April 2025

Simulating Orbital Motion for Analysis of the Solar System

Merle In der Rieden

Abstract

This simulation of the solar system found accurate values for the orbital periods for the planets tested. I found that the Direct Euler integration scheme was the least effective in conserving energy, and observed interesting oscillations about some mean total system energy for both the Euler Cromer and Beeman integration methods, with the Beeman method showing far lower deviation. I observed relatively frequent planetary alignments and tested for alignments with ten and five degrees of allowance.

1. Introduction

In order to understand the behaviour of large multiple body systems, we must be able to simulate them. This way, we can verify results without needing highly complicated measuring equipment, or waiting years for orbits to complete or planets to align. Despite the necessity for approximation by numerical integration, simulations are still a powerful tool capable of high levels of accuracy. In this simulation, I test the orbit periods of the planets in our solar system, compare the effectiveness of various numerical integration schemes, and detect instances of planetary alignment; I define planetary alignment as planets being within some allowance of their mean angle, without any restriction on the side of the sun which they lie on. I hope to observe results for orbit periods which are close to those published by NASA, and to observe a frequency of planetary alignment which corresponds to known data.

2. Methods

2.1. Set Up

I set up my simulation by declaring two classes: Planet, which held the methods for computing acceleration, updating position and velocity, and checking the orbital period, and Simulation, which contained methods for energy calculation, motion simulation, and alignment detection.

2.2. Acceleration

Mainly, I used numpy arrays to calculate the total acceleration on a planet. I did this by first initialising a *masses* array, containing the mass of each body in the simulation system except for the planet being defined by the current instance of the Planet class.

$$\text{Masses} = \begin{pmatrix} m_1 \\ m_2 \\ \vdots \\ m_i \end{pmatrix}$$

I then created a matrix containing the positions of all other planets using the numpy 'stack' function:

$$\text{P} = \begin{pmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_i & y_i \end{pmatrix}$$

I stored the displacements between the position of self and the planet represented by each row in P in a matrix:

Then, the Euclidean distance d from self to every other planet is calculated. Using Newton's law, I calculated the acceleration magnitude:

$$a = \frac{GM}{r^2}$$

However, I needed a vector $\vec{a}$ to represent acceleration. The direction unit vector is given by

$$\hat{r} = \frac{\vec{r}}{|\vec{r}|}$$

where $\vec{r}$ is the displacement vector. So, I found the acceleration as follows:

$$\vec{a}_i = \frac{GM}{d_i^2}\frac{\vec{r}_i}{|\hat{r}_i|} = \frac{GM}{d_i^2}\frac{\vec{r}_i}{d_i} = \frac{GM}{d_i^3}\vec{r}_i$$

So, the total acceleration on self was

$$\vec{a} = \sum_i \frac{GM}{d_i^3}\vec{r}_i$$

2.3. Initial Conditions

As numerical integration was used to execute the simulation, it was necessary to declare initial conditions. The initial positions of the planets were aligned with the sun, with their x positions according to their orbital radii as given in the json file, which is unrealistic. I set the initial velocity vector $\vec{v}$ as

$$\begin{pmatrix} 0 \\ \frac{GM_{\text{sun}}}{R_{\text{orbit}}} \end{pmatrix}$$

for planets. Again, this is quite unrealistic. The initial velocity for the sun was set as zero, and acceleration was initialised as a zero vector.

2.4. Energy

The total energy of the system is the sum of the kinetic and potential energies of all bodies. I calculated kinetic energy using the standard formula:

$$KE = \frac{1}{2}\sum m(\vec{v}\cdot\vec{v})$$

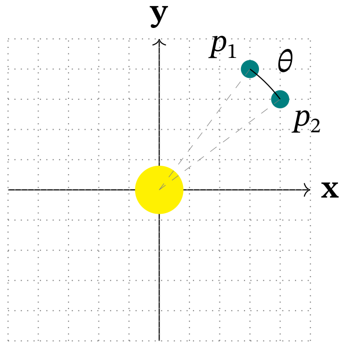
I calculated this for each body in the system and stored the total kinetic energy. Finding the gravitational potential energy acting on one body involves other system bodies, so there was a potential issue of double counting. For one planet, the GPE is as follows:

$$GPE = \sum \frac{GMm_i}{r}$$

where M is the planet's mass and m_i the masses of the other planets. I used enumerate to ensure energy was not double counted, and stored the system energy at some time instance as the sum of the KEs and GPEs at that instance.

2.5. Orbital Periods

To detect an orbit, I needed to compute the time at which a planet had rotated 2π around the sun. To do this, I repositioned the planet position vector every timestep to adjust for movement of the sun and still consider the sun as the origin, and calculate the angular travel per timestep. This angle was added to a variable storing total rotation. The first time instance at which this total was at least 2π was recorded as the orbital period.



2.6. Energy Conservation and Alternative Integration Methods

As numerical integration schemes are approximate, it is necessary to consider which scheme better simulates the system. One way to compare different methods is to check to what extend energy is conserved under each method, which is what I aimed to do in experiment 2.

Beeman:

$$\vec{r}(t+\Delta t) = \vec{r}(t) + \vec{v}(t)\Delta t + \frac{1}{6}[4\vec{a}(t) - \vec{a}(t-\Delta t)]\Delta t^2$$

$$\vec{v}(t+\Delta t) = \vec{v}(t) + \frac{1}{6}[\vec{a}(t+\Delta t) + 5\vec{a}(t) - \vec{a}(t-\Delta t)]\Delta t$$

Euler-Cromer:

$$\vec{v}(t+\Delta t) = \vec{v}(t) + \vec{a}(t)\Delta t$$
$$\vec{r}(t+\Delta t) = \vec{r}(t) + \vec{v}(t+\Delta t)\Delta t$$

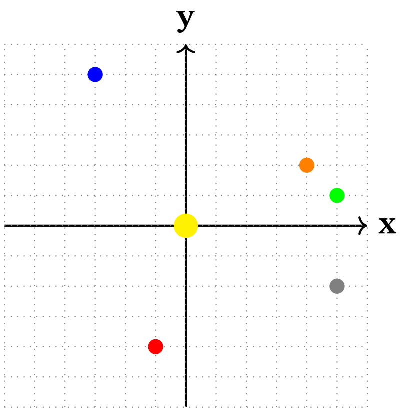
Direct Euler:

$$\vec{r}(t+\Delta t) = \vec{r}(t) + \vec{v}(t)\Delta t$$
$$\vec{v}(t+\Delta t) = \vec{v}(t) + \vec{a}(t)\Delta t$$

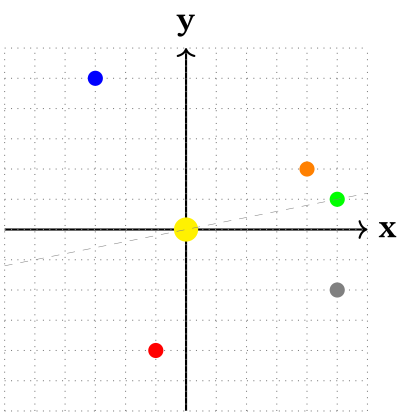
This required writing new sections in the methods for updating position and velocity and altering my simulation method. I checked and compared energy conservation graphically.

2.7. Planetary alignment

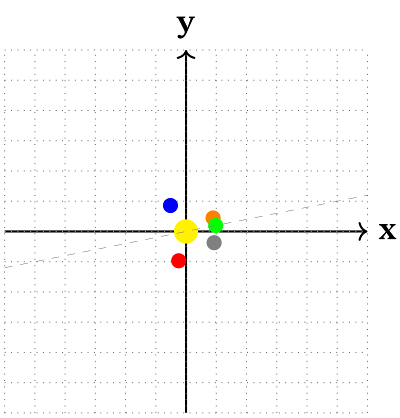
In this simulation, a planetary alignment is defined as planets being within some given angular range of the mean angle of the five innermost planets, either side of the sun. My method to detect an instance of alignment was as follows:



The mean vector is $\begin{pmatrix} 0.98058 \\ 0.19612 \end{pmatrix}$, so the mean line has equation $y = 0.20000x$



I then normalised the position vectors for the planets:



Then I checked the perpendicular distance between these points and the mean line, using the equation:

$$d = \left| \frac{x \cdot a - y \cdot b}{||(a, b)||} \right|$$

Here, (x, y) is the point and (a, b) is the unit direction defining the mean line. If this distance is less than $\sin\theta$ for some angle of allowance θ, then the point is within acceptable range of the mean angle. If all five points satisfy this condition, then an instance of planetary alignment is recorded.

3. Results

3.1. Animation

Accessing each planet's position every timestep allowed me to return an animation of the solar system. One frame of this animation appears below:

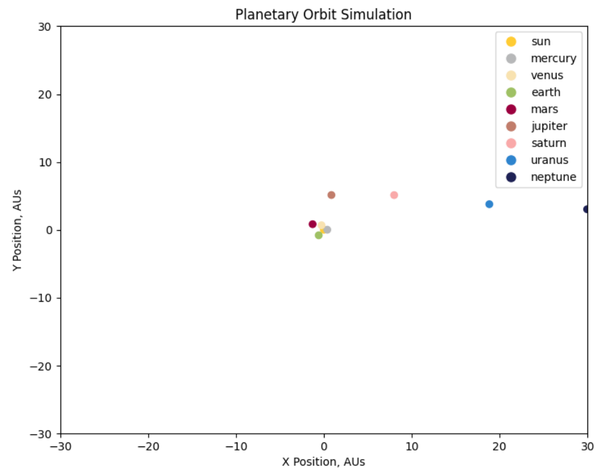


Figure 1. Solar System Animation

The frame shows the planets of the solar system orbiting the Sun at one time instance.

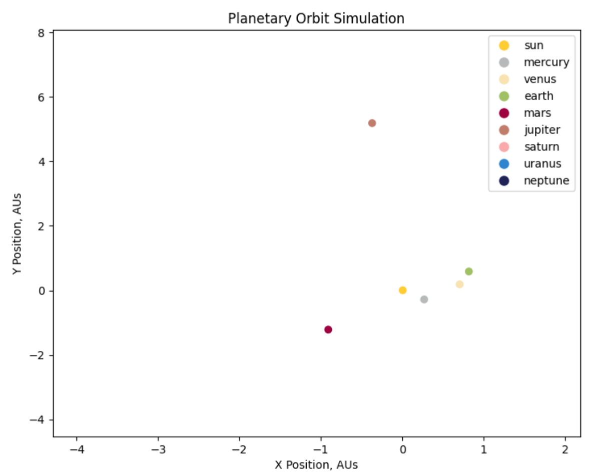


Figure 2. Five Innermost Planets

This figure better shows the distances between the innermost five planets. Zooming in on these planets in the animation itself shows the elliptical nature of the orbits.

3.2. Orbital Periods

Planet	Orbital Period	True Value
Mercury	0.241	0.241
Venus	0.615	0.615
Earth	1	1
Mars	1.881	1.881
Jupiter	11.842	11.859
Saturn	29.067	29.428
Uranus	83.209	83.760
Neptune	163.13	163.746

Figure 3. Simulated Orbital Periods and NASA's Orbital Periods, in Earth years

Figure 3 shows the orbital periods as found in the simulation as well as NASA's figures for orbital periods [1]. We see that the numbers are very close, only differing slightly for the planets beyond Mars.

3.3. Energy Conservation

By running the simulation using the three different integration methods and saving the energy every 100 time steps, I plotted the total the system energies to determine whether energy was conserved.

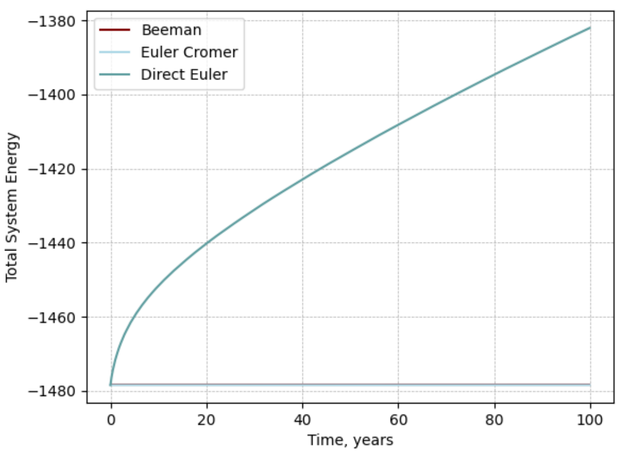


Figure 4. System Energies ($\times 1.34 \times 10^{32}$ J)for the Beeman, Euler Cromer, and Direct Euler methods in the first 100 years

Note that the Beeman line lies below the Euler Cromer line in figure 4. Figure 4 demonstrates that energy is certainly not conserved when using the Direct Euler integration method, but may be conserved for the Beeman and Euler Cromer methods. To gain better insight on this, I plotted the graphs of only Beeman and only Euler Cromer:

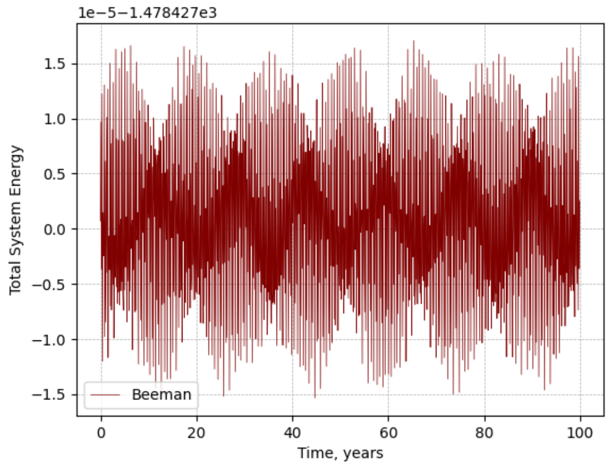


Figure 5. Beeman System Energy, first 100 years

Although there is some deviation, meaning that energy is not exactly conserved, we see that the magnitude of this deviation is incredibly small and tightly clustered about the central line.

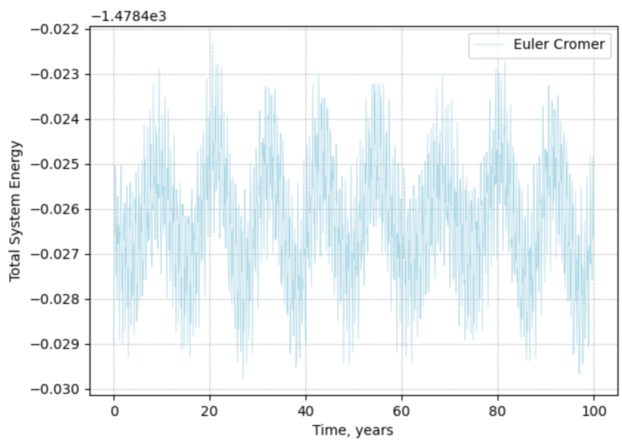


Figure 6. Euler Cromer System Energy, first 100 years

Euler Cromer integration introduces stronger sense of deviation and is far less tightly packed, demonstrating worse variance. The difference in deviation between Beeman and Euler Cromer integration is much clearer when the two are plotted together, as below.

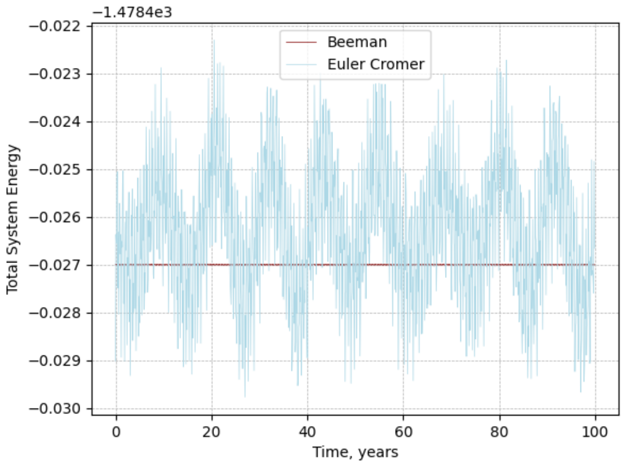


Figure 7. Beeman & Euler Cromer System Energies, first 100 years

These figures demonstrate that the Beeman integration method is, by far, best for this simulation as energy in the system should be conserved.

3.4. Planetary Alignment

$\pm 10°$ $\pm 5°$	0 0	22	95	139	161	234 234	256 256
$\pm 10°$ $\pm 5°$	278	283	306	329	351	467	490 490
$\pm 10°$ $\pm 5°$	512	540	562	585	746	768	796 796
$\pm 10°$ $\pm 5°$	818 818	841	957	979			

Figure 8. Occurrences of Planetary Alignment, Years

Observe that alignments with an allowance of ten degrees either side of the mean angle are far more frequent than with an allowance of five degrees.

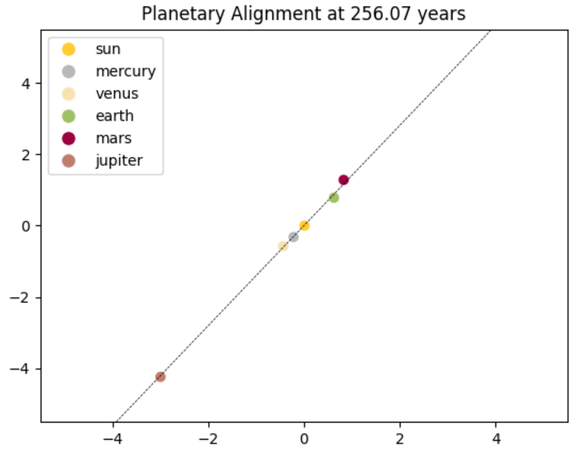


Figure 9. Planetary Alignment at 257 years

Figure 9 demonstrates my method of repositioning the planets such that the sun is at the origin, and constructing a mean vector (represented by a line passing the mean point and the origin). Above is an instance of planetary alignment with a $\pm 5°$ allowance.

4. Discussion

4.1. Animation

The animation was a successful demonstration of the planetary orbits, and constructing the animation such that zooming was possible allowed me to better observe the movements of the innermost planets. Although enabling this setting meant that the animation was slightly slower regardless of the simulation method, it was still a successful tool.

4.2. Orbital Periods

The results were quite successful. They showed matching orbital periods for the simulation and NASA's recorded values to 3 decimal places until Jupiter onwards. The percentage error is still very small so these discrepancies are not of particular concern, and are likely due to the various differences between this simulation and the actual solar system. For example, we do not take into account the additional forces due to moons and Pluto. The integration scheme as well as the initial positions and velocities may have impacted the orbital periods such that they were slightly different. To determine the period I recorded the first time at which the total rotation about the sun was more than 2π, meaning there could have been some slight delay. Additionally, I took the time period of the first orbit as the orbital period for the planet, rather than recording multiple orbital periods and calculating a mean.

4.3. Energy Conservation for different Integration Schemes

Figure 4 demonstrates that the Direct Euler integration scheme appears to cause a logarithmic increase of energy in the system; the simulation should conserve the system energy, so this shows that Direct Euler is not a useful method for the simulation. From figure 4, it appears that both Beeman and Euler Cromer integration conserve the system energy at just above $-1480 \times 1.32 \times 10^{32}$J. However, closer analysis showed variation in system energy for both of these methods. In figure 5, we see deviation for the Beeman scheme of approximately $\pm 1.5 \times 10^{-5}$, with periods of about 20 years for these deviation cycles. For Euler Cromer integration we observe a similar deviation cycle period of about 20 years, but the amplitude of the deviation is about 2 orders of magnitude larger; this difference is highlighted in figure 6 where both integration schemes are plotted together.

This occurs because using a numerical integration scheme will not produce an exact solution like an analytical solution could. The deviation will increase over time, using smaller timesteps lessens the deviation. This closer look at the Beeman and Euler Cromer energies demonstrates that Beeman integration is the better scheme for conserving energy and hence for a more accurate simulation.

4.4. Planetary Alignments

The experiment showed alignments occurring quite often, especially when I allowed for ten degrees of give either side of the mean angle. Given that I only tested for alignment of the five innermost planets and did not restrict the planets to having to be on the same side of the sun, this does not seem unreasonable. Unfortunately I was unable to find reliable sources to compare these results to.

5. Conclusion

This project aimed to simulate the solar system effectively enough such that the three experiments performed would be informative and would confirm known facts. The orbital periods experiment was very successful and showed only slight differences to the true orbital periods, likely due to the simpler nature of the simulation compared to the actual solar system. The closer investigation of integration schemes was also useful, and demonstrated interesting properties of the schemes as well as being informative on how a simulation could lessen deviation and so mitigate the potential problems caused by system energy not being conserved. Lastly, the planetary alignments experiment demonstrated the infrequency of alignment at more restricted angle allowances, and animating the results showed that most alignment results occurred not on the same side of the sun, suggesting that a restriction would produce far fewer alignment instances.

References

[1] D. R. Williams. “Planetary fact sheet - ratio to earth values”. (2025), [Online]. Available: https://nssdc.gsfc.nasa.gov/planetary/factsheet/planet_table_ratio.html.