

Vega Launchers' Trajectory Optimization Using a Pseudospectral Transcription

G. Di Campli Bayard de Volo^{*}, M. Naeije^{**}, C. Roux[†] and M. Volpi[‡]

^{*}AVIO Spa on behalf of aizoOn Technology Consulting
Strada del Lionetto 6, 10146, Turin, Italy · giuseppe.dicampli@service.avio.com

^{**}Delft University of Technology - Faculty of Aerospace Engineering
Kluyverweg 1, 2629 HS Delft, The Netherlands · m.c.naeije@tudelft.nl

[†]AVIO Spa - Via degli Esplosivi 1, 00034, Colleferro (Rome), Italy
christophe.roux@avio.com · matteo.volpi@avio.com

Abstract

The objective of the presented research is to assess advantages and disadvantages of applying a pseudospectral method to practical trajectory optimization by taking the Vega launcher as test case. In particular, the work focuses on the formulation of the launcher ascent problem, including flight constraints, control variables, initial guesses and cost function. Compared to the results obtained with standard guidance laws, the proposed approach opens up to slightly higher launcher performances, which is paid in terms of problem complexity. On the other hand, the flexibility of this implementation is confirmed by applications to different target orbits and launcher configurations.

1. Introduction

Given the current technological limitations, launch missions are extremely expensive: rockets are huge, being about 100 times more massive than the payload, and most of the times expendable. Consequently, trajectory optimization plays a major role in reducing the launch cost. Universities, companies, agencies and research centres have their own optimization software and continuously try to improve the existing methods. In this context, the main objective of the presented research is to assess advantages and disadvantages of using a pseudospectral approach for practical ascent trajectory optimization by applying it to the Vega launcher.

The software GPOPS¹⁵ (General Pseudospectral OPTimal control Software) is chosen for the transcription of the continuous optimal control problem and SNOPT¹³ (Sparse Nonlinear OPTimizer) for the solution of the associated non-linear programming problem. The launcher ascent problem is formulated considering all the main constraints: angle of attack, pitch rate, aerodynamic loads and heat flux. In addition, the dynamic model includes thrust profiles and drag curves. The control to be optimized is the direction of the thrust and it is formulated in two different ways, allowing to select the most suited for the specific application. The results are compared with the validated trajectory data provided by AVIO, Vega prime contractor.

Disclaimer: this paper is an extract of the MSc Thesis research of the prime author, carried out at the Delft University of Technology⁷.

2. The Optimization Problem

The objective of an optimization problem is to find the *control* that minimizes a certain *cost function*, subject to state equations and boundary constraints, with given initial conditions. Most of the methods used to solve an optimal control problem can be classified as *direct* or *indirect methods*. The reader can refer to the work of Betts³ for a detailed description of these two approaches. For what is presented hereafter, it is important to highlight that direct methods are the most widely used for aerospace applications because of their robustness and simple problem formulation, although indirect methods usually lead to a higher accuracy of the solution.

Pseudospectral Methods

The action of transforming the time-continuous optimal control problem into a discrete one, hereafter *non-linear programming* problem (NLP), is called *transcription* or *collocation*. In the *pseudospectral* methods, which are part of the wider family of *direct collocation* methods, the transcription of the time-continuous optimal control problem to NLP is done by “parameterizing the state and control using global polynomials and collocating the differential algebraic equations using nodes from a Gaussian quadrature” (Garg et al⁹).

There are three main pseudospectral (PS) methods, characterized by different set of collocation points: the *Legendre-Gauss* (LG), the *Legendre-Gauss-Radau* (LGR) and the *Legendre-Gauss-Lobatto* (LGL) points. Using the same notation proposed by Garg et al^{10,11}, being $P_k(\tau)$ the k^{th} -degree Legendre polynomial, the LG, LGR and LGL points are the roots of $P_k(\tau)$, $P_{k-1}(\tau) + P_k(\tau)$ and $\dot{P}_{k-1}(\tau)$ respectively. Depending on the choice of collocation points, the methods are named *Gauss*, *Radau* and *Lobatto* PS methods. These methods have been developed through the years by a small group of researchers, composed by Garg, Patterson, Hager, Rao, Benson and Huntington. Some of them, namely Rao, Benson, Patterson and Huntington, are also the authors of the software GPOPS, which will be presented in a later section.

Going back to the definition of direct and indirect methods, Benson² has demonstrated that, using the Gauss points as collocation points, the Karush-Kuhn-Tucker (KKT) conditions of the direct method are equal to the discretized form of the necessary conditions of the indirect method (scheme in Figure 1). In other words, if the discretization is done at the LG points, the direct and indirect approaches lead to the same solution. This result has been extended to the Radau method by Garg et al⁹, except for the initial costate, while it does not hold for the LGL points. As a consequence, direct methods based on collocation at LG and LGR points offer the advantages of both direct and indirect approaches: a fast and simple problem set-up and an accurate result. This unique feature is the reason for the current interest of the aerospace community in the application of this approach to the trajectory optimization problem.

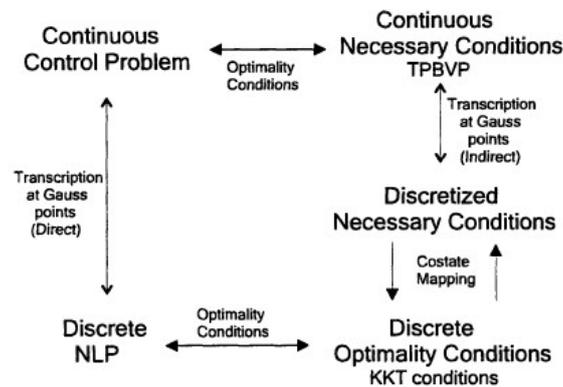


Figure 1: direct and indirect approaches, Gauss PS discretization, Benson²

Optimization Software

The best known software implementing PS methods is GPOPS, a Matlab-based optimization software that uses Gauss or Radau PS transcription of the optimal control problem, developed by Rao et al⁹. The version 3.3 of GPOPS, which was free-of-charge and openly available, has been used for this work. The new upgraded version GPOPS II can be purchased via the official website. The software SNOPT, developed by Gill et al¹² to solve large-scale constrained optimization problems, has been selected to solve the associated NLP problem. A set of MATLAB mex files for SNOPT was already included in the GPOPS SW package, for use with GPOPS only, according to the user manual¹⁵.

3. The Vega Launcher

Vega is a four-stage launch vehicle, with the first three being all-solid motors: P80, Z23 and Z9. The liquid-fueled and restartable upper stage AVUM is capable of precise orbit injection. Details of the four stages are given in Table 1.

The drag coefficient of Vega is obtained from an angle of attack-Mach table. Starting from these values, the interpolating curve is used to determine the C_D in every phase of the flight. In a similar way, thrust, mass flow and specific impulse profiles were obtained from tables through spline interpolation.

Table 1: Vega stages specifics (credits: AVIO website)

Name	Unit	P80	Z23	Z9	AVUM
Inert Mass	[ton]	10	2.8	1.4	0.56
Propellant Mass	[ton]	88	24	10.5	0.57
Thrust	[kN]	2200	1122	314	2.42
Specific Impulse	[s]	280	287	295	314.6
Burn time	[s]	110	77	117	723

The P/L fairing, which protects the payload from aerodynamic, thermal and acoustic environment during atmospheric flight. Since it adds inert mass to the launcher, it is jettisoned as soon as possible to increase the performance. The two separated halves then re-enter the atmosphere and splash in the Ocean.

3.1 Mission Profile

As reported in the *Vega User's Manual*,¹ a typical flight can be divided in three parts: in the first part P80, Zefiro 23, Zefiro 9 and AVUM burn in succession; in the second, orbital maneuvers and payload injection are performed; in the third, AVUM is disposed. The number of AVUM ignitions depends on the target orbit: for example, it is not possible to reach a circular polar orbit at 800 km altitude with a single boost, whereas for certain low-altitude orbits only one is necessary. A typical ascent trajectory with two AVUM boosts is showed in Figure 2.



Figure 2: Vega typical SSO mission (credits: Arianespace)

The ascent trajectory is optimized for each mission, taking into account specific constraints and requirements. However, four common phases can be distinguished: 1st stage vertical ascent, pitch-over and gravity turn flight; 2nd stage short coast and powered flight; 3rd stage short coast and powered flight; 4th stage orbit raising and plane changes with two or more boosts; de-orbit manoeuvre, in accordance with the regulations on space debris.

As it will be shown later, the first two stages, P80 and Z23, once jettisoned, re-enter and splash in the Atlantic Ocean, whereas AVUM performs a controlled re-entry after payload injection. On the other hand, Z9 has sub-orbital velocity at burnout, requiring special precautions to ensure a safe impact. It is also important to note that the most critical phases

VEGA TRAJECTORY OPTIMIZATION

of the mission, like payload separation and boosts, are carried out within the visibility of one of the ground stations. In this way, it is possible to monitor in real time all the events, orbital parameters and separation conditions. Third stage impact point and ground station coverage constitute two practical constraints to be considered in mission design and trajectory optimization.

Vega reaches the highest performance, in terms of payload mass, with an elliptical, near-equatorial orbit: according to the User's Manual¹, it can deliver up to 1.9 tons to a 1500x200 km altitude elliptical orbit, with 5.4 deg inclination. However, the design reference orbit is a polar orbit of 700 km altitude: in this case Vega can deliver about 1.4 tons.

4. Problem Formulation

In ascent trajectory optimization, the parameters to be minimized can be either the propellant mass consumption or the flight time, or a combination of the two. On the other hand, one could decide to maximize the payload mass. Given the nature of the Vega launcher, flight time minimization was not considered. The cost function is then defined as:

$$J = -m^{(P)}(t_f) \quad (1)$$

where $m^{(P)}$ is the total mass (propellant, inert and payload) at last phase P and t_f is the final time, i.e. the moment of payload injection into orbit. The minus sign means that the final mass has to be maximized.

The so-formulated cost function is applicable to either payload or leftover propellant maximization: in both cases, the final objective is to obtain the greatest mass possible at mission end. The difference stands in the partition of this mass: in fact, when maximizing the payload mass, all the propellant is burned to lift the heaviest payload. In that case, the final mass is composed by payload, inert and some propellant reserve (if considered). On the other hand, when maximizing the leftover propellant, the payload mass is given (fixed) and the final mass is composed by some *extra* propellant as well.

Depending on the application, one of the two methods is preferred: for example, when defining the performances of a new launcher, the payload mass is maximized; on the other hand, propellant reserve maximization is adopted in real-life trajectory design, in which payload mass is defined by the customer and the design objective is to spare the greatest amount of propellant to counteract undesired effects and ensure mission success.

4.1 Dynamics Equations

The motion of a launcher is fully described by six variables, three for the position and three for the attitude, meaning that the motion has six degrees of freedom (6DOF). However, in trajectory optimization only the 3DOF of position are considered. Still, the attitude of the launcher is determined by the optimal control, which is a perfect one: there are neither transients nor control errors.

Seven equations of motion are needed to solve for the seven state variables, three for the position, three for the velocity and one for the mass:

$$\begin{aligned} \dot{\vec{r}} &= \vec{v} \\ \dot{\vec{v}} &= \frac{\vec{F}_g}{m} + \frac{T}{m}\hat{u} + \frac{\vec{D}}{m} \\ \dot{m} &= -\frac{T}{I_{sp}g_0} \end{aligned} \quad (2)$$

where $\vec{r} = [r_x \ r_y \ r_z]^T$ is the position vector, $\vec{v} = [v_x \ v_y \ v_z]^T$ the velocity vector and m the total mass. The unit vector \hat{u} defines the direction of the thrust. Since the latter is assumed to be always directed along the roll axis (as in Fig. 3), the unit vector \hat{u} defines also the attitude of the vehicle. Referring to the same figure, angle of attack and flight path angle are defined as:

$$\alpha = \arccos\left(\frac{\vec{v}_{air} \cdot \hat{u}}{\|\vec{v}_{air}\| \cdot \|\hat{u}\|}\right) = \arccos\left(\frac{v_{air,x}u_x + v_{air,y}u_y + v_{air,z}u_z}{\|\vec{v}_{air}\|}\right) \quad (3)$$

$$\gamma = \arccos\left(\frac{\vec{v} \cdot \vec{r}}{\|\vec{v}\| \cdot \|\vec{r}\|}\right) = \arccos\left(\frac{v_x r_x + v_y r_y + v_z r_z}{\|\vec{v}_{in}\| \cdot \|\vec{r}\|}\right) \quad (4)$$

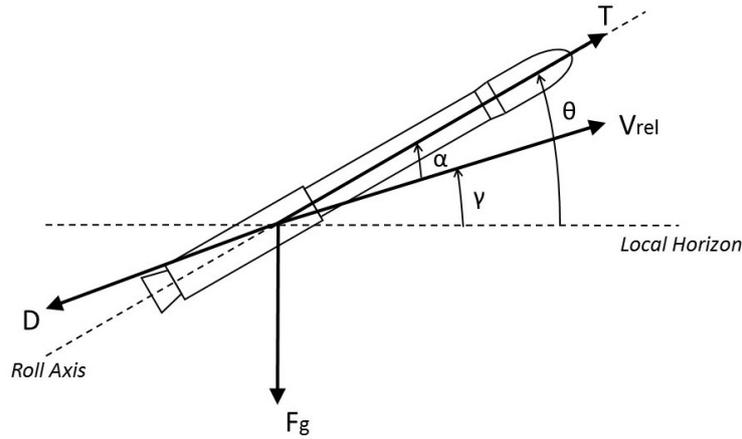


Figure 3: 2D sketch of pitch, flight path angle and angle of attack

Gravity. The adopted gravity model comprises the central-body and J_2 terms only, which is common practice in (preliminary) trajectory optimization:

$$\vec{F}_g = -m \frac{\mu}{r^3} \vec{r} + m \vec{f}_{J_2}(\vec{r}) = \begin{bmatrix} -m \frac{\mu}{r^3} r_x - \frac{3}{2} \mu J_2 \frac{R^2}{r^5} r_x (1 - 5 \frac{r_z^2}{r^2}) \\ -m \frac{\mu}{r^3} r_y - \frac{3}{2} \mu J_2 \frac{R^2}{r^5} r_y (1 - 5 \frac{r_z^2}{r^2}) \\ -m \frac{\mu}{r^3} r_z - \frac{3}{2} \mu J_2 \frac{R^2}{r^5} r_z (3 - 5 \frac{r_z^2}{r^2}) \end{bmatrix} \quad (5)$$

Rocket Thrust can be expressed as:

$$\vec{T} = [m \cdot v_e + (p_e - p_a) \cdot A_e] \hat{u} = [T_{vac} - p_a A_e] \hat{u} \quad (6)$$

where $T_{vac} = T_{vac}(t)$ is the vacuum thrust, $p_a = p_a(h)$ is the atmospheric pressure at a certain altitude and A_e the nozzle area at exit. The term $[-p_a A_e]$, called pressure loss, is negligible for flight at high altitudes, but it substantially reduces the total thrust at lift-off. Vacuum thrust is taken from tables in case of the SRMs and is assumed constant for the liquid upper stage.

Drag. Being the lift quite small in real-life Vega flights, it has been neglected in the simulations. Therefore, the only aerodynamic force considered is the drag:

$$\vec{D} = -\frac{1}{2} \rho C_D S |\vec{v}_{air}| \vec{v}_{air} \quad (7)$$

where $\rho = \rho(h)$ is the atmospheric density and depends on the chosen atmospheric model (exponential, CIRA-2012⁶); $C_D = C_D(Ma)$ is the drag coefficient, taken from the α -Ma tables; S is the reference surface area; \vec{v}_{air} is the air speed, which is assumed to be equal to the local relative speed \vec{v}_{rel} , by neglecting the wind term

$$\vec{v}_{air} = \vec{v}_{in} - \vec{\omega}_E \times \vec{r} - \vec{v}_w \approx \vec{v}_{in} - \vec{\omega}_E \times \vec{r} = \vec{v}_{rel} \quad (8)$$

In the equation above, the atmosphere is assumed co-rotating with the Earth as a rigid body.

4.2 Flight Phases

The launch ascent trajectory optimization is of the multi-phase type, primarily because of the discrete mass change at stage separations. Furthermore, several flight phases can be distinguished, based on the specific dynamics and control law: *vertical lift-off*, *pitch-over*, *pitch-constant* and *gravity turn*. In addition to those, *coast phases*, *stages re-entry* and final *de-orbiting* have been taken into account in this research.

Vertical Lift-off. Right after the first stage ignition, most of the launch vehicles follow a vertical trajectory. This phase is obliged by launch pad safety regulations: since the engine hot plumes could damage the ground facilities, the launcher has to fly vertically until a certain safe altitude is reached, whence it can start steering the trajectory. Vertical

VEGA TRAJECTORY OPTIMIZATION

flight is characterized by a 90 deg pitch angle, whereas yaw is not defined.

Pitch-Over. The objective of a launch vehicle is to gain orbital speed. To do that, rockets start flying horizontally as soon as possible, thus reducing gravitational losses. Clearly, vertical flight is energetically inefficient because large gravity losses occur. However, sometimes the optimal trajectory starts to steer a bit after the constraint of vertical flight becomes inactive: in fact, even if vertical ascent is locally inefficient because of gravity losses, gaining altitude and speed can be beneficial for the subsequent pitch-over, thus increasing the overall performance. During pitch-over, pitch decreases from 90 deg to a certain value, with a rate that is usually constrained by conditions of rocket stability and controllability.

Pitch-Constant. During pitch-over, the angle of attack increases up to a maximum at phase end. In order to reduce the angle of attack, a pitch constant phase comes after pitch-over: during this short phase, thrust points at a fixed direction, until the relative velocity vector aligns to it (see again Figure 3). Once thrust and relative velocity are aligned, gravity turn starts.

Gravity Turn is a particular flight condition during which the angle of attack is zero. It allows to reduce the aerodynamic loads, while gravity steers the trajectory downwards, as explained by Cornelisse et al⁴. In reality, angle of attack will always differ from zero, for example because of the presence of wind. Therefore, it is important to design the trajectory with the lowest angle of attack possible, in order not to exceed the aerodynamic loads constraints when crosswind occurs. On the optimality of the gravity turn manoeuvre, some considerations will be drawn from the numerical results.

Coast Phases occur when the engine is off, for example after stage separation or between two consecutive burns of the same stage. Coast can be fundamental to fulfill trajectory constraints, like visibility and dry stages impact point, and can even improve launcher performances. Note that stage separations are assumed instantaneous and not causing any changes in velocity due to the separation systems.

Dry Stages Re-entry is always included in the launcher mission analysis: safety regulations impose that impact points must be in uninhabited areas, like oceans or desert zones. Moreover, the footprint, i.e. the zone with highest probability of impact, has to be small and contained in the selected area.

For some launchers these constraints are more stringent than for others, like in the case of Vega: due to the fact that the first three stages are solid-propelled, it is impossible to shut down the engine when needed and the residual thrust is often unknown. In addition, Vega's third stage Z9 follows a suborbital trajectory that is tough to predict with precision. As a consequence, the problem of dry stages impact becomes one of the main design drivers.

De-Orbiting. Current regulations on space debris mitigation oblige a safe disposal of each rocket stage, together with payload fairing and upper stage. The latter needs additional maneuvering to de-orbit and burn into the atmosphere. Therefore, a certain percentage of propellant is allotted to de-orbit the upper stage. According to disposal requirements, the target perigee altitude after de-orbit burn has to be lower than -50 km.

4.3 Control Variables

An application of GPOPS to the Delta-III launcher trajectory optimization is fully described in the user manual¹⁵. The authors use three control variables, i.e. the x-y-z ECI coordinates, to define the direction of the thrust:

$$\vec{u} = [u_x \ u_y \ u_z]^T \quad (9)$$

with the additional constraint on the module, to ensure \vec{u} being an unit vector:

$$\|\vec{u}\| = 1 \Rightarrow \vec{u} = \hat{u} \quad (10)$$

This approach, adopted by several authors like Coskun⁵ and Gabrielli⁸, was also the starting point for the presented research. However, after a deeper analysis it was clear that the three degrees of freedom represented by the variables $\{u_x, u_y, u_z\}$, reduce to only two because of the constraint on the module. In other words, it is possible to define a direction with two variables only, i.e. two angles, like *pitch* and *yaw*.

To do that, a Body-Fixed (BF) reference frames was introduced, together with the relative transformation matrix. In fact, the numerical integration is performed in the ECI reference frame with $[u_x \ u_y \ u_z]^T$, whatever the choice of control

variables is. Therefore, the transformation from $[\theta \ \psi]^T$ to $[u_x \ u_y \ u_z]^T$ was obtained by applying the $A_{BF \rightarrow ECI}$ rotation matrix:

$$\hat{u}_{ECI} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = \begin{bmatrix} -\sin lat \cos Az \cos \theta \cos \psi + \cos lat \sin \theta - \sin lat \sin Az \cos \theta \sin \psi \\ \sin Az \cos \theta \cos \psi - \cos Az \cos \theta \sin \psi \\ \cos lat \cos Az \cos \theta \cos \psi + \sin lat \sin \theta + \cos lat \sin Az \cos \theta \sin \psi \end{bmatrix} \quad (11)$$

where lat is the latitude and Az the launch Azimuth. It has been experienced that the choice of pitch and yaw is convenient for the computational speed and for the simple use, specially in the set-up of guesses and constraints.

4.4 Constraints

The constraints can be divided in *limit*, *path*, *event* and *linkage constraints*, although the dynamics equations in Eq. (2) are also considered as *dynamic constraints* by some authors.

Limit constraints define maximum and minimum values of position, velocity and control at begin, at end and during each phase. These limits are reasonably large, so that no possible optimal solution is excluded a priori. In addition, the position at liftoff is determined by latitude, longitude and altitude of the launch pad:

$$\vec{r}_0 = \begin{bmatrix} r_{x,0} \\ r_{y,0} \\ r_{z,0} \end{bmatrix} = \begin{bmatrix} (R_E + h_0) \cdot \cos(lat_0) \cdot \cos(lon_0) \\ (R_E + h_0) \cdot \cos(lat_0) \cdot \sin(lon_0) \\ (R_E + h_0) \cdot \sin(lat_0) \end{bmatrix} \quad (12)$$

where the coordinates of Kourou, French Guiana, are $lat_0 \approx 5.16^\circ$, $lon_0 \approx -52.64^\circ$ and $h_0 \approx 0$ m (assumption). Earth rotation provides also a kick-off inertial velocity, which is expressed as:

$$\vec{v}_0 = \begin{bmatrix} v_{x,0} \\ v_{y,0} \\ v_{z,0} \end{bmatrix} = \begin{bmatrix} 0 & -\omega_E & 0 \\ +\omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{x,0} \\ r_{y,0} \\ r_{z,0} \end{bmatrix} = \begin{bmatrix} -\omega_E \cdot r_{y,0} \\ +\omega_E \cdot r_{x,0} \\ 0 \end{bmatrix} \quad (13)$$

In order to leave some degrees of freedom to the optimizer and make the process more robust, the desired conditions at orbit injection are defined as *event constraints*.

Path Constraints. The main inequality and equality path constraints relevant to the ascent trajectory are described hereafter:

1. *Pitch* is constrained during vertical flight as

$$\theta^{(1)}(t_k) = 90^\circ \quad (14)$$

where ⁽¹⁾ stands for the first phase.

2. *Pitch Rate* is constrained during pitch-over maneuver as

$$-3.5^\circ/s \leq \dot{\theta}^{(2)}(t_k) \leq +3.5^\circ/s \quad (15)$$

3. *Angle of Attack* should be as small as possible during atmospheric flight (early phases). However, in this work small angles are accepted:

$$-5^\circ \leq \alpha^{(p)}(t_k) \leq +5^\circ \quad (16)$$

In exo-atmospheric flight the angle of attack is left free.

4. *Dynamic Pressure* is constrained, although the propellant grain of the first stages is specifically designed to keep it as low as possible

$$q^{(p)}(t_k) \leq 35 \text{ kPa} \quad (17)$$

5. The by-product of angle of attack and dynamic pressure, $q \cdot \alpha$, is an indicator of aerodynamic loads acting on the launcher. Therefore, it is common practice to set a limit to this value, rather than to the dynamic pressure itself. For Vega acceptable values are:

$$-40 \text{ kPa}^\circ \leq q^{(p)}(t_k) \cdot \alpha^{(p)}(t_k) \leq +40 \text{ kPa}^\circ \quad (18)$$

6. *Heat Flux* is constrained in order to protect the payload after fairing separation:

$$\dot{Q}^{(p)}(t_k) = \frac{1}{2} \rho v_{air}^3 \leq 900 \text{ W/m}^2 \quad (19)$$

VEGA TRAJECTORY OPTIMIZATION

7. *Visibility* of the ground stations is mandatory during critical ascent phases, like stage separation and burn arcs. Being the visibility a bit out of the scope of the research, this constraint is left to further studies.

Event Constraints apply to state and control variables at start or end of a phase. For example, orbital parameters at payload injection are set as event constraints:

$$\begin{aligned} h_p^{(P)}(t_f) &= h_{p,target} \\ h_a^{(P)}(t_f) &= h_{a,target} \\ i^{(P)}(t_f) &= i_{target} \end{aligned} \quad (20)$$

where h_p is the perigee altitude, h_a the apogee altitude, i the inclination and P the last phase. Note that right ascension of the ascending node Ω , argument of perigee ω and true anomaly θ are left free, so that they will result from the optimization. Event constraints are used to set the minimum altitude at the end of the vertical ascent and the latitude and longitude of the dry stages impact point.

Linkage constraints define the discrete changes of state variables from a phase p to a phase $p + 1$

$$\begin{bmatrix} \vec{r}^{(p+1)} \\ \vec{v}^{(p+1)} \\ m^{(p+1)} \end{bmatrix} - \begin{bmatrix} \vec{r}^{(p)} \\ \vec{v}^{(p)} \\ m^{(p)} \end{bmatrix} = \begin{bmatrix} \Delta \vec{r}^{(p)} \\ \Delta \vec{v}^{(p)} \\ \Delta m^{(p)} \end{bmatrix} \quad (21)$$

where p indicates also the number of linkage, which is numbered like the first phase of the pair. This formulation is particularly useful in defining the discrete mass change at stage separation. The linkage constraint for a solid rocket stage separation is:

$$\begin{bmatrix} \Delta \vec{r}^{(p)} \\ \Delta \vec{v}^{(p)} \\ \Delta m^{(p)} \end{bmatrix} = \begin{bmatrix} \vec{0}_{3 \times 1} \\ \vec{0}_{3 \times 1} \\ -(m_{dry} + m_{res}) \end{bmatrix} \quad (22)$$

which means that position and velocity remain the same (assumption of ideal separation), whereas the stage dry mass and some residual propellant are jettisoned. The same expression can be used at fairing separation, substituting the term at the last row with the term $-m_{fairing}$.

Initial Guesses. In general, a good initial guess is beneficial for convergence and computational speed. However, except for the mass and the epoch of ignition/cut-off of the SRMs, which are quite straightforward, it is usually hard to predict the values that a certain variable will take on. In this cases a rough guess is accepted. Once a first solution is obtained, this can be used as initial guess for next loops, thus saving CPU time.

5. Software & Method Validation

In this Section, three different validation approaches are presented, each with a specific scope: first, an orbital transfer with plane change is optimized, allowing to compare the numerical results with analytical computations, thus verifying reliability and accuracy of the solution; second, an ideal two-stage-to-orbit vehicle is considered to cross-check the results obtained with GPOPS/SNOPT (hereafter named GP/SN) and those taken from an in-house software; third, the same trajectory is used to test the accuracy of the discretization method with a procedure taken by literature.

5.1 Orbital Transfer

A simple orbital transfer is considered: starting from a circular orbit, an Hohmann-like transfer is performed to rise the altitude from 200 to 400 km and change the inclination from 0 to 5 deg. The remaining orbital elements are left free. The spacecraft consists of Vega's upper stage AVUM and a payload of 200 kg.

In the ideal transfer, two impulses are executed at the ascending and descending nodes, respectively: the first boost changes inclination and injects the spacecraft in an elliptical transfer orbit; the second one "circularizes" the orbit and completes the change of inclination. The objective is to find the value of the transfer orbit inclination i_t^* that minimizes the total ΔV :

$$\begin{aligned} \Delta V_{tot} &= \Delta V_1 + \Delta V_2 = \\ &= \sqrt{V_{c1}^2 + V_{tp}^2 - 2V_{c1}V_{tp} \cos(i_t - i_1)} + \sqrt{V_{c2}^2 + V_{ta}^2 - 2V_{c2}V_{ta} \cos(i_2 - i_t)} \end{aligned} \quad (23)$$

where, following the notation of Wakker¹⁷, the subscripts 1, 2, t , a , p stand for initial orbit, final orbit, transfer orbit, apogee and perigee, respectively. The minimum of the function $\Delta V_{tot}(i_t)$ cannot be found in a closed-analytical form. Therefore, the MATLAB function `fminsearch` was used. The resulting optimal transfer orbit inclination is $i_t^* \approx 1.68^\circ$, meaning that most of the plane change is executed with the second boost. Starting with this value of inclination, ΔV_1 and ΔV_2 are uniquely determined. Moreover, propellant consumption can be estimated by applying Tsiolkovsky equation, together with burn time and coast duration.

Following the problem formulation already presented, GPOPS/SNOPT (hereafter GP/SN) is used to optimize the orbital transfer. In particular, a four-phase optimization is considered: coast-boost-coast-boost. In Table 2 the analytical results are compared to those obtained with GP/SN: optimal transfer orbit inclination differs of 0.22° , which is very good considering that the associated error on the $\Delta V_{tot}(i_t^*)$ is only 0.3 m/s; coast time is longer in the GP/SN calculation because it takes into account the duration of the boosts, which last about 5 minutes in total; for the same reason, the manoeuvres start before and finish after node passages, resulting in a +3% propellant consumption.

Table 2: Orbital Transfer Results Comparison

Symbol	Unit	Theoretical	GP/SN
i_t^*	[deg]	1.68	1.44
$\Delta V_{tot}(i_t^*)$	[m/s]	682.8	683.1
t_{b1}	[s]	131	101
t_{b2}	[s]	170	209
t_{coast}	[min]	45.3	39.6
M_{prop}	[kg]	283	292

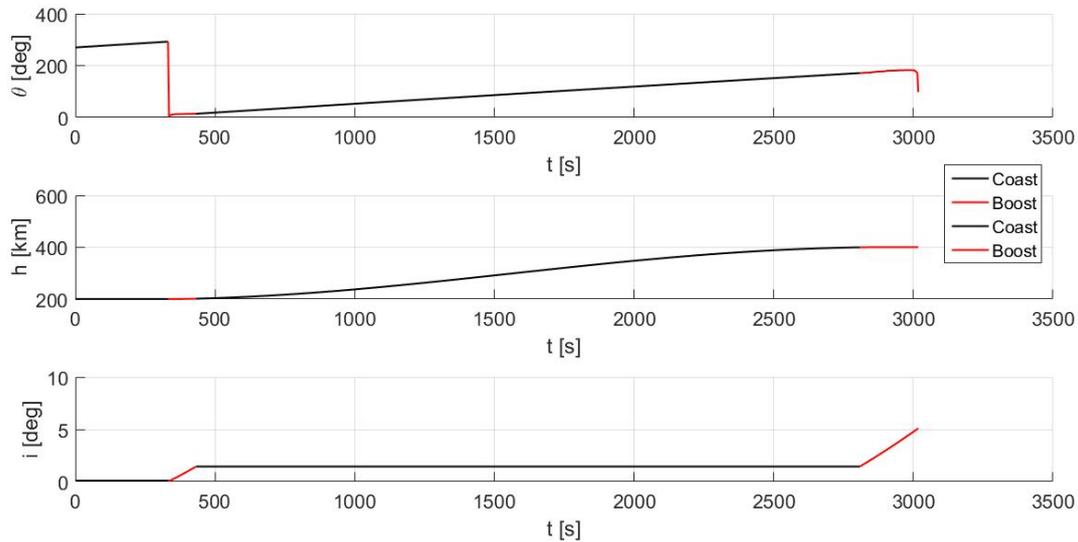


Figure 4: Top-down: true anomaly, altitude and inclination in time

Plots in Figure 4 are particularly useful in verifying the optimality of the solution: the first shows that the boosts are executed at perigee and apogee of the transfer orbit, i.e. 0° and 180° of true anomaly. The third confirms that change in inclination is mainly achieved with the second boost. It is important to note that the computation time, starting with a rough initial guess, ranges from 5 to 15 minutes. Once the first result is obtained, the latter can be used as initial guess to reduce the CPU time down to tens of seconds.

5.2 Two-Stage-To-Orbit

The PSO method has also been tested with a dummy two-stage-to-orbit vehicle, with a solid-propelled first stage and a liquid upper stage. Selected target orbit is a Sun-Synchronous Orbit (SSO) at 400 km altitude and 97° inclination. The ascent trajectory is divided into 6 phases: vertical lift-off, first stage powered flight, coast, upper stage first boost, coast, upper stage second boost. The resulting optimal trajectory is then compared to the one computed with AVIO in-house optimization tool. Looking at the Table 3, it is clear that the trajectories computed by the two tools are similar, above all in terms of payload mass and conditions at injection. AVIO performance is 2.8 kg higher, but in the simulation with GP/SN leftover propellant is 2.1 kg. Given also the (very) small difference of about 1.3 minutes in the mission duration, it can be concluded that the overall launch performance is absolutely equal.

Table 3: comparison of numerical results for a two-stage-to-orbit, 400 km SSO

Name	Unit	GP/SN	AVIO
Payload Mass	[kg]	809.2	812.0
Prop. for De-Orbit	[kg]	52.9	49.7
Leftover Propellant	[kg]	2.1	0.0
Final Altitude	[km]	400.0	400.0
Final Inclination	[deg]	97.03	97.03
Final RAAN	[deg]	1.7	1.7
Final Argument of Lat.	[deg]	191.8	186.5
Mission Duration	[min]	50.5	49.2
CPU Time	[min]	8-10	2-3

The optimization methods used, i.e. pseudospectral in GP/SN and differential evolution in AVIO software, are essentially different. In addition, the guidance strategy adopted by the two software is also different, since AVIO SW makes use of well-known guidance laws during certain parts of the flight: for example, assuming a linear pitch law during pitch-over, only slope and duration have to be optimized. On the contrary, the control variables, namely pitch and yaw, are optimized at each collocation point in GP/SN. Clearly, such a method is more computationally expensive, as confirmed by the simulations, but at the same time it allows to find new solutions outside the predetermined guidance law, often resulting in a smoother control.

5.3 Numerical Integration

Taking the same two-stage-two-orbit configuration, the accuracy of the discretization is verified, as suggested by many authors in case of pseudospectral methods: Josselyn et al¹⁴ and Ross et al¹⁶ have proposed to check the possibility of obtaining the same optimal trajectory by propagating the state (off-line) using the control history generated by the software, thus confirming the accuracy of the discretization at the LG/LGR collocation points.

For the simulations, a 300 km SSO target orbit is selected, with fixed payload of 850 kg. The validation procedure is described step by step hereafter:

1. the optimal pitch and yaw profiles computed by GP/SN are interpolated;
2. the dynamic model (J2, drag, atmosphere, etc.) used in GP/SN is copied and tailored for direct integration;
3. starting with the initial state vector at launch pad, dynamics equations are integrated via the MATLAB built-in ode45 function.

The final conditions are compared in Table 4 on the left, while maximum errors are reported on the right. Being the first ones very similar and the second ones negligible, the accuracy of the adopted discretization is confirmed.

6. Vega Trajectory

The selected target orbit is a SSO at 785 km altitude. The SSO mission is often taken as reference in describing the launcher performances, being the standard for Earth observation satellites, which constitute Vega's main target. For these missions payload mass is fixed, according to the existing performance maps, and the leftover propellant is

Table 4: final conditions (left) and max errors (right), GP/SN vs ode45

Name	Unit	GP/SN	ode45	Name	Unit	Max Error
Final Altitude	[km]	300.8	300.9	Altitude	[m]	13.5
Final Inclination	[deg]	96.68	96.68	Velocity	[cm/s]	14
Final RAAN	[deg]	1.61	1.61	Mass	[g]	1.8
Final Argument of Lat.	[deg]	104.5	104.5			

maximized. This approach was preferred in order to make the example as similar as possible to practical (i.e. industrial) trajectory optimization.

6.1 Sun-Synchronous Orbit - 785 km

The SSO at 785 km altitude was the target orbit of Sentinel-2A (lift-off mass 1130 kg), launched with Vega flight VV05. Trajectory is divided in 8 phases, each with a particular constraint. 16 collocation points are used for the shortest phases (vertical flight, first coast, second AVUM boost), 31 for the others. Knowing the number of phases P , collocation points $N_{points}^{(p)}$, state and control variables (N_{state} and $N_{control}$ respectively), it is possible to compute the total number of nonlinear variables as

$$N_{variables} = \sum_{p=1}^P N_{points}^{(p)} \times (N_{state} + N_{control}) = (16 \times 3 + 31 \times 5) \times (7 + 2) = 1827 \quad (24)$$

which gives an idea of the dimension of the optimization problem. The results for the fully-constrained case are reported in Tables 5 and 6: using the same payload mass of Sentinel-2A, the leftover propellant is about 6 kg.

Table 5: Performances (left) and mission parameters (right) - Vega - SSO 785 km

Name	Unit	Value	Name	Unit	Value
Payload Mass	[kg]	1130	Final Altitude	[km]	785
Prop. for De-Orbit	[kg]	62.2	Final Inclination	[deg]	98.54
Leftover Propellant	[kg]	5.9	Final RAAN	[deg]	1.69
CPU Time (simulation #1)	[s]	847.3	Final Argument of Lat.	[deg]	184.56
CPU Time (simulation #2)	[s]	5.6	Mission Duration	[min]	50.8

Taking a look at the Figures, the following conclusions can be drawn: first coast lasts 72 s, which is around 10 times the minimum duration, allowing the first AVUM boost to be executed at the best orbital position (Figure 5); constraints of maximum pitch rate, $q \cdot \alpha$ and heat flux are active (Figure 7), whereas angle of attack constraint is inactive (Figure 6); the most demanding constraint is the maximum heat flux after fairing separation (Figure 7 right), due to the fact that in this work it is assumed that fairing is jettisoned together with dry Z23. Therefore, the entire flight of Z9 has to be high in the atmosphere to keep the heat flux low. As a result, the trajectory becomes much steeper and Z9 ignites at 126 km altitude.

6.2 Dry Stages Re-Entry

Figure 8 shows the impact points of P80, Z23 and Z9. Clearly, due to the strategic position of the Guiana Space Center, the first two stages splash in the Atlantic Ocean for all the trajectories heading North or East (SSO, polar, equatorial orbits). Unfortunately, this is not true for the Z9 stage: in fact, because of the altitude and speed at stage burnout, Z9 impact point is thousands of miles away from the launch pad and, most importantly, it is hard to predict with precision. Therefore, in order to be sure that Z9 will eventually re-enter over Arctic Sea instead of Canada, some precaution is needed.

Table 6: flight phases - Vega - SSO at 785 km

Phase	Description	Time (duration) [s]	Alt. [km]	Constraints
1	Vertical Ascent	0 - 3.6 (3.6)	0 - 0.049	$\theta = 90^\circ$ $h(t_f) \geq 45 \text{ m}$
2	P80 Powered Flight	3.6 - 122.8 (119.2)	0.049 - 64.6	$-3.5^\circ/\text{s} \leq \dot{\theta} \leq 3.5^\circ/\text{s}$ $\alpha \leq 5^\circ$ $q \cdot \alpha \leq 40 \text{ kPa}^\circ$
3	Z23 Powered Flight	122.8 - 197.9 (75.0)	64.6 - 126.2	
4	Z9 Powered Flight	197.9 - 312.1 (114.2)	126.2 - 183.0	$\dot{Q} \leq 900 \text{ W/m}^2$ $75^\circ \leq \text{lat}_{\text{impact}} \leq 90^\circ$
5	AVUM 1st Coast	312.1 - 384.8 (72.8)	183.0 - 205.5	$t_f - t_0 \geq 7 \text{ s}$
6	AVUM 1st Boost	384.8 - 747.2 (362.4)	205.5 - 281.6	
7	AVUM 2nd Coast	747.2 - 2946.6 (2199.4)	281.6 - 784.6	
8	AVUM 2nd Boost	2946.6 - 3050.5 (103.9)	784.6 - 785.1	$h_p(t_f) = 784 \text{ km}$ $h_a(t_f) = 786 \text{ km}$ $i(t_f) = 98.54^\circ$

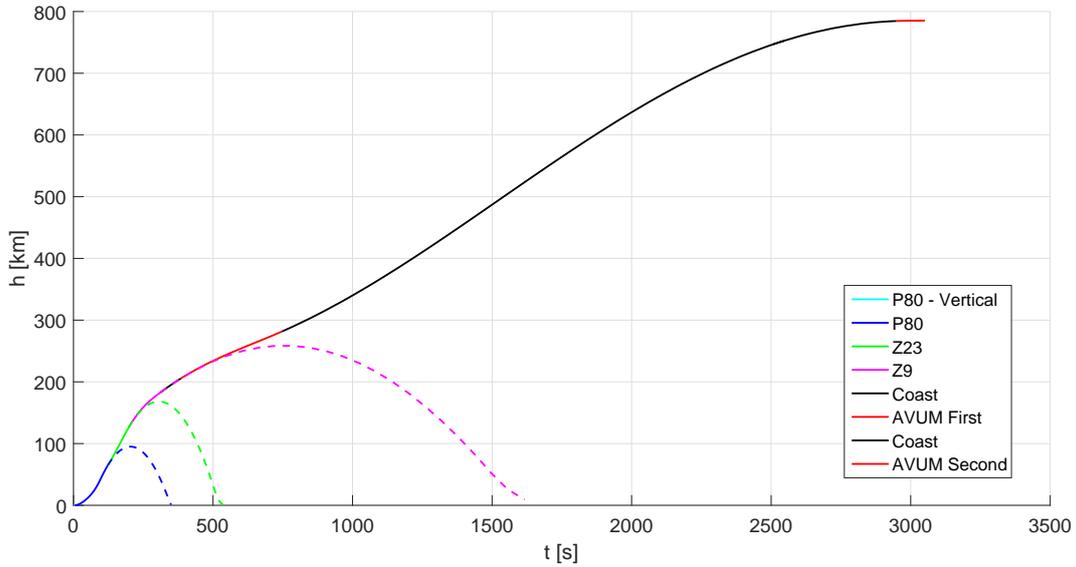


Figure 5: altitude vs time - Vega, SSO 785 km

Ballistic Re-entry

Starting from the orbital conditions at Z9 burnout $\{a_b, e_b, i_b, \Omega_b, \omega_b\}$ and with the fundamental equation of conic sections, the true anomaly of the impact point is derived by setting $r = R_E$:

$$\theta_{\text{impact}} = \arccos\left(\frac{a_b(1 - e_b^2)}{e_b R_E} - \frac{1}{e_b}\right) \quad (25)$$

Then, the ECI coordinates $\{x_{\text{imp}}, y_{\text{imp}}, z_{\text{imp}}\}$ are computed:

$$\begin{aligned} x_{\text{imp}} &= R_E \cdot \left[\cos(\omega_b + \theta_{\text{impact}}) \cdot \cos(\Omega_b) - \sin(\omega_b + \theta_{\text{impact}}) \cdot \cos(i_b) \cdot \sin(\Omega_b) \right] \\ y_{\text{imp}} &= R_E \cdot \left[\sin(\omega_b + \theta_{\text{impact}}) \cdot \cos(i_b) \cdot \cos(\Omega_b) + \cos(\omega_b + \theta_{\text{impact}}) \cdot \sin(\Omega_b) \right] \\ z_{\text{imp}} &= R_E \cdot \sin(\omega_b + \theta_{\text{impact}}) \cdot \sin(i_b) \end{aligned} \quad (26)$$

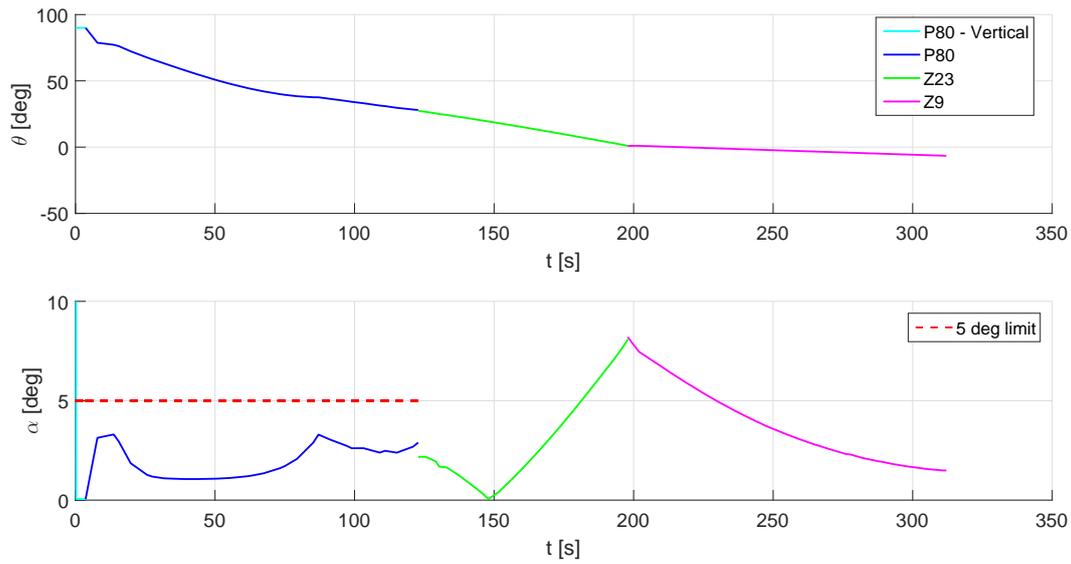
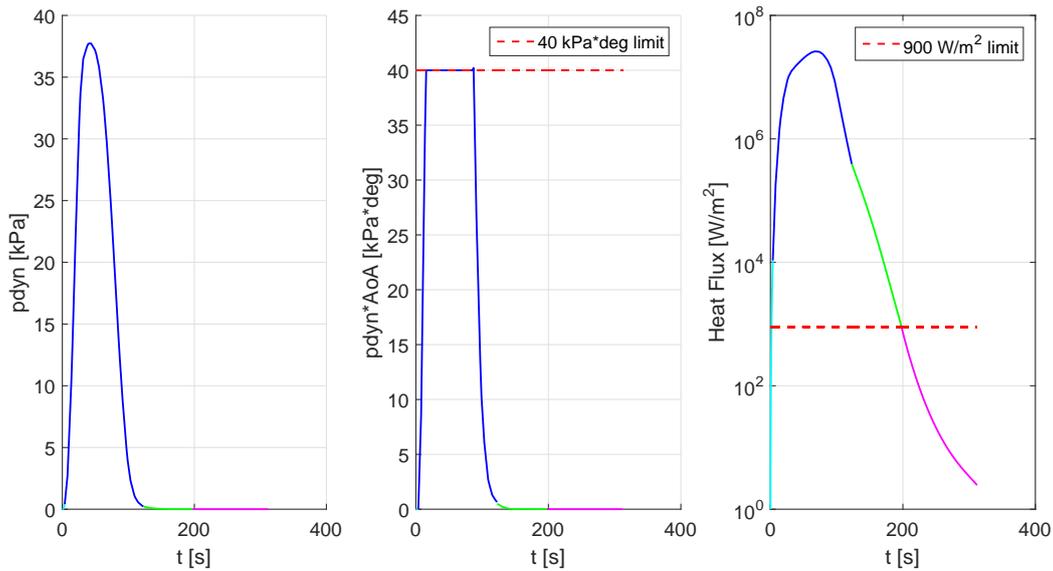


Figure 6: pitch and angle of attack - Vega - SSO 785 km

Figure 7: dynamic pressure, $q \cdot \alpha$ and heat flux - Vega - SSO 785 km

Finally, latitude and longitude of the impact point are:

$$\begin{aligned} \text{lat}_{imp} &= \arcsin\left(\frac{z_{imp}}{R_E}\right) \\ \text{lon}_{imp} &= \arctan2\left(\frac{y_{imp}}{R_E \cdot \cos(\text{lat}_{imp})}, \frac{x_{imp}}{R_E \cdot \cos(\text{lat}_{imp})}\right) \end{aligned} \quad (27)$$

Therefore impact point is constrained in a box defined by max and min latitude and longitude. The software then finds the optimal orbital elements at Z9 burnout to fulfill this constraint, through the equations above. In the particular case of the SSO at 785 km, it is enough to set the maximum and minimum values of the latitude to ensure that the stage splashes in the Arctic sea. The real impact point will be different from the one computed with the ballistic assumption: in general, because of drag, the down-range will be shorter. This effect, visible in Figure 8, must be taken into account when setting the constraint.

VEGA TRAJECTORY OPTIMIZATION

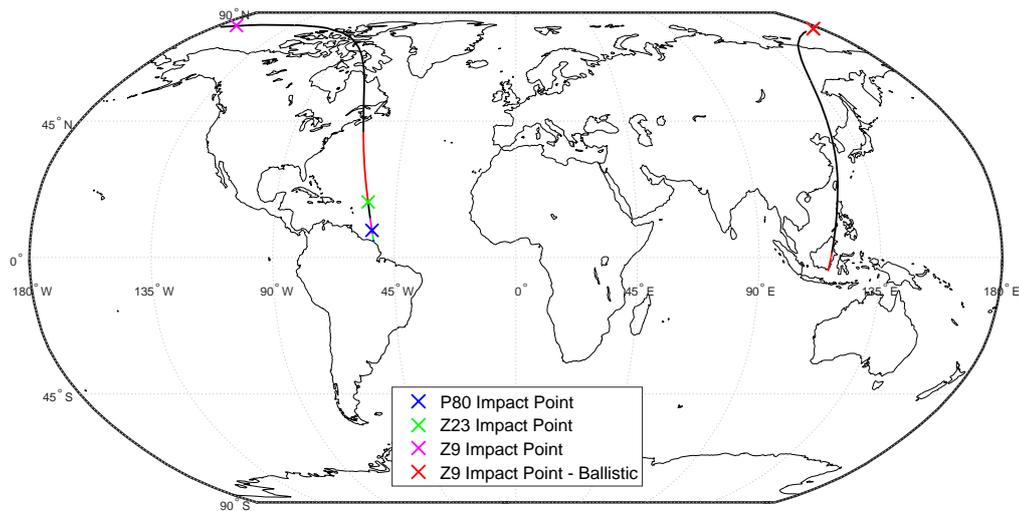


Figure 8: ground track - Vega - SSO 785 km

7. Conclusions

The presented research focuses on the application of PS methods to the launch trajectory optimization and on the formulation of the ascent problem. The main outcomes of this work can be summarized as follows:

1. Gauss and Radau PS methods have been selected because direct transcription of the optimal control problem using the LG and LGR collocation points is characterized by an easy implementation, typical of direct methods, but at the same time by high accuracy, typical of indirect methods, as demonstrated by Benson² and Garg et al⁹ ;
2. The dynamic model includes Vega stages' thrust profiles and drag curves. The control to be optimized is the direction of the thrust and it is formulated in two different ways, i.e., Cartesian x-y-z components and pitch-yaw angles, allowing to select the one that is most suited for the specific application. Thanks to a general expression of the cost function, it is possible to maximize the propellant reserve (knowing the payload) or, by adding an extra variable to the problem, to maximize the (unknown) payload mass. The main flight constraints are taken into account, such as maximum angle-of-attack, pitch rate, aerodynamic loads and heat flux;
3. The implementation of launcher dynamics, constraints and guesses is validated with realistic mission scenarios. The obtained performances, in terms of payload mass, are often slightly higher than those provided by AVIO; in addition, pitch-yaw curves obtained with GPOPS are smoother, at the expense of a higher computational effort. In fact, while the point-by-point optimization performed by GPOPS allows for more accurate solutions, the standard guidance laws (e.g. gravity turn or linear pitch-over) used in real-life trajectory design drastically lower the number of variables, as well as the complexity of the problem;
4. The flexibility of the proposed implementation is tested with different target orbits (Sun-synchronous, polar, equatorial, escape, etc.) with promising results: all the typical constraints are included and fulfilled; problem formulation can be easily tailored to all kinds of configurations (number of stages, types of motors, etc.) and number of phases (single/multiple boosts, coast arcs, etc.); moreover, it allows for both single and multi-payload missions with change of orbital plane; lastly, the performances are improved with respect to those obtained with common guidance laws, despite a higher computational effort.

Few open points have been identified to improve the presented approach: the accuracy of the simulations can be enhanced by including the constraints of ground station visibility and aspect angle, together with the lift component of the aerodynamic force; moreover, computational effort can be reduced by implementing an algorithm for fast guess computation and a set of simple (e.g. linear) control laws, which would drastically decrease the number of variables of the problem.

In conclusion, the problem formulation could also be extended to more complex applications: for example, thrust magnitude can be added to the control variables for launchers with throttling capabilities. In the same way, the 6DoF trajectory can be directly optimized by including the attitude dynamics, with the related extra state and control variables (and extra problem complexity).

References

- [1] Arianespace. *Vega User's Manual, Issue 4 Revision 0*. Arianespace, April 2014.
- [2] D. Benson. *A Gauss Pseudospectral Transcription for Optimal Control*. PhD Thesis - Massachusetts Institute of Technology, 2005.
- [3] J.T. Betts. *Practical Methods for Optimal Control Using Nonlinear Programming*. SIAM, 2001.
- [4] Cornelisse, Schoyer, and Wakker. *Rocket Propulsion and Spacecraft Dynamics*. Pitman, 1979.
- [5] E. C. Coskun. *Multistage Launch Vehicle Design with Thrust Profile and Trajectory Optimization*. Middle East Technical University, Ankara, 2014.
- [6] COSPAR International Reference Atmosphere. *Models of the Earth Upper Atmosphere*. CIRA, 2012.
- [7] G. Di Campi Bayard de Volo. *Vega Launchers Trajectory Optimization Using a Pseudospectral Transcription*. MSc Thesis - Delft University of Technology, Faculty of Aerospace Engineering, Kluyverweg 1, 2629 HS Delft, The Netherlands, 2017.
- [8] A. Gabrielli. *Ottimizzazione di Traiettorie di un Piccolo Lanciatore*. Sapienza University of Rome - School of Aerospace Engineering, 2008.
- [9] D. Garg, M. A. Patterson, W. W. Hager, A. V. Rao, D. Benson, and G. T. Huntington. *Direct Trajectory Optimization and Costate Estimation of General Optimal Control Problems Using a Radau Pseudospectral Method*. AIAA Guidance, Navigation, and Control Conference 10 - 13 August 2009, Chicago, Illinois, 2009.
- [10] D. Garg, M. A. Patterson, W. W. Hager, A. V. Rao, D. Benson, and G. T. Huntington. *An overview of Three Pseudospectral Methods for the Numerical Solution of Optimal Control Problem*. AAS 09-332, 2009.
- [11] D. Garg, M. A. Patterson, W. W. Hager, A. V. Rao, D. Benson, and G. T. Huntington. *A Unified Framework for the Numerical Solution of Optimal Control Problems Using Pseudospectral Methods*. Automatica Elsevier, 2010.
- [12] P.E. Gill, W. Murray, and M. A. Saunders. *SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization*. Society for Industrial and Applied Mathematics, 2005.
- [13] P.E. Gill, W. Murray, and M. A. Saunders. *Users Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming*. Department of Mathematics, University of California, San Diego, La Jolla, CA 92093-0112, USA, 2007.
- [14] S. Josselyn and I.M. Ross. *A Rapid Verification Method for the Trajectory Optimization of Reentry Vehicles*. Journal of Guidance, Control and Navigation, 2003.
- [15] A. Rao, D. Benson, G.Y. Huntington, C. Francolin, C.L-Darby, and M. Patterson. *User Manual for GPOPS Version 1.0: A MATLAB Package for Dynamic Optimization Using the Gauss Pseudospectral Method*. 2011.
- [16] I.M. Ross, C. D'Souza, F. Fahroo, and J. B. Ross. *A Fast Approach to Multi-Stage Launch Vehicle Trajectory Optimization*. AIAA Guidance, Navigation and Control Conference and Exhibit, Austin, Texas, 2003.
- [17] K.F. Wakker. *Fundamentals of Astrodynamics*. Faculty of Aerospace Engineering, Delft University of Technology, 2015.