Computational Physics - Assessment 2 - N-Body Investigation

[redacted]

December 2023

1 Introduction

The 'N-body problem' describes the physical challenge of evolving the motion of a system of masses under gravity. As it currently lacks an analytical solution, a computational approach must be employed to facilitate its investigation. One effective method that is often used, known as an 'Eulerian approach', projects the future position of a particle by extrapolating from its positional derivatives. To enhance precision, derivatives can be calculated at midpoints between time steps, rendering it 'semi-implicit'. This particular method falls under the category of 'symplectic integrators,' ensuring conservation of energy and angular momentum within the system. This study specifically evaluates the validity of a semi-implicit Eulerian computational method, known as a 'Velocity-Verlet integration', in its capacity to evolve a system of N-bodies while preserving computational stability.

2 Code Description:

Initial conditions are stored within text files that provide mass, position and velocity coordinates. These files contain data corresponding to either the Solar System or a predefined set of stable 3-body orbits. The code base enables a user to choose between options of the different initial conditions that will govern the evolution of the system. Upon user selection, the relevant file is read in to initialise the system. To evolve the system, the code first calculates the initial acceleration of the masses within the system as prescribed in 2.1. Then, it applies the velocity-verlet integration method in 3 dimensions to advance the system in time as prescribed in 2.2. The code then calculates the total energy and angular momentum of the system as in 2.3 and 2.4. This processes is then iterated a pre-prescribed number of times. Finally, the program prints the data to a text file, where a Python script plots the required positions, energy and angular momentum throughout the simulation.

2.1 Force Calculation

In a given time-step, the program first calculates the force vectors on a given mass by summation using of Newtons Law of Gravitation

$$F_{grav} = \frac{Gm_1m_2}{r^2}, \tag{1}$$

where F_{grav} is the gravitational force between the two masses m_1 and m_2, G is the gravitational constant, and r is their separation. To account for the fact that the masses are described as point masses, a gravitational softening parameter ϵ is implemented as

$$F_{grav} = \frac{Gm_1m_2}{r^2 + \epsilon}, \tag{2}$$

which limits the convergence of the gravitational force if separation tends to 0. The code stores each mass interaction, calculating the net force on a given mass, and then decomposing it into its component force vectors x, y and z, using the positional angles between the masses θ and ϕ as in figure 1:

$$\theta = \arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right); \phi = \arctan\left(\frac{((y_2 - y_1)^2 + (x_2 - x_1)^2)^{1/2}}{z_2 - z_1}\right) \tag{3}$$

$$F_x = F_{grav}\cos\theta \tag{4}$$

$$F_y = F_{grav}\sin\theta \tag{5}$$

$$F_z = F_{grav}\cos\phi \tag{6}$$

To improve computational efficiency, the code leverages Newton's Third Law. Specifically, when calculating the force between, for instance, particle i and particle j as F_{ij}, the code simultaneously stores the force F_{ji} as equal and opposite.

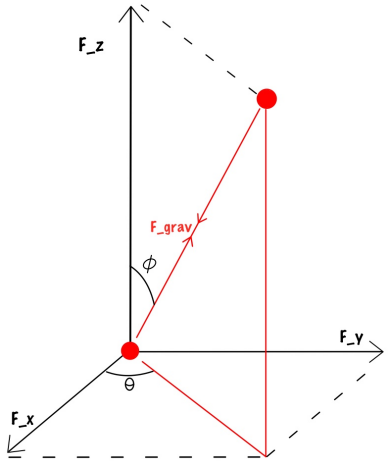

Figure 1: Force diagram showing gravitational force decomposed into vector quantities

By executing this for each mass interaction and summing the components on a given mass, we gain a net force in each direction, which when divided by the respective mass, gives the resulting acceleration vectors.

2.2 Position Calculation

To evolve the positions and velocities of the system within a given time-step, the code employs the velocity-verlet integrator. This involves first updating the velocity of all the particles at half the given time step Δt using:

$$v\left(t + \frac{\Delta t}{2}\right) = v(t) + a(t) \cdot \frac{\Delta t}{2} \tag{7}$$

Then, a full-step update for positions is applied:

$$x(t + \Delta t) = x(t) + v\left(t + \frac{\Delta t}{2}\right) \cdot \Delta t \tag{8}$$

Then, with the updated positions, the accelerations are re-calculated as in section 2.1. Finally, the velocities are calculated at the end of the time step using:

$$v(t + \Delta t) = v\left(t + \frac{\Delta t}{2}\right) + a(t + \Delta t) \cdot \frac{\Delta t}{2} \tag{9}$$

This method allows for time-synchronisation through calculation of velocity vectors twice per time-step.

2.3 Energy Calculation

The code then calculates the total energy of the system as

$$E_{tot} = E_{kin} + E_{grav}. \tag{10}$$

The total kinetic energy of the system, E_{kin}, is calculated by summing all individual kinetic energies of the particles

$$E_{kin} = \sum_{i=1}^{i=N} \frac{1}{2} m_i (v_{x_i}^2 + v_{y_i}^2 + v_{z_i}^2), \tag{11}$$

where N is the total number of bodies in the system, and v_{x_i}, v_{y_i} and v_{z_i} are the x, y and z velocity components of a specific mass. The total gravitational potential energy of the system, E_{grav}, is calculated by summing all the individual gravitational potential interactions between the particles:

$$E_{grav} = \sum_{i=1}^{i=N} \sum_{j=i+1}^{j=N} -\frac{G m_i m_j}{r}. \tag{12}$$

The condition for the second summation is necessary to ensure that each gravitational interaction is only counted once.

2.4 Angular Momentum Calculation

The program then calculates the total angular momentum of the system, L_{tot}, by summing all the individual angular momentum' within a given time-step.

$$L_{tot} = \sum_{i=1}^{i=N} L_i. \tag{13}$$

Within the program, each individual angular momentum is calculated as the cross product of the positional vector of the given mass from the origin r, and its linear momentum vector ρ

$$L = \vec{r} \times \vec{\rho}. \tag{14}$$

Expanded and simplified, this becomes

$$L = \begin{bmatrix} \hat{x} & \hat{y} & \hat{z} \\ x & y & z \\ \rho_x & \rho_y & \rho_z \end{bmatrix} = (y\rho_z - z\rho_y)\hat{x} + (z\rho_x - z\rho_z)\hat{y} + (x\rho_y - y\rho_x)\hat{z} \tag{15}$$

for which the magnitude can be calculated.

3 Results

In this section the code is tested to reproduce known solutions, assessing the performance and accuracy of the velocity Verlet integrator in the context of N-body simulations. The validation process is centered around three critical physical parameters: Position, Energy, and Angular Momentum.

3.1 Sun - Earth Orbit

We first analyse the Sun-Earth 2-body system through 5e5 runs with a 64-second time step, simulating a complete Earth orbit around the Sun and then comparing the result to the Cartesian equation for an ellipse

$$y = \pm\sqrt{(a^2 - x^2)(1 - e^2)}, \tag{16}$$

where a is the semi major axis of Earth, 1.496e11, and e is the eccentricity of Earths orbit, 0.017. The initial conditions for this simulation were acquired from the JPL Horizons on-line solar system data and ephemeris computation service, which provided initial velocities for each body, as well as barycentric coordinates [1].

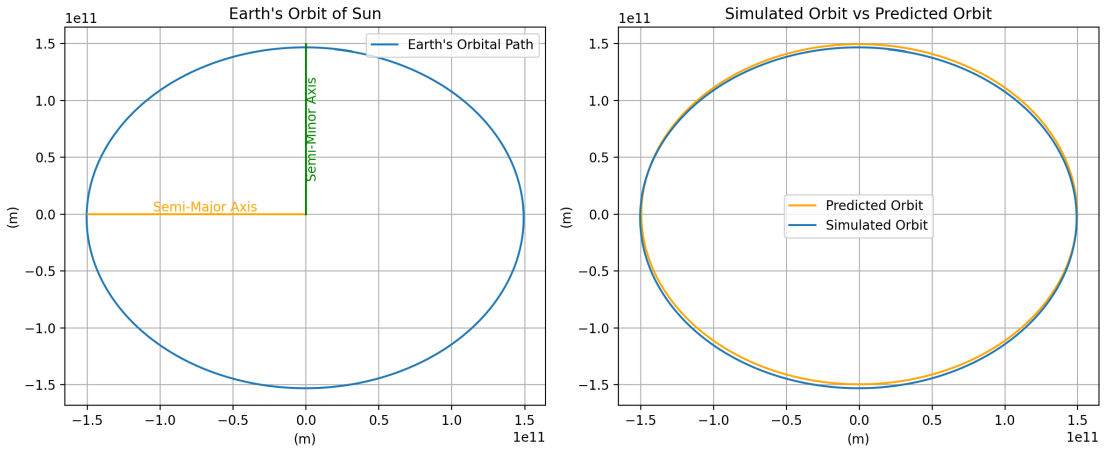

Figure 2: 2a) Two dimensional plot of the simulated orbit of Earth around the Sun in the 2-body case. 2b) Comparison of the simulated Orbit of the Earth with the elliptical equation predicted orbit.

Figure 2a illustrates the attainment of a closed orbit. The simulated orbit closely aligns with the predicted trajectory as shown in figure 2b, exhibiting near-perfect correspondence, with the only noticeable deviation being a slight tilt attributed to the initial conditions not precisely centering the ellipse.

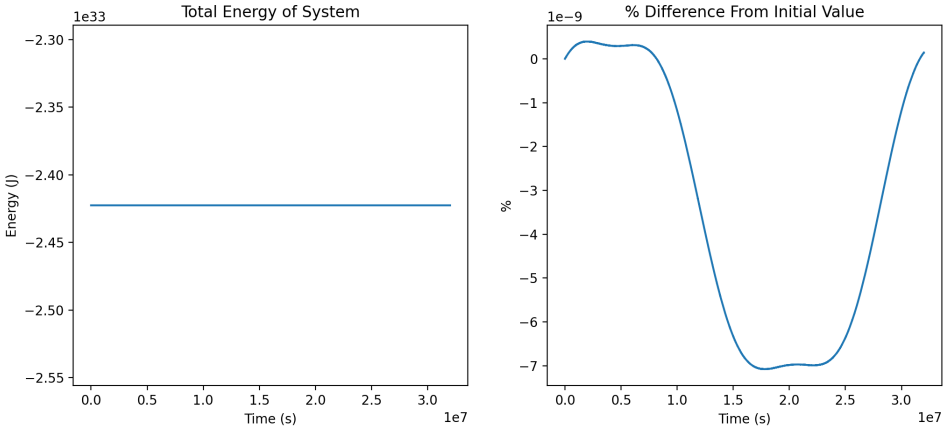

Figure 3: 3a) Total energy of the Sun-Earth system during one orbit. 3b) The percentage difference in energy of the system away from the starting energy of the system during one orbit.

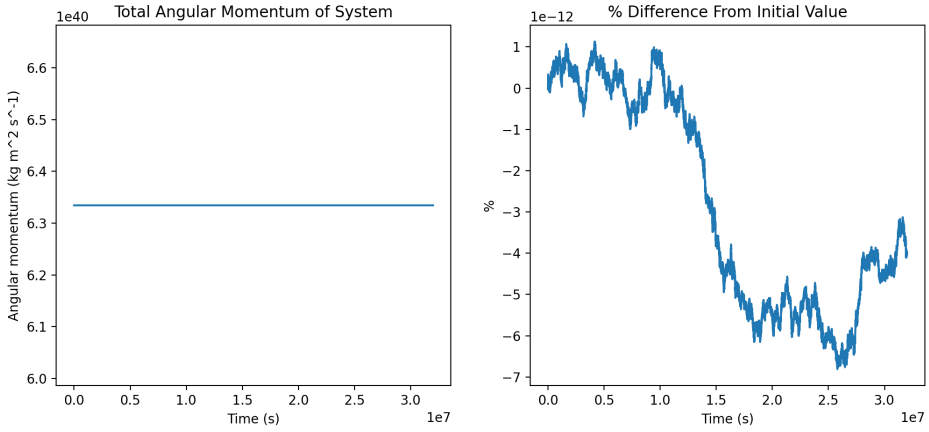

Figure 4: 4a) Total angular momentum of the Sun-Earth system during one orbit. 4b) The percentage difference of angular momentum of the system away from the starting angular momentum of the system during one orbit

Adherence to conservation laws is apparent, as illustrated in Figures 3 and 4. Throughout the orbit, the total energy and angular momentum of the system exhibit almost constant values, with a percentage error of -7e9% and -7e12%, respectively, indicating they are well conserved throughout the simulation.

The system was further simulated in three dimensions, as depicted in Figure 5. Here it becomes evident that the initial conditions impart a net velocity to the center of mass, and hence the Earth does not return to its origin in space. This departure from a closed system highlights the significance of considering three-dimensions for complete capture of the orbit.

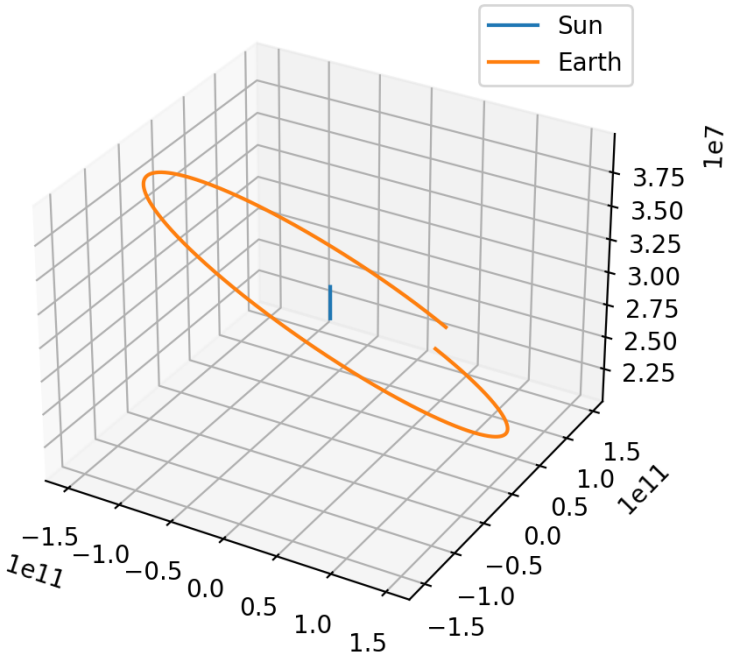

Figure 5: Three dimensional plot of the simulated orbit of Earth around the Sun in the 2-body case.

3.2 Solar System

After confirming the reliability of the method for the standard 2-body problem, the same approach was applied to simulate the entire solar system. The outcome of this simulation, conducted with a time step of 15e3 seconds, is illustrated in figure 6.

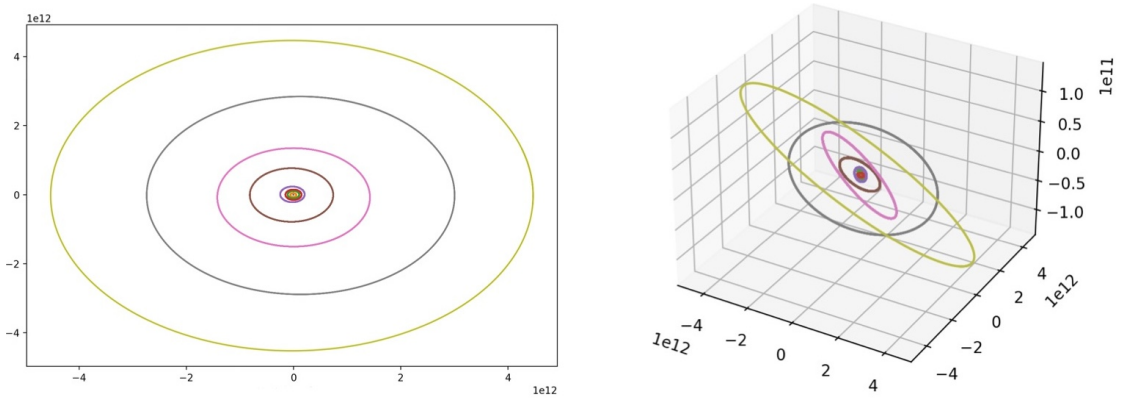

Figure 6: Simulated orbits for all 8 planets in the solar system with initial conditions set using data collated from the JPL Horizons on-line solar system data and ephemeris computation service [1]

.

The orbits depicted in figure 6 exhibit distinct elliptical shapes, confirming the validity of the method as this aligns with Kepler's First Law, which describes that planets orbit in ellipses with the sun at one of the foci. Despite the large time step required for Neptune to complete its orbit, both energy and angular momentum exhibit consistent conservation as depicted in figures 7 and 8, with the energy percentage error remaining within a narrow range of ±0.002%, and the angular momentum undergoing a maximum change of 2e-5%.

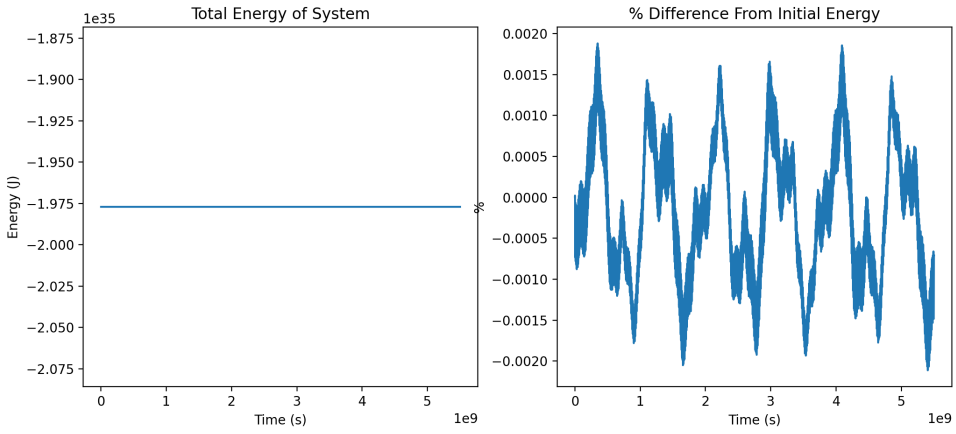

Figure 7: 7a) Total energy of the Solar System system for one Neptune orbit. 7b) The percentage difference in energy of the system away from the starting energy of the system during orbit.

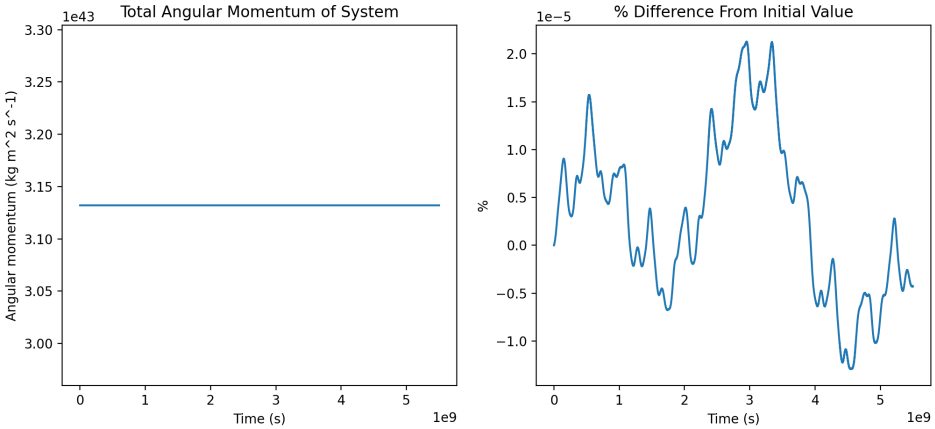

Figure 8: 8a) Total angular momentum of the Solar System system for one Neptune orbit. 8b) The percentage difference in angular momentum of the system away from the starting angular momentum of the system during orbit.

To further assess the method's validity, the orbital periods of each planet were determined. This was done by calculating the time taken for a planet to return to within a tolerance level of its starting point. The tolerance in this calculation was set to be 0.5% of the length of its semi-major axis. These periods are presented in table 1. These values were then tested to fit Kepler's third law. This states that The square of a planet's orbital period is proportional to the cube of the length of the semi-major axis of its orbit, where if the period is in years, and semi-major axis in au it becomes

$$p^2 = a^3. \tag{17}$$

Planet	Semi-major axis (au)	Orbital period (yr)	Kepler constant
Mercury	0.387	0.239	1.015
Venus	0.723	0.613	1.006
Earth	1.000	1.000	1.000
Mars	1.524	1.876	1.006
Jupiter	5.204	11.846	1.004
Saturn	9.572	29.350	1.018
Uranus	19.164	83.790	1.002
Neptune	30.180	164.351	1.018

Table 1: Table that displays known semi-major axis data, orbital periods derived from the simulation and the Kepler constants calculated using p^2/a^3 for each planet in the solar system.

Table 1 highlights the precision of the model, with the calculated Kepler constants closely aligning with the expected value of 1, exhibiting a maximum error of 1.8%. Similarly, Figures 9a and 9b affirm the model's accuracy, with the best fitted gradient displaying a 1.7% error in alignment with the expected gradient of 1. Additionally, the y-intercept exhibits negligible deviation from the expected value of 0. This conformity provides support for the simulation's adherence to Kepler's third Law.

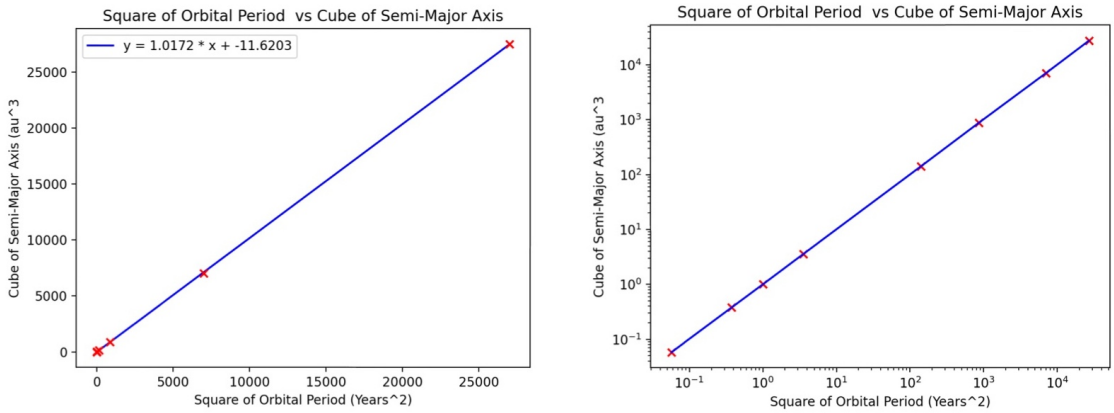

Figure 9: 9a) Plot of orbital period squared against the cube of the semi-major axis as presented in table 1. 9b) Plot of orbital period squared against the cube of the semi-major axis displayed on a log plot.

3.3 Known Stable Orbits

In this subsection, we test the velocity verlet integration against known 3-body stable orbits in order to further determine its validity. Here we investigate the methods sensitivity to different time steps, by simulating the stable orbits for one full period using varied time intervals.

3.3.1 Broucke A 1 [2]

The first orbit we look at is Broucke A 1 discovered in 1975. Known data about this orbit is presented in table 2, and plots of the orbital path, energy and angular momentum throughout the period are presented in figures 10 to 12.

Period (s)	Energy (J)	Angular Momentum (kgm^2s^{-1})
6.28	-0.85	1.10

Table 2: Table that displays period, energy and angular momentum data about the system [2].

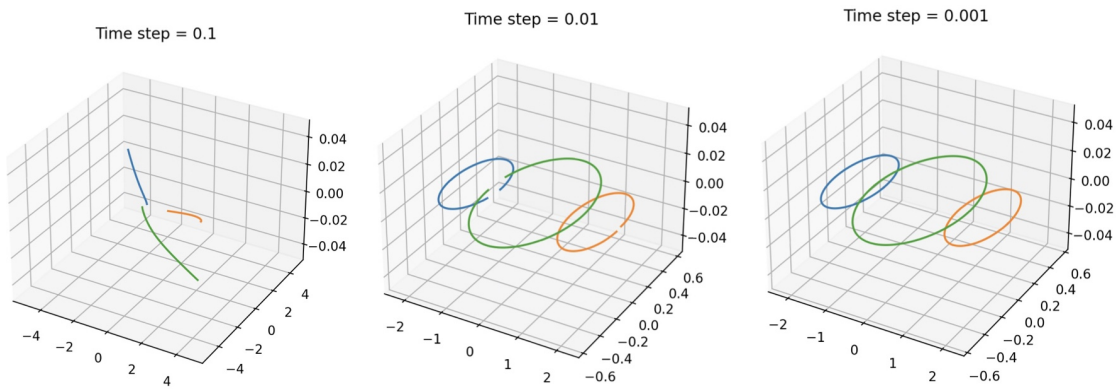

Figure 10: Broucke A 1 simulated orbital path with varying time steps. 10a) time step = 0.1. 10b) time step = 0.01. 10c) time step = 0.001.

The significance of employing a small time step in modeling interactions, especially when particles come into close proximity, is illustrated in Figure 10. Figure 10a reveals deviations from the anticipated path, indicating the sensitivity of the simulation to larger time steps. In contrast, Figure 10c, employing a 100 times smaller time step, displays a closed system with an orbital path aligning with expectations.

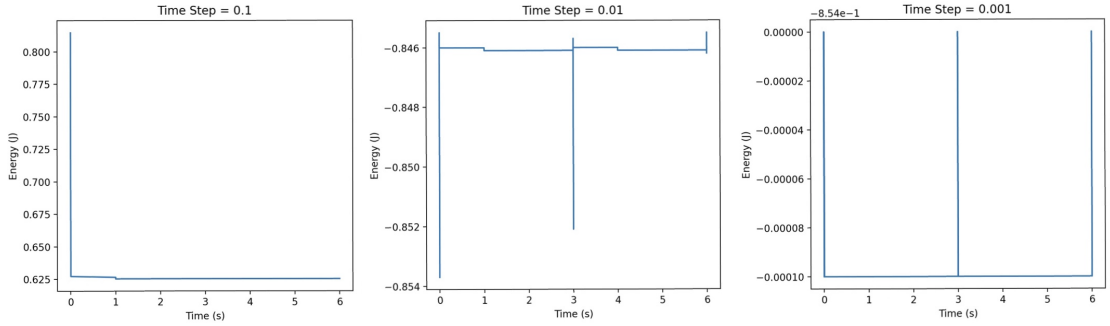

Figure 11: Broucke A 1 system energy throughout orbit with varying time steps. 11a) time step = 0.1. 11b) time step = 0.01. 11c) time step = 0.001.

Figure 11 provides insight into the orbital path deviations at larger time steps. In Figure 11c, where the time step is considerably smaller, the energy closely aligns with the true system energy outlined in Table 2. Additionally, a smaller time step results in significantly improved energy conservation throughout the orbit, with a maximum error of 1e-4 J compared to the higher errors observed in Figure 10a (0.2) and Figure 10b (1e-3), underscoring the direct relationship between time step size and energy conservation.

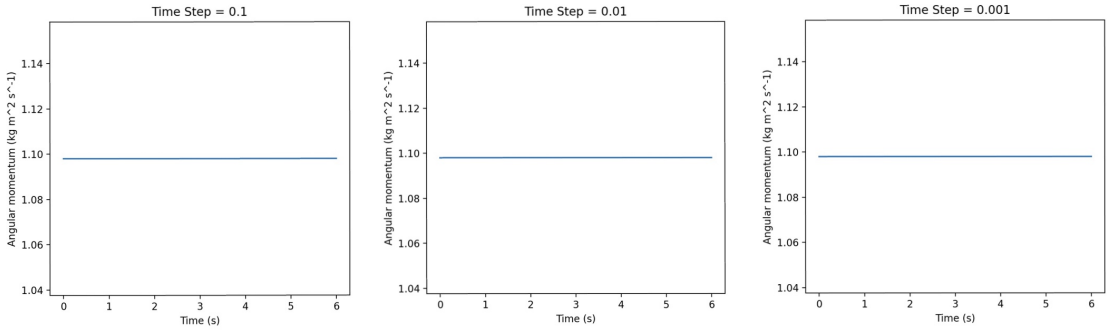

Figure 12: Broucke A 1 system angular momentum throughout orbit with varying time steps. 12a) time step = 0.1. 12b) time step = 0.01. 12c) time step = 0.001.

Figure 12 reveals consistent angular momentum conservation across all systems. This could be influenced by the relatively short total simulation time. To thoroughly assess angular momentum conservation, extending the simulation duration could be necessary as longer simulations may uncover subtle deviations in conservation trends not immediately apparent in the current time frame.

3.3.2 Figure of 8 [3]

The Figure of 8 stable orbit was also used in this analysis. Known data about this orbit is presented in table 3, and plots of the orbital path, energy and angular momentum throughout the period are presented in figures 13 to 15.

Period (s)	Energy (J)	Angular Momentum (kgm^2s^{-1})
6.32	-1.29	0.00

Table 3: Table that displays period, energy and angular momentum data about the system [3].

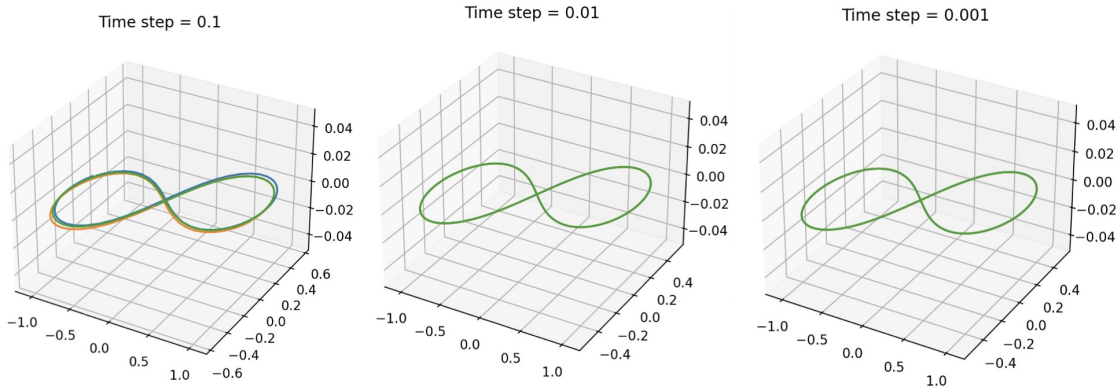

Figure 13: Figure of 8 simulated orbital path with varying time steps. 13a) time step = 0.1. 13b) time step = 0.01. 13c) time step = 0.001

Figure 13 demonstrates the importance of utilising a time step significantly smaller than the total period of the orbit to achieve accurate modelling. This is evident in the comparison between Figure 13a and Figure 13c, where the latter exhibits a considerably tighter orbit due to the implementation of a smaller time step.

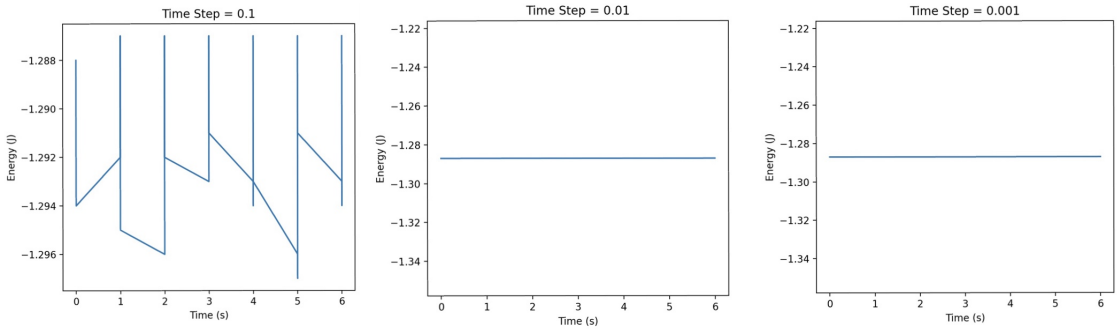

Figure 14: Figure of 8 system energy throughout orbit with varying time steps. 14a) time step = 0.1. 14b) time step = 0.01. 14c) time step = 0.001.

Figures 14 and 15 collectively emphasize the method's accuracy, as both the calculated energies and angular momenta consistently align with the expected values. It is evident from figure 14 that energy conservation is better achieved with a smaller time step. However, an intriguing observation emerges in the context of this specific orbit when examining angular momentum conservation, as illustrated in Figure 15. Surprisingly, the angular momentum appears to be better conserved with a larger time step, however, this could be attributed to the limited simulation time.

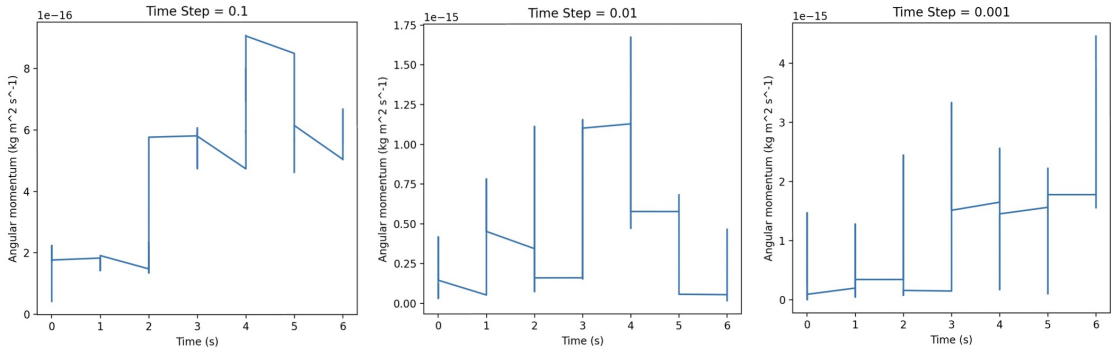

Figure 15: Figure of 8 system angular momentum throughout orbit with varying time steps. 15a) time step = 0.1. 15b) time step = 0.01. 15c) time step = 0.001.

3.3.3 Additional Orbits

This subsection examines the method's performance under more intricate orbital patterns. The tests were configured with the minimum time step necessary to complete one full orbit. Table 4 provides the known data for all orbits presented in this section.

Name	Period (s)	Energy (J)	Angular Momentum (kgm^2s^{-1})
Butterfly	6.24	-2.17	0.00
Moth	25.84	-1.63	0.00
Dragonfly	21.27	-1.44	0.00
Broucke R 12	17.02	-1.44	0.00

Table 4: Table that displays period, energy and angular momentum data for each system [2, 4]

Figure 16 demonstrates the validity of the method at small time steps as the orbits depicted present stable, closed systems that align closely with expected behavior. The success of the method in reproducing expected orbital patterns confirms its reliability in capturing N-body gravitational interactions when using suitably small time steps. However, in investigating these more intricate orbits, it was discerned that angular momentum and energy were not fully conserved. This suggests that over extended simulation duration's, there exists the potential for destabilisation and the onset of chaotic behavior within the simulated systems.

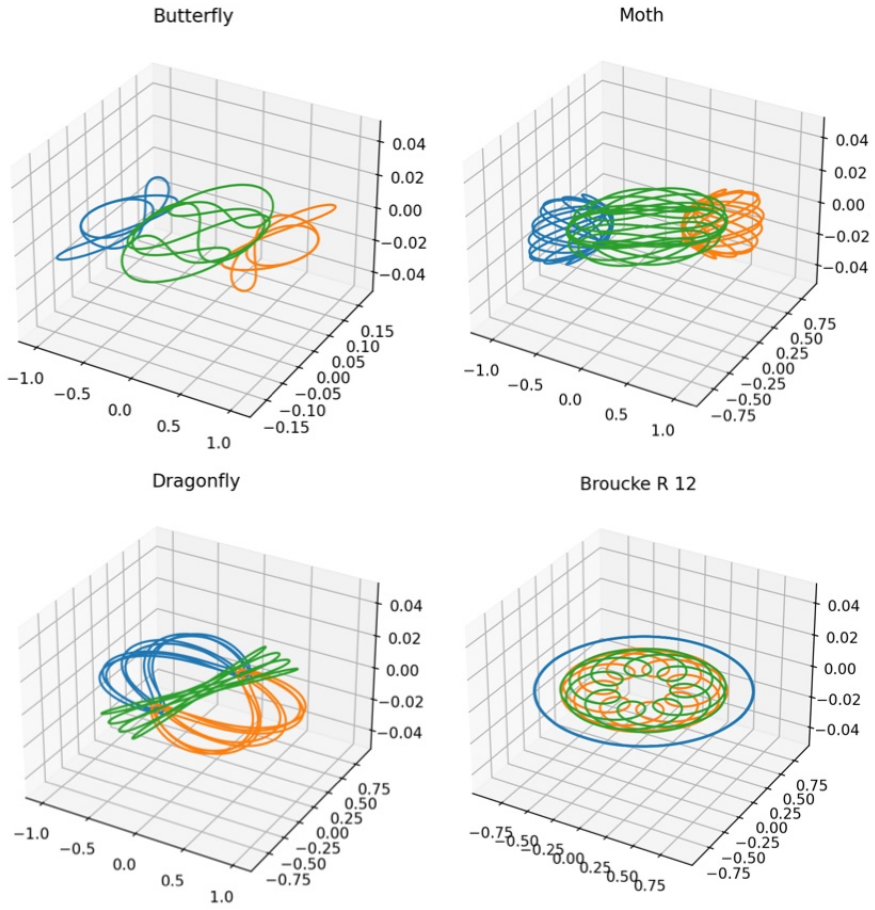

Figure 16: Simulated orbital paths for the stable orbits named in table 4.

References

[1] N. J. P. Laboratory, "Jpl horizons on-line solar system data and ephemeris computation service." `https://ssd.jpl.nasa.gov/horizons/`.

[2] R. Broucke, "On relative periodic solutions of the planar general three-body problem," *Celestial mechanics*, vol. 12, no. 4, pp. 439–462, 1975.

[3] A. Chenciner and R. Montgomery, "A remarkable periodic solution of the three-body problem in the case of equal masses," *Annals of Mathematics*, pp. 881–901, 2000.

[4] M. Šuvakov and V. Dmitrašinović, "Three classes of newtonian three-body planar periodic orbits," *Physical review letters*, vol. 110, no. 11, p. 114301, 2013.