Software User's Manual

Description, usage and instruction

Clément VELLU
14/04/2023

Table of content

I- Software general description

a. General purpose

The software is a python script that computes the trajectory, speed, altitude and mass of a rocket over time, using parameters from Ariane 5 rocket[1]. The script is usable in 3 different modes (for more details, see the document delivered with the software) :

- Mode 0 : Simulation with constant pitch angle variation (with adjustable slope)
- Mode 1 : Simulation with gravity turn only
- Mode 2 : Simulation with constant pitch angle (with adjustable constant)

The software outputs plots of the quantities indicated above.

The software requires the use of *Numpy* 1.24.2 and *Matplotlib* 3.7.1 and was coded with *Python* 3.10

b. User input and editable parameters

In order to run, the user has to choose between one of the three modes by typing '0', '1' or '2' depending on the requested mode.

User can also edit parameters inside the main code. These parameters are concentrated in an unique location that starts under line 42 : # *User modifiable parameters* # and finishes at line 52 : # *End of User modifiable parameters* #.

These are the following :

- `end_time` : Time in *seconds* at which the simulation shall stop
- `dt` : Interval in *seconds* between two time steps
- `pitch_slope_time` : Time in seconds for the pitch angle of the rocket to linearly transition from 0 rad to $-\frac{\pi}{2}$ rad (for mode 0 only : see details further)
- `init_pitch_diff` : Initial pitch angle difference between the vertical axis and the rocket axis in *radians* (for mode 1 only : see details further)
- `constant_pitch` : Pitch angle in *radians* between the horizontal axis and the rocket axis (for mode 2 only : see details further)

One shall not edit other parameters as some unexpected behaviour can be achieved and the software might not output correct data.

c. Other parameters

The software uses other parameters :

- `g0` : acceleration of gravity (in $m.s^{-2}$)
- `earth_radius` : Radius of the Earth (in m)
- Rocket parameters : masses, thrust, specific impulse inputted from Ariane 5 technical data[1] and other computed parameters.

[1] ArianeSpace, *Ariane 5 user's manual*, https://www.arianespace.com/wp-content/uploads/2016/10/Ariane5-users-manual-Jun2020.pdf, accessed: 2023-04-13 (2020)

II- Software technical description

a. Acceleration computation

The physics model relies on Newton's Second Law only with gravity and thrust and is used in all three modes of the software.

The following assumptions are made :

- Rocket is a point with non-null mass
- Earth is a perfect sphere with no atmosphere with $R_{Earth} = 6378{,}00\ km$ and $g = 9{,}80665\ m.s^{-2}$
- Each rocket stage provides a constant thrust and mass flow rate $\dot{m} = -\frac{dm}{dt} = constant$
- Rocket is Ariane 5 with 3 stages
- Rocket burns from $t = 0$ until burn-out and ditches stage when current stage propellant is exhausted
- Time is discretized with time step dt that can be changed by user (see I.b.)
- Thrust is in the direction of the rocket $\overrightarrow{x_r}$
- Gravity is in the direction $-\overrightarrow{e_r}$

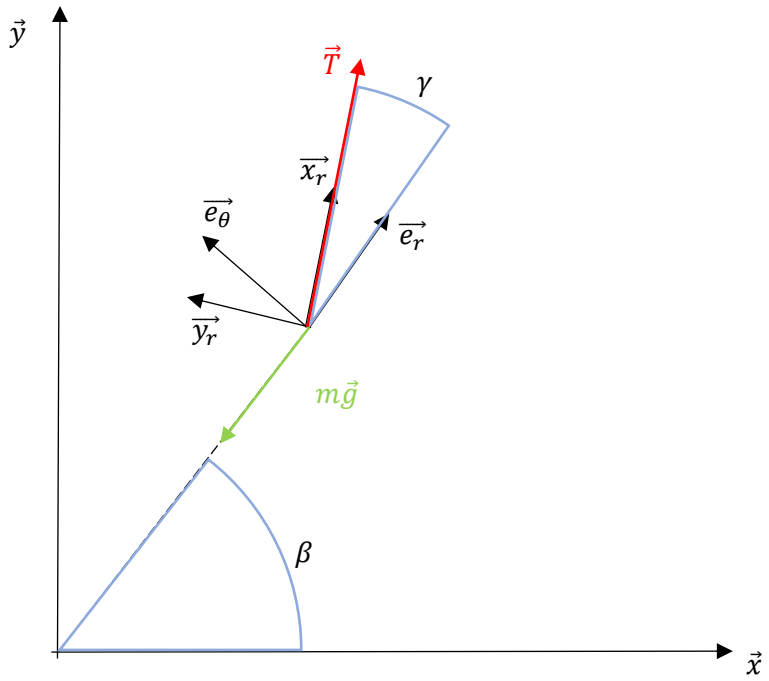

Figure 1 – Angle definitions and forces directions

We can then compute the acceleration at the k-th step in the fixed reference frame (x, y) with Newton's 2nd law

$$\overrightarrow{a_k} = \left[\frac{T}{m_k} \cdot \cos(\beta + \gamma) - g\cos(\beta)\right] \vec{x} + \left[\frac{T}{m_k} \cdot \sin(\beta + \gamma) - g \sin\beta\right] \vec{y}$$

b. Speed and position computation

Speed and position at the k-th step are calculated using Euler's integration scheme :

$$\overrightarrow{v_{k+1}} = \overrightarrow{v_k} + dt \cdot \overrightarrow{a_k}$$

$$\overrightarrow{p_{k+1}} = \overrightarrow{p_k} + dt \cdot \overrightarrow{v_k}$$

c. Software Modes

i. Mode 0

Mode 0 enables the user to select a pitch angle control law (where γ is the relevant angle) that linearly change from 0 to $-\frac{\pi}{2}$ in the trigonometric direction.

The time it takes to reach $-\frac{\pi}{2}$ can be adjusted via the `pitch_slope_time` parameters as represented in the figure below.

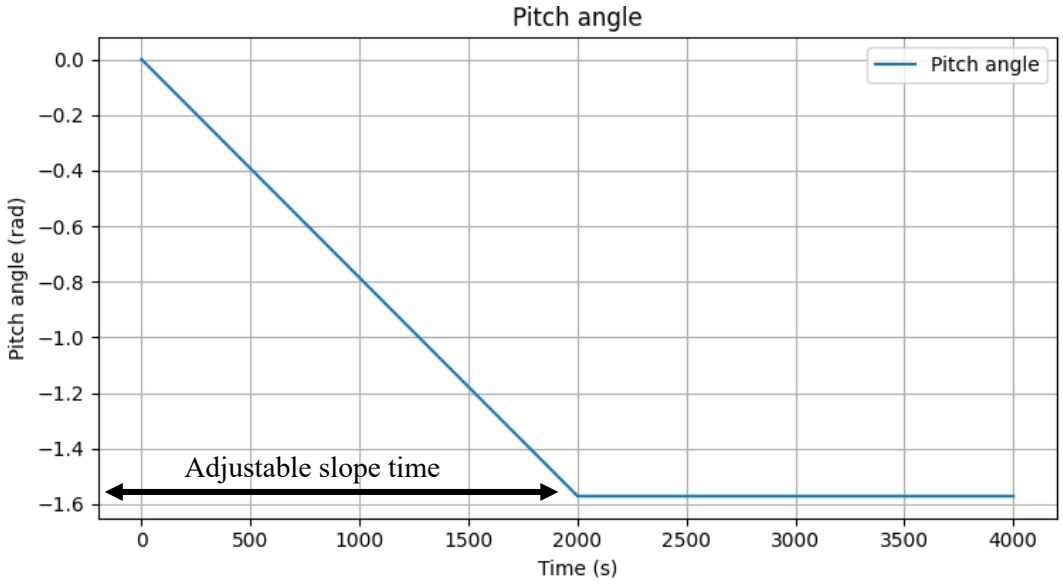

Figure 2 – Pitch angle slope control

ii. Mode 1

Mode 1 enables the user to visualize the trajectory of the rocket with gravity turn only and no pitch control. The user can adjust `init_pitch_diff` the initial difference between the true vertical rocket and the actual rocket direction relative to the vertical.

Most of the value used lead to a collision with Earth's surface in the following seconds. This is indicated by the prompt "Crash" and the display of the computed trajectory up to the crash time.

The pitch angle $\theta = \gamma + \beta$ of the rocket from the fixed reference frame is computed using the following formula :

$$\theta = \tan^{-1}\left(\frac{v_y}{v_x}\right)$$

The *Numpy* function `arctan2()` is designed to give the angle in the right quadrant to avoid discontinuities.

iii. Mode 2

Mode 2 enables the user to choose a constant θ pitch angle and force the rocket to remain at this angle.

The angle can be chosen by setting the variable `constant_pitch`.

III- Software usage and instructions

a. How to run

The software is ready to run by launching the main.py script. The requirements can be found in I.a.

The software prompts the user to choose a mode between 0 and 2. The user shall not input any other value as the software will raise an Exception

b. Expected outputs

The following figures represent the expected output for mode 0 to mode 2 in the out-of-box configuration for mass plot in a separated window and the 4 subplot in another window.

Mode 0 :

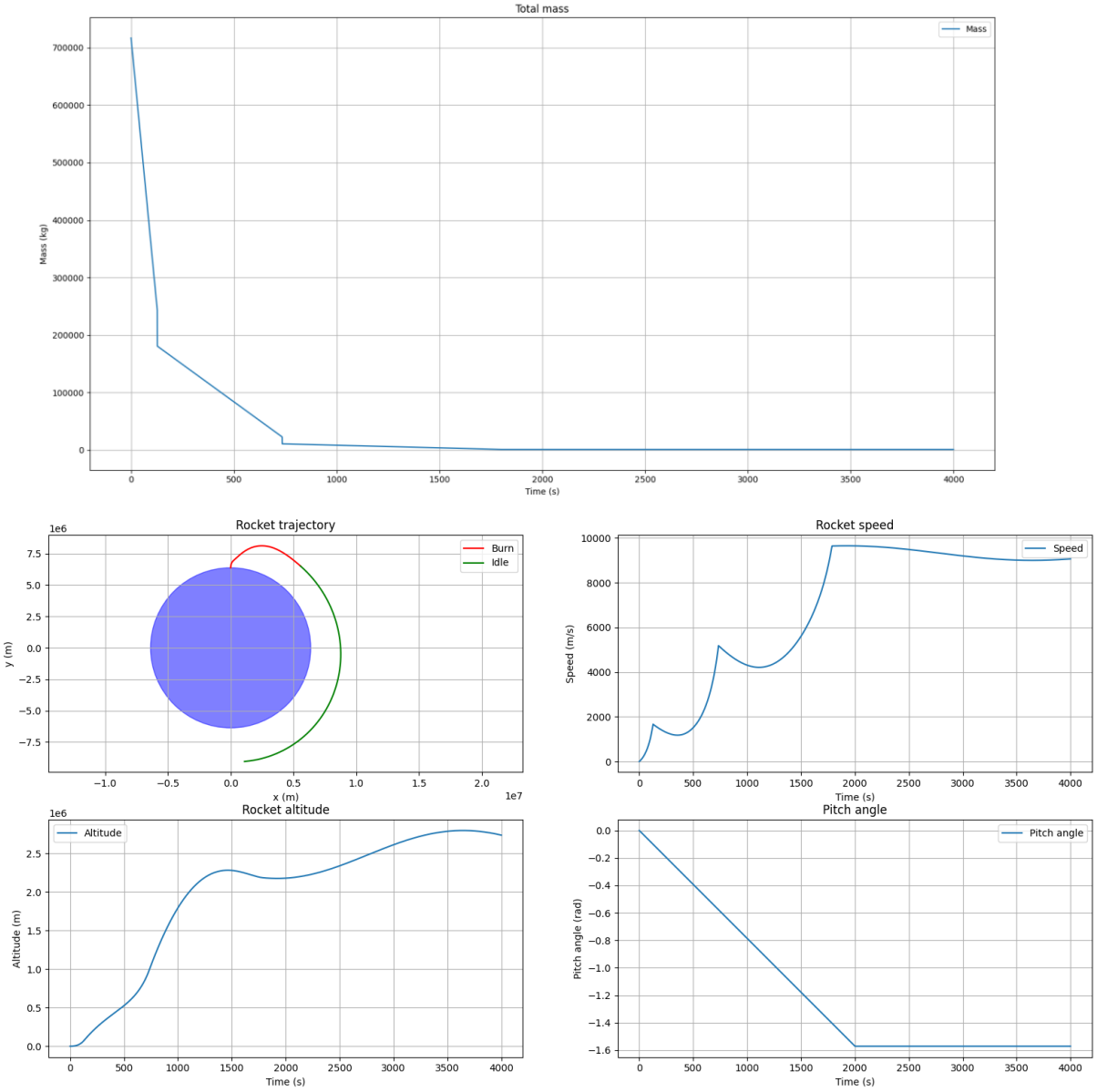

Figure 3 – Expected Mode 0 outputs

Mode 1 : Prompts « Crash » in the console

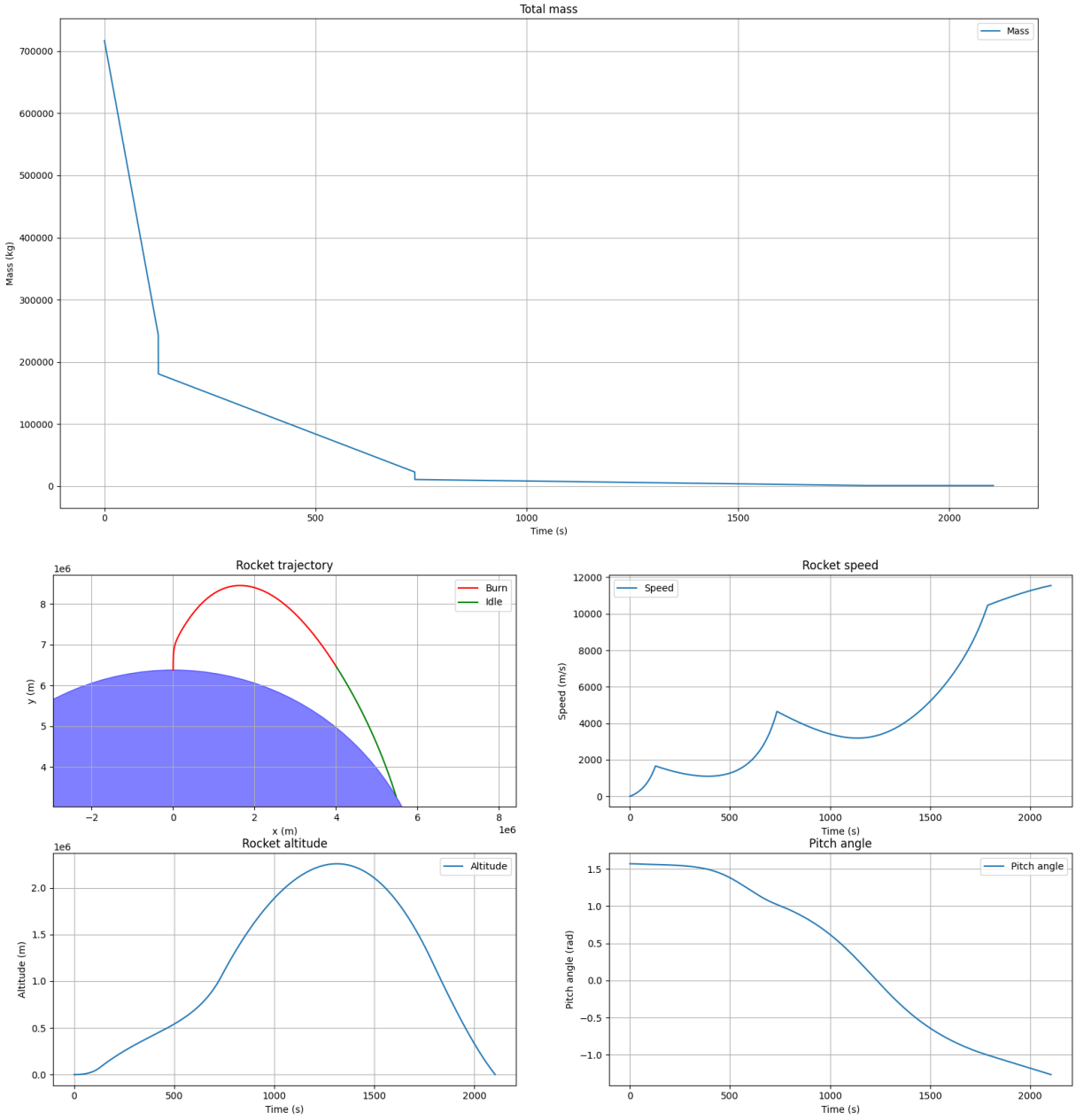

Figure 4 – Expected Mode 1 outputs

Mode 2 : Prompt « Crash » in the console

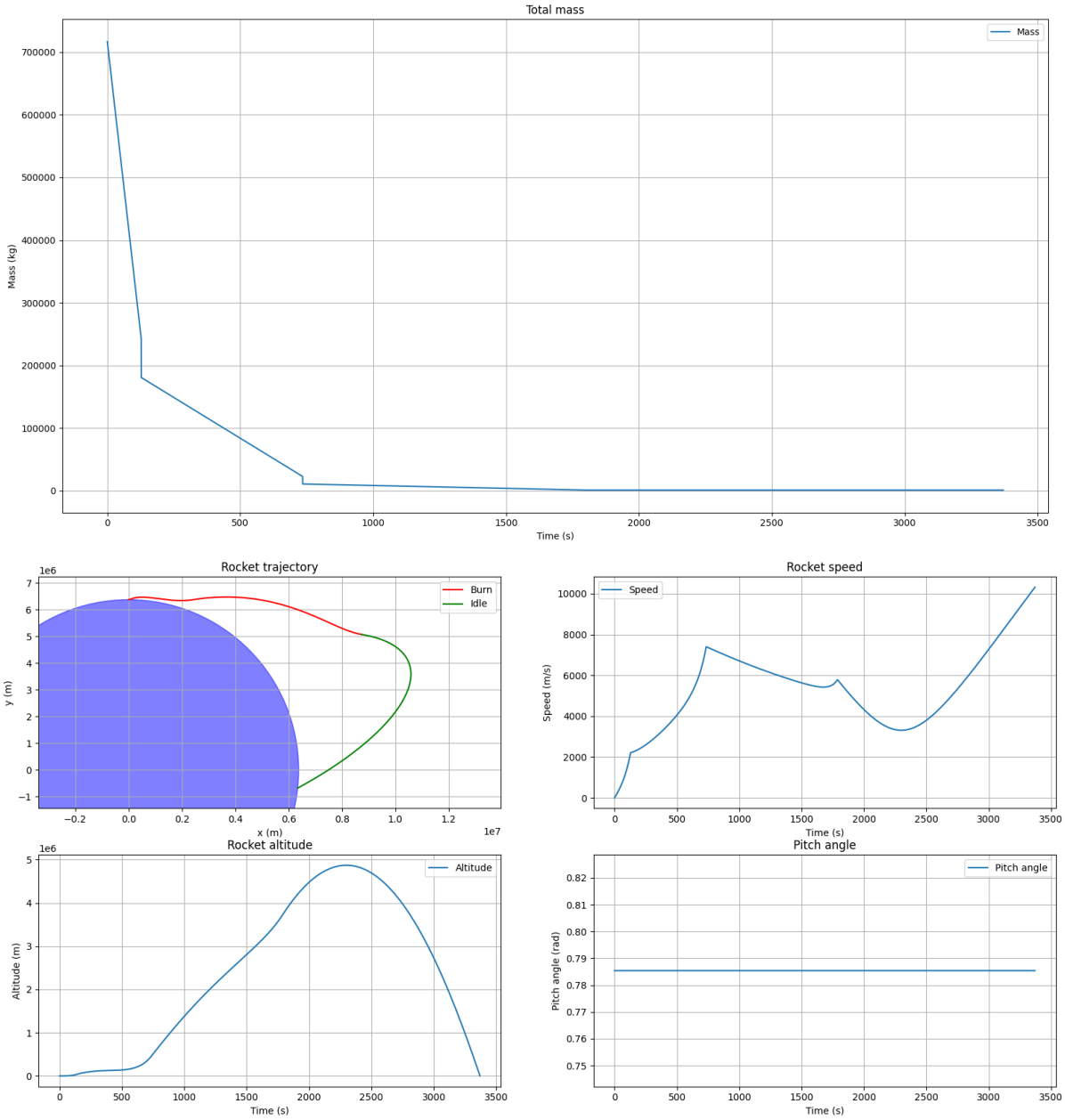

Figure 5 – Expected Mode 2 outputs

Any other input : Raise an exception