Matthew Kenzinger

Dennis Mboya

PHY-234L-01

Mechanics Lab Final Project: *"This little maneuver's gonna cost us 51 years"*

Introduction:

This project is inspired by *Interstellar* (2014), a sci-fi film known for its attention to detail regarding realistic physics. The scene in particular we would like to model consists of a spacecraft ("The Endurance") using a sling shot maneuver to swing around a black hole ("Gargantua") and get to another planet. In our case we model a rocket ("The Endurance", **1kg**) initially in a circular orbit around a massive planet ("Gargantua", **2000 kg**). The rocket begins with an **initial speed of 1 m/s (tangential to radius)** and is equipped with forward-facing engines that expel mass at a tunable exhaust velocity **(vel_ex)**. We examine how increasing our "thrust" affects the orbital trajectory and eccentricity over time. Specifically, with a fixed burn **time of 30 seconds** and an initial rocket mass of **1 kg (90% of which is fuel)**, we ask: **What exhaust velocity is required for the Endurance to escape orbit?**

Getting our Orbit:

Our first step in modeling the system was to get the rocket into a stable circular orbit around the large central mass. To do this, we followed a procedure similar to the one used in our previous orbits lab. The code we used was originally adapted from a three-body system, and instead of fully removing the third mass, which caused our derivative function to break, we found it more stable to simply set that mass equal to zero. The governing equations were primarily taken from Chapter 8 of the Classical Mechanics textbook, especially Sections 8.6 (bounded orbits) and 8.7 (unbounded orbits). One important derivation worth highlighting is our calculation of orbital eccentricity, which was based on those textbook equations. Here we define the mass of our spaceship (Endurance) as **m1** and the mass of our planet (Gargantua) as **m0** for ease of formula understanding

Here is our derivation for the eccentricity of our orbits:

1. Definitions and Constants

Reduced mass:
$\boldsymbol{\mu = m_0 m_1 / (m_0 + m_1)}$
where m_0 is the mass of the central body and m_1 the mass of the orbiting object.

Specific angular momentum:
$\boldsymbol{\ell = \mu\, r^2\, \dot{\phi}}$

where $\dot{\phi}$ **= v/r,** and was calculated using **v = sqrt(Vx + Vy)** and **r = sqrt(x1 + y1),** describing the aircraft's angular momentum as evidenced by the use of **x1** and **y1** rather than **x0** and **y0**

Orbit constant **C**:
$\boldsymbol{C = \ell^2 / (G\, m_0\, m_1\, \mu)}$

2. Pericenter and Apocenter Radii

For a Keplerian orbit with eccentricity ε:
$r_{min} = C / (1 + \varepsilon)$
$r_{max} = C / (1 - \varepsilon)$

3. Bounding Eccentricity from Noisy Data

Suppose measured radii differ slightly from the ideal, yielding $\tilde{r}_{min}$ and $\tilde{r}_{max}$.
Lower-bound estimate from $\tilde{r}_{min}$:

$$\varepsilon_{min} = C / \tilde{r}_{min} - 1$$

Upper-bound estimate from $\tilde{r}_{max}$:

$$\varepsilon_{max} = 1 - C / \tilde{r}_{max}$$

Where $\tilde{r}_{max} = \max(r)$ and $\tilde{r}_{min} = \min(r)$

4. Defining a Robust ('Real') Eccentricity

To mitigate noise or unmodeled forces:

$$\varepsilon_{real} = (\varepsilon_{min} + \varepsilon_{max}) / 2$$

5. Practical Steps

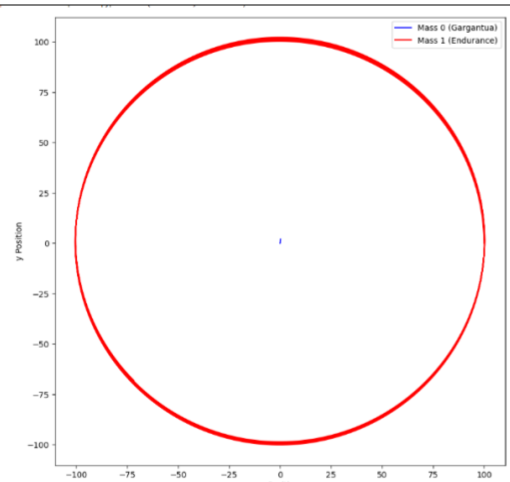
1. Compute ℓ from trajectory data $(r, \dot{\phi})$.
2. Form $C = \ell^2 / (G m_0 m_1 \mu)$.
3. Measure approximate $\tilde{r}_{min}$ and $\tilde{r}_{max}$.
4. Calculate:
   $\varepsilon_{min} = C / \tilde{r}_{min} - 1$
   $\varepsilon_{max} = 1 - C / \tilde{r}_{max}$
5. Set:
   $\varepsilon_{real} = (\varepsilon_{min} + \varepsilon_{max}) / 2$

Note: This procedure assumes that deviations of $\tilde{r}_{min}$ and $\tilde{r}_{max}$ from their ideal values are small and symmetrical.

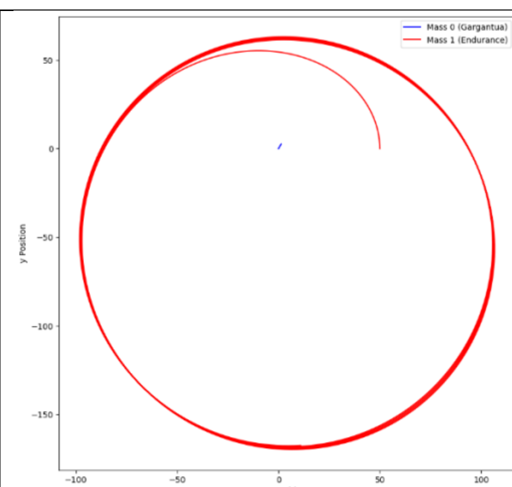
We calculated $\tilde{r}_{min}$ and $\tilde{r}_{max}$ after the first 30 seconds to not account for the orbit mid-burn. This way, we can calculate the eccentricity of the steady-state orbit and disregard rocket-induced orbits

Ultimately, we are looking for a robust value of e, which we call ε_{real} and define as $\varepsilon_{max} + \varepsilon\ min/2$ to account for any weirdness in our orbits we failed to account for. We can then assess these eccentricity values by comparing them with the textbook.

Testing our model and eccentricity calculation:



For this initial model we simply have the Endurance in a circular orbit around

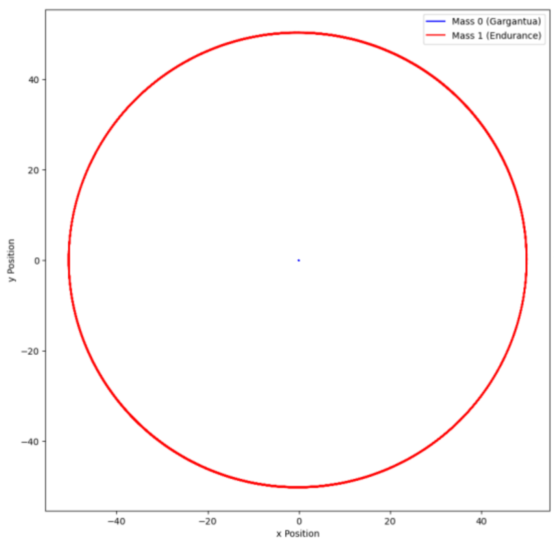


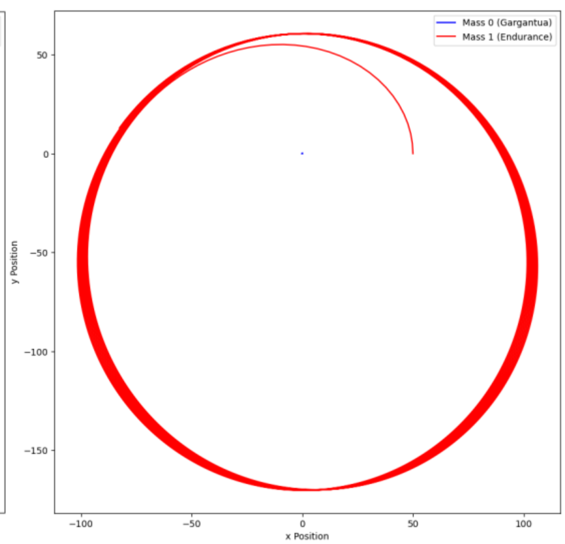
For this model we changed the exhaust velocity to 2m/s and our burn time to 30s and

Gargantua with no exhaust velocity or exhaust time. When we run our code, we get a value of 0.014 for the eccentricity, and because eccentricity for a circular orbit should be zero (Chapter 8.7), we know our calculation is correct within a very small degree of uncertainty.	as we expected we see a change in the orbit to an ellipse shape. Our eccentricity calculation gave us a value of approximately 0.52 which according to chapter 8.7 of the textbook, is in the viable range we expect.

Please note that our exhaust velocity values were forced to be negative by the relation between **dmass_dt** and **burn_rate**. Given that these two have a negatively correlated relationship (namely **dmass_dt = -1*burn_rate**) then we must define exhaust velocity as **-vel_ex** to account for this relationship

This calculation was also done in the home frame. You notice that in the above plots the large mass (Gargantua) is moving, and this causes the orbit to shift at well. We fix this problem by plotting in the center of mass frame as we did in the “three-body lab”. These fixed COM plots are shown below:





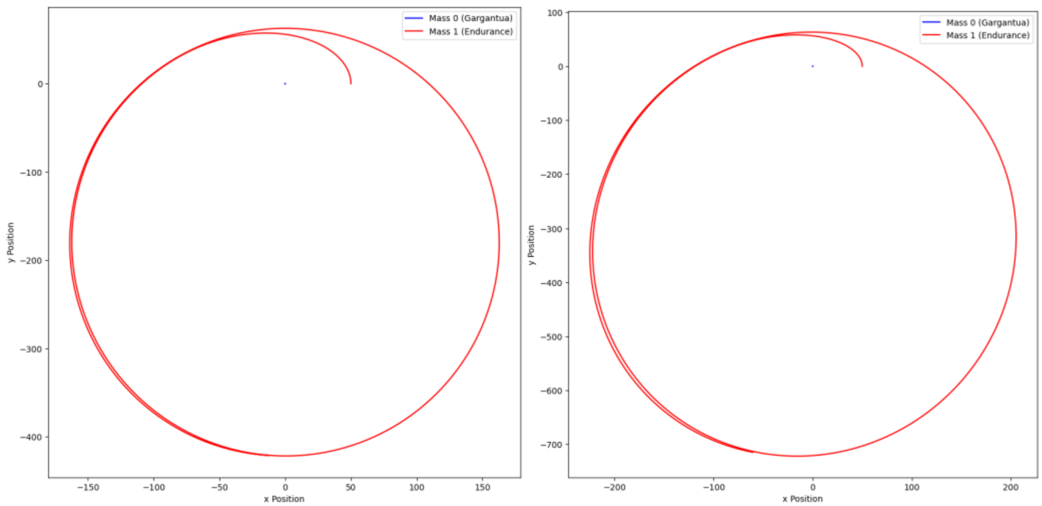
The main takeaway here is that our central mass is no longer in motion within our frame of reference. Once we are certain that our code works and is reasonable, we begin our eccentricity analysis process.

Analyzing multiple cases of eccentricity variation with a change in exhaust velocity:

Exhaust Velocity (vel_ex)	**eccentricity (e_real)**	**Energy (E)**
0	0.05831274306	-6305.720303
0.25	0.03509031457	-6089.946505
0.5	0.07978795733	-5840.407056
0.75	0.1311998983	-5551.228572
1	0.1892112609	-5231.967561
1.25	0.2563462812	-4846.407979
1.5	0.3327137152	-4402.796602
1.75	0.4209672455	-3874.673651
2	0.5296226859	-3193.697173
2.25	0.6691415815	-2284.06987
2.5	0.8597708236	-988.6643753
2.75	1.136228856	989.0351646
3	0.9184211549	-473.4949378
3.25	3.949501829	21645.05949

3.5	12.99859564	135984.4626
3.75	28.29017916	432518.4666
4	152.0077976	5325459.317

Special Case: The path of **vel_ex = 2.75** is a closed orbit, but the eccentricity is larger than 1. Its plot in the COM frame **(left),** as well as that of **vel_ex = 3 (right),** is shown below:



We are unsure why this happens, but its presence is worth noting despite our perplexity concerning the eccentricity of **vel_ex = 2.75** orbit being classed as parabolic as per the textbook.

System Validation:

This part was difficult as the exact orbit our system is supposed to be taking is largely unknown. We checked our trajectories using the parameters for eccentricity from the textbook, making sure our values for eccentricity matched with the orbit. We also doubled checked our calculations thoroughly to ensure our mathematics were sound.

Conclusion:

In our code, we were able to accurately model a spacecraft orbiting a massive central body and visualize how its trajectory evolves under the influence of sustained thrust. By introducing a tunable exhaust velocity (**vel_ex**) and fixing the burn duration, we observed how thrust magnitude affects orbital shape over time. Varying **vel_ex** allowed us to track the corresponding changes in eccentricity, revealing a clear transition from circular to elliptical and, eventually, to hyperbolic escape trajectories.

The orbital energies we computed matched theoretical expectations: they were negative for bound (circular or elliptical) orbits, zero for parabolic escape, and positive for hyperbolic motion. Notably, **we found that the system reliably escapes orbit when vel_ex exceeds approximately 3 m/s, precisely addressing the central question posed at the start of our analysis.**

These results confirm that the simulation captures both the qualitative behavior and quantitative thresholds of gravitational mechanics under thrust and offers a strong foundation for modeling realistic orbital maneuvers.

Citation:

Taylor, John R. *Classical Mechanics*. University Science Books, 2005.