AE 502 Homework 1

Thane Gesite

February 2023

Note: I was unable to get the code running for my contour plots in time, so I have no plots. See the unfinished code attached in my GitHub: https://github.com/agesite2/ae502.git

1 'Oumuamua

Below are the two pork chop plots for rendezvous ($\Delta V < 50$ km/s) fly-by missions ($\Delta V < 20$ km/s) to 'Oumuamua.

1.1 Initial State Vectors

The initial state vectors for 'Oumuamua were given as the following:

$$r_{1I} = [3.515868886595499 \times 10^{-2}, -3.162046390773074, 4.493983111703389]$$
$$v_{1I} = [-2.317577766980901 \times 10^{-3}, 9.843360903693031 \times 10^{-3}, -1.541856855538041 \times 10^{-2}]$$

Using the Python function *orbital_elements*, we can convert these vectors to the 6 Keplerian orbital elements. To determine whether 'Oumuamua is interstellar, we only need to look at the eccentricity, e, and the semi-major axis, a. These two elements help us determine the type of orbit. The eccentricity is 19619963.95061179, which is much larger than 1, and the semi-major axis is $-3.90299659840782 \times 10^{-8}$ AU, which is obviously less than 0. By definition, this means that 'Oumuamua is in a hyperbolic orbit, which proves that it's interstellar.

2 Borisov

Below are the two pork chop plots for rendezvous ($\Delta V < 60$ km/s) fly-by missions ($\Delta V < 20$ km/s) to Borisov.

2.1 Initial State Vectors

The initial state vectors for Borisov were given as the following:

$$r_{2I} = [7.249472033259724, 14.61063037906177, 14.24274452216359]$$
$$v_{2I} = [-8.241709369476881 \times 10^{-3}, -1.156219024581502 \times 10^{-8}, -1.317135977481448 \times 10^{-2}]$$

Using the same method denoted for 'Oumuamua, we can see that Borisov's orbit is also hyperbolic, and that it is of interstellar origin.

3 Credits

I used the following libraries/wrappers in my code: NumPy, Math, SpiceyPy, Matplotlib.
Credit to David Yaylali for his porkchop plot code written in MATLAB that I heavily referenced, as well as to Alfonso Gonzalez's YouTube videos on porkchop plots and software implementation.