Satellite fly-by

Aron Dahlberg

August 12, 2021

1 Introduction

1.1 General idea of the code

This code is used along-side of Mario's asteroid-image-generator [1].
It calculates a linear trajectory path for a satellite with user given velocity and fly-by time. After this, generates coordinate-points from the trajectory for image rendering. The data is outputted in both Cartesian and Spherical coordinates. A file containing the phase angle φ shown in Figure 1, distance from the satellite to object of each time step is also generated.
Note that the data will be outputted into the folder in which the code is located.

1.2 Parameters for user to define

- Closest approach distance d in kilometers.
- Velocity v of the satellite in km/s
- t_detailed = Fly-by time of the detailed interval, where images are taken more frequently.
- t_out = Fly-by time of interval further away from closest approach. Frequency of images most likely is smaller during this.
- Seconds in between two images during an interval. The code contains a parameter for this in both intervals "t_out" and "t_detailed", which the user can modify.
- Sun_angle = Sun angle in terms of the closest approach vector. This is shown in Figure 4 as ±45°.
- "gamma" = Inclination of the trajectory plane (above/below sun)
- delta_restrictions = Two numbers, which signify the restrictions on angle delta (see Figure 1). First number restricts delta on the left side of closest approach and second number on the right side.
- Other parameters the user can switch but does not have to:
  Sun_irr = Sun irradiance (kW/m^2)
  rot = object rotation in degrees.

1.3 Diagrams and further descriptions of the parameters

Figure 1 displays the angle δ, which can be restricted. For example, choosing the first number of delta_restrictions to be 30 results in coordinates on the left side of closest approach being outputted only if $\delta > 30$.

Figure 2 indicates, that the closest approach location is at $t = 0$. Thus if t_out $= \pm 300$s and t_detailed $= \pm 120$s, the total fly-by time will be from $t = [-420, 420]$ seconds.

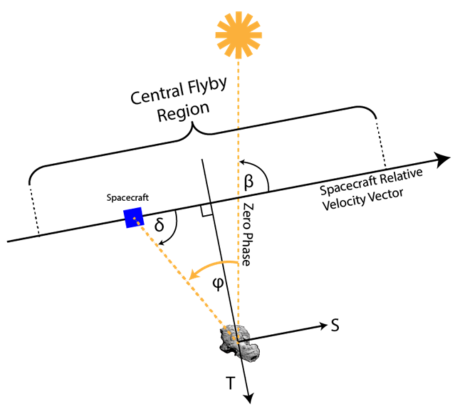

Figure 1: General situation.

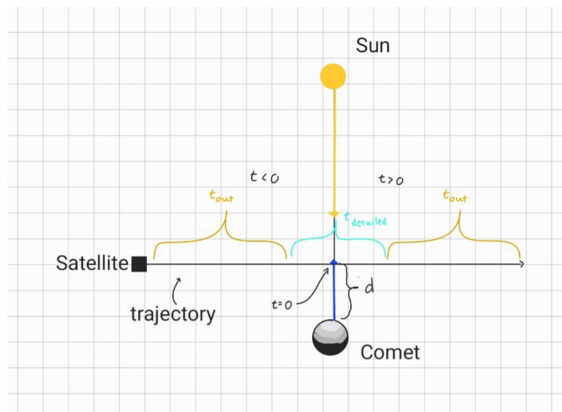

Figure 2: Diagram showing the fly-by time parameters.

Figure 4 shows what the parameter "sun_angle" does to the system, it rotates the sun vector by sun_angle degrees. Negative values rotate it to the left and positive values to the right. A slight insignificant side-note is, however that in the code, the sun actually stays in place and everything else is rotated. This is because in Mario's code the sun acts as a coordinate vector. Both situations are totally analogous so the user does not have to worry about this at all.

Figure 4 exhibits what changing the parameter "gamma" does to the code. With gamma $= 0$, the trajectory stays in XY-plane. Negative values lower it below sun and positive values lift it above the sun.

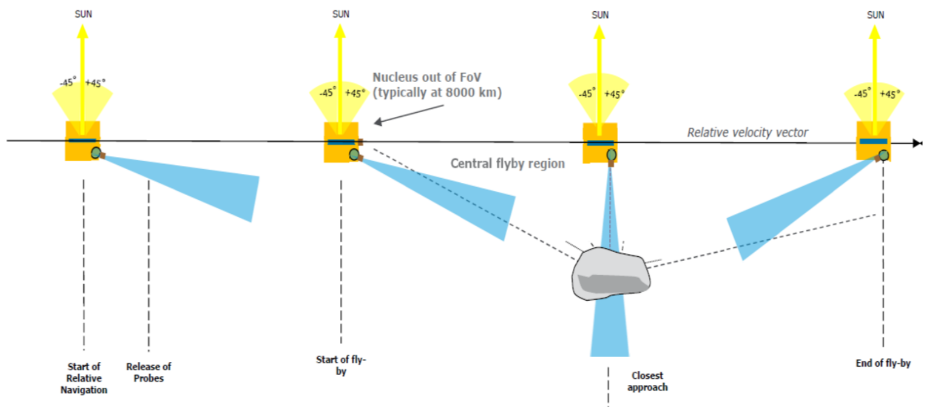

Figure 3: Rotation of the sun in relation to the closest approach vector.

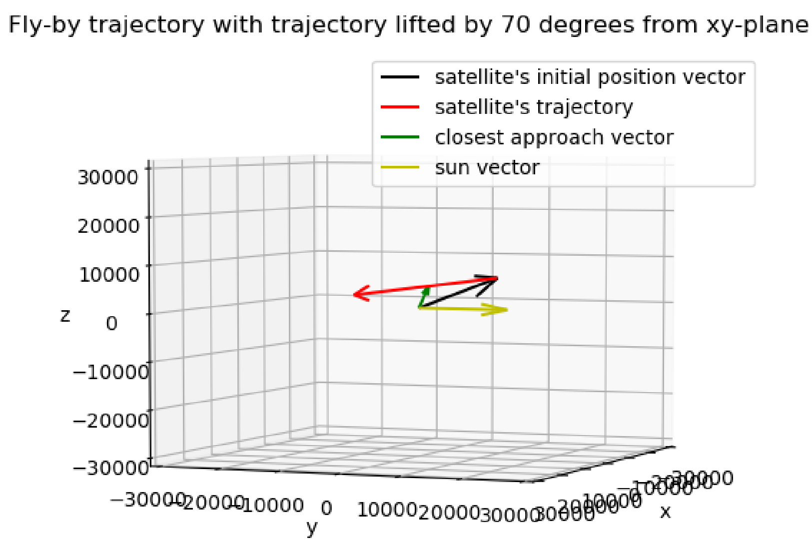

Figure 4: Trajectory lifted above the sun.

2 Description of the math behind the code

Define the closest approach vector as: $\vec{p} = p_{\mathrm{x}}\vec{i} + p_{\mathrm{y}}\vec{j} + p_{\mathrm{z}}\vec{k}$

Asteroid is located at origin (0,0,0). Sun vector is a scaled version of the closest approach vector.

Let's call the vector perpendicular to $\vec{p}$, as $\vec{w} = \mathrm{a}\vec{i} + \mathrm{b}\vec{j} + \mathrm{c}\vec{k}$. The vector $\vec{w}$ is defined with the following equations:

$$\begin{cases} \vec{w} \cdot \vec{p} = a\vec{p}_x + b\vec{p}_y + c\vec{p}_z = 0 & (1) \\ \vec{w} \cdot \vec{w} = a^2 + b^2 + c^2 = 1^2 & (2) \\ c = 0, & (3) \end{cases}$$

thus signifying that it is perpendicular to $\vec{p}$ and of unit-length. Note that the third equation could have been any of the components, but $\mathrm{c}\vec{k}$ was chosen.

Rotation of the sun is then accomplished by everything in the system, except the sun, by a rotation matrix:

$$\begin{bmatrix} \cos(\text{sun\_angle}) & \sin(\text{sun\_angle}) \\ -\sin(\text{sun\_angle}) & \cos(\text{sun\_angle}) \end{bmatrix}$$

This matrix rotates everything by sun_angle degrees to the clockwise direction.

The inclination of trajectory is achieved by rotating the closest approach parameter around the trajectory vector by "gamma" degrees. This is a special case of Rodrigues' rotation formula [2], where the vector being rotated is perpendicular to the axis of rotation. The rotation is achieved with the following steps:

1) Create a help-vector $\vec{h}$ perpendicular to both the axis of rotation(vector $\vec{w}$) and closest approach vector $\vec{p}$. 2) Rotate the vector with the following equation:

$$\vec{p}_{rot} = \vec{p}\cos(-\text{gamma}) + \vec{h}\sin(-\text{gamma}) \quad (4)$$

The negative signs of gamma are to obtain lift with positive angles and lowering with negative angles.

References accessed: 27.07.2021

References

[1] Mario F. Palos. Asteroid image generator. https://bitbucket.org/mariofpalos/asteroid-image-generator/wiki/Home.

[2] Carlo Tomasi. Vector representation of rotations. *Computer Science*, 527, 2013.