# Computational Final

geoffreyshennessy

May 2025

## 1 Introduction

We were tasked to create a working model of gravity within Python and then use that model to generate orbits for Ceres, Jupiter, and Saturn. To do this, we pulled real-world position and velocity vectors for all three bodies from NASA's JPL Horizons, ignoring $\hat{z}$ components. In addition, we also verified our model was working by plotting the total energy of the system. Once we had a working gravitation model, we also updated our model to simulate a stable three-body orbit.

## 2 Planetary Orbits

### 2.1 The Code

The main principle used for the planetary orbits was Newton's 2nd law:

$$\Sigma\vec{\mathbf{F}} = m\frac{\partial\vec{\mathbf{V}}}{\partial t}$$

The other principle used was Newton's Law of Gravitation:

$$\vec{\mathbf{F}} = mMG\frac{(\vec{\mathbf{x}}_m - \vec{\mathbf{x}}_M)}{|\vec{\mathbf{x}}_m - \vec{\mathbf{x}}_M|^3}$$

Note that $\vec{\mathbf{x}}_{m,M}$ are generalized position vectors. That means that the numerator $(\vec{\mathbf{x}}_m - \vec{\mathbf{x}}_M)$ is not normalized, so the denominator needs to be raised to the third power.

The logic of this choice was that we can calculate the net force that any body would experience at any given time based on its distance from the other bodies and the mass of the other bodies. We could then use the net force in Newton's 2nd law to get the change in velocity for that body. This was done component-wise. So, for each time step we knew the position, velocity, and acceleration of the body. We then plugged the positions and velocities of each body into a vector and used a fourth-order Runge-Kutta method to update the vector in accordance to Newton's laws for each time step. Since body orbits are on the order of years, we used a 6 hour time step. Note that *this was only done for forces from the Sun.* The forces of the other bodies were ignored. In addition, we assumed *that the Sun stays fixed at the origin at all time steps* and has a mass of $1,989e30$ kg. Finally, we chose the following initial parameters for each body from JPL:

| | m (kg) | x (m) | y (m) | $V_x$ (m/s) | $V_y$ (m/s) |
|---|---|---|---|---|---|
| Ceres | 1.9e20 | 4.13e11 | -1.45e11 | 5.19e3 | 1.57e5 |
| Jupiter | 18.98e26 | 1.667e10 | 7.757e11 | -1.321e4 | -9.06e2 |
| Saturn | 5.68e26 | 1.424e12 | -1.626e-11 | 5.952e2 | 9.576e3 |

## 2.2 Results

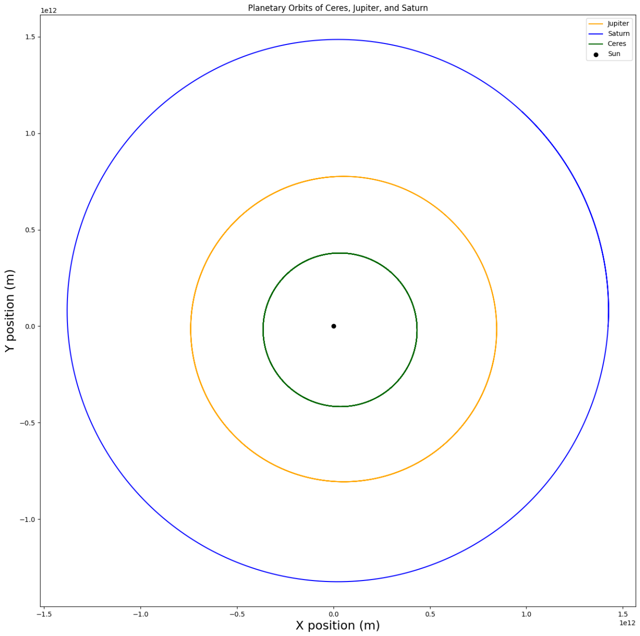

Figure 1: Orbits of Ceres, Jupiter, and Saturn

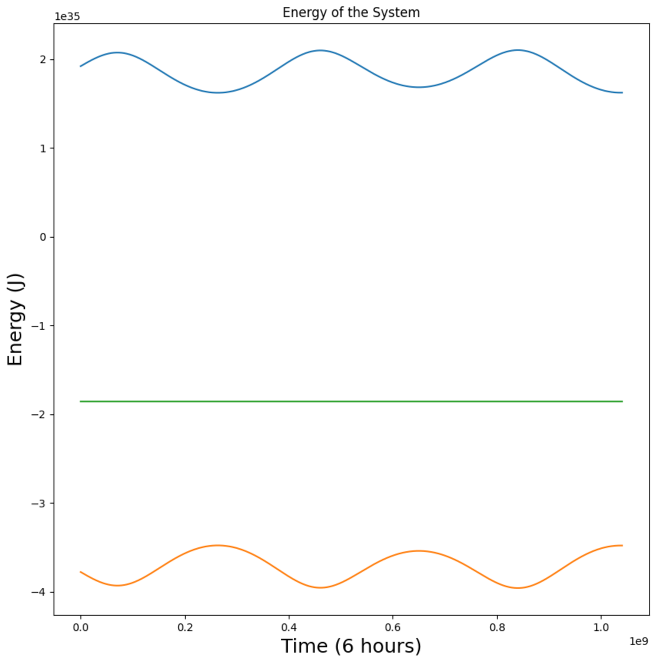

Figure 2: Energy of the system

## 2.3 Discussion

These initial conditions show some interesting behavior. To start, note that the orbits aren't centered. This showcases the elliptical nature of each orbit. You can also see how certain bodies like Jupiter and Saturn show less eccentricity than others like Ceres. In addition to this, the kinetic and potential energy of the system are cleanly periodic. This makes sense because of each body's periodic potential. However, what is interesting is how smooth it is. Since each body has it's own period, I was expecting a wavefunction that looks like a combination of waves. But figure 2 looks like one wavefunction. This might be a behavior that's true for orbits in general. It also could be because Saturn is in my system, and Saturn has a comparatively much larger energy. So, it's energy period dominates the others.

# 3 Stable Three-Body Orbit

## 3.1 The Code

We generated the stable three-body system using the same gravitation model from the orbits. However, during this simulation, we factored in the forces from all objects simultaneously. This involved expanding our $\vec{r}$ vector to include the positions of the other two bodies. This was used to calculate the sum of forces from each body, which was then used to calculate change in our positions and velocities during the Runge-Kutta. Notably, the Runge-Kutta only updated the positions and velocities of one body. So, we had to repeat it for each body. This gave us the change in all three bodies for each time step. An important consideration is that each body has to be updated *simultaneously*.

For our values, we set the gravitation constant G to 1. The mass of each object was 1. The positions of each body was set to $(-0.9892620043, 0), (2.2096177241, 0), (-1.2203557197, 0)$ respectively. The velocities $(V_x, V_y)$ were set to $(0, 1.9169244185), (0, 0.1910268738), (0, -2.1079512924)$ respectively. Finally, we ran the simulation three different times, 34 seconds each, with a time step of 10ms, 1ms, and 0.1ms for each trial.

## 3.2 Results

These are the various plots for each time step.

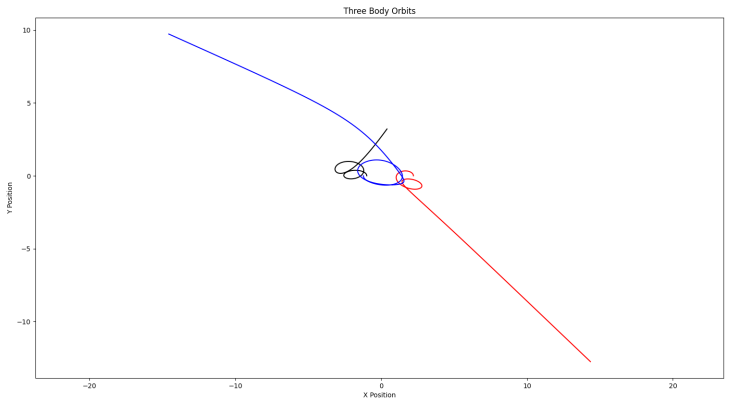

Figure 3: Three-Body System dt = 10ms

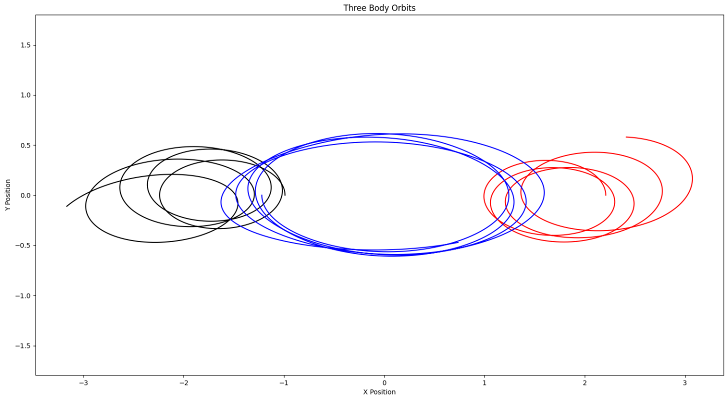

Figure 4: Three-Body System dt = 1ms

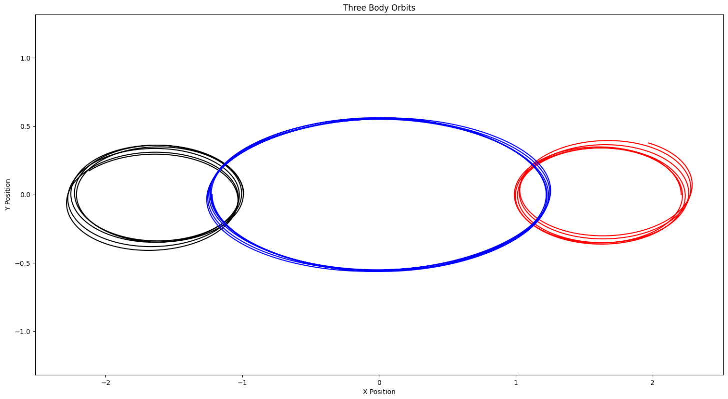

Figure 5: Three-Body System dt = 0.1ms

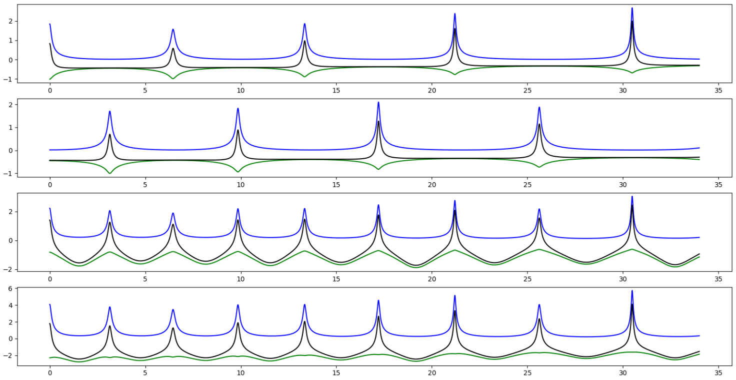

Figure 6: Kinetic, Potential, and Total Energy of Bodies 1,2,3

## 3.3 Discussion

The simulation demonstrates the chaotic nature of three-body systems. If the time step is too large, even relative to the total time, the system can diverge and become unstable. As you decrease the timestep, the system becomes more and more stable. However, for these simulations, decreasing the timestep also makes it more computationally expensive. We did not include the figures here, but we also tried different stable three-body systems with different initial parameters. These systems had much more intricate paths, and were also much more sensitive to time steps. We had one path we tried to create where we set the time step to 1ns, and it still diverged, despite the initial conditions being a known stable three-body system. The energy almost behaves ideally. It shows behaviors that match the system such as periodic spikes in kinetic and potential as the objects get close to each other. But the total energy of the system doesn't remain constant. We believe this is due to floating point error. With chaotic systems, even small changes can lead to noticeable changes. However, this doesn't explain why the total energy is also periodic. That remains unexplained.

## References

http://three-body.ipb.ac.rs/