ASTOS® 9

Conventional Launcher Tutorial

Version: 9.16.0

Printed in Germany.

December 2020

Preface

This tutorial presents a step by step procedure to create and optimize a conventional launcher (VEGA) starting from the available design information. In case the user has never used *ASTOS* before, it is suggested to follow the instructions presented in the Getting Started book first.

The time required for this tutorial is around three hours (advanced modifications not considered). In case of problems, the user can find the "final" status of the scenario in `%ASTOS%\examples\Optimization and Design\Launch\Vega.gtp`, but the suggestion is to try to follow the procedure without using the prepared scenario.

Please note that this tutorial uses the **Optimization** feature of *ASTOS*. In case this feature is not present in your license, the tutorial can still be followed with the exception of the chapter relative to the optimization of the launcher.

PREFACE

Table of Contents

1 Introduction **10**

2 Create the Model **11**

2.1 Create a Folder 11
2.2 Define the Environment 11
2.3 Define the Actuators 12
2.4 Define the Aerodynamics 14
2.5 Define the Components 16
2.6 Vehicles and POIs Definition 18
2.7 Phase Configuration 21
2.7.1 Phase Definition 22
2.7.2 Initial Phase Configuration 24
2.7.3 Pitch Over Phase Configuration 25
2.7.4 Constant Pitch Phase Configuration 26
2.7.5 P80 Burn Phase Configuration 27
2.7.6 Zefiro23 Burn Phase Configuration 27
2.7.7 Coast with Fairing Phase Configuration 27
2.7.8 Zefiro9 Burn Phase Configuration 28
2.7.9 First AVUM Burn Phase Configuration 29
2.7.10 Upper Stage Coast Arc Phase Configuration 29
2.7.11 Second AVUM Burn Phase Configuration 30
2.7.12 Check of the Phase Sequence 30
2.8 Add Constraints 31
2.8.1 Initial Constraints 32
2.8.2 Final Constraints 33
2.8.3 Path Constraints 36
2.9 Cost Functions 37

3 Initialize the Launcher **39**

4 Optimize the Launcher **42**

4.1 Optimization Settings 42
4.2 Grid Nodes 44
4.3 Run the Optimization 46

5 Visualize the Results **48**

6 Summary and Advanced Modifications **51**

TABLE OF CONTENTS

6.1 Target Orbit 51
6.2 Station Visibility 52
6.3 Staging Constraint 53
6.4 Mass Flow Profile 54

List of Figures

Fig. 2.1:	*New Scenario* window	11
Fig. 2.2:	Settings for the propulsion system of the first stage of VEGA	13
Fig. 2.3:	Settings for the P80 stage of VEGA	17
Fig. 2.4:	Final state of the *Vehicle Structure* panel	19
Fig. 2.5:	Vega vehicle preview.	21
Fig. 2.6:	Settings for the attitude control of pitch over phase	26
Fig. 2.7:	Initial altitude constraint for VEGA	33
Fig. 2.8:	Cost function settings for VEGA	38
Fig. 3.1:	Initial trajectory of VEGA in satellite view	39
Fig. 3.2:	Initial trajectory of VEGA altitude profile	40
Fig. 3.3:	Example of Scenario Summary for VEGA (extract)	41
Fig. 4.1:	*Optimization Settings* for the VEGA scenario	43
Fig. 4.2:	*Diagnostic Output* window for *CAMTOS*	43
Fig. 4.3:	Adding constraint evaluation grid nodes using the *Initial Grids* panel	45
Fig. 4.4:	Iteration information of the optimization of VEGA (iteration steps > 4 have been cut)	46
Fig. 5.1:	Optimized altitude profile of VEGA	49
Fig. 5.2:	Example of Result Summary for VEGA (phases 2-10 are not displayed)	50

List of Tables

Table 2.1: Simplified drag force coefficient profile based on Ariane 4 15
Table 6.1: Mass flow profile for P80_engine 54
Table 6.2: Mass flow profile for Z23_Engine 55
Table 6.3: Mass flow profile for Z9_Engine 55

1 Introduction

One of the main applications of *ASTOS* is the optimization of launcher vehicles. This tutorial describes all necessary steps to build an *ASTOS* optimization model of a launcher from scratch. The selected launcher is VEGA (Vettore Europeo di Generazione Avanzata, European Advanced Generation Carrier). It is an expendable launch system jointly developed by the Italian Space Agency and the European Space Agency since 1998, with the first launch on 13 February 2012. VEGA is designed to launch small payloads (300 to 2000 kg satellites) for scientific and Earth observation missions to polar and low Earth orbits.

VEGA is a single body launcher with three solid rocket stages: the P80 first stage, the Zefiro23 second stage, the Zefiro9 third stage, and a liquid rocket upper module called AVUM.

All information about the VEGA launcher are taken from [1], [2], [3], [4] and [5].

Note: These data need to be intended as indicative and not as "real" VEGA data.

The presented scenario is the "design" mission of VEGA: place the maximal payload in a polar orbit at 700 km altitude.

2 Create the Model

This chapter contains all the steps required to create the optimization scenario.

2.1 Create a Folder

Open *ASTOS* and select *New Scenario* from the *Application menu* (in the tutorial, such operation is denoted as *Application menu*→ *New Scenario*). The *New Scenario* window automatically opens to let the user specify the *Name of the scenario* and the *Parent path* (see Fig. 2.1). Insert the desired name (with the relative path) of the folder that identifies the new scenario. Then press the **Create** button. The newly opened *Viewer* window can be closed. A short description of the *Viewer* is given in a subsequent chapter of this tutorial.

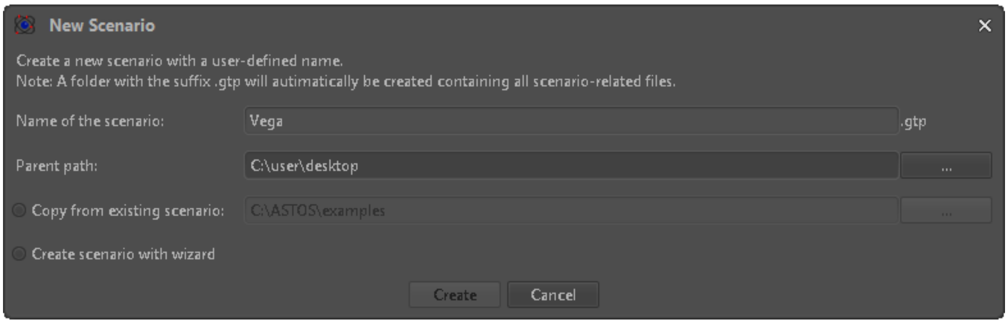


Fig. 2.1: New Scenario window

ASTOS stores each scenario in a folder and **Create** generates the required files (and subfolders) in the specified folder. For detailed information about the creation of a new scenario, refer to the Scenario Menu in User Manual

2.2 Define the Environment

An empty scenario has been created. Now the launcher data has to be set up in an environment model. For this purpose the *Modelling* tree on the left side of the *ASTOS* main window is used.

For detailed information about the *Modelling* tree, please refer to the Modelling of Scenarios or to the complete Model Reference.

- First select the name of this newly created scenario which is the first entry in the *Modelling* tree.
- Optionally, a brief *Description* of the scenario can be added in the corresponding cell (e.g. *VEGA into polar orbit, 700 km altitude*). This description gets included in the result summary which will be created later in this tutorial.
- In the *Active features* section, enable the *Optimization* field. Then in the *Automatic simulation options* section, in the *Simulate after initialize* setting, select *Multiple Shooting*. In the *Optimization*, *Check Bounds*, uncheck the *Connected flags* in case of checking the bounds. The latter will cause that phase connect value violations, which are inevitable in the first iterations of preparing a scenario, will not be displayed.
- Expand the *Environment* node and select *Celestial Bodies* in the *Modelling* tree. Make sure that the Earth is already defined as default. If not, select *Add*→ *Environmental Models*→ *Celestial Body* from the ribbon, insert an *Identifier* (e.g. "Earth") and press the **Create** button.
- The Earth model should now be present in the *Celestial Bodies* list. In the configuration panel go to the *Shape* section. Select *Spheroid* from the *Type* drop-down list and enter the *Equatorial radius* (*6378.137 Kilo-Meter*) radius and the *Polar radius* (*6356.752 Kilo-Meter*). In the *Gravity* section select *Ellipsoid* from *Type*, and enter *398600.4 Kilo-Meter**3/ Second**2* for the *Gravity parameter*, *0.0010827* for the *J2*, and *0.0* for *J3*, *J4*, *J5* and *J6* coefficients.
- Select ribbon task *Add*→ *Environmental Models*→ *Atmosphere*.
- Insert an *Identifier* (e.g. *us_standard_76*) and select *US_Standard* and *US_Std_76* from the *Type* and *Subtype* drop-down lists. Press the **Create** button.
- Finally, navigate to the *Modelling* tree → *Default Environment*. Select *Earth* as *Central body* drop-down, and *us_standard_76* as *Atmosphere* drop-down.

Note: All object names are case-sensitive and need to be unique.

Tip: *ASTOS* uses the information stored after the **Save** button has been clicked. Always save before initializing to update the model, otherwise the previously saved values are used.

2.3 Define the Actuators

VEGA contains four propulsion engines, three solid engines and one liquid engine. As for the stages later on, only the creation of the first engine is presented in detail. The procedure can be repeated analogously for the remaining engines. The data comes from Appendix 1 in [4], while the vacuum thrust is computed from the propellant mass, the specific impulse (Isp) and the burn duration.

- Select ribbon task *Add→ Vehicle Models→ Actuator*.
- Enter *P80_Engine* as *Identifier*, choose *Rocket* from the *Type* drop-down list and *Profile* from *Subtype* respectively. Confirm using the **Create** button.

Navigate to the new item *Vehicle Parts & Properties -> Actuators -> P80_engine* in the *Modelling* tree on the left and set the following values in respective section of the configuration panel as custom (leave disabled all other settings):

- *Nozzle Ae*: *3.0 Meter**2*
- *Vacuum Thrust*: *2272.0 Kilo-Newton*
- *Vacuum Isp*: *280.0 Second*

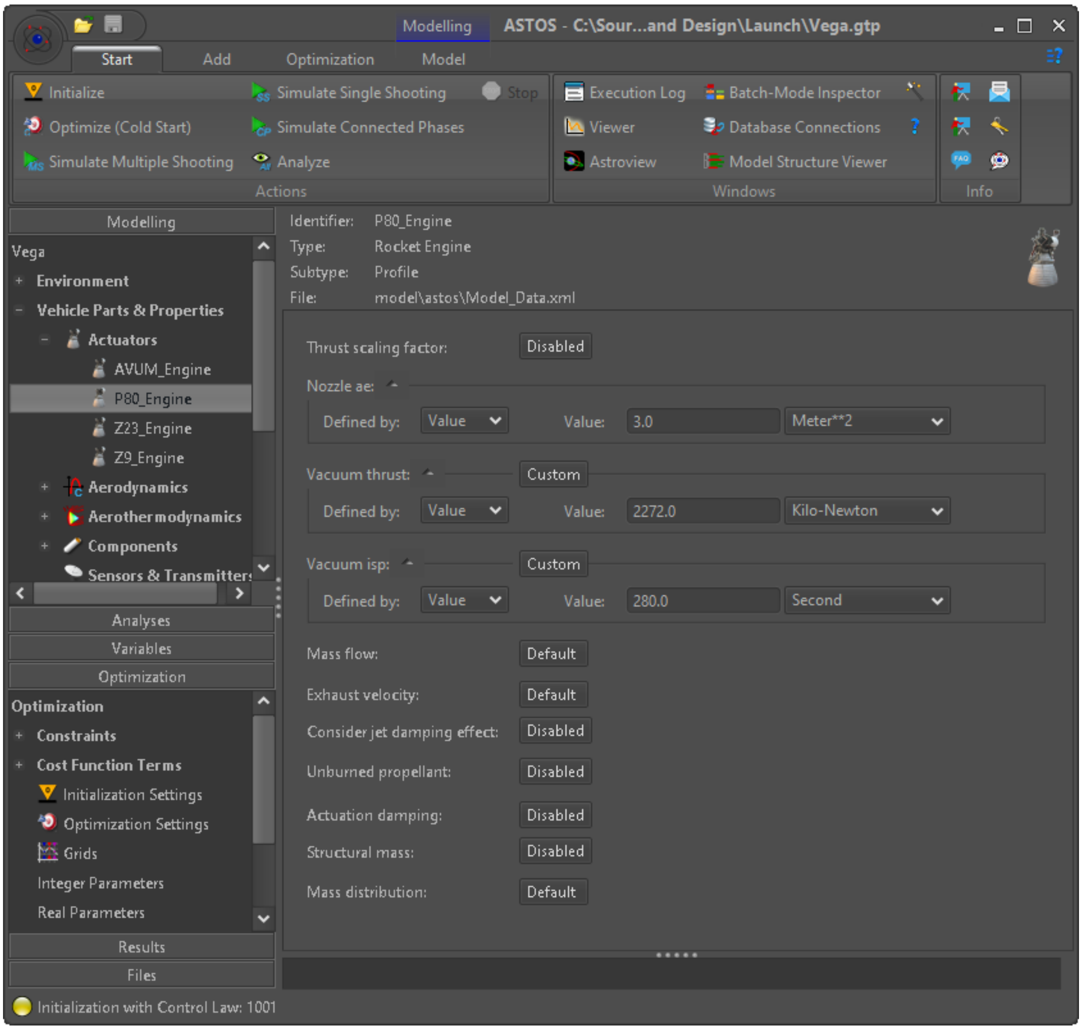


Fig. 2.2: Settings for the propulsion system of the first stage of VEGA

Set up the remaining three propulsion systems in the same manner (*Rocket / Profile*) using the parameters listed below. Again, leave disabled all other variables:

- Zefiro23 engine [4]:

  - *Identifier*: *Z23_Engine*
  - *Nozzle Ae*: *1.7 Meter**2*
  - *Vacuum Thrust*: *944.88 Kilo-Newton*
  - *Vacuum Isp*: *289.0 Second*
- Zefiro9 engine [4]:
  - *Identifier*: *Z9_Engine*
  - *Nozzle Ae*: *1.257 Meter**2*
  - *Vacuum Thrust*: *250.1 Kilo-Newton*
  - *Vacuum Isp*: *295.0 Second*
- AVUM engine [4]:
  - *Identifier*: *AVUM_Engine*
  - *Nozzle Ae*: *0.07 Meter**2*
  - *Vacuum Thrust*: *2.45 Kilo-Newton*
  - *Vacuum Isp*: *315.5 Second*
- Press **Save**.

Tip: Instead of creating a new propulsion system, it is also possible to clone an existing one (right-click the actuator element in the *Modelling* tree and select **Clone**).

2.4 Define the Aerodynamics

After the environment and propulsion is fixed, it is necessary to define the vehicle stages with their aerodynamics and masses.

- Click on *Add*→ *Vehicle Models*→ *Aerodynamics*. Insert an *Identifier*, (e.g. *vega_aero*), select *Tabular* as *Type* and press the **Create** button.
- Select the newly created aerodynamics model in the *Modelling* tree (expand the *Aerodynamics* element).
- In the configuration panel, expand the *Reference Area* sub-pane, select *Diameter* as *Type* and enter a *Reference diameter* of *3.025 Meter* [4].
- In the *Coefficients* pane click on the *Add force coefficient* button. The *Add force coefficient* window appears. There, please perform the following configuration

  - for *Frame* select *Air path (A)*
  - for *Axis* select *Drag direction*
  - for *Character* select *Absolute value*

  and click on the **Add** button.

- In the *Coefficients* table click on the table row containing the newly created force coefficients. In the *Selected coefficient* pane, in the *Absolute value* sub-pane enter the following settings.
  - for *Defined by* select *Profile*. In the *Profile* pane enter the following settings:
    - for *Interpolation type* select *Linear*
    - for *Ordinate data scale* select *Linear to linear*
    - for *Data source* select *Local*
    - for *Out of bounds action* select *Nearest value*
    - for *Scaling factor* enter 1.0
  - In the profile data table in the GUI transfer the content of the following Table 2.1 :

Table 2.1: Simplified drag force coefficient profile based on Ariane 4

	1	**Data**
Name	Mach Number	
Unit	None	None
1	0.0	0.4
2	0.9	0.4
3	1.1	1.0
4	4.0	0.5
5	5.0	0.5

  - In the *Perturbation (uncertainty) sub-pane* set *Bias* to *Disabled* and *Gain* to *Disabled*
- Save the scenario by pressing the **Save** button.

Different aerodynamics models can be defined and activated for the different phases of the mission. For sake of simplicity, one aerodynamics model is considered valid for this tutorial from begin to end - even though after the fairing separation there is a change of the vehicle diameter. However at the related altitude (> 100 km) the atmosphere does not play an important role anymore.

Note: *ASTOS* supports complex aerodynamics definition with interpolated coefficient functions - also for several independent variables.

2.5 Define the Components

VEGA is made of three main stages plus an upper stage. The *ASTOS* model is composed of four engines, four associated stages, a fairing and a payload. The fairing is defined first, since it has the simplest model:

- Select ribbon task *Add*→ *Vehicle Models*→ *Component*.
- Insert *Fairing* as *Identifier*.
- Select *Auxiliary* from the *Type* drop-down list.
- Press **Create**.

- Select the *Components* object just created in the *Modelling* tree (*Fairing*) and enter *490.0 Kilogram* as *Total Mass* [4].
- Activate the *Dimensions* section and set *X*, *Y* and *Z* to *7.8 Meter*, *2.6 Meter* and *2.6 Meter* respectively.
- Select *Cylinder* in the *Shape* drop-down with *X* as *Cylinder axis*.
- Select the *Fairing* object, right-click and press **Clone**. Change the name of *Fairing_CLONE* to *AVUM_deorbit* (right-click and press **Rename**).
- Set the mass of *AVUM_deorbit* to *100.0 Kilogram* (this is the propellant mass used to de-orbit the AVUM stage).
- In the *Dimensions* section set *X*, *Y* and *Z* to *0.1 Meter*, *1.9 Meter* and *1.9 Meter* respectively.

Only the creation of the first stage is presented in detail, the procedure can be repeated for the other three stages analogously:

- Select ribbon task *Add*→ *Vehicle Models*→ *Component*.
- Enter *P80_Stage* as *Identifier*.
- Select *Basic Vehicle Stage* from the *Type* drop-down list and press the **Create** button.
- Select the *P80_Stage* model in the *Modelling* tree and enter in the respective panels [4]:
  - *Structural Mass*: *7416.0 Kilogram*
  - *Propellant Mass*: *88380.0 Kilogram*
  - Leave the *Filling Ratio* as Common.
  - Activate the *Dimensions* section and set *X*, *Y* and *Z* to *10.5 Meter*, *3.0 Meter* and *3.0 Meter* respectively.

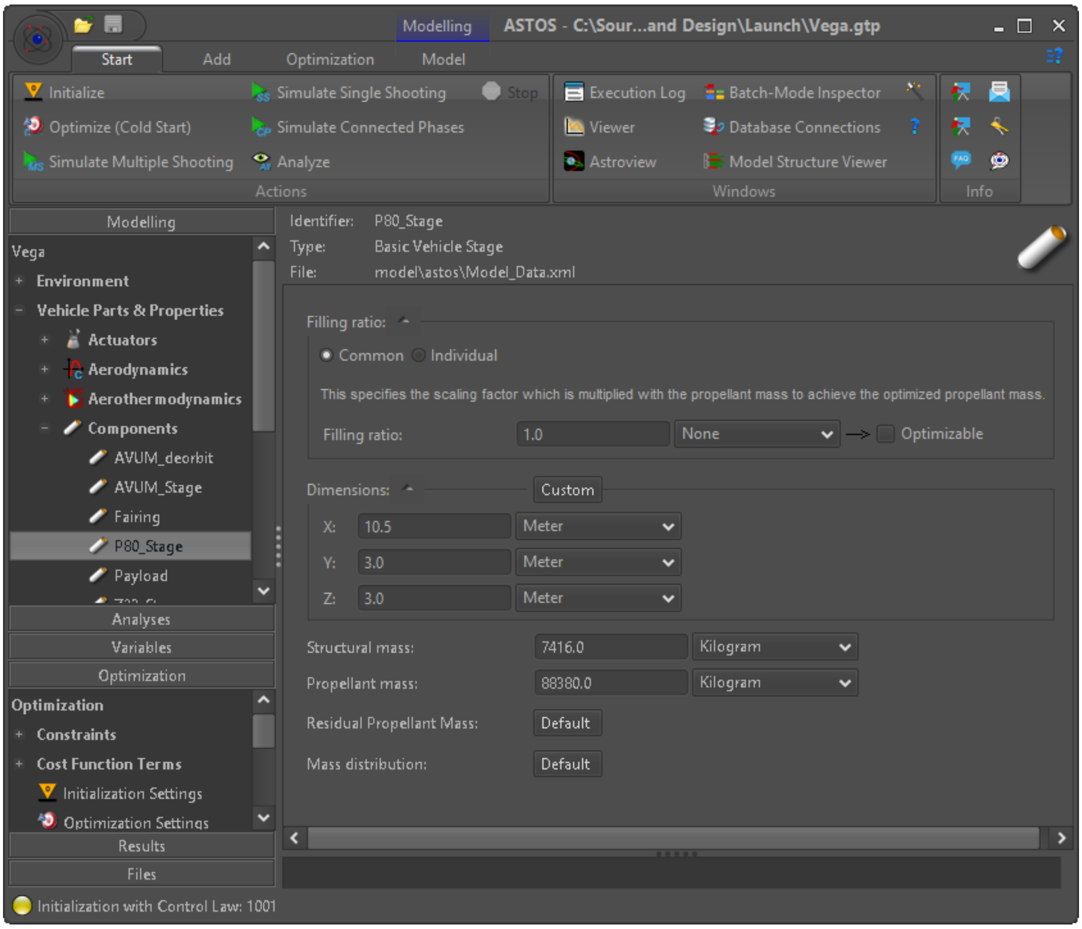


Fig. 2.3: Settings for the P80 stage of VEGA

Tip: Instead of creating a new component, it is also possible to clone an existing one (right-click the component element in the *Modelling* tree and select **Clone**).

In the next step, create the remaining three *Basic Vehicle Stages* (with the filling ratio kept at 1.0) [4].

- Zefiro23:
  - *Identifier*: *Zefiro23_Stage*
  - *Structural mass*: *1845.0 Kilogram*
  - *Propellant mass*: *23906.0 Kilogram*
  - *Dimensions* (*X/Y/Z*): *7.5 / 1.9 / 1.9 Meter*
- Zefiro9:
  - *Identifier*: *Zefiro9_Stage*
  - *Structural mass*: *833.0 Kilogram*
  - *Propellant mass*: *10115.0 Kilogram*
  - *Dimensions* (*X/Y/Z*): *3.85 / 1.9 / 1.9 Meter*
- AVUM (structural mass includes also the payload adapter):

- *Identifier*: *AVUM_Stage*
- *Structural mass*: *478.0 Kilogram*
- *Propellant mass*: *450.0 Kilogram*
- *Dimensions* (*X/Y/Z*): *1.74 / 1.9 / 1.9 Meter*

Finally, the payload needs to be implemented:

- Select ribbon task *Add→ Vehicle Models→ Component*.
- Select *Payload* as *Identifier* and *Payload* as *Type*.
- Enter a *Nominal mass* equal to *1500.0 Kilogram*.
- Leave *1.0* as *Payload scaling factor*.
- Check the *Optimizable* box and define the *Lower/upper bound* as *0.5* and *2.0* in order to allow *ASTOS* to optimize the trajectory for a payload between 750kg and 3000kg.

> **Note:** In case the *Optimizable* box is not visible, navigate to the top of the *Modelling* tree and verify that *Optimization* is enabled in the *Active Features* section. If activation is not possible, your license does not cover the **Optimization** feature. Simply skip this step in that case.

- Activate the *Dimensions* section and set *X*, *Y* and *Z* to *5.0 Meter*, *2.0 Meter* and *2.0 Meter* respectively.
- Press **Save**.

2.6 Vehicles and POIs Definition

The operations performed so far have created all the required models for the mission. Now it is necessary to activate the models and to associate each propulsion system to a stage:

- Select ribbon task *Add→ Vehicles & Other Entities→ Vehicle*.
- Enter *Vega_Rocket* as *Identifier*.
- Select *Rocket* from the *Type* drop-down list and press the **Create** button.
- In the *Vehicle Structure* section press the **Add Assembly** button (). In the pop-up window, set *Assembly Type* to *Stage 1*, name *P80* as *Assembly Identifier* and click **Add Assembly**
- Select the *P80* assembly, now visible in the *Vehicle Structure* tree, and press the **Add Element** button (). In the pop-up window select the *P80_Stage* from the *Components* list and click **Add Element**.
- Repeat the last step, but this time add the *P80_Engine* to the *P80* assembly.
- Add the Zefiro23, Zefiro9 and AVUM assembly (as 2nd to 4th stage) to the vehicle by using the same steps as for the P80 assembly described above. Do not forget to add the remaining propellant for the deorbit burn of the AVUM stage to the AVUM assembly.

- In the *Vehicle Structure* section press the **Add Assembly** button again. In the pop-up window, set *Assembly Type* to *Fairing*, enter *Vega_Fairing* as *Assembly Identifier* and click **Add Assembly**.
- Select the *Vega_Fairing* assembly and press the **Add Element** button. In the pop-up window select *Fairing* from the *Components* list and click **Add**.
- In the *Vehicle Structure* section press the **Add Assembly** button again. In the pop-up window, set *Assembly Type* to *Payload*, enter *Vega_Payload* as *Assembly Identifier* and click **Add Assembly**.
- Select the *Vega_Payload* assembly and press the **Add Element** button. In the pop-up window select *Payload* from the *Components* list and click **Add**.
- Press **Save**.

Fig. 2.4 shows how the *Vehicle Structure* settings should appear after all steps have been performed.

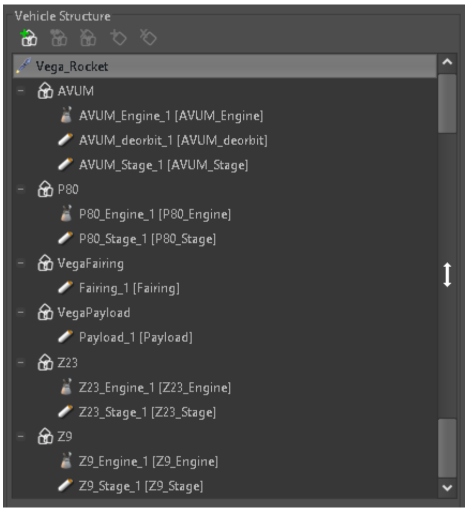


Fig. 2.4: Final state of the Vehicle Structure panel

Finally all components have to be assembled properly (engines on the bottom of their corresponding stage, stage 2 on top stage 1 etc.):

- Navigate to the P80 stage (*P80_Stage*) of the P80 assembly (*P80*) in the *Vehicle Structure* tree and click it.
  Options for that element should now appear in the *Vehicle Element* section below.
- Set the *Anchor Node* to *Bottom (-x)* and leave the *Reference element* as *Global*.
- Now select the P80 engine (*P80_Engine*) from the *Vehicle Structure* tree. In the *Vehicle Element* section select the *P80_Stage* as *Associated Tank*.
- Anchor the engine to the stage by selecting *P80_Stage* as *Reference element* and *Bottom (-x)* as *Reference node*.
- Select the Zefiro23 stage (*Zefiro23_Stage*) from the *Zefiro23* assembly by clicking it in the *Vehicle Structure* tree.
- Set the *Anchor Node* to *Bottom (-x)* here as well. But in order to place it on the top of the P80 stage select the *P80_Stage* as *Reference element* and set the *Reference node* to *Top (x)*.

- Now select the *Z23_Engine* from the *Vehicle Structure* tree. In the *Vehicle Element* section select the *Zefiro23_Stage* as *Associated Tank*.
- Anchor the engine to the stage by selecting the *Zefiro23_Stage* as *Reference element* and set *Bottom (-x)* as *Reference node*.
- Repeat these steps for the Zefiro9, AVUM and the fairing. Keep in mind to place them on top of each other in the right order by selecting the stage below it as *Reference element*.
- For *AVUM_deorbit*, leave the *Anchor Node* to *Center*, set the *Reference element* to *AVUM_Stage* and the *Reference node* to *Center*.
- Select the *Payload* from the payload assembly by clicking it in the *Vehicle Structure* tree
- Set the *Anchor Node* to *Center* in order to center it within the fairing. Select the *Fairing* as *Reference element* and set the *Reference node* to *Center*.
- Press **Save**.

Once the vehicle assemblies are created and correctly set, please use the *Vehicle Preview* on the right to visualize the result:

- Select *Use data from simulation*.
- Press the **Projection** to rotate the view.

The *Vehicle Preview* on the right should show a figure similar to Fig. 2.5.

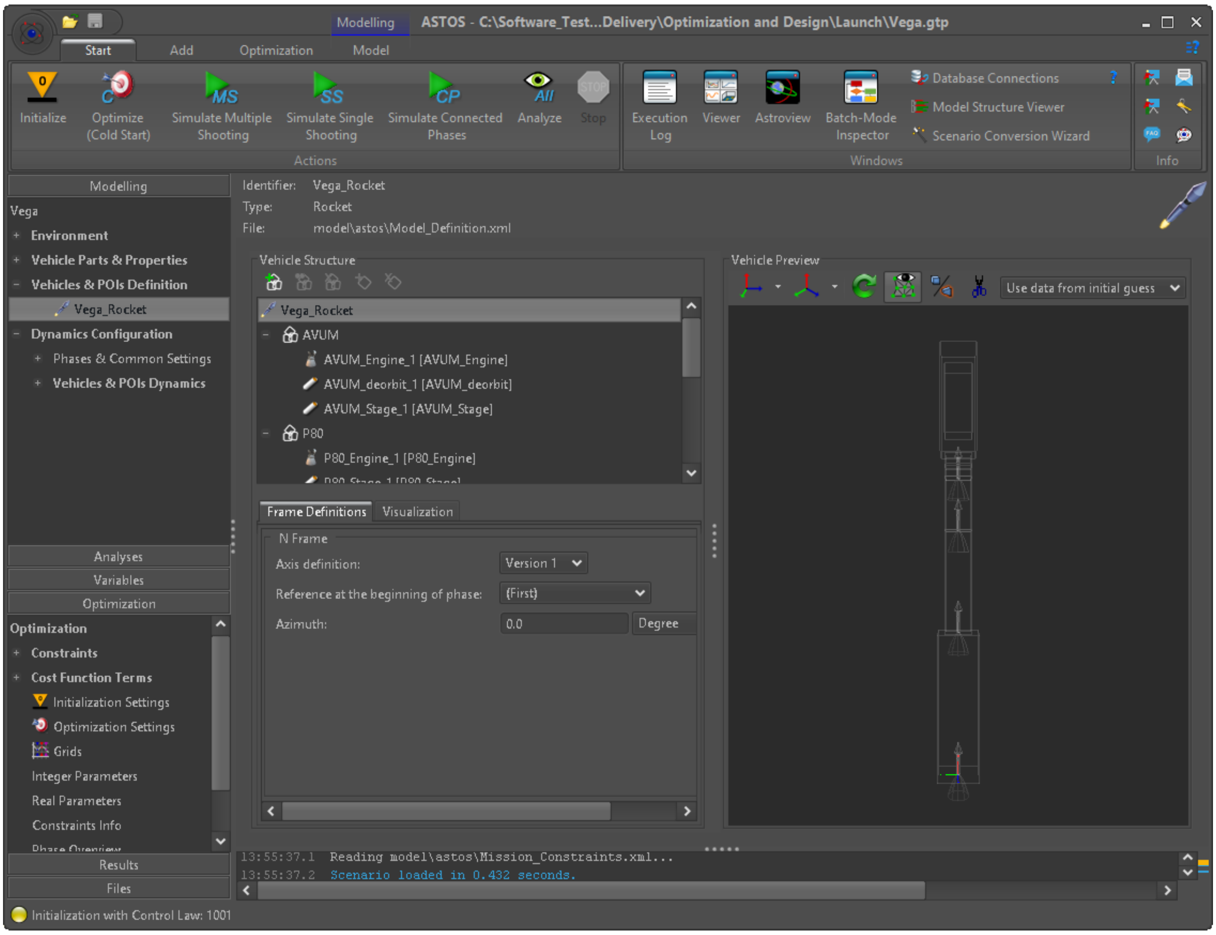


Fig. 2.5: Vega vehicle preview.

2.7 Phase Configuration

The phase configuration is one of the most complex tasks related to the creation of a launcher model. A certain experience is essential to be able to "see" a launcher mission segmented in all the necessary phases. But why are there multiple phases?
At each discontinuity in the launcher schedule, a new phase must be defined. There is no upper limit to the number of specified phases as be able to cover any configuration and launcher schedule of a vehicle. The following list states some typical reasons for discontinuities:

- Switching between different attitude control strategies (vertical ascent, pitch over, pitch constant, gravity turn, required-velocity guidance or optimizable)
- Switching the aerodynamics model
- Jettisoning of a component
- Switching the constraint conditions

The typical sequence for a vertically launched rocket is:

- (Launcher locked at the pad during the ignition)
- Vertical ascent with constant yaw pointing in the desired direction (90° orbit inclination) until a certain altitude is reached to avoid the collision with the launch pad.
- The pitch over maneuver is used to give the rocket a direction. It uses a linear pitch law and a constant yaw law. The pitch rate is characteristic for specific launchers: heavy launchers (Ariane 5) around -1°/sec, small launchers up to -3°/sec or more.
- An intermediate constant pitch phase is required to meet the gravity turn conditions: flight path angle equal to pitch angle or angle of attack equal to zero. The yaw angle is constant.
- The gravity turn phase follows until the aerodynamic forces are small enough for high angles of attack. In the beginning the yaw angle remains constant. It depends on the launcher, how many phases fly with gravity turn and at which time point yaw control is allowed.
- For sure after jettisoning the fairing (low dynamic pressure) the attitude control is free. During the initial guess, required-velocity guidance can be used. An optimizable configuration normally uses optimizable controls for yaw and pitch.
- Coast arcs should be modeled using a linear law.

Most conventional launchers take off vertically. Since the air path bank angle is not well defined for vertical flight, aerodynamic angles or load factor attitude controls cannot be used for this phase. In the exo-atmospheric phases no aerodynamic forces apply to the vehicle. Therefore only the orientation of the thrust vector, which is mostly aligned with the vehicle's forward axis, is important. For these two reasons, the Euler angle controls were devised for conventional launchers.

Additional to the launch-pad clearance, there is another reason why a vertical phase is required before the pitch over maneuver: some launchers have too little thrust to accelerate and the gravity turn results in a reduction in flight path angle which is irrecoverable (the rocket turns too fast towards the planet surface). For these rockets a short phase must be introduced, in which the pitch is held at 90°. Only then can the pitch-over program set in.

2.7.1 Phase Definition

First the number and schedule of each mission phase have to be defined in the *ASTOS* GUI. This is done in the *Dynamics Configuration* node of the *Modelling* tree in the to *VEGA* launcher scenario.

Set the time to be normalized by switching the *Normalized* option next to the independent variable to *Enabled*. The following phases have to be defined:

Lift Off

- Select the first *Phases* node in *Dynamics Configuration*, the default name should be *Phase_1*.

- Right-click and rename it to *LiftOff*.
- In the configuration panel, insert a proper *Description* (e.g. *Lift Off*).
- Insert a *Final Time* of *5.0 Second*, select the *Bounds* box and insert *3.0* as lower bound and *5.5* as upper bound.

Pitch Over

- Select the phase node *LiftOff* in *Dynamic Configuration -> Phases* again.
- Right-click, and select **Clone**). A new phase appears in the tree (ignore the upcoming message by pressing **OK**).
- Select and rename it to *PitchOver*.
- In the configuration panel, insert a proper *Description* (e.g. *Pitch Over*).
- Insert a *Final Time* of *6.3 Second* (lower bound *6.0*; upper bound *7.0*).

Constant Pitch

- Repeat above-mentioned steps to create a new phase (i.e. clone phase *LiftOff* or *PitchOver*) named *Constant_Pitch* with a proper *Description* (e.g. *Constant Pitch*).
- Insert a *Final Time* of *8.5 Second* (lower bound *7.1*; upper bound *15.0*).

P80 Burn

- Repeat above-mentioned steps to create a new phase named *P80_Burn* with a proper *Description* (e.g. *P80 Burn*).
- Insert a *Final Time* of *106.8 Second* [4] and deactivate *Bounds*.

Zefiro23 Burn

- Repeat above-mentioned steps to create a new phase named *Z23_Burn* with a proper *Description* (e.g. *Zefiro23 Burn*).
- Change the type of *Phase span defined by* to *Phase Duration* and insert *71.7 Second* [4], no bounds.

Coast with Fairing

- Repeat above-mentioned steps to create a new phase named *Coast_with_Fairing* with a proper *Description* (e.g. *Separation of Zefiro23 before fairing jettison*).
- Change the type of *Phase span defined by* to *Final* and insert a *Final time* of *239.0 Second*, activate *Bounds* and insert *180.0* and *260.0* as lower and upper bound, respectively.

Zefiro9 Burn

- Repeat above-mentioned steps to create a new phase named *Z9_Burn* with a proper *Description* (e.g. *Zefiro9 Burn*).
- Insert a *Phase Duration* of *117.0 Second* [4], no bounds.

First AVUM Burn

- Repeat above-mentioned steps to create a new phase named *First_AVUM_Burn* with a proper *Description* (e.g. *First AVUM Burn*).
- Insert a *Final Time* of *500 Second* with *400.0* and *800.0* as bounds [4].

Upper Stage Coast Arc

- Repeat above-mentioned steps to create a new phase named *Coast_AVUM* with a proper *Description* (e.g. *Upper Stage Coast Arc*).
- Insert a *Final Time* of *2000.0 Second* [4] with *1000.0* and *3300.0* as lower and upper bound, respectively.

Second AVUM Burn

- Repeat above-mentioned steps to create a new phase named *Second_AVUM_Burn* with a proper *Description* (e.g. *Second AVUM Burn*).
- Switch the *Phase span defined by* to *Duration* and insert a value of *400.0 Second* with *10.0* and *600.0* as lower and upper bound [4].
- Save the model using the **Save** button.

Note: The overlapping bounds between different phases is managed by the normalized time setting. If several warnings are printed during the initialization, please check this setting in the scenario name of the *Modelling* tree.

In the subsequent chapters, all individual phases are configured in detail.

2.7.2 Initial Phase Configuration

The first phase contains a vertical ascension to avoid any collision with the launch pad.

Tip: In case the simulation reveals that the thrust is smaller than the weight in the first seconds (during ignition), insert a phase before *LiftOff* with *Launch Pad* as equation of motion (see Launch Pad in Model Reference) to fix the vehicle at the launch pad during this period.

- Navigate to *Dynamics Configuration* in the *Modelling* tree and expand the *Vehicles and POIs Dynamics* node. Select the *Vega_Rocket* vehicle
- On top of the configuration panel, now all user-defined phases should be present as individual tabs together with two default tabs called *Initial State* and *Default Settings*.
- Select the *Initial State* tab. Set the *State type* to *Position and Velocity*. Change the *Frame* to *PCPF* and the *Representation* type to *Polar*. Make sure the *Altitude type* is set to *Altitude* and the *Latitude type* to *Latitude*.
- Enter the initial state data [5]:

  - *Altitude*: *0.0 Kilo-Meter*
  - *Longitude*: *-52.7744 Degree*
  - *Latitude*: *5.2356 Degree*
  - Make sure that the *Reference Frame* is set to *Relative PCPF*, the *Representation Frame* is set to *L* and the *Representation* type is set to *Cartesian*; in the *Velocity* section leave the radial, north and east velocities in this panel at zero.
- Select the *Default Settings* tab. Leave *Environment* in the quick-selection at the top as *Default*. Set the *Aerodynamics configuration* to *vega_aero* and the *Equation of Motion* type to *Inertial Velocity*.
- Switch to the *LiftOff* tab. Keep all options in the quick selection list at *Default* except for *Attitude* which has to be set to *Individual*.
- Ignore *Jettisoned assemblies*.
- In the *Active propulsion systems* table select the *P80_Engine_1*. Make sure no other engine is active.
- Scroll down to the *Attitude* section and expand it. Expand the *Yaw Angle* section and set the *Control Law* to *Constant Law*. Insert a *Yaw* value of *0.0* to specify the north direction. Activate the *Optimizable* option as *Parameter* and set the lower and upper bound to *0.0* and *91.5 Degree*.
- Expand the *Pitch Angle* section and change the *Control Law* to *Vertical Take Off*.
- Press **Save**.

Note: In case the *Optimizable* boxes are not visible, navigate to the top of the *Modelling* tree and verify that *Optimization* is checked in the *Active Features* section. If activation is not possible, your license does not cover the **Optimization** feature. Simply skip the corresponding steps in that case.

2.7.3 Pitch Over Phase Configuration

In the second phase VEGA starts to rotate with a fixed pitch rate. The yaw angle is kept constant.

- Select the *PitchOver* tab to define the settings for the pitch over phase.
- Leave all options in the quick selection as *Default* except for *Attitude* which has to be set to *Individual*.
- Select the *P80_Engine_1* as active engine in the *Active propulsion system* table and make sure no other engine is active.
- Modify the attitude by scrolling down to the *Attitude* section and expand it. Activate the *Use final value from previous phase* box for the yaw angle and leave all other *Yaw Angle* settings as they are (*Constant Law*).

- For the *Pitch Angle* change the *Control Law* to *Linear Law*. Activate the *Use final value from previous phase* box again and change the *Slope defined by:* to *Rate*. Insert a *Pitch Rate* of *-1.5 Degree/Second*.
- **Save**.

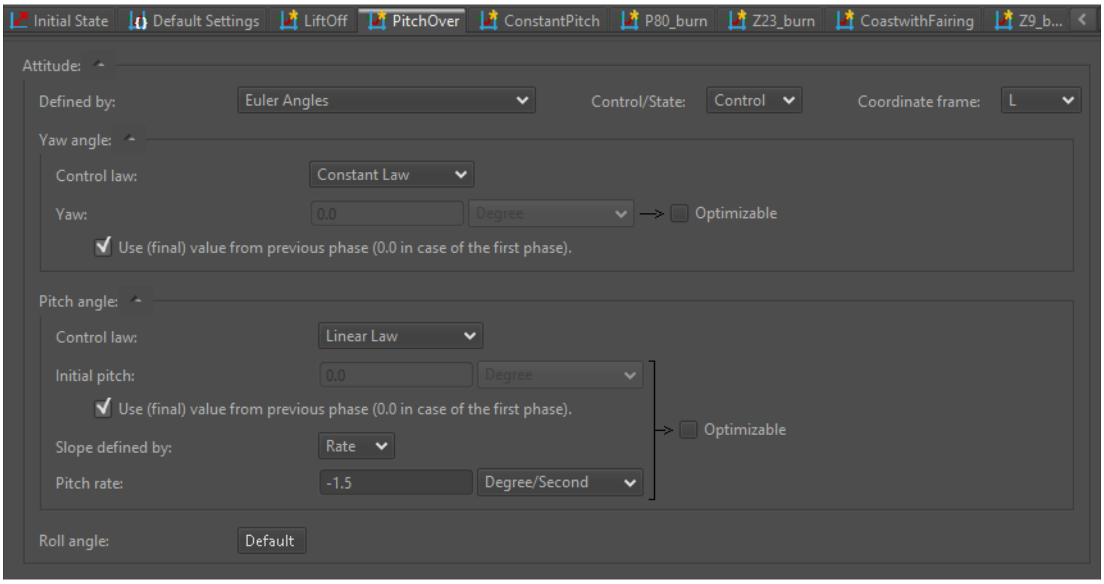


Fig. 2.6: Settings for the attitude control of pitch over phase

2.7.4 Constant Pitch Phase Configuration

Before starting the gravity turn phase, a pitch constant phase is required to reach an angle of attack of zero (gravity turn condition). The yaw direction is again kept constant.

- Select the *Constant_Pitch* tab to define the settings for the constant pitch phase.
- Leave all options in the quick selection as *Default* except for *Attitude* which has to be set to *Individual*.
- Select the *P80_Engine_1* as active engine in the *Active propulsion system* table and make sure no other engine is active.
- Modify the attitude by scrolling down to the *Attitude* section and expand it. Activate the *Use final value from previous phase* box for the *Yaw Angle* and leave all other settings as they are (*Constant Law*).
- For the *Pitch Angle* change the *Control Law* to *Constant Law* and activate the *Use final value from previous phase* box.
- **Save**.

2.7.5 P80 Burn Phase Configuration

This phase contains the first stage burn with a gravity turn control and a constant yaw orientation.

- Select the *P80_Burn* tab to define the settings for the P80 burn phase.
- Leave all options in the quick selection as *Default* except for *Attitude* which has to be set to *Individual*.
- Select the *P80* assembly in *Jettisoned assemblies* table and the *P80_Engine_1* as active engine in the *Active propulsion system* table. Make sure no other engine is active.
- Modify the attitude by scrolling down to the *Attitude* section and expand it. Activate the *Use final value from previous phase* box for the *Yaw Angle* and leave all other yaw settings as they are (*Constant Law*).
- For the *Pitch Angle* change the *Control Law* to *Gravity Turn*.
- **Save**.

2.7.6 Zefiro23 Burn Phase Configuration

This phase contains the second stage burn with a gravity turn control and constant yaw orientation.

- Select the *Z23_Burn* tab to define the settings for the Zefiro23 burn phase.
- Leave all options in the quick selection as *Default* except for *Attitude* which has to be set to *Individual*.
- Select the *Zefiro23* assembly in *Jettisoned assemblies* table and the *Z23_Engine_1* as active engine in the *Active propulsion system* table. Make sure no other engine is active.
- Modify the attitude by scrolling down to the *Attitude* section and expand it. Activate the *Use final value from previous phase* box for the *Yaw Angle* and leave all other yaw settings as they are (*Constant Law*).
- For the *Pitch Angle* change the *Control Law* to *Gravity Turn*.
- **Save**.

2.7.7 Coast with Fairing Phase Configuration

This phase contains a coast arc between the burnout of the Zefiro23 and the jettison of the fairing [4].

- Select the *Coast_with_Fairing* tab to define the settings for the coast phase with fairing.
- Leave all options in the quick selection as *Default* except for *Attitude* which has to be set to *Individual*.

- Select the *Vega_Fairing* assembly in *Jettisoned assemblies* table and make sure no engine is active in *Active propulsion system*.
- Modify the attitude by scrolling down to the *Attitude* section and expand it. Activate the *Use final value from previous phase* box for the *Yaw Angle* and leave all other yaw settings as they are (*Constant Law*).
- For the *Pitch Angle* change the *Control Law* to *Gravity Turn*.
- **Save**.

2.7.8 Zefiro9 Burn Phase Configuration

This phase contains the third stage burn with optimizable control to a constant pitch and yaw orientation.

- Select the *Z9_Burn* tab to define the settings for the Zefiro9 burn phase.
- Leave all options in the quick selection as *Default* except for *Attitude* which has to be set to *Individual*.
- Select the *Zefiro9* assembly in *Jettisoned assemblies* and the *Z9_Engine_1* as active engine in *Active propulsion system*. Make sure no other engine is active.
- Modify the attitude by scrolling down to the *Attitude* panel and expand it. Activate the *Use final value from previous phase* box for the *Yaw Angle* and leave all other yaw settings as they are (*Constant Law*).
- Activate *Optimizable* box and set *Optimizable as* to *Control*. Insert a *Lower/upper bound* of *-360.0 / 360.0 Degree*.
- For the *Pitch Angle* change the *Control Law* to *Constant Law* and activate the *Use final value from previous phase* box.
- Activate *Optimizable* box and set *Optimizable as* to *Control*. Insert a *Lower/upper bound* of *-90.0 / 90.0 Degree*.
- **Save**.

Note: In case the *Optimizable* boxes are not visible, navigate to the top of the *Modelling* tree and verify that *Optimization* is checked in the *Active Features* section. If activation is not possible, your license does not cover the Optimization **Optimization**. Simply skip the corresponding steps in that case.

Note: By selecting the *Profile Law* option instead of *Constant Law* as *Control Law* it would be possible to define a profile (function of time or altitude) to generate an initial guess, but since no information is provided by the references, a constant zero value is used instead (see Control Laws in Model Reference).

2.7.9 First AVUM Burn Phase Configuration

Since the target is a circular orbit at 700 km altitude, two upper stage burns are foreseen to achieve it. This phase contains the first burn of the AVUM stage. The upper stage phase has an optimized constant attitude control.

- Select the *First_AVUM_Burn* tab to define the settings for the first AVUM burn phase.
- Leave all options in the quick selection as *Default* except for *Attitude* which has to be set to *Individual*.
- Select the remaining *AVUM_Engine_1* as active engine in *Active propulsion system*. No assembly is jettisoned.
- Modify the attitude by scrolling down to the *Attitude* panel and expand it. Activate the *Use final value from previous phase* box for the *Yaw Angle* and leave all other yaw settings as they are (*Constant Law*).
- Activate *Optimizable* box and set *Optimizable as* to *Control*. Insert a *Lower/upper bound* of *-360.0* / *360.0 Degree*.
- For the *Pitch Angle* change the *Control Law* to *Constant Law* and activate the *Use final value from previous phase* box.
- Activate *Optimizable* box and set *Optimizable as* to *Control*. Insert a *Lower/upper bound* of *-90.0* / *90.0 Degree*.
- **Save**.

Note: In case the *Optimizable* boxes are not visible, navigate to the top of the *Modelling* tree and verify that *Optimization* is checked in the *Active Features* section. If activation is not possible, your license does not cover the **Optimization** feature. Simply skip the corresponding steps in that case.

2.7.10 Upper Stage Coast Arc Phase Configuration

This phase contains a long coast arc between the two AVUM burns while the vehicle passes over the North Pole.

- Select the *Coast_AVUM* tab to define the settings for the upper state coast phase.
- Leave all options in the quick selection as *Default* except for *Attitude* **and** *Equations of motion* which have to be set to *Individual*.
- Unselect the *AVUM_Engine_1* as active engine in *Active propulsion system*. No assembly is jettisoned.
- **Important**: Change the *Equation of motion* type to *Inertial Cartesian*, due to the singularity present over the North Pole for the classical equations of motion.
- Modify the attitude by scrolling down to the *Attitude* panel and expand it. Activate the *Use final value from previous phase* box for the *Yaw Angle* and change the *Control Law* to *Linear Law*.

- Set the *Final yaw* to *180.0* degree.
- Activate *Optimizable* box and set *Optimizable as* to *Parameter*. Insert a *lower/upper bound* of *-360.0 / 360.0 Degree* for both the initial and final bounds.
- Apply the same settings for the *Pitch Angle* with *Final pitch* set to *0.0* degree.
- **Save**.

2.7.11 Second AVUM Burn Phase Configuration

This phase contains the second burn of the AVUM stage, it has an optimized attitude control with a guidance for the initial guess (thrust in south direction).

- Select the *Second_AVUM_Burn* tab to define the settings for the second AVUM burn phase.
- Leave all options in the quick selection as *Default* except for *Attitude* which has to be set to *Individual*.
- Select the remaining *AVUM_Engine_1* as active engine in *Active propulsion system*. No assembly is jettisoned.
- Modify the attitude by scrolling down to the *Attitude* section and expand it. For the *Yaw Angle* change the *Control Law* to *Target Orbit*. Insert *90.0 Degree* for the *Target inclination*, set the *Orbit leg* to *Automatic*, enable *Optimizable* and insert a *Yaw lower/upper bound* of *-360.0 / 360.0 Degree*.
- Select *Required velocity* as *Control Law* in the *Pitch Angle* section. Set this law to achieve an orbit with *Target periapsis altitude* and a *Target apoapsis altitude* of *700.0 Kilo-Meter*. Define a *Default pitch* of *10.0 Degree* and the *Min. thrust to effective weight ratio* to *1.1*. Enable the *Optimizable* area and enter a *Pitch lower/upper bound* of *-90.0 / 90.0 Degree*.
- **Save**.

2.7.12 Check of the Phase Sequence

The phase sequence has been defined in the model till the end of the mission: the payload deployment. The configured data can be used to initialize and simulate the trajectory of VEGA. This procedure reveals potential problems or errors in the model.

- In the *ASTOS* ribbon, press the *Start→ Actions→ Initialize* button.
- In spite of Murphy's law (squared) no error should be present: `Process terminated with status 0`
- If the simulation is not automatically performed (this information can be found in the execution log), press the ribbon task *Start→ Actions→ Simulate Multiple Shooting* button.
- Follow the instruction presented in Chapter 5 to visualize the trajectory.

The list of functions and properties that should be checked is case-dependent, but some guidelines could be provided on a generic basis:

1. Altitude (position, relative) vs. flight-time; to verify the trajectory profile.
2. Flight path angle (velocity, relative) and pitch_I (attitude) vs. flight-time; to check the first 10 seconds of the mission (vertical take-off, pitch over and constant pitch) and the gravity turn condition: pitch = flight path angle.
3. Thrust to effective weight ratio (performance) vs. flight-time; this is an important issue in the first seconds of the mission (to check if the thrust is larger than the weight).
4. Propulsion consumption of a stage (PROP_TANK@XX_Stage) vs. flight-time; to verify the correctness of the burn duration, i.e. all but not more propellant is burnt.
5. Map-plot or Satellite plot; in case of polar orbits, better to use Stereographic or Orthographic projection instead of Mercator.
6. Heat flux density (Aerodynamics, Thermal) vs. flight-time; to check the fairing separation.
7. Perigee and apogee altitude vs. flight-time; to verify the two-burn strategy.

ASTOS automatically saves the current set of graphs, but it is possible to store this information in a file (`my_name.gavc`) with the *File*→ *Save as...* command in the *Viewer* window. To load a saved set of graphs simply select *File*→ *Load* in the *Viewer* window.

Tip: The name of components and propulsion systems affect the name of the auxiliary functions. In case a component name changes, the auxiliary function (or state) needs to be reselected in an existing plot.

2.8 Add Constraints

In case of a trajectory simulation, the required steps are the definition of the environment, components, aerodynamics and phase sequence. An add-on could be to add some phase conditions: a condition used to stop a phase not at a pre-defined time but using another function (e.g. altitude or heat-flux-density). These conditions are set togeter with the phase time, but not used in this tutorial. For details on how to use them please refer to chapter Additional Phase End Conditions in the User Manual book.

Constraints are required in case of optimization: during this process the parameters are modified in order to minimize a cost function (see Section 2.9), but the parameter space is limited from a physical point of view. In order to restrict such modifications, the user can define bounds for the parameters or constraints of computed functions (e.g. altitude).

ASTOS includes four types of constraints: initial, final, path and parameter constraints. The first group is evaluated at the beginning of the phase, the second at the end and the third one is evaluated during the complete phase (i.e. at the constraint refinement grid nodes). *ASTOS* neither has a limitation on the number of constraints that could be set, nor in the number of phases in which a constraint can be active. For further details, please refer to the Constraints in the User Manual.

2.8.1 Initial Constraints

ASTOS is capable of modifying the initial state during the optimization process to improve the vehicle performance. Since this is not desired within this tutorial, the respective six states need to be constrained.

- Select ribbon task *Add*→ *Optimization*→ *Constraint*.
- Insert *Initial_Altitude* as *Identifier*.
- Select *All* from the *Type* drop-down list, *Altitude* as *Subtype* and press the **Create** button.
- Add two more constraints for Longitude and Latitude in a similar way as described above. Call them *Initial_Longitude* and *Initial_Latitude* with appropriately chosen *Subtype* selections.
- Navigate to the *Optimization* tree below the *Modelling* tree (enlarge it by dragging or double-clicking if necessary). Expand the *Constraints* node and select the newly created *Initial_Altitude* constraint.
- Insert a description (optional).
- Select the *Vega_Rocket* as *Vehicle ID* and check the *LiftOff* phase in the *Specified* column.
- Set it up as *Initial Boundary* constraint.
- Insert *0.0 Kilo-Meter* as reference value.
- Repeat these steps for the longitude (*-52.7744 Degree*) and latitude (*5.2356 Degree*) constraint.

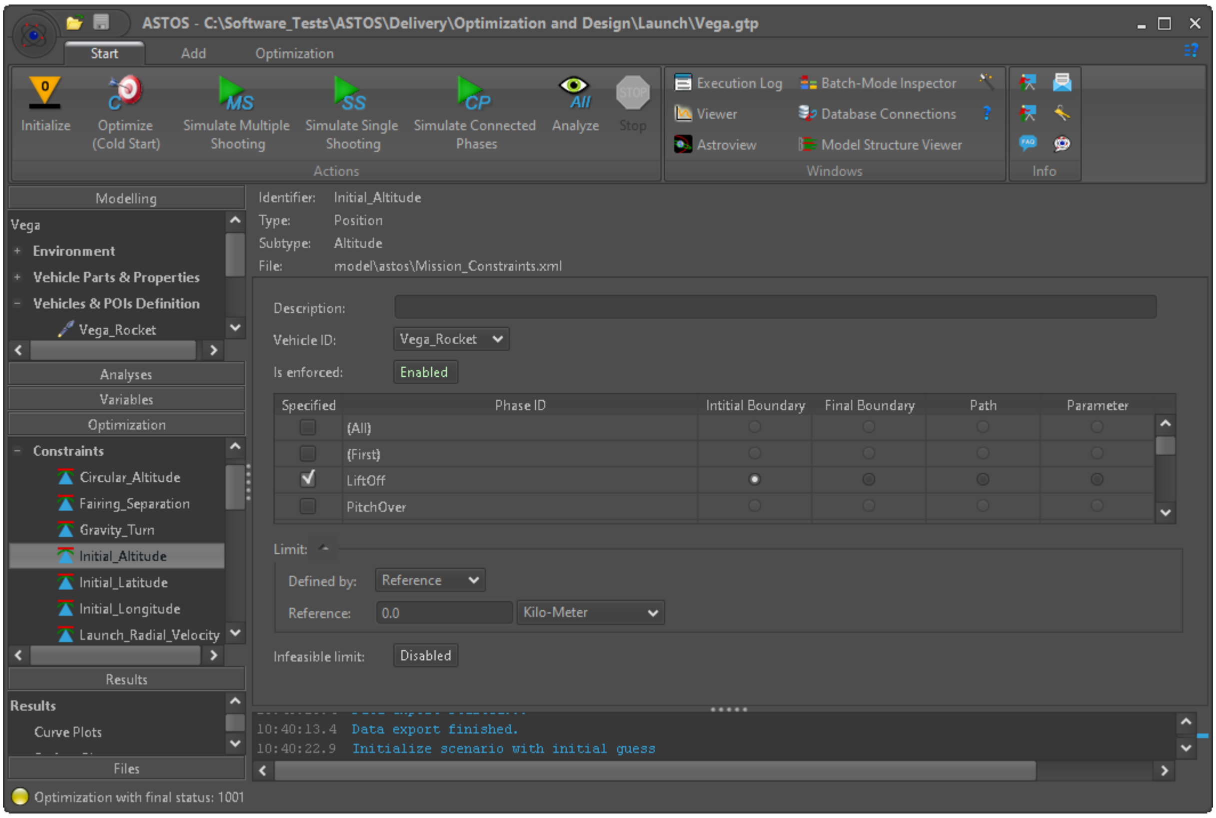


Fig. 2.7: Initial altitude constraint for VEGA

Analogously, create the following three constraints using *Add→ Optimization→ Constraint* with reference values equal to zero and assigned as *Initial Boundary* to the *LiftOff* phase:

- *Initial_Radial_Velocity* as *Identifier* and *Radial_Velocity* as *Subtype*
- *Initial_Rel_East_Velocity* as *Identifier* and *Rel_East_Velocity* as *Subtype*
- *Initial_North_Velocity* as *Identifier* and *North_Velocity* as *Subtype*
- **Save**

With these six constraints, the initial state is fixed and the optimizer retains the values.

2.8.2 Final Constraints

Apart from the mission requirements - final circular orbit at 700 km altitude and 90° inclination - there are other constraints derived from physical properties or safety issues. The complete procedure to define a final constraint is presented for one constraint only. The remaining ones can be set up analogously.

- Select *Add→ Optimization→ Constraint*.
- Enter *Launchpad* as *Identifier*.
- Select *All* as *Type*, *Altitude* as *Subtype* and press the **Create** button.

- Select the just created constraint (usually it should already be selected).
- Insert a description (optional).
- Select the *Vega_Rocket* as *Vehicle ID*.
- Check the *LiftOff* phase in the *Specified* column and set it up as *Final Boundary*.
- Select *Lower Limit* from the *Limit_Type* drop-down list.
- Enter *0.06 Kilo-Meter* (twice the height of VEGA) in the *Lower limit* field.
- **Save**

This constraint ensures a minimum altitude of 60 m (0.06 km) at the end of the first phase just before the pitch over maneuver for launch tower clearance reason.

Gravity Turn

- *Identifier*: *Gravity_Turn*
- *Type*: *Controls*
- *Subtype*: *Gravity Turn Condition*
- Specified in phase: *Constant_Pitch*
- Boundary Type: *Final Boundary*
- *Limit Type*: *Reference*
- *Reference*: *0.0 Degree*

This constraint assures the restoring of the condition (zero angle of attack) before the phases with gravity turn control.

Residual Propellant
ASTOS automatically realizes a final boundary constraint on the residual propellant (> 0) when a stage is jettisoned: this checks that the burn duration has not been too long. In this mission, the burn duration of P80, Z23 and Z9 are fixed number provided by the reference [4].

Fairing Separation

- *Identifier*: *Fairing_Separation*
- *Type*: *Various*
- *Subtype*: *Heat Flux*
- Specified in phase: *Coast_with_Fairing* and *Z9_Burn*
- Boundary Type: *Final Boundary*
- *Limit Type*: *Upper Limit*
- *Upper Limit*: *1135.0 Watt/Meter**2* [4]

The fulfillment of this constraint decides about the jettisoning of the fairing (and consequently about the related phase durations).

Periapsis Altitude

- *Identifier*: *Periapsis_Altitude*
- *Type*: *Orbit*
- *Subtype*: *Periapsis_Altitude*
- Specified in phase: *First_AVUM_Burn*
- Boundary Type: *Final Boundary*
- *Limit Type*: *Lower Limit*
- *Lower Limit*: *100.0 Kilo-Meter*

This constraint is not mandatory, but it increases the general safety of the mission, as it implies a higher perigee before the long coast arc of the AVUM upper stage.

Orbit Altitude

- *Identifier*: *Orbit_Altitude*
- *Type*: *All*
- *Subtype*: *Altitude_Equatorial*
- Specified in phase: *Second_AVUM_Burn*
- Boundary Type: *Final Boundary*
- *Limit Type*: *Reference*
- *Reference*: *700.0 Kilo-Meter*

This constraint assures that the final altitude (over the Equator) corresponds to the desired target altitude.

Circular Altitude

- *Identifier*: *Circular_Altitude*
- *Type*: *Orbit*
- *Subtype*: *Circular Altitude*
- Specified in phase: *Second_AVUM_Burn*
- Boundary Type: *Final Boundary*
- *Limit Type*: *Reference*
- *Reference*: *700.0 Kilo-Meter*

This constraint assures that the final inertial velocity corresponds to the one required for a circular orbit at a certain altitude.

Radial Velocity

- *Identifier*: *Radial_Velocity*
- *Type*: *All*
- *Subtype*: *Radial Velocity*
- Specified in phase: *Second_AVUM_Burn*

- Boundary Type: *Final Boundary*
- *Limit Type*: *Reference*
- *Reference*: *0.0 Meter/Second*
- *Infeasible Limit*: *1000.0 Meter/Second*

This constraint assures that the final radial velocity corresponds to the one required for a circular orbit (i.e. zero). *Infeasible Limit* represents a scale factor for the error, see Phase-End Conditions and Optimization Constraints of the Model Reference.

Orbit Inclination

- *Identifier*: *Orbit_Inclination*
- *Type*: *Orbit*
- *Subtype*: *Orbit Inclination*
- Specified in phase: *Second_AVUM_Burn*
- Boundary Type: *Final Boundary*
- *Limit Type*: *Reference*
- *Reference*: *90.0 Degree*
- *Reference frame*: *J2000*
- **Save**.

This constraint assures that the final orbit orientation corresponds to the desired polar orbit.

2.8.3 Path Constraints

The heat flux density has been constrained at the jettisoning of the fairing (end of the coast phase with fairing and the next phase). But it is also important to control the heat flux until deployment of the payload or at least in the next phases, as the latter is unprotected.

- Navigate to the *Constraints* node in the *Optimization* tree
- Select the *Fairing_Separation* constraint and clone it.
- Select the created *Fairing_Separation_CLONE* constraint and rename it to *Path_Heat_Flux*.
- In the configuration panel, deselect the *Specified* phase *Coast_with_Fairing* and select the phase *Firs_AVUM_burn* instead (*Z9_Burn* stays selected).
- Change the constraint type to *Path* for the two active phases.
- Leave the *Limit Type* and *Upper Limit* unchanged.
- **Save**.

With this constraint, *ASTOS* checks the values of the heat flux density throughout the two phases after the jettison of the fairing.

Note: A path constraint is only evaluated at certain "Constraint Evaluation Nodes" which are defined later in Section 4.2.

2.9 Cost Functions

The cost function is defined in the *Optimization* tree. The general settings that were present in mission definition (i.e. normalized time) are now moved to the top end of the *Modelling* tree. The possibility to activate the integration of auxiliary states (i.e. DeltaV) are instead present in the *Default settings* inside *Vehicles & POIs Dynamics*. For further details, please refer to Scenario General Settings and Auxiliary States paragraph in User Manual.

- Navigate to *Cost Function Terms* node in the *Optimization* tree and select it.
- Click on *Add*→ *Optimization*→ *Cost Function*. Insert an *Identifier*, (e.g. *Max_Payload*) select *Max Payload* as *Type* and press the **Create** button.
- Select the newly created cost function by clicking on *Max_Payload* now present below the *Cost Function Terms* node in the *Optimization* tree.
- Select *Vega_Rocket* as *Vehicle ID* and enter *1.0E-4* as *Scaling* value: since the payload is defined in kilogram (limited between 750 kg to 3000 kg), this scaling produces a cost function with magnitudes between 0.1 and 1.0 (absolute values), the best range for optimization purpose.
- Leave the other settings unchanged.
- **Save**.

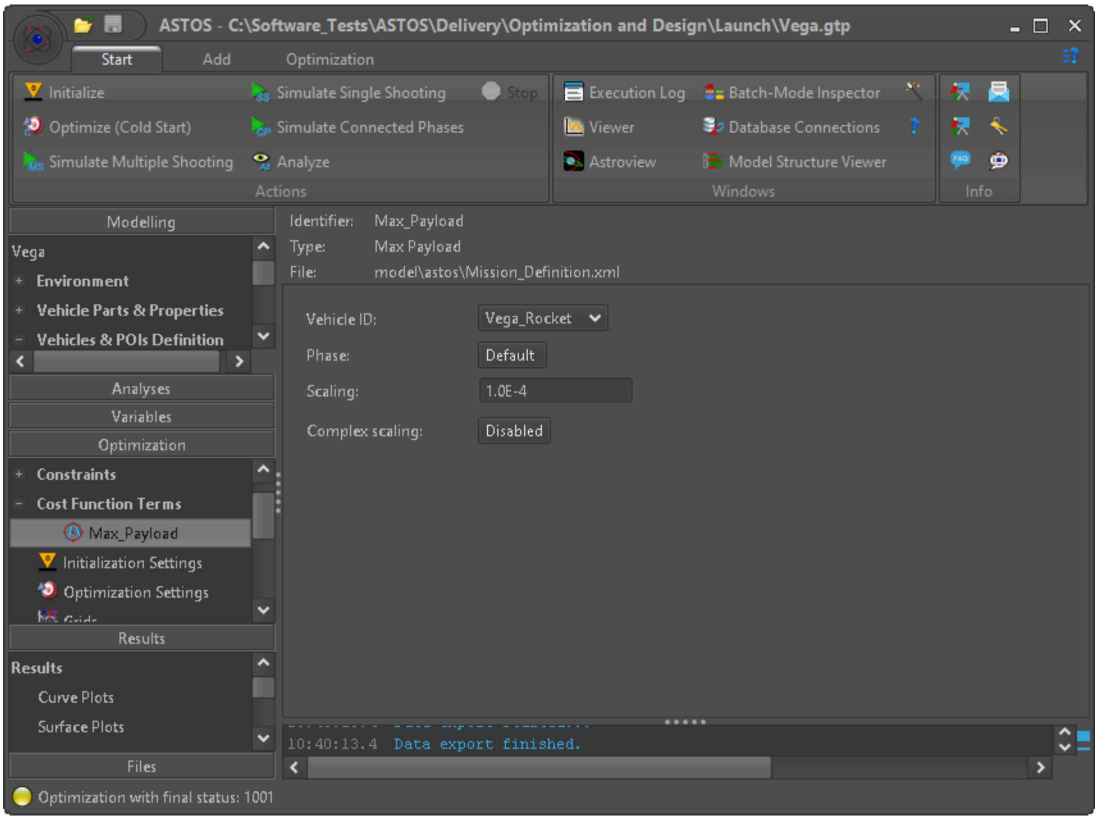


Fig. 2.8: Cost function settings for VEGA

3 Initialize the Launcher

At this point, the complete model for has been defined in *ASTOS*. The data can now be used to initialize and simulate the trajectory of VEGA.

- In the *ASTOS* ribbon, press the *Start*→ *Actions*→ *Initialize* button.
- If all the preceding steps are properly realized, no error should be present (`Process terminated with status 0`).
- Simulate the trajectory with *Start*→ *Actions*→ *Simulate Multiple Shooting*.
- Follow the instruction presented in Chapter 5 to visualize the trajectory and to use the Result Summary.

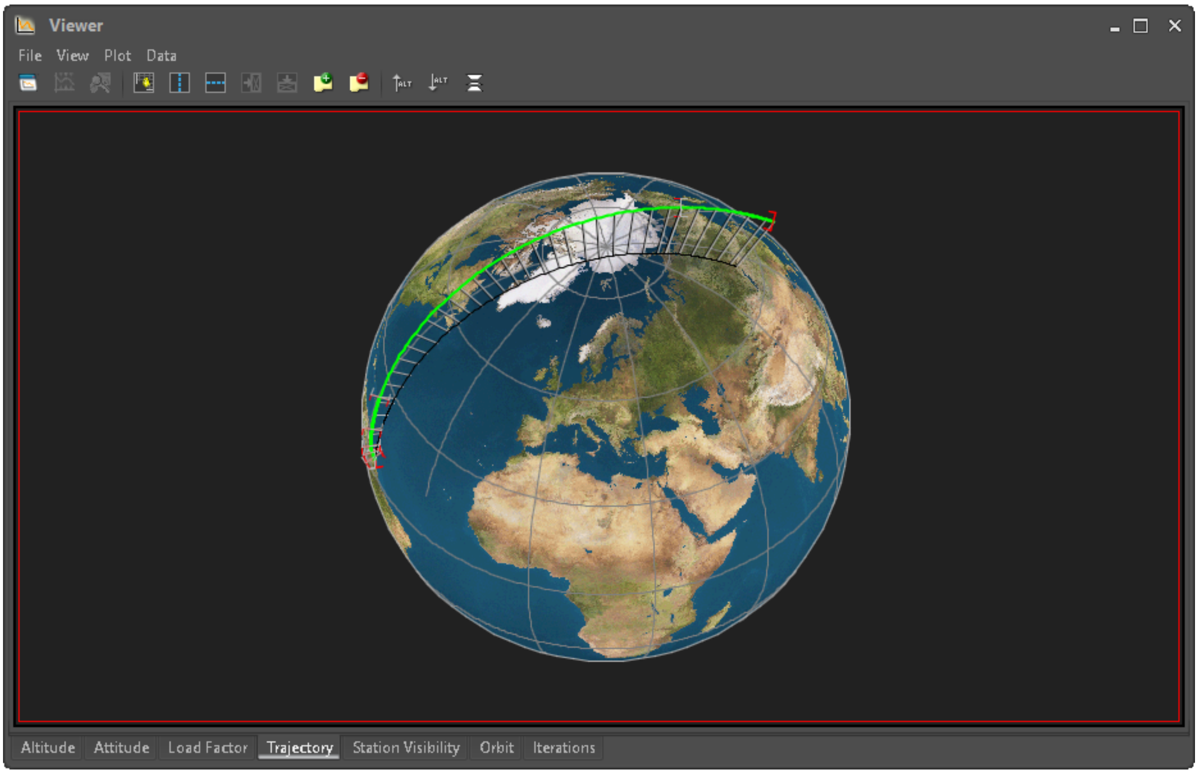


Fig. 3.1: Initial trajectory of VEGA in satellite view

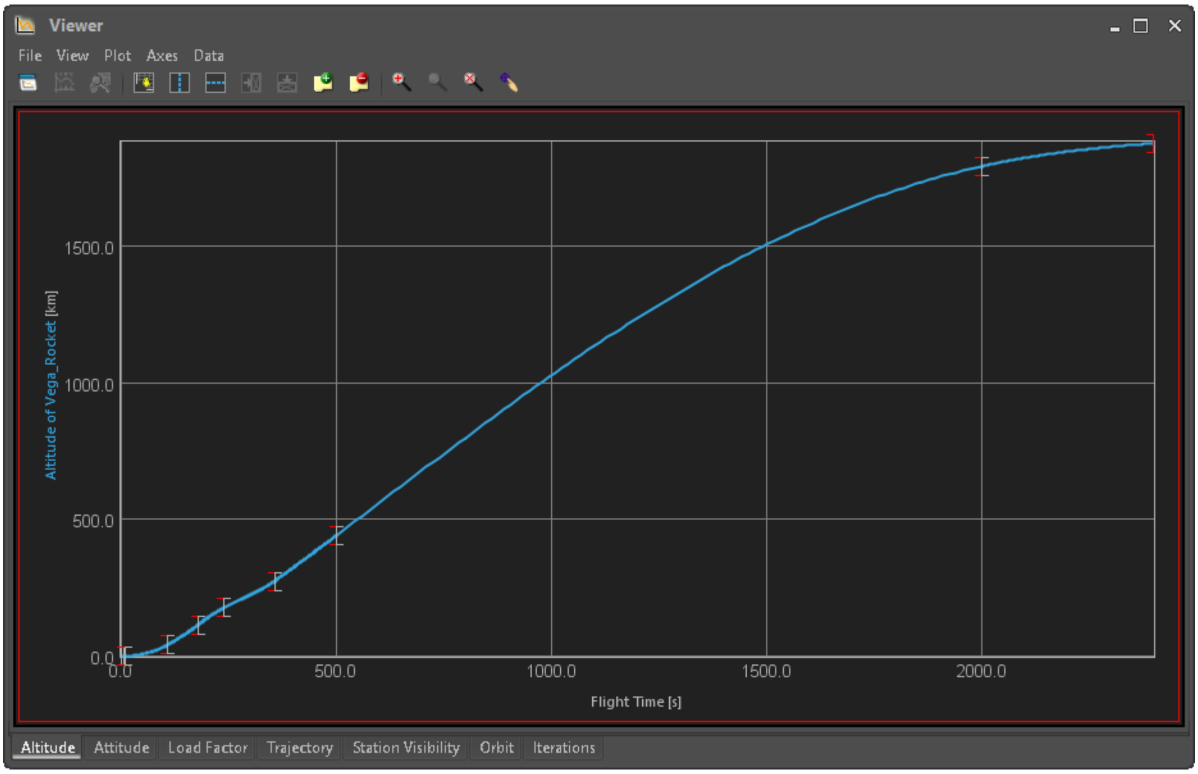


Fig. 3.2: Initial trajectory of VEGA altitude profile

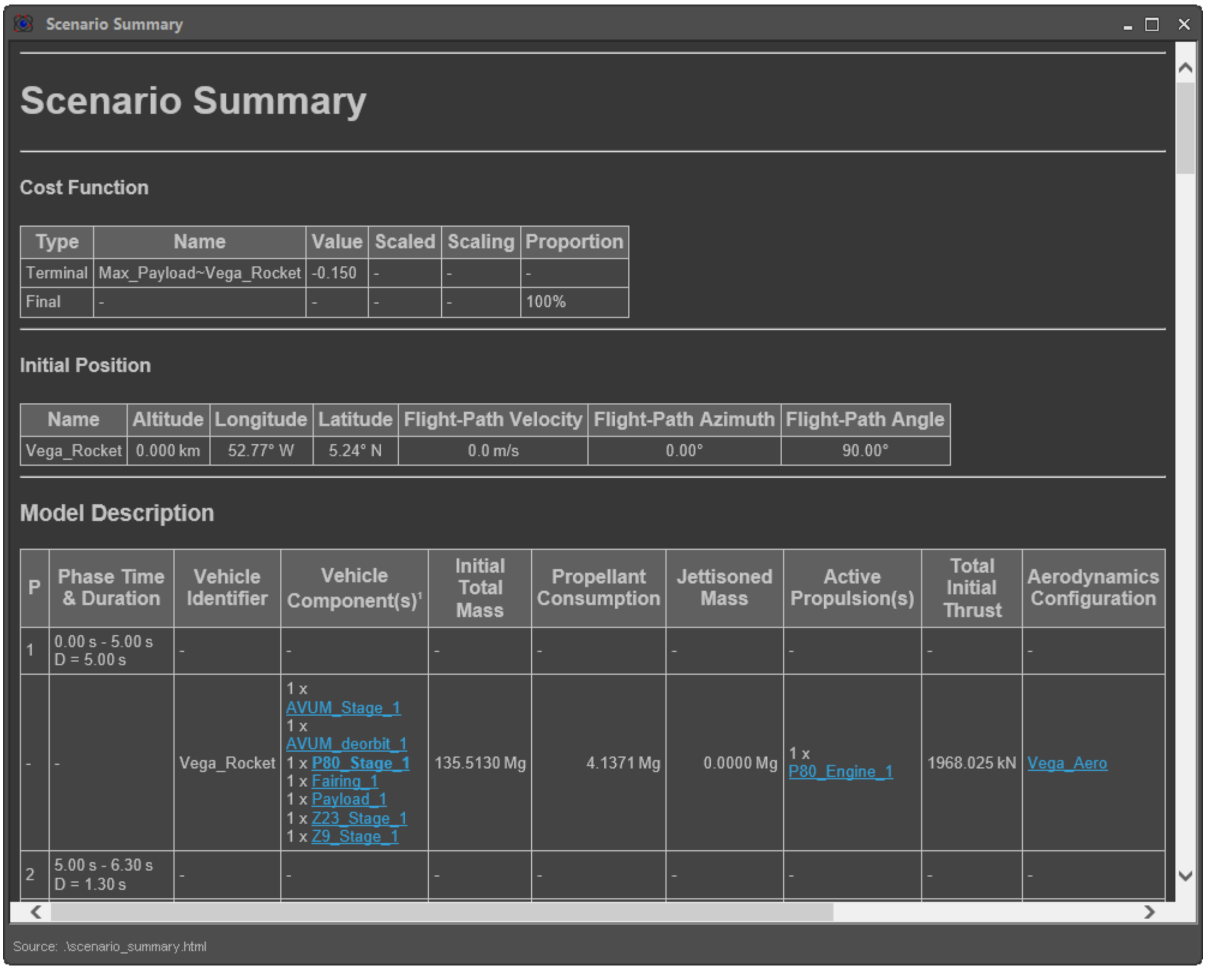
Scrolling through the Result Summary printout in the *Execution Log*, it is possible to note that violations are only present in the target orbit and in the perigee altitude: the initial guess is not violating any mission constraint. This is a good starting point for an optimization.

> **Tip:** In case there are large violations in the results, there might be errors in the constraint definition, please double check your entries.

Summary

There is an additional functionality in *ASTOS* which supports the identification of model errors: the Summary. Select ribbon task *Start*→ *Info*→ *Scenario Summary*. A properties file (`scenario_summary.html`) is created in the `.gtp` folder which contains a summary of defined phases, components, environment, constraints and correlations between the elements that compose the mission.

> **Tip:** On *Linux* platforms the opening of the `scenario_summary.html` file with the default web browser requires the installation of the *Gnome* library. In the case of problems, please open this file manually.


Scenario Summary

Scenario Summary

Cost Function

Type	Name	Value	Scaled	Scaling	Proportion
Terminal	Max_Payload~Vega_Rocket	-0.150	-	-	-
Final	-	-	-	-	100%

Initial Position

Name	Altitude	Longitude	Latitude	Flight-Path Velocity	Flight-Path Azimuth	Flight-Path Angle
Vega_Rocket	0.000 km	52.77° W	5.24° N	0.0 m/s	0.00°	90.00°

Model Description

P	Phase Time & Duration	Vehicle Identifier	Vehicle Component(s)[1]	Initial Total Mass	Propellant Consumption	Jettisoned Mass	Active Propulsion(s)	Total Initial Thrust	Aerodynamics Configuration
1	0.00 s - 5.00 s D = 5.00 s	-	-	-	-	-	-	-	-
-	-	Vega_Rocket	1 x AVUM_Stage_1 1 x AVUM_deorbit_1 1 x P80_Stage_1 1 x Fairing_1 1 x Payload_1 1 x Z23_Stage_1 1 x Z9_Stage_1	135.5130 Mg	4.1371 Mg	0.0000 Mg	1 x P80_Engine_1	1968.025 kN	Vega_Aero
2	5.00 s - 6.30 s D = 1.30 s	-	-	-	-	-	-	-	-

Source: .\scenario_summary.html

Fig. 3.3: Example of Scenario Summary for VEGA (extract)

This tutorial presents a "well-defined" scenario. The numbers used (phase durations, component models, etc.) are already chosen to produce a good initial guess. In a "normal" situation, the design of an initial guess for launcher is not a straight forward process: after each phase insertion, it is the best practice to initialize and simulate the trajectory to identify immediately a design error.

4 Optimize the Launcher

Before the optimization can be started, some preparatory steps are necessary. This involves certain optimization settings and the insertion of the required nodes.

4.1 Optimization Settings

This tutorial only describes how to define the specific settings selected for this scenario. It does not provide any background information on why certain methods and values are chosen. For a detailed description of the different methods and theory, please refer to the book Optimization Theory and Description of Methods or to the section Optimization Settings in User Manual).

- Navigate to the *Optimization* tree and select *Optimization Settings*.
- Check that *CAMTOS* is selected as *Optimization method*.
- Check that *WORHP* is selected as *NLP Solver* (middle right in the panel).
- Change the value for the *Max. Iterations* cell to *200*.
- Change the value of the iteration output frequency for the *Execution log* cell to *1*.
- Verify that all other values are as depicted in Fig. 4.1.
- Press **Select Diagnostic Output** and verify the settings are as in Fig. 4.2 (deactivate *Show a list of all constraints*).

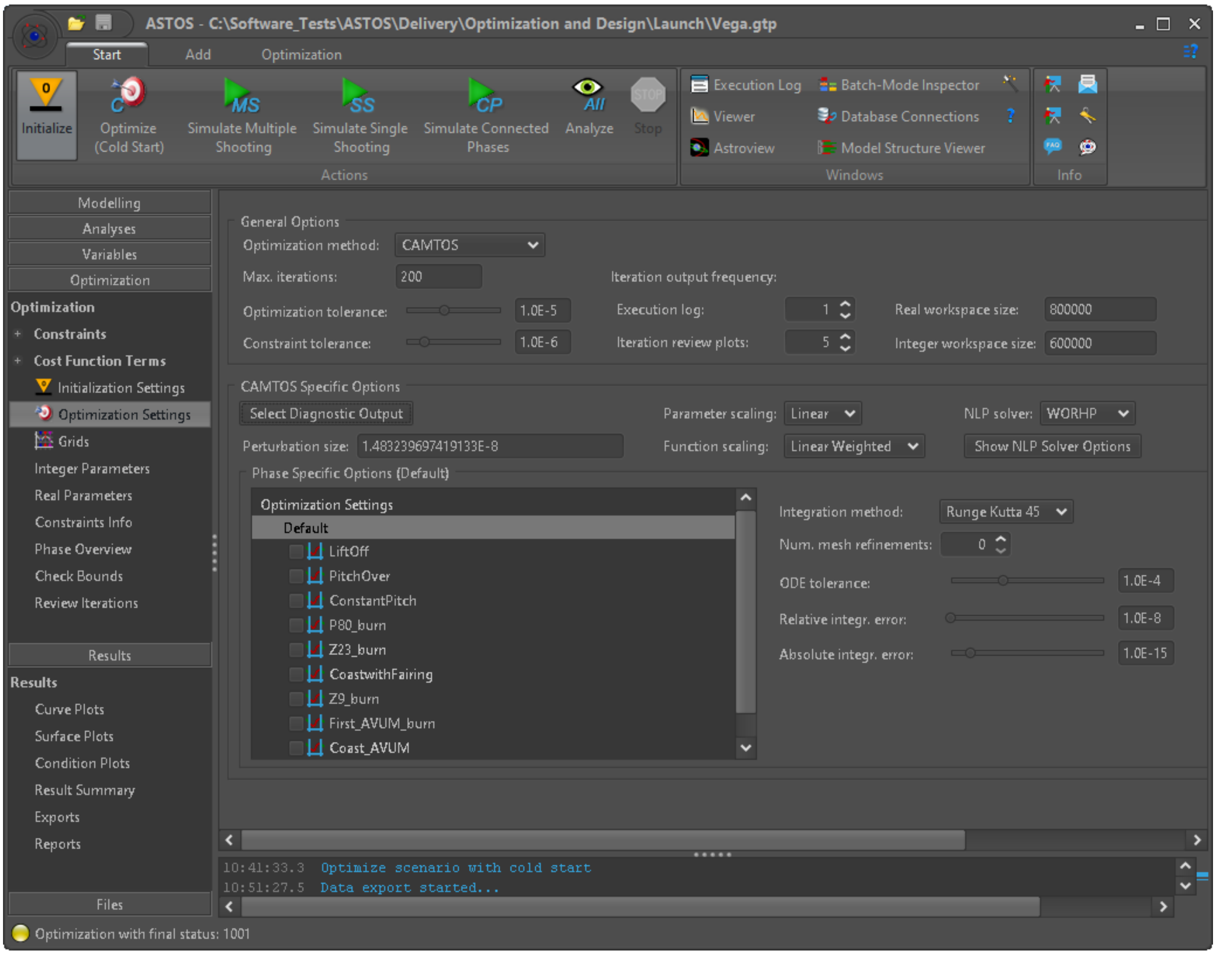


Fig. 4.1: Optimization Settings for the VEGA scenario

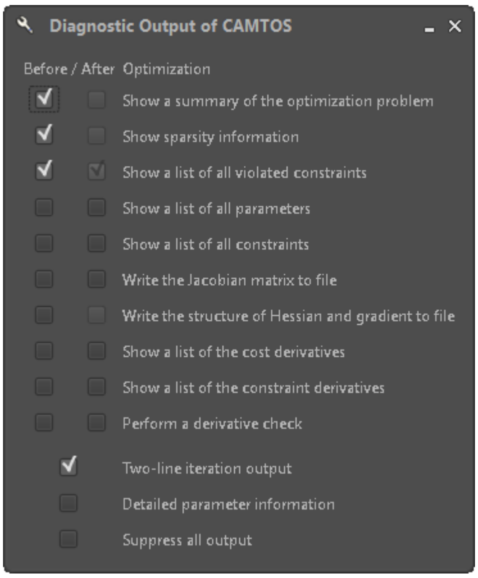


Fig. 4.2: Diagnostic Output window for CAMTOS

4.2 Grid Nodes

For the presented scenario, a multiple shooting transcription method is used (refer to Direct Multiple Shooting Method in Optimization Theory and Description of Methods). As a general guideline in case of multiple shooting transcription methods, only very long phases require additional major grid nodes. However, constraint evaluations and the control refinement nodes for some specific phases always need to be defined.

Major Grid

The tutorial mission comprises a long coast phase between the two burns of the AVUM upper stage.

1. Navigate to the *Optimization* tree and select *Initialization Settings*.
2. Press the **Update Model Data** button to read the phase names from the Model.
3. Press the "+" at the *Coast_AVUM* node to expand it.
4. Select the *Major Grid* flag to activate the respective area in the right panel (red framed).
5. Insert *4* in the first cell of the right panel to insert four equidistant node.
6. Press ENTER (keyboard).
7. Repeat the same procedure (3-6) for the *P80_Burn* (1 major grid node) and the *First_AVUM_burn* (2 major grid nodes) phases.

Constraint Evaluation

ASTOS evaluates the path constraints only at some specific points, i.e. at constraint evaluation nodes. For this mission, there is a path constraint (heat flux density) only in the Zefiro9 and the first AVUM burn phase. Thus, nodes need to be defined only for these phases.

1. Navigate to the *Optimization* tree and select *Initialization Settings*.
2. Press the "+" at the *Z9_Burn* to expand it.
3. Select the *Constraint Evaluation* flag to activate the respective area in the right panel (red framed).
4. Insert *9* in the first cell of the right panel to insert 9 equidistant nodes.
5. Press ENTER (keyboard).
6. Repeat the same procedure (3-6) for the *First_AVUM_Burn* phase.

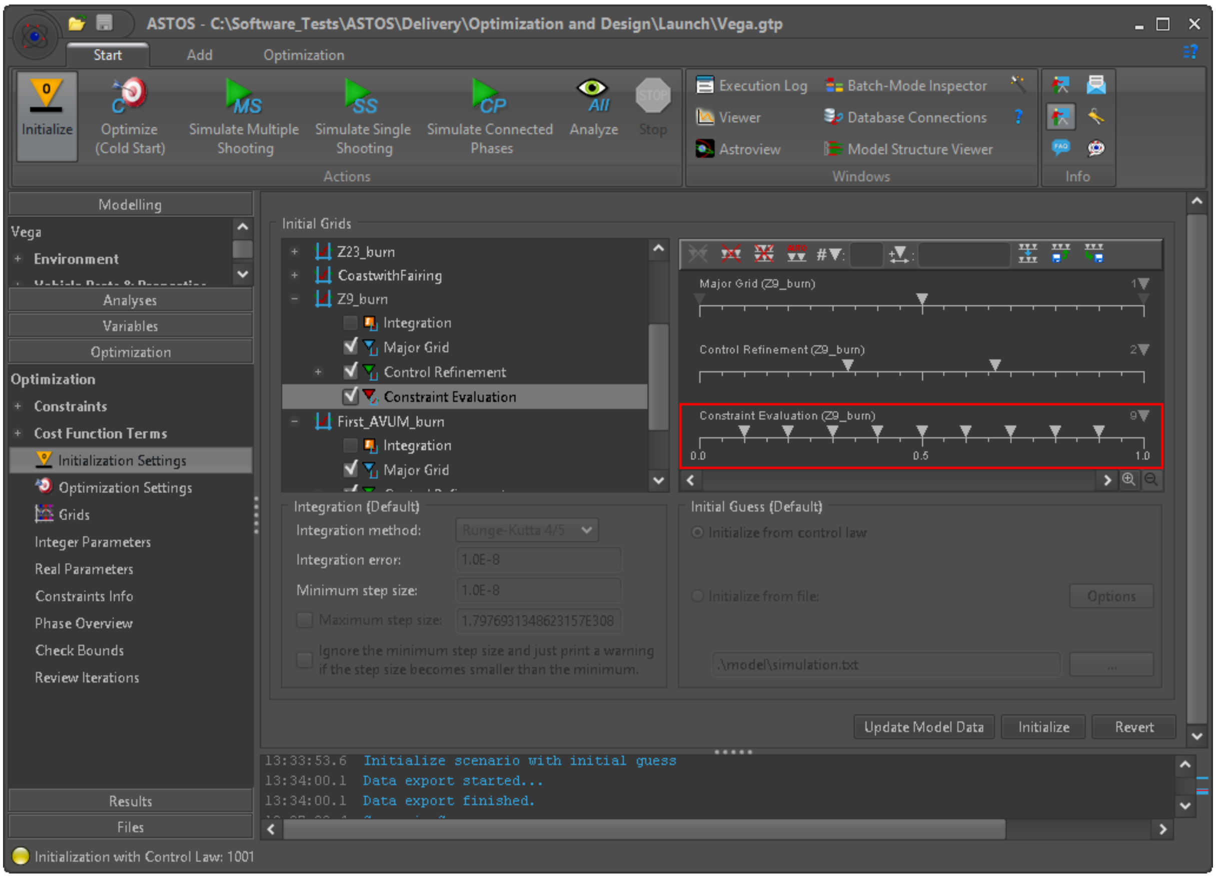


Fig. 4.3: Adding constraint evaluation grid nodes using the Initial Grids panel

Control Refinement

As explained in the Direct Multiple Shooting Method chapter of the Optimization Theory and Description of Methods book, the selected control law is linearly interpolated between the major and control refinement grid nodes. In Section 2.7 an optimizable control has been selected for the Zefiro9 and the two upper stage burn phases. Thus, it is necessary to define some nodes which allow generation of a control profile. The lower stage phases, with gravity turn, do not require any control refinement node (analytical law).

The procedure is similar to the definition of constraint evaluation nodes:

- Navigate to the *Optimization* tree and select *Initialization Settings*.
- Select *Z9_Burn* phase which should still be expanded.
- Select the *Control Refinement* flag to activate the respective area in the right panel (red framed).
- Insert *2* in the first cell of the right panel to insert 2 equidistant nodes.
- Press ENTER (keyboard).
- Repeat the same procedure for the *First_AVUM_Burn* and the *Second_AVUM_Burn* phase, but create ***4*** equidistant nodes for these phase.

4.3 Run the Optimization

To include the previously created grid nodes in the model, the scenario has to be initialized again. Press the **Initialize** button at the bottom of the *Initial Grids* panel or use ribbon task *Start*→ *Actions*→ *Initialize*.

Start the optimization with *Start*→ *Actions*→ *Optimize (Cold Start)*. The *Execution Log* first displays a summary of the optimization problem. After `ITERATION INFORMATION` a matrix is printed where each optimization step corresponds to a row, while the columns are intermediate information produced by *CAMTOS* (see Fig. 4.4).

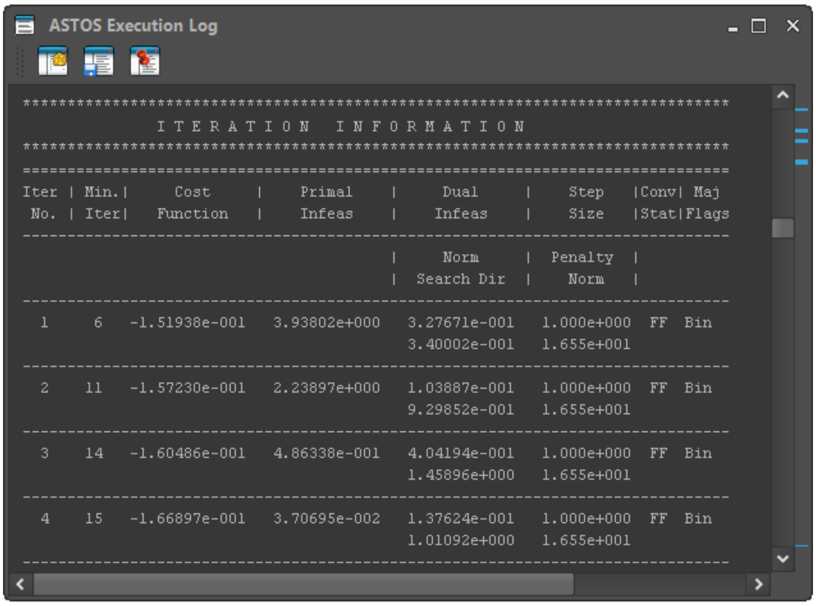


Fig. 4.4: Iteration information of the optimization of VEGA (iteration steps > 4 have been cut)

A detailed description of the meaning of all the quantities printed out during the optimization process is presented in the *ASTOS* User Manual (Monitoring Optimization Iterations in Optimization Theory and Description of Methods). The most important ones are the `Cost Function` values (3rd column) and the `Conv Stat` (a set of last two capital letters on the right). In particular when these letters are TT (true-true) the scenario is successfully optimized: the constraint violation (4th column) is below a specified tolerance and the result found is below the specified tolerance (see the Optimization Settings in User Manual)).

For this scenario, the optimization stops with an optimal solution for the cost function (maximum payload). This is indicated by the green ball appearing in the status bar at the bottom of the main window. This status information can alternatively be retrieved from the *Execution Log* >>> `Optimal solution found <<< ( 0)` printed below `FINAL INFORMATION`.

Non-Optimal Results

In general, it may occur that a final solution is found, but not optimal. In this case, `Feasible solution, but the requested accuracy not achieved( 4)` is printed below `FINAL`

INFORMATION in the *Execution Log* or indicated in the status bar. One reason for such result might be an improper initial scaling of the cost function.
If no Violated Constraints are present or the violations are very small, save the TOPS result file as result.tops (*Optimization→ TOPS→ Save TOPS as...*) and select *Optimization→ Actions→ Optimize (Cold Start)* again. This restart now uses the results of the previous iteration at initial guess. Very often the final optimal solution is found within a couple of steps. Overwrite result.tops to store the final status of the second optimization. As default the optimized solution is saved in the working file (input.tops) and at each action (initialization, simulation, optimization) the final status is saved in this file overwriting the previous solution. To save an important result permanently (the optimal one), use *Optimization→ TOPS→ Save TOPS as...* and insert a filename (e.g. result.tops). At any time it is possible to reload this file with the *Optimization→ TOPS→ Open TOPS* command.

Tip: To reproduce an old result, after reloading the TOPS file, do not forget to simulate the trajectory.

Apart from above-mentioned, the final status produced by *CAMTOS* can also reach other conditions:

- If the iteration number reaches the defined maximum (200 here), the optimization stops with the final status is: Too many iterations. In this situation restart the optimization with cold start.
- The final status is Current point cannot be improved but the constraint violations are small: this is usually caused by a bad scaling of a bad initial guess, restarting the optimization with cold start could solve the problem.

5 Visualize the Results

ASTOS automatically stores all necessary trajectory data (independent variables, states, controls, auxiliary data). These data can be visualized using the built-in tools of *ASTOS* or exported to a file. In particular, *ASTOS* allows displaying simulation results in various plotting styles (2D, 3D, map plots, etc.) in the *Viewer* window or printing customizable sets of data to a text file using the *Result Summary* in the *Results* tree.

All related *ASTOS* tools are explained in detail in the ASTOS User Manual, so only basic functionality required for tutorial is be presented here.

Curve Plots

Select *Curve Plots* in the *Results* tree. The configuration panel now contains three sections: *Data Source*, *Simulation Data* and *Plot*.

- In the *Plot* panel, select *2D-Plot* as *Plot type*.
- Scroll down the tree below *Simulation Data* and double-click *flight_time*.
- Then, select *Position -> Relative -> altitude* and double-click.

> **Tip:** It is possible to drag and drop items from the *Simulation Data* tree to the *Plot* table.

- Click on the **Show** button.
- A new window (*Viewer*) is automatically opened and displays the trajectory profile.

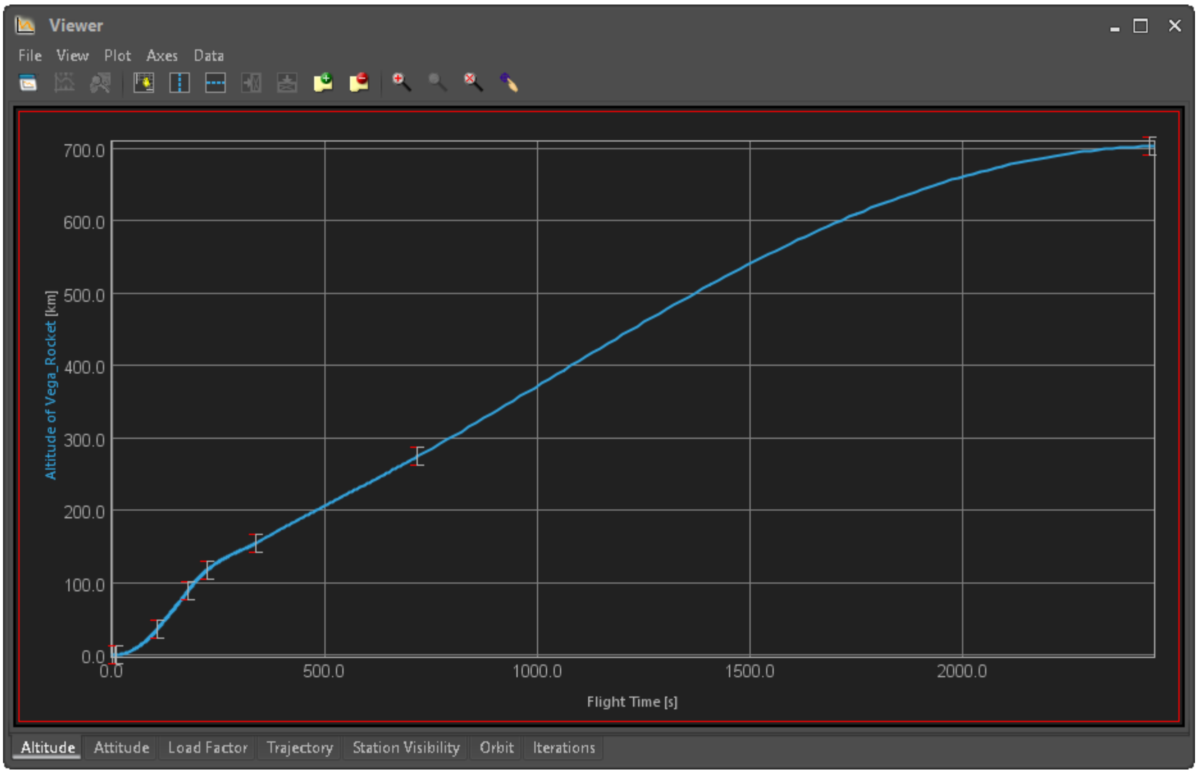


Fig. 5.1: Optimized altitude profile of VEGA

It is important to note that the final altitude of VEGA is slightly different from the target orbit altitude: the latter is defined as equatorial altitude, while the first is the exact altitude at that latitude. This difference is not present in case a spherical Earth model is used.

Repeat the same procedure for each function of interest (e.g. flight path speed vs. flight time, see also Section 2.7.12 for list of examples) and for creating map plots. In case a *Map-Plot* is selected as *Plot type*, the **Add Defaults** button automatically pads typical data.

Tip: The *Viewer* automatically saves created plots. Moreover, it is possible to save/load multiple sets of visualizations to/from dedicated files.

Result Summary

Navigate to the *Results* tree and select *Result Summary*. Click the **Summary** button in the configuration panel. Without further selections, this produces a printout of the most important characteristics of the mission in the *Execution Log* (see Fig. 5.2). The summary contains the initial/duration/final time, parameters value and constraint violations for each phase together with the overall cost function result of the optimization process (maximum payload in this scenario).

Optionally, the user can arbitrarily extend the Result Summary by choosing data from the *Item Selection*. For more details, please refer to the Result Summary chapter in the *ASTOS* User Manual.

```
******************************************************************************
*                                                                            *
*  Result Summary - Scalar Information              Oct 19, 2017 12:43:27 PM *
*                                                                            *
*                                                                            *
******************************************************************************

* COST FUNCTIONS *************************************************************
                                                          Nominal       Final
  Cost is Negative Total Payload Mass of Vega_Rocket                -2095.33446

* PERFORMANCE INFO ***********************************************************
                                              Desired    Achieved %-Deviation
  Circular Orbit Altitude of Vega_Rocket        700.0  699.999999      1.3E-07
  Altitude_Equatorial of Vega_Rocket            700.0  700.000000        1E-07
  Orbit Inclination of Vega_Rocket               90.0          90      5.7E-09
  Radial velocity of Vega_Rocket                  0.0   5.953E-07
  Normalized ResidualPropellant Mass of AVU       0.0   7.286E-08

* SCALAR QUANTITIES OF PHASE 1 ***********************************************
  Description:
  Phase Times (initial + duration = final): 0.0 + 1.0 = 1.0 -

  Scalars                                                    Value         Unit
    Altitude of Vega_Rocket                              3.863E-06           km
    Latitude of Vega_Rocket                              5.2356007            °
    Longitude of Vega_Rocket                              -52.7744            °
    Radial velocity of Vega_Rocket                       8.055E-15          m/s
    North velocity of Vega_Rocket                       -6.217E-15          m/s
    Relative east velocity of Vega_Rocket               -1.137E-13          m/s
    Delta of InitialPropellant Mass of P80_Stage_1 of V    88380.0           kg
    Altitude of Vega_Rocket                              0.0565617           km
    Normalized ResidualPropellant Mass of P80_Stage_1 o 84373.2523           kg
    Normalized Independent Variable Constraint of        4.8424191            s
    Ignition Time of P80_Engine_1 of Vega_Rocket of Veg -2.599E-40            s
    structural mass of Vega_Rocket                      13257.3344           kg
    Payload mass of Vega_Rocket                         2095.33446           kg
    structural mass of AVUM of Vega_Rocket                   578.0           kg
    structural mass of P80 of Vega_Rocket                   7416.0           kg
    structural mass of VegaFairing of Vega_Rocket            490.0           kg
    structural mass of VegaPayload of Vega_Rocket       2095.33446           kg
    structural mass of Z23 of Vega_Rocket                   1845.0           kg
```

Fig. 5.2: Example of Result Summary for VEGA (phases 2-10 are not displayed)

Note: The optimized value for the maximum payload for this scenario is 2076 kg. The negative sign of the cost function shown in the *Execution Log* stems from a property of the optimization process (i.e. a maximization of a parameter corresponds to a minimization of the same parameter with inverted sign) and can be ignored.

Note: This value for the optimized payload is larger than the payload of the "real" VEGA (1500 kg). Reasons for this deviation are the fictitious aerodynamics present in the scenario (no real data are available) and the unrealistic constant thrust value of the VEGA booster (real thrust profile not available).

Scenario Summary

It is possible to update the information present in the existing scenario summary with the optimized results. Please repeat the instructions presented in the Summary paragraph of to create it.

6 Summary and Advanced Modifications

The aim of this tutorial is to provide the user with a basic procedure on how to create a launcher scenario in *ASTOS*. Several modules could be added to prepare a more realistic and complex scenario using this simple tutorial as a starting platform.

The following more advanced modifications are described in this chapter:

1. Modify the target orbit.
2. Add a station visibility constraint.
3. Add a staging constraint (final constraint as "splash down" applied to the burn phases using a combined average drag constraint).
4. Add a profile for the mass flow (or the vacuum thrust) of the solid propulsion engines
5. Visualize the optimization steps with Review Iterations.

6.1 Target Orbit

The modification of the target orbit could be an easy or difficult task depending on how different the initial and final target orbit are chosen.

- In the *Optimization* tree, expand *Constraints* and select *Orbit_Altitude* constraints.
- Enter a new *Reference* value of *1500.0 Kilo-Meter*.
- Modify the *Circular_Altitude* constraint accordingly.
- **Save** the scenario.
- Optimize the scenario by clicking on *Start*→ *Actions*→ *Optimize (Cold Start)* or *Optimization*→ *Actions*→ *Optimize (Cold Start)*.
- After around 50 iterations, a new optimal solution is found with a payload around 1600 kg

Usually, this kind of modification can be realized without another initialization: simply start up the optimization from the previous result. In some special cases the previous solution is so "stuck" in the optimal area, that it is impossible to converge to the required new solution. In this case, an initialization is mandatory.

6.2 Station Visibility

For creating constraints for station visibility, the definition of desired ground stations is required for which the station visibility constraint shall be evaluated. In case of a polar orbit of VEGA these ground stations are: Kourou, Bermuda, Antigua, Pare Pare, St. Pierre and Perth.

- Select ribbon task *Add→ Vehicles & Other Entities→ Ground Station*.
- Enter *Kourou* as *Identifier* and press the **Create** button.
- In the *Modelling* panel of the GUI main window, enter the *Dynamics Configuration* node, click on the *Vehicles & POIs Dynamics* section and select *Kourou*. In the *Initial State* tab enter
  - *0.0 Kilo-Meter* for Altitude,
  - *-52.6 Degree* for Longitude,
  - *5.1 Degree* for Latitude.

Repeat these steps for the remaining ground stations:

Bermuda

- *Identifier*: *Bermuda*
- Altitude: *0.0 Kilo-Meter*
- Longitude: *-64.658 Degree*
- Latitude: *32.351 Degree*

Antigua

- *Identifier*: *Antigua*
- Altitude: *0.0 Kilo-Meter*
- Longitude: *-61.774 Degree*
- Latitude: *17.137 Degree*

Pare Pare

- *Identifier*: *Pare_Pare*
- Altitude: *0.0 Kilo-Meter*
- Longitude: *119.643 Degree*
- Latitude: *-4.02 Degree*

St. Pierre

- *Identifier*: *St_Pierre*
- Altitude: *0.0 Kilo-Meter*

- Longitude: *-56.17 Degree*
- Latitude: *46.77 Degree*

Perth

- *Identifier*: *Perth*
- Altitude: *0.0 Kilo-Meter*
- Longitude: *115.885 Degree*
- Latitude: *-31.8025 Degree*

To get an output variable for the station visibility a path constraint has to be created.

- Select *Add→ Optimization→ Constraint* from the ribbon.
- Enter *Station_Visibility* as *Identifier*.
- Select *All* as *Type*, *Station Visibility* as *Subtype* and press the **Create** button.
- Insert a description (optional).
- Select the *Vega_Rocket* as *Vehicle ID*.
- Disable the *Is enforced* button.
- Check the *(All)* Phase ID in the *Specified* column and set it up as *Path*.
- Select *Reference* from the *Limit_Type* drop-down list.
- Enter *0.0 Degree*
- Check the recently created ground stations in the *Access objects* list.
- **Save** the scenario.

With this constraint *ASTOS* creates an output variable of the station visibility. Since this constraint is not enforced, it is not evaluated at any Constraint Evaluation Nodes. Thus it does not influence the optimization process and is only used to compute the output function.

6.3 Staging Constraint

To add and visualize a splash down constraint the following steps are required:

- Select ribbon task *Add→ Optimization→Constraint*.
- Enter *Splash_Down_Drag* as *Identifier* and press the **Create** button.
- Insert a description (optional).
- Select the *Vega_Rocket* as *Vehicle ID*.
- Disable the *Is enforced* button.
- Check the 4 Phase ID's of the burn phases *P80_burn, Z23_burn, CoastwithFairing* and *Z9_burn* in the *Specified* column and set it up as *Final Boundary*.

- Select *Combined Average Aero* from the *Drag computation* drop-down list and select *Vega_Aero* for *Aerodynamics ID*.

To visualize the impact points of the jettisoned components, perform again a simulation with Multiple Shooting. Afterwards in the *Viewer* window:

- Select *2D-Plot*.
- Select *Data*→ *Auxiliary Items...* from the main menu.
- Check the box of *Splash_Down_Drag_Final* and choose your desired symbol with the according colors (e.g. Star with orange fill color)
- Apply the changes.

Now the impact points of the jettisoned components are visualized with the selected symbol in the 2D-Plot.

6.4 Mass Flow Profile

To use a mass flow profile instead of a constant value the already created actuators for the solid propulsion engines have to be modified:

- In the *Modelling* panel on the left in the GUI main window, navigate to *Vehicle Parts & Properties -> Actuators -> P80_engine*.
- Change the *Vacuum thrust* from *custom* to *default*.
- Change the *Mass flow* from *default* to *custom*.
- In the *Defined by* drop-down list select *Profile*.
- In the *Data source* drop-down list select *Local*.
- Enter the data in the table below for *Burn_Time* and *Mass flow*.

Table 6.1: Mass flow profile for P80_engine

Burn_Time [s]	Mass flow [kg/s]
0.0	75.0
7.0	87.5
8.5	88.25
10.0	88.25
15.0	85.8
32.0	61.0
37.0	60.0
56.0	65.0
72.0	65.6
75.0	65.0

Burn_Time [s]	Mass flow [kg/s]
85.0	57.0
91.5	54.0
95.0	31.0
97.0	20.0
100.0	10.0
103.0	6.0
110.0	2.0
120.0	0.0

- Enable *Scale to fit* and select *All Propellant* from the *Fit to* drop-down list.
- Repeat these steps for the actuators *Z23_Engine* and *Z9_Engine* with the corresponding data in the tables below.

Table 6.2: Mass flow profile for Z23_Engine

Burn_Time [s]	Mass flow [kg/s]
0.0	91.5
0.7	89.1
2.9	94
3.3	93.3
7.0	93.5
9.0	94.6
17.0	90.6
23.5	85.0
30.0	81.3
46.0	63.7
55.0	52.5
64.0	43.8
70.0	40.0
71.8	20.0
73	10
75.0	2.0
80.0	0.3
85.0	0.0

Table 6.3: Mass flow profile for Z9_Engine

Burn_Time [s]	Mass flow [kg/s]
0.0	46.5
9.5	75
11.3	76.3
13.6	76.3

Burn_Time [s]	Mass flow [kg/s]
20.0	73.3
23.0	73.0
36.0	77.0
41.0	76.0
55.0	68.0
94.8	47.0
105.1	40.0
110.0	35.0
111.2	30.0
113.0	10.0
115.0	2.0
120.0	0.0

- Since the mass flow of the P80 Engine peaks in the first 10 seconds (and thus is higher than the previous constant mass flow) the timings of the first three phases have to be slightly modified. Otherwise the Initial Guess will not lead to a converging Optimization. To modify the timings:
- In the *Modelling* panel, navigate to *Dynamics Configuration -> Phases* and modify the following:
    - In Phase *LiftOff* set *Final Time* to *3.3 Seconds*
    - In Phase *PitchOver* set *Final Time* to *7.0 Seconds*
    - In Phase *ConstantPitch* set *Final Time* to *11.0 Seconds*.
- Navigate to *Dynamics Configuration* in the *Modelling* panel and expand the *Vehicles and POIs Dynamics* node. Select the *Vega_Rocket* vehicle, select the *PitchOver* tab and change the *Pitch rate* to *-3.0 Degree/Seconds*.
- **Save** the scenario.
- In the *ASTOS* ribbon, press the *Start→ Actions→ Initialize* button. This is necessary for the scenario changes to be taken into account.
- If no error occurred, start the optimization with *Start→ Actions→ Optimize (Cold Start)*.

References

[1] European Space Agency, *ESA Launch Vehicles*, http://www.esa.int/esaMI/Launchers_Access_to_Space/SEMH3E67ESD_0.html

[2] European Space Agency, *VEGA Small Launcher*, http://esamultimedia.esa.int/multimedia/vega/index_vega.html

[3] European Space Agency, *The Small Launcher for Europe*, http://esamultimedia.esa.int/docs/VEGA.pdf

[4] Arianespace, *VEGA, Users' Manual*, Issue 3/Revision 0, March 2006, http://www.arianespace.com/site/documents/VEGAUsersManual.pdf

[5] http://www.astronautix.com/sites/kourou.htm

[Page intentionally left blank]