$</td>
<td>9 MB</td>
<td>18 MB</td>
<td>45 MB</td>
<td>90 MB</td>
<td>180 MB</td>
<td>900 MB</td>
</tr>
</tbody>
</table>

VI. Conclusions and Recommendations

The goal of this paper was to develop and test a dedicated guidance system for abort entry missions, focusing on flights that originate from a low Earth orbit. The abort guidance architecture presented in this paper features an Adaptive Multivariate Pseudospectral Interpolation (AMPI) on-board trajectory planner, which feeds off a large database of optimal trajectories computed off-board. The selection of the appropriate database sub-space is made according to the projected entry conditions at the moment of the abort.

The developed guidance system is capable of planning a reference trajectory in the order of a few milliseconds, proving its capability to run on-board. Furthermore, the system can handle multiple abort scenarios, since the trajectory planner is capable of selecting the most appropriate reference trajectories from the database based on information provided by the navigation system. Thorough testing of the system showed that while the constraint compliance is marginally affected by the density of the trajectory database, the position dispersions at the terminal point are strongly correlated with the chosen density. Results show that increasing the spacing between the stored EIPs in the database increases the mean and spread of the terminal position dispersions. Despite this, the AMPI algorithm employed in the trajectory planner allows for a loss-less compression of the database. Additional Monte Carlo campaigns showed that the compression factor does not have a significant impact on the dispersions at the terminal point. Consequently, manageable database sizes can be achieved even for fine spacings, ranging from only 18 MB for a parameter spacing of 5 deg to 900 MB for a spacing of 0.1 deg.

In summary, the results shown in this paper serve as solid proof that a lightweight autonomous adaptive abort guidance system can be successfully implemented by means of combining a trajectory planner based on trajectory interpolation and a trajectory tracker. Despite this, testing showed that in a limited number of cases the LQR fails to track well the altitude and groundspeed in the early portions of the flight, causing the vehicle to land excessively far from the landing site. Implementing a trajectory tracker where the traveled range is directly tracked could improve the robustness to dispersions. Furthermore, a full system test where the choice for landing site is left free is recommended, where a similar dispersions analysis to the one presented in this paper would have to be made.
Acknowledgements

The authors would like to thank Marco Sagliano from DLR for his kind help deciphering the inner workings of his AMPI algorithm. A special thanks goes to Francesco Cremaschi from ASTOS Solutions for his unattended help and for providing an academic ASTOS license to work with.

References


