

Project 5

For the course FYS3150

Erik Grammeltvedt, Erlend Tiberg North and Alexandra Jahr Kolstad

December 19, 2019
Week 50 - 51

Abstract

This project gives an astonishingly accurate and good looking model of the solar system. Using a 3D graphing tool in python the project has mapped the planets position for several years to come. This was possible due to the numerical Velocity Verlet method. The methods potency of mapping solar system is so good due to it's ability to conserve energy, which is very important when mapping the planetary motion. This makes the method more favorable over other methods such as the Euler-Cromer method which does not conserve energy. After writing the necessary code in order to simulate the solar system for the Euler-Cromer method and the Velocity Verlet method the conclusion is that Velocity Verlet is superior. Recognising that for some planets the relativistic effects on them affect their trajectory, a new plot that accounted for the relativistic effects was made. Having made a seemingly functioning model one could now go on to experiment with different values. Figuring out by trial and error how big the Earths initial velocity needs to be in order to escape the Suns gravitational pull. The opportunities that present themselves once on such a mode is endless. Introducing new objects to the system for instance or having fun with a self made space craft. This model is a great building block on which one can expand on.

Contents

1	Introduction	3
2	Theory	3
2.1	Important calculations for this project	3
2.2	Velocity Verlet method	3
2.3	Euler method	4
3	Method	4
3.1	Euler-Cromer implementation	5
4	Results	5
4.1	Sun-Earth solar system	6
4.1.1	Testing the methods and their stability	6
4.1.2	Testing the escape velocity	9
4.1.3	Testing different values of β	15
4.2	Sun-Mercury solar system	26
4.3	Sun-Earth-Jupiter solar system	28
4.3.1	Massfactor 1	28
4.3.2	Massfactor 10	35
4.3.3	Massfactor 1000	42
4.4	Dynamic sun	49
4.5	Ten-body solar system	50
4.5.1	Static sun	50
4.5.2	Dynamic sun	51
5	Discussion	52
5.1	Stability of the Sun-Earth solar system given by the Euler-Cromer method and the Velocity Verlet method	52
5.2	The escape velocity of the Earth in the Sun-Earth solar system	52
5.3	Varying β for the Sun-Earth solar system	53
5.4	Stability of the Sun-Earth-Jupiter solar system	53
5.5	Varying mass for Jupiter in the Sun-Earth-Jupiter solar system	53
5.6	Moving Sun in the in the Sun-Earth-Jupiter solar system	53
5.7	Model for all planets of the solar system	53
5.8	Perihelion precession of Mercury	54
6	Conclusion and perspective	54
7	References	55
A	Appendix	55
A.1	Mass-conversion	55
A.2	Explaining the calculation of a planets escape velocity	55
A.3	Rewriting Newtons second law of motion	56
A.4	Adjusting Newtons method for relativity	57
A.5	Measuring frequencies	58

1 Introduction

Starting with Newtons second law of motion and Newtons law of universal gravity one can simulate the solar system with quite high accuracy. First by discretizing the differential equations that one gets when wanting to solve Newtons second law of motion for position and velocity. To do this in the best possible way the project compares two different numerical methods. The methods used was Velocity Verlet and Euler-Cromer. The Velocity Verlet method proved to be superior both in theory and from the results acquired from the method. This is mainly due to the Velocity Verlet method's ability to conserve energy. In order to simulate the entire solar system the program need to have object oriented code to make the code more fluid, and also because it gave a much better arrangement of the code. This lead to our code being separated into three different types of programs that were needed to make the full simulation. Making a program for the objects seemed like the best solution in order to keep the program tidy. Then the `main.cpp` program uses the numerical methods in order to generate the data. The data is then put into several different python programs for plotting and getting the 3D simulations that one can find in the results. Using these programs this project has been able to simulate the solar system over a period of ten years with ease. The planets move around in quite accurate 3D orbitals. This study is supposed to show how one would approach a problem like this and overcoming the obstacles one might face. The study starts by explaining the main theory of both methods used and presenting how to use them to solve the problems and simulate the solar system. From there it explains the programs that were used to get the results. For instance by trial and error we found that the escape velocity of the earth is quite similar to the theoretically calculated escape velocity from the appendix, see section [A.2](#). That was for a two planetary system only.

2 Theory

2.1 Important calculations for this project

Using the section [A.2](#) in the appendix one can calculate the escape velocity of the Earth from the Suns gravity field. The distance from the Earth to the Sun is 1 AU (astronomical unit). G is given as $4\pi^2$. M and m is the mass of the Sun and the planet, which in this case is the Earth. M is given as $1 M_\odot$ and m is equal to $\frac{1}{33333} M_\odot$. Using the theoretical expression [\(13\)](#) one can calculate the earths escape velocity. For this two-body system the escape velocity is

$$v_p = 8.886 \text{ m/s} \quad (1)$$

2.2 Velocity Verlet method

The Velocity Verlet method is a differential equation solver, which was used in this project to solve Newtons second law, given by equation [\(14\)](#). Equation [\(14\)](#) will be used to calculate the next position of the planet as a Taylor expansion. The advantage of using the Velocity Verlet method is that it conserves energy.

Deriving the equations used in this method from Newtons second law of motion is done in section [A.3](#). The method starts by using the old acceleration, velocity and position to calculate the new position. Using equation [\(2\)](#), which is the discretized version of equation [\(19\)](#).

$$x_{i+1} = x(i) + hv(x, t) + \frac{h^2}{2}a(x, t) + (O)h^4(8) \quad (2)$$

The thing that separates this method from the Euler method is that in order to calculate the new velocity the Velocity Verlet method first calculates a new acceleration using the newly gained position, giving $a_{i+1} = a_i(x_{i+1})$.

The newly calculated acceleration is now put into the expression for the new velocity.

$$v_{i+1} = v_i + \frac{h}{2}(a_i + a_{i+1}) \quad (3)$$

The equations (2) and (3) makes up the Velocity Verlet method, and allows for conservation of energy, which is important when working with the solar system.

2.3 Euler method

The Euler method is in many ways similar to the Velocity Verlet method. However, instead of calculating a new acceleration and using that to find the velocity, the method uses the old position to calculate the acceleration and uses that acceleration to find the new velocity. The old velocity is also used to find the new position. Another important difference is that the position is not calculated using the acceleration. Other than that the methods are quite identical.

The equations used are derived from Newtons second law of motion. This was done in the appendix under, see section A.3.

First one calculates the velocity using the old position.

$$v_{i+1} = v_i + a(x_i) h \quad (4)$$

Using the newly gained velocity one can now find the new position.

$$x_{i+1} = x_i + v_i(x_i) h \quad (5)$$

This method is less precise compared to the Velocity Verlet method and it does not conserve energy.

3 Method

There are three different types of programs that are important for the results that have been acquired in this project. The first program in the section generates objects that will be callable in the calculation program. The calculation program finds all the future values for position and acceleration of the planets. Then there are plotting programs that plots the movement of the planets in three dimensions.

The first file is the `body.h` file. This program reads a `.txt` document with all the initial planetary data from NASA. It stores the data as objects that are callable through the planets name. It contains information, such as the initial position and velocity.

The second code, `main.cpp` simulates the solar system in the future using different types of numerical methods. First the program is set up with several different functions, with different tasks. Then there is a set of functions that calculates different parts of the Euler-Cromer method and the Velocity Verlet method. These functions will be called on in the main function. The main function starts by reading a file that contains initial values such as number of years. Reading it this way allows for quick changes in a `.txt` file instead of changing such values in the program. Then the program proceeds to calculate the values for position, velocity and acceleration. These values are now stored and sent to the third stage.

The third stage is the plotting stage, where all the date can be analyzed. This project has used several different python plotting programs in order to get the data presented in the best looking way possible.

3.1 Euler-Cromer implementation

We use the Euler-Cromer method instead of the Euler method in this project. This is because Euler-Cromer is a better approximation than Euler, and is also easily implemented. Euler-Cromer is a better approximation because it uses the next step in velocity in the current step in position. Because of these advantages we decided to use the Euler-Cromer method instead of the plain Euler method. The Euler-Cromer method is given by the equations (6) and (7) below.

$$x_{i+1} = x_i + v_{i+1} h \quad (6)$$

$$v_{i+1} = v_i + a_i h \quad (7)$$

Here we can see that the main difference between the two methods are the next velocity step in the equation for the position, (6). However, the Euler-Cromer method does not conserve energy as well.

One thing to also note is that in multiple figures and animations the name Euler-Cromer has been mistaken for Forward Euler. The process of fixing this would take too much time compared to the benefit, therefore it is stated here that we are aware that these two methods are different, but did not have the time to fix it.

4 Results

All plots which do not have the number of iterations specified have 3660 iterations. All plots which do not have the number of years specified have 10 years.

By reading the PDF for the project a little more thoroughly we discovered that in the latter parts from testing the algorithms, it says to only implement the Velocity Verlet method. However, considering how time consuming the making of the data, plots and figures in L^AT_EX we will leave the plots in this PDF, but choose not to comment them further than if the results are interesting.

See also our GitHub for some animations of the solar systems. These proved to be too complicated and time consuming to implement in the L^AT_EX file. For instance for the results in 4.1.1 there are some animations. See this [link](#) for the folder img for animations and pictures. The animations are in 3D and are given by the initial letters ani3D. For instance S_E in the filename gives the Sun-Earth solar system and F gives the Forward Euler, also mistakenly known as Euler-Cromer.

4.1 Sun-Earth solar system

4.1.1 Testing the methods and their stability

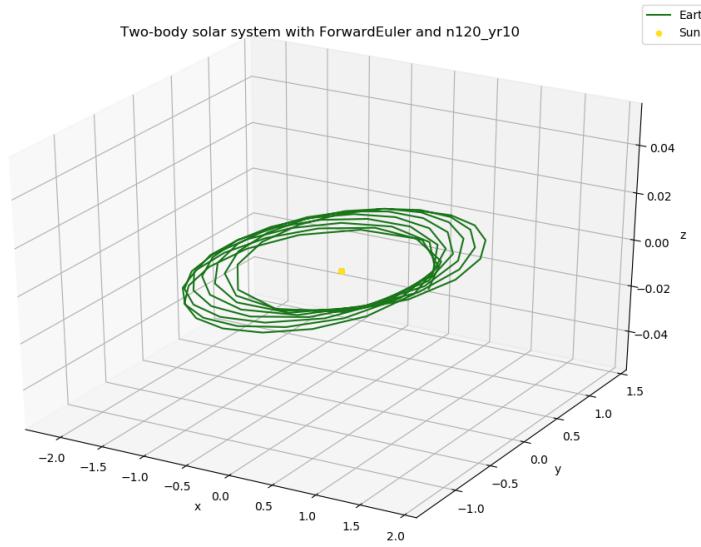


Figure 1: Plot of the Sun-Earth solar system with 120 iterations over 10 years, made with Euler-Cromer.

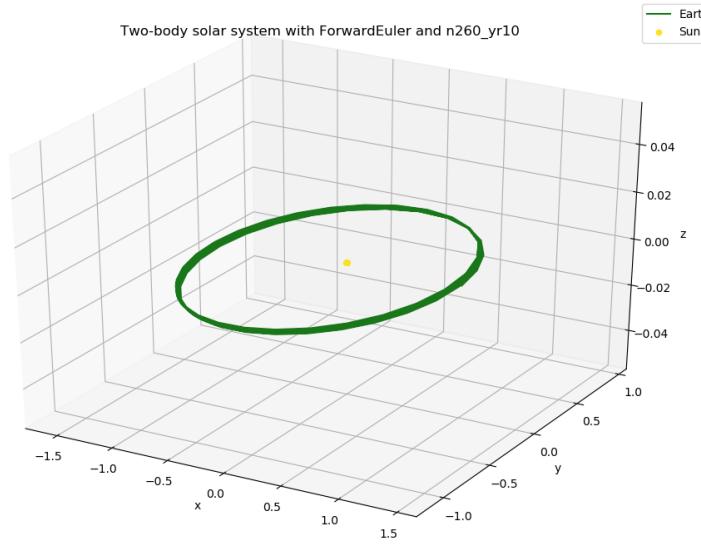


Figure 2: Plot of the Sun-Earth solar system with 260 iterations over 10 years, made with Euler-Cromer.

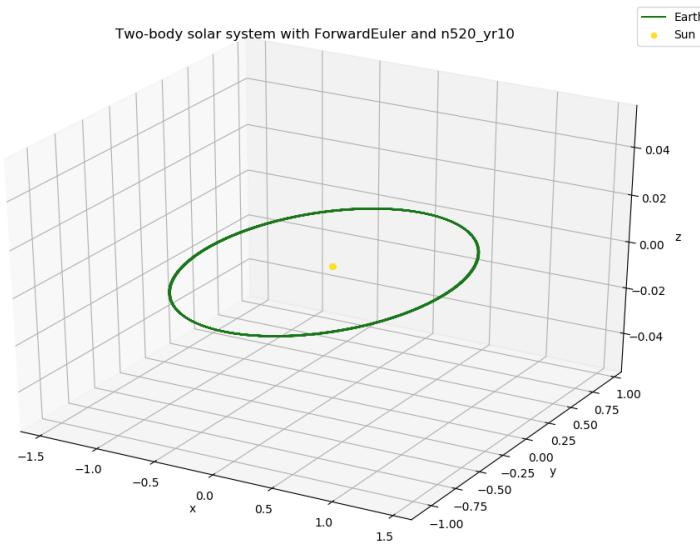


Figure 3: Plot of the Sun-Earth solar system with 520 iterations over 10 years, made with Euler-Cromer.

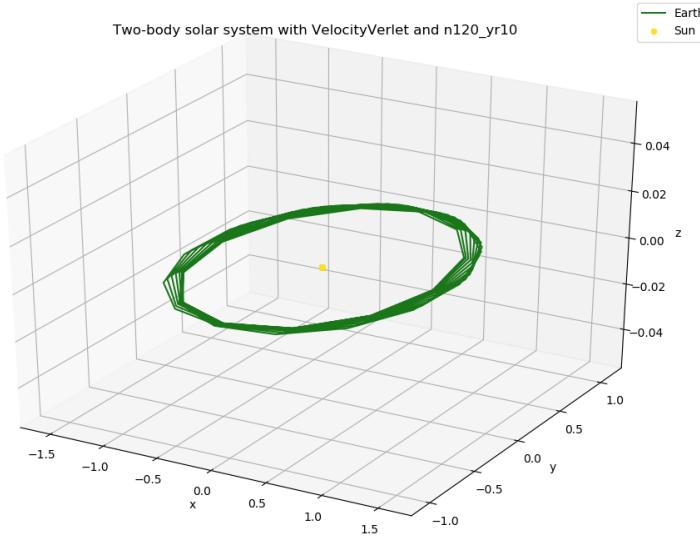


Figure 4: Plot of the Sun-Earth solar system with 120 iterations over 10 years, made with Velocity Verlet.

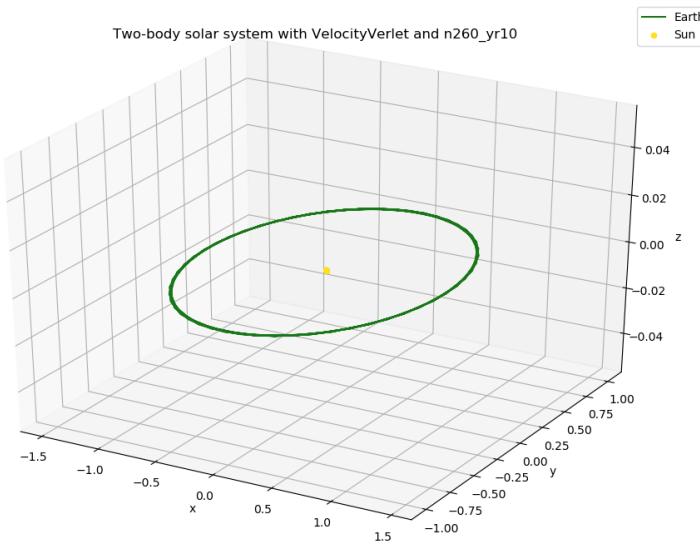


Figure 5: Plot of the Sun-Earth solar system with 260 iterations over 10 years, made with Velocity Verlet.

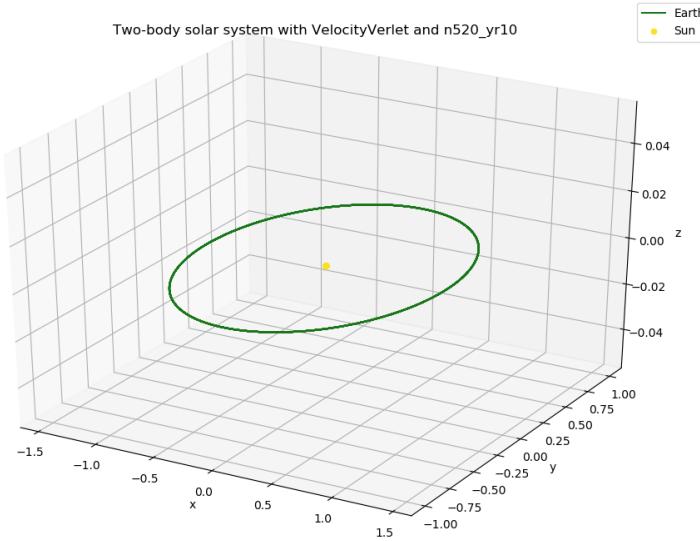


Figure 6: Plot of the Sun-Earth solar system with 520 iterations over 10 years, made with Velocity Verlet.

Euler-Cromer time [s]	Velocity Verlet time [s]
25.8553	34.0578
30.0186	30.5725

Table 1: Timings for the two methods for the ten-body solar system with 3660 iterations over 10 years.

4.1.2 Testing the escape velocity

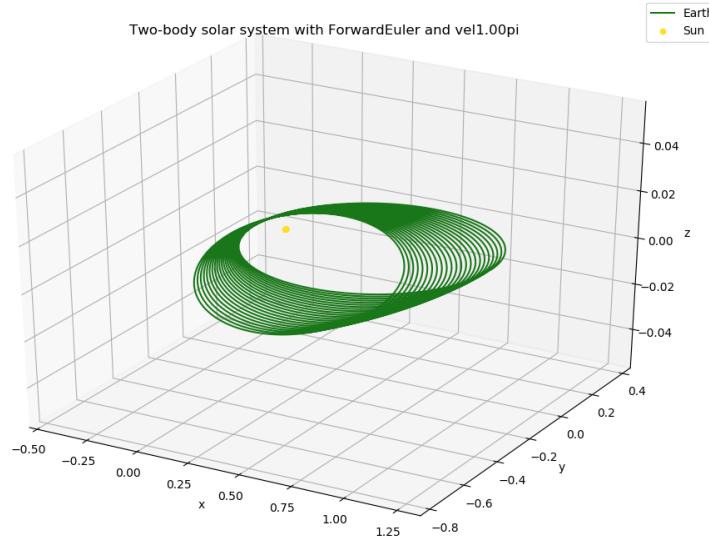


Figure 7: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 1.00 \pi$ m/s, made with Euler-Cromer.

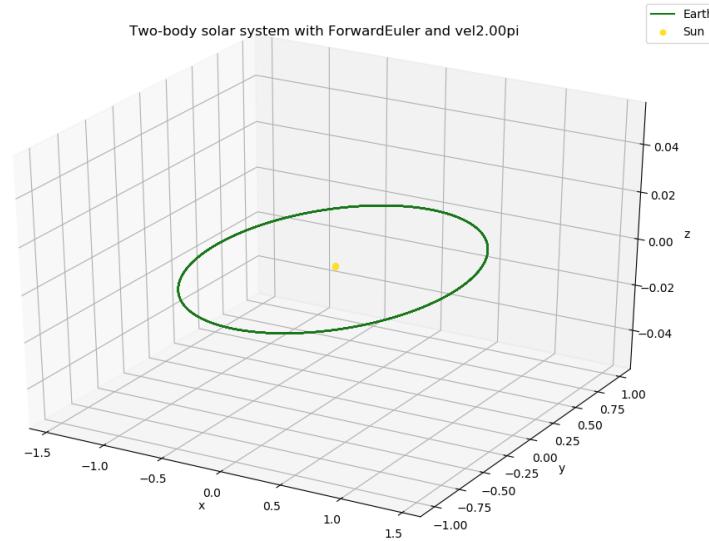


Figure 8: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 2.00 \pi$ m/s, made with Euler-Cromer.

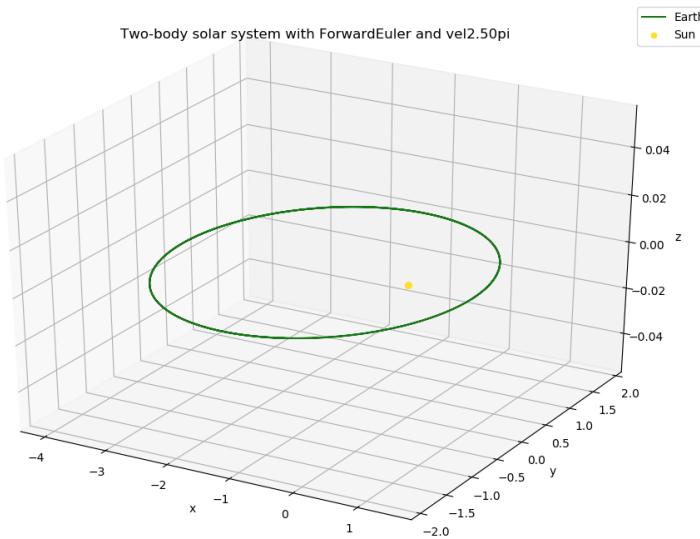


Figure 9: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 2.50 \pi$ m/s, made with Euler-Cromer.

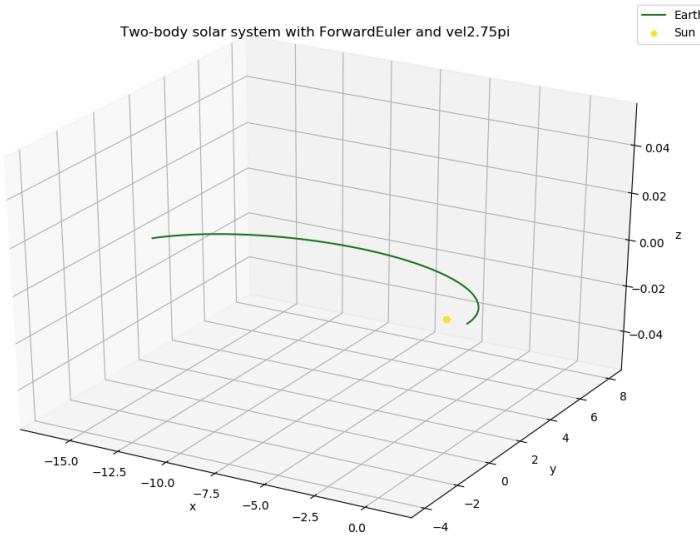


Figure 10: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 2.75 \pi$ m/s, made with Euler-Cromer.

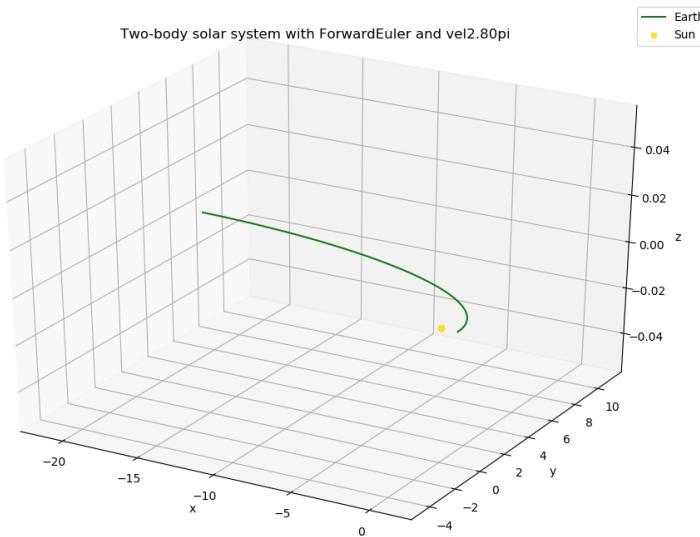


Figure 11: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 2.80 \pi$ m/s, made with Euler-Cromer.

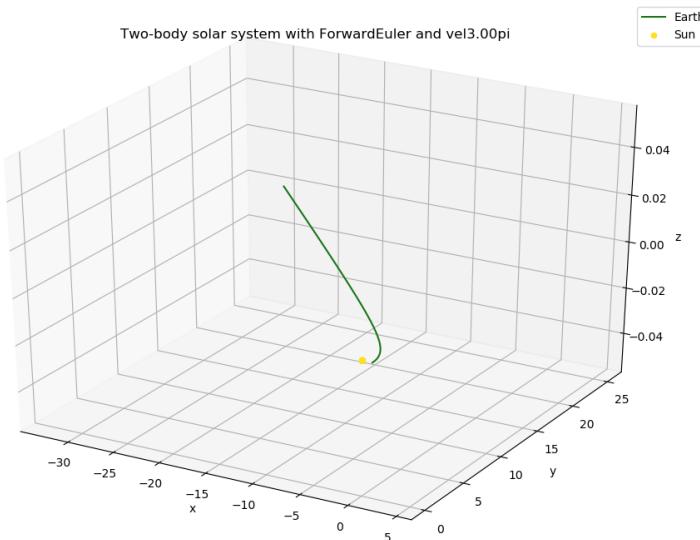


Figure 12: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 3.00 \pi$ m/s, made with Euler-Cromer.

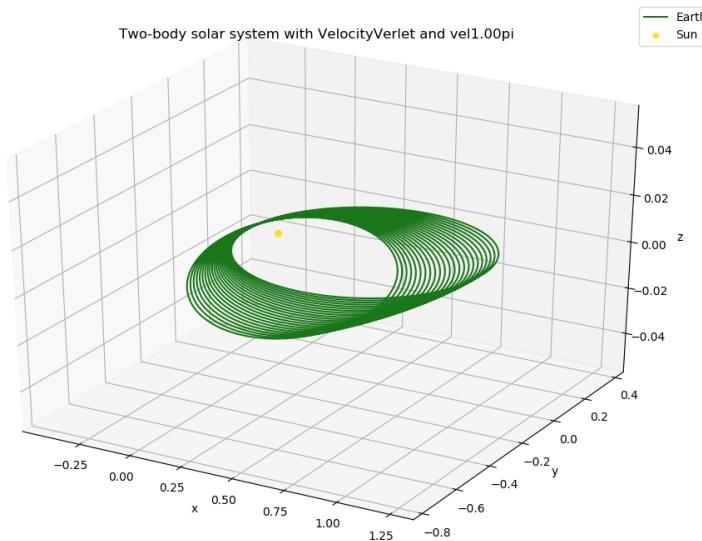


Figure 13: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 1.00 \pi$ m/s, made with Velocity Verlet.

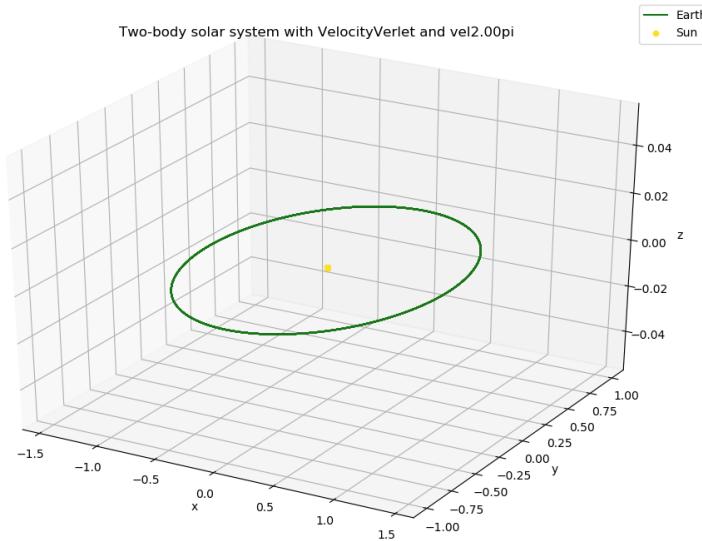


Figure 14: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 2.00 \pi$ m/s, made with Velocity Verlet.

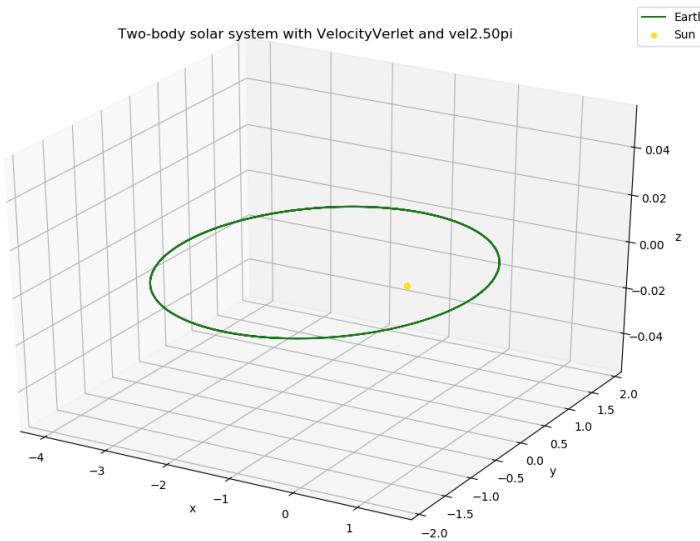


Figure 15: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 2.50 \pi$ m/s, made with Velocity Verlet.

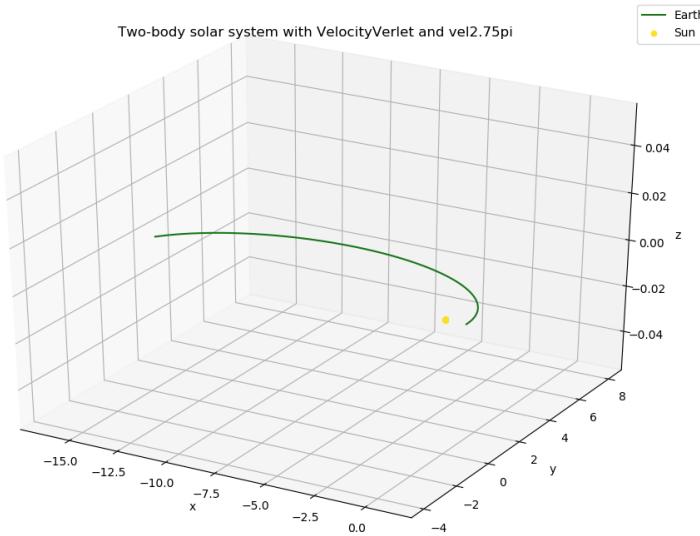


Figure 16: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 2.75 \pi$ m/s, made with Velocity Verlet.

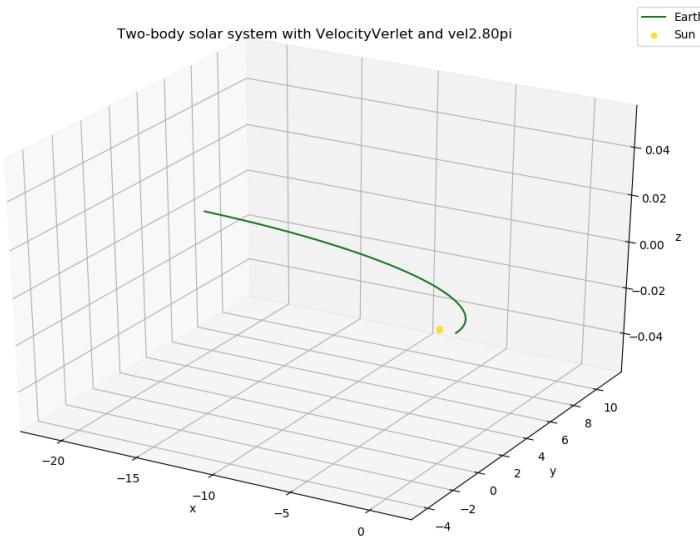


Figure 17: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 2.80 \pi$ m/s, made with Velocity Verlet.

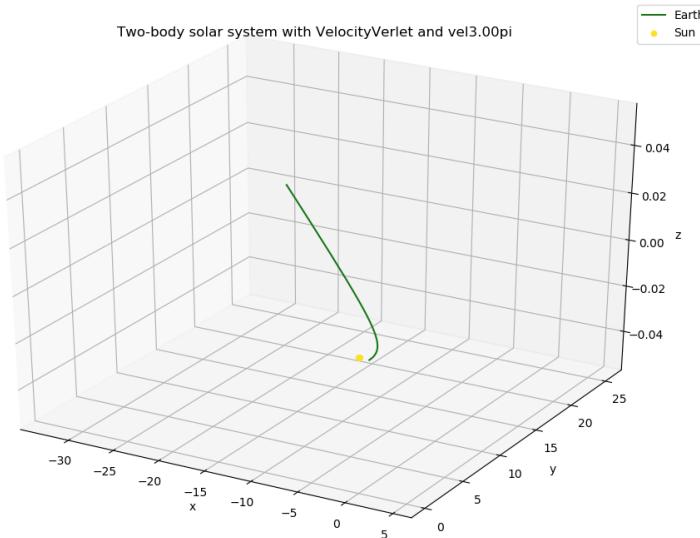


Figure 18: Plot of the Sun-Earth solar system with an escape velocity of $v_p = 3.00 \pi$ m/s, made with Velocity Verlet.

4.1.3 Testing different values of β

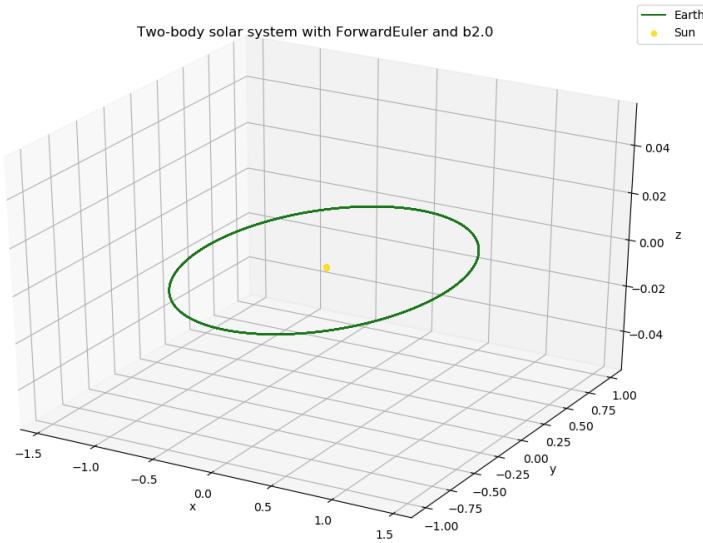


Figure 19: Plot of the Sun-Earth solar system with $\beta = 2.0$, made with Euler-Cromer.

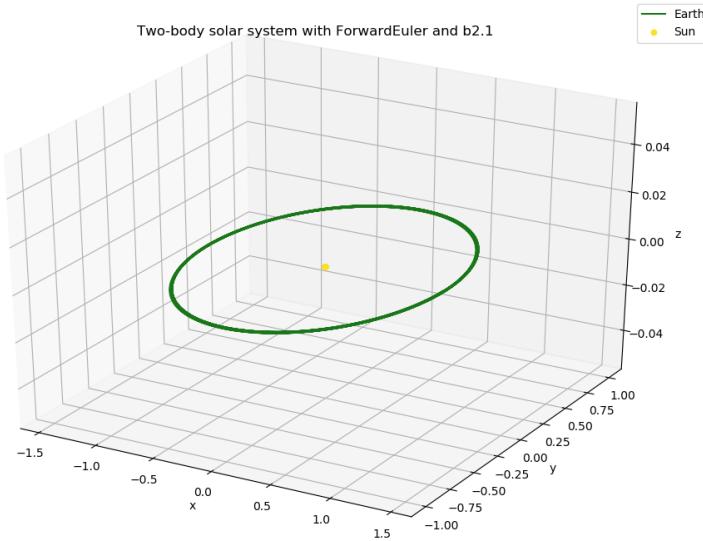


Figure 20: Plot of the Sun-Earth solar system with $\beta = 2.1$, made with Euler-Cromer.

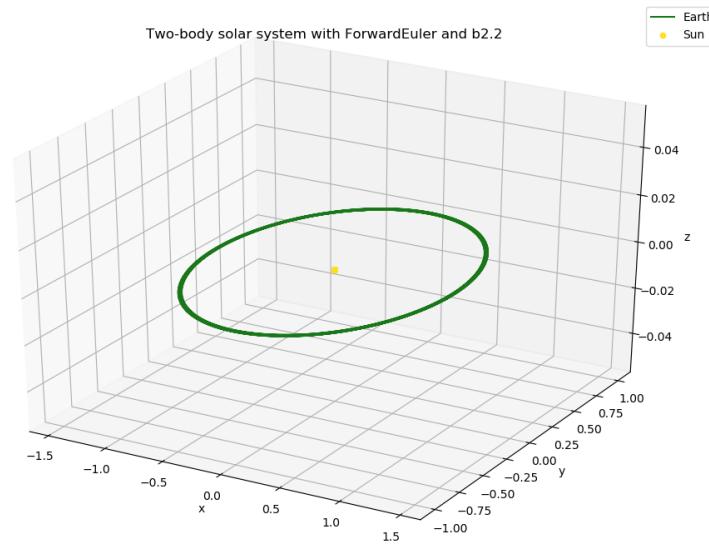


Figure 21: Plot of the Sun-Earth solar system with $\beta = 2.2$, made with Euler-Cromer.

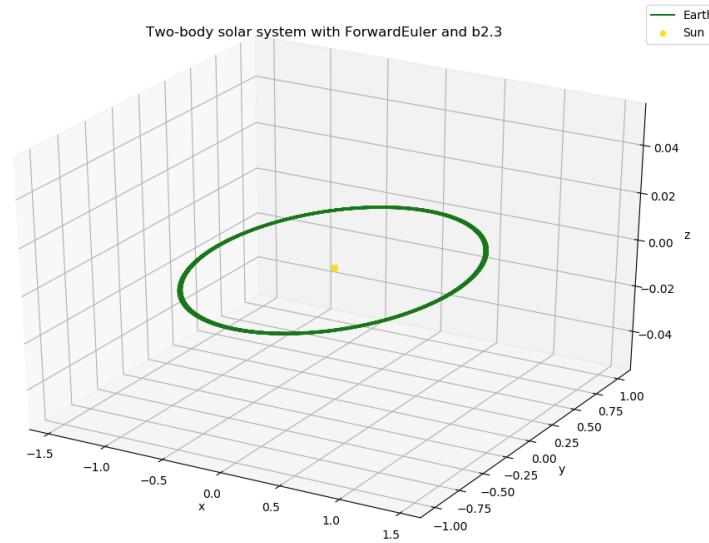


Figure 22: Plot of the Sun-Earth solar system with $\beta = 2.3$, made with Euler-Cromer.

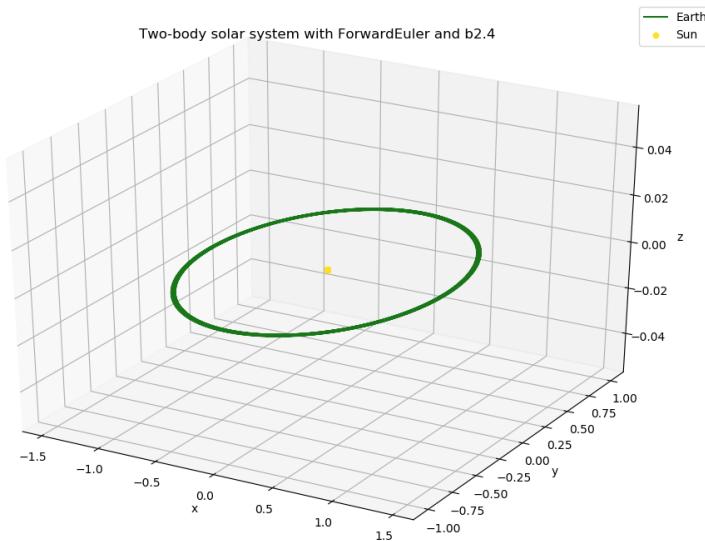


Figure 23: Plot of the Sun-Earth solar system with $\beta = 2.4$, made with Euler-Cromer.

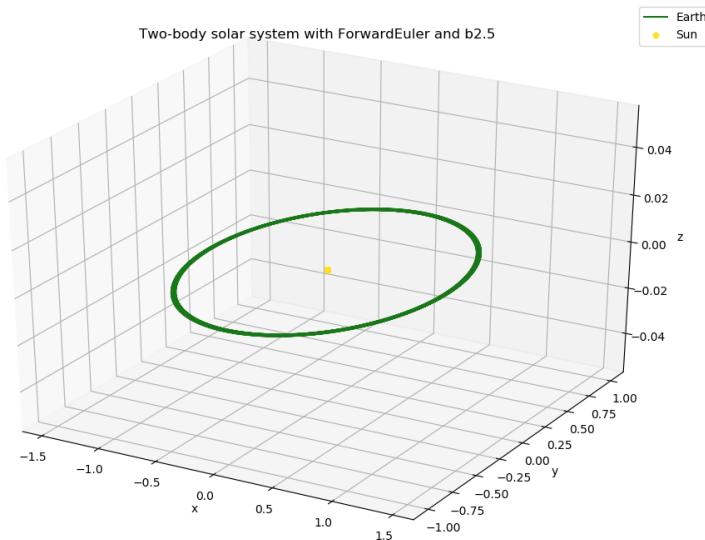


Figure 24: Plot of the Sun-Earth solar system with $\beta = 2.5$, made with Euler-Cromer.

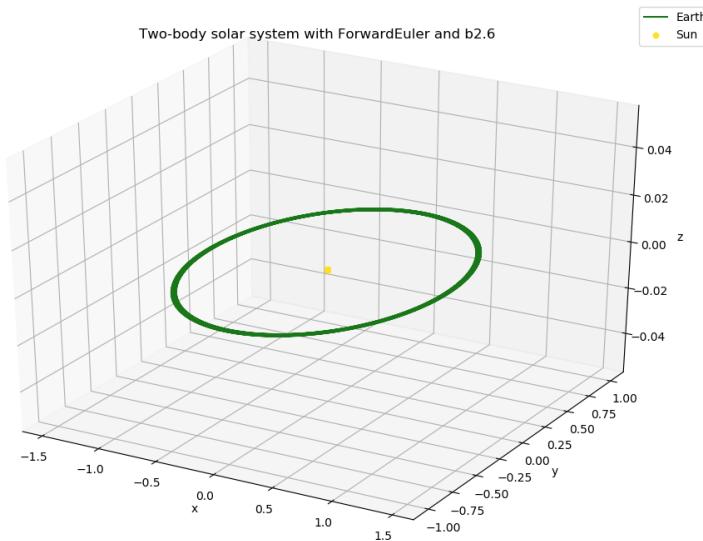


Figure 25: Plot of the Sun-Earth solar system with $\beta = 2.6$, made with Euler-Cromer.

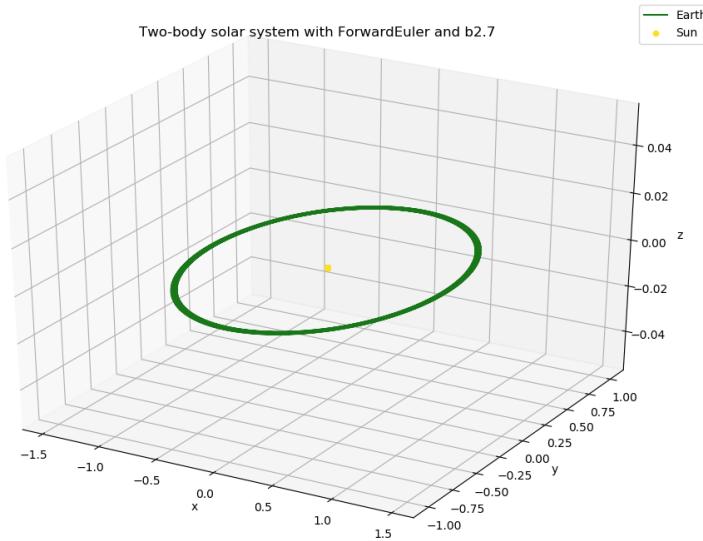


Figure 26: Plot of the Sun-Earth solar system with $\beta = 2.7$, made with Euler-Cromer.

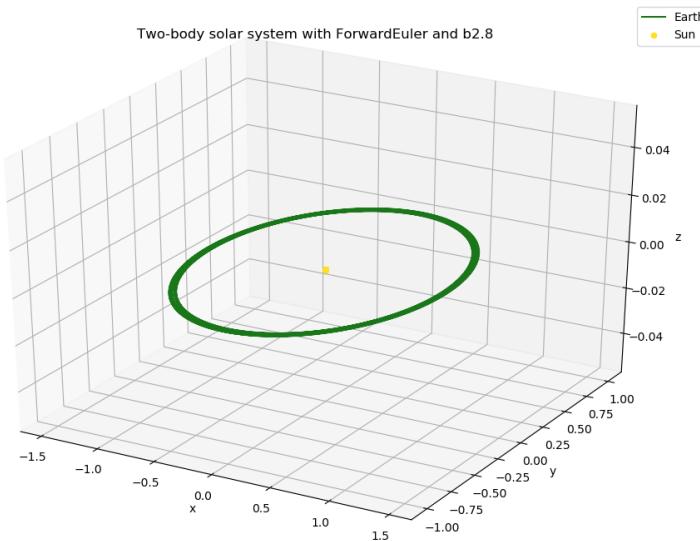


Figure 27: Plot of the Sun-Earth solar system with $\beta = 2.8$, made with Euler-Cromer.

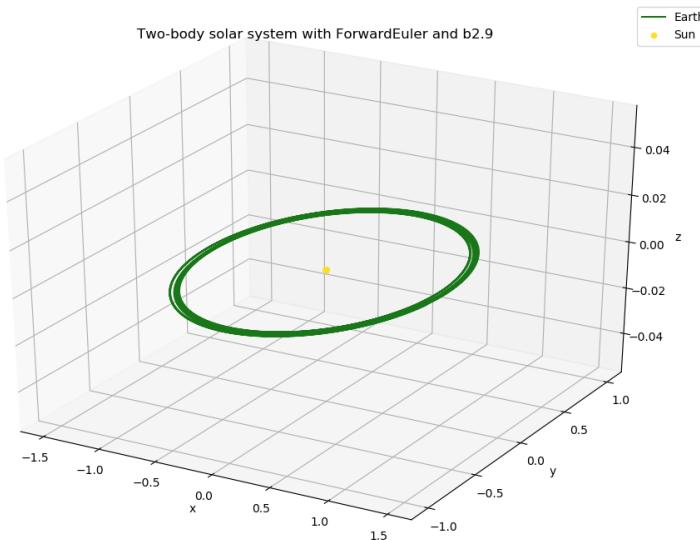


Figure 28: Plot of the Sun-Earth solar system with $\beta = 2.9$, made with Euler-Cromer.

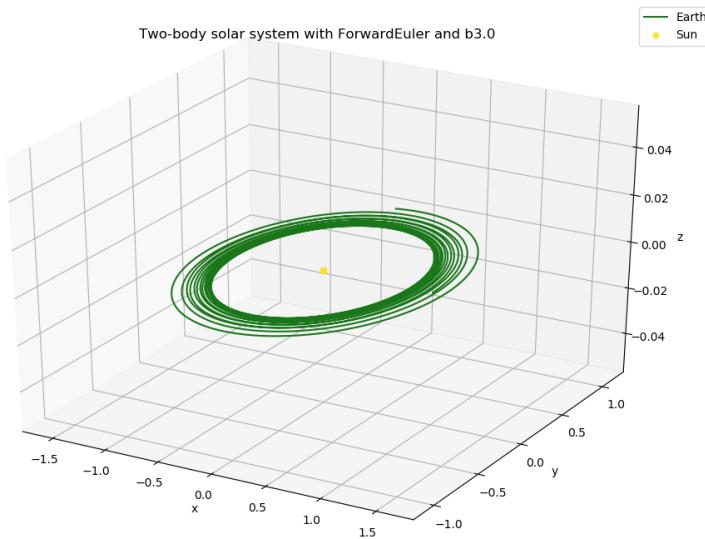


Figure 29: Plot of the Sun-Earth solar system with $\beta = 3.0$, made with Euler-Cromer.

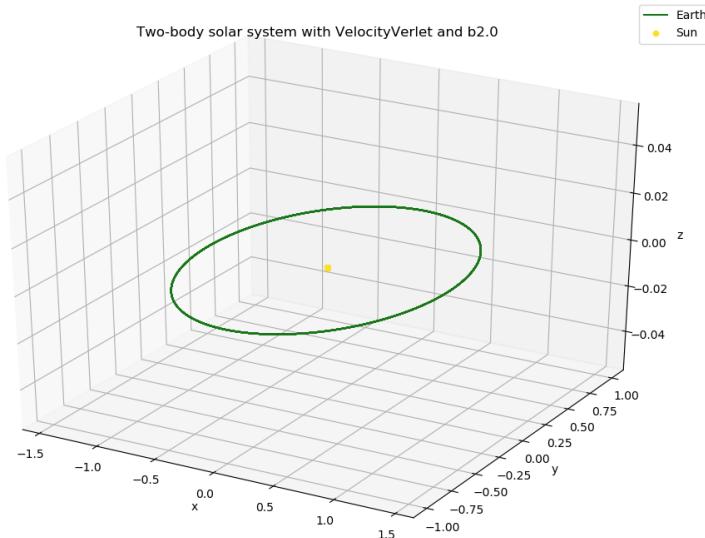


Figure 30: Plot of the Sun-Earth solar system with $\beta = 2.0$, made with Velocity Verlet.

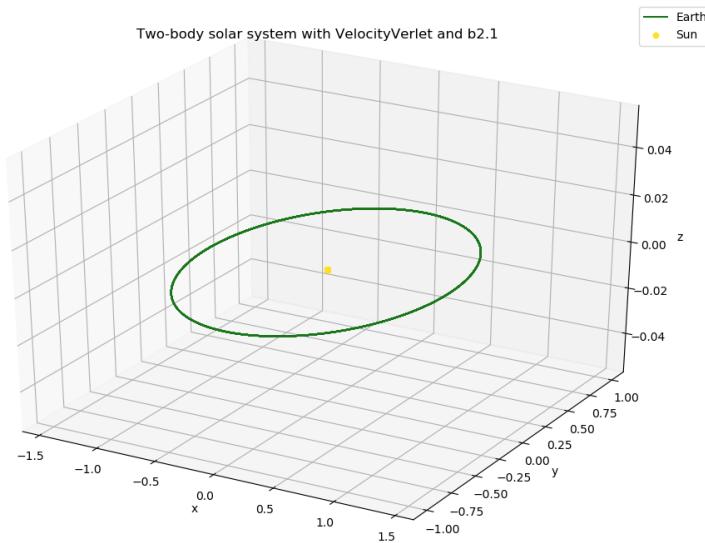


Figure 31: Plot of the Sun-Earth solar system with $\beta = 2.1$, made with Velocity Verlet.

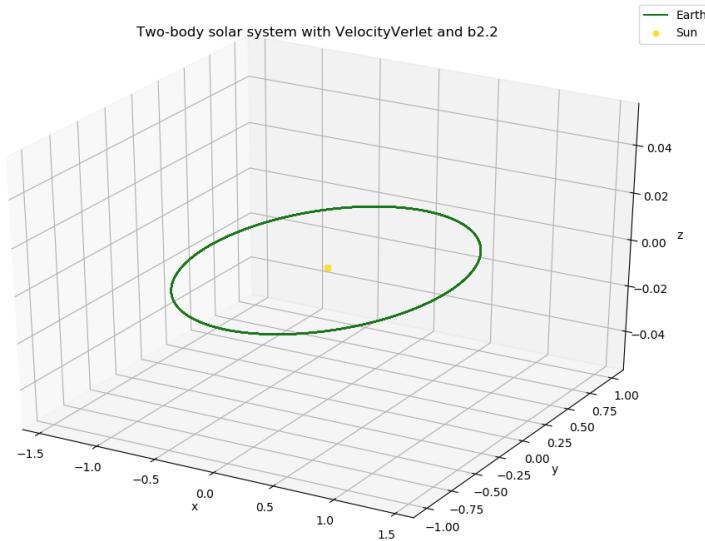


Figure 32: Plot of the Sun-Earth solar system with $\beta = 2.2$, made with Velocity Verlet.

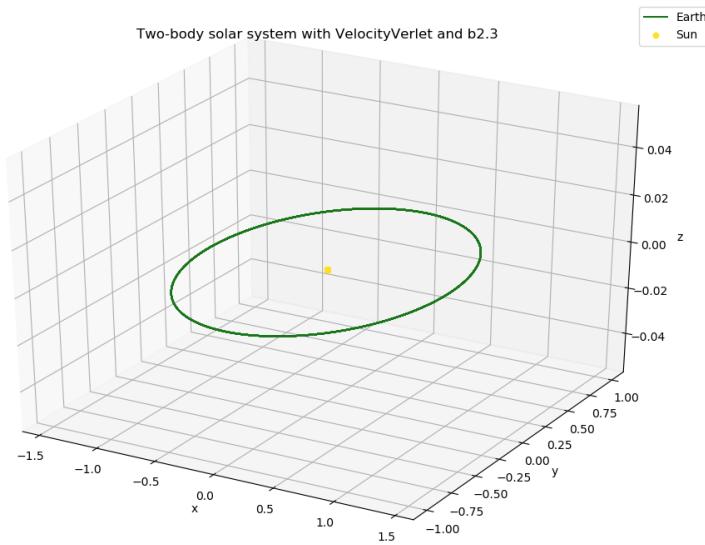


Figure 33: Plot of the Sun-Earth solar system with $\beta = 2.3$, made with Velocity Verlet.

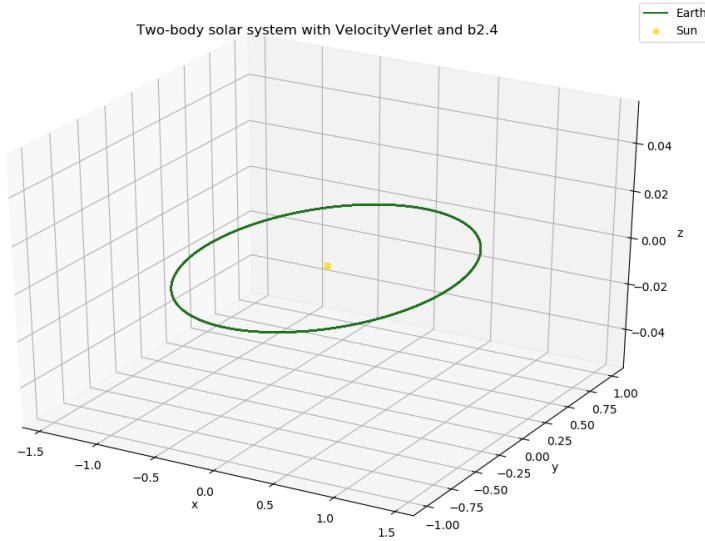


Figure 34: Plot of the Sun-Earth solar system with $\beta = 2.4$, made with Velocity Verlet.

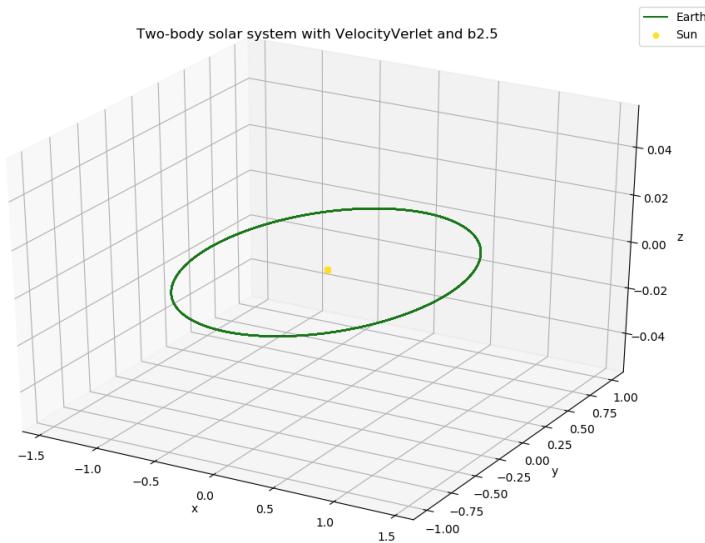


Figure 35: Plot of the Sun-Earth solar system with $\beta = 2.5$, made with Velocity Verlet.

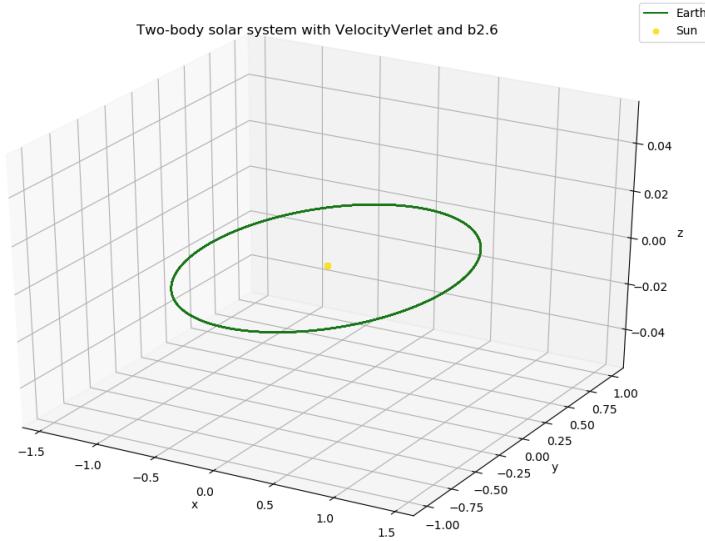


Figure 36: Plot of the Sun-Earth solar system with $\beta = 2.6$, made with Velocity Verlet.

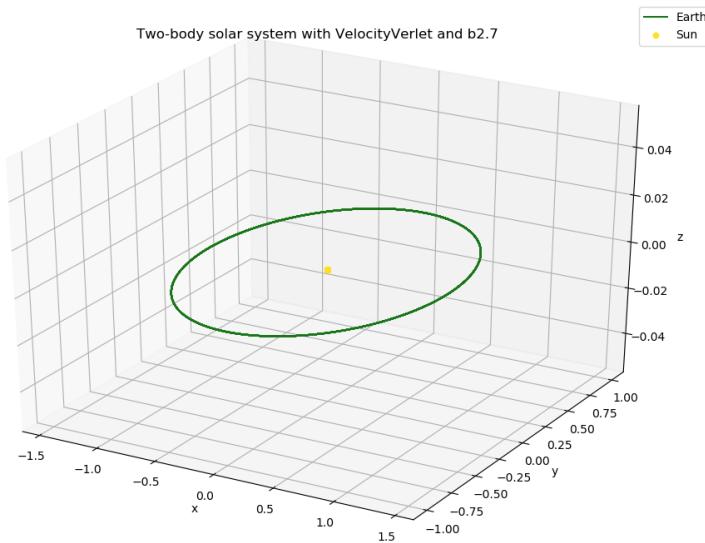


Figure 37: Plot of the Sun-Earth solar system with $\beta = 2.7$, made with Velocity Verlet.

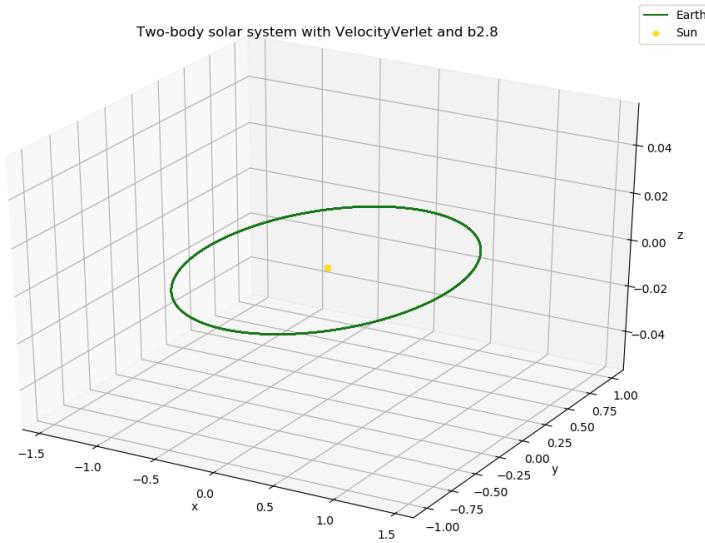


Figure 38: Plot of the Sun-Earth solar system with $\beta = 2.8$, made with Velocity Verlet.

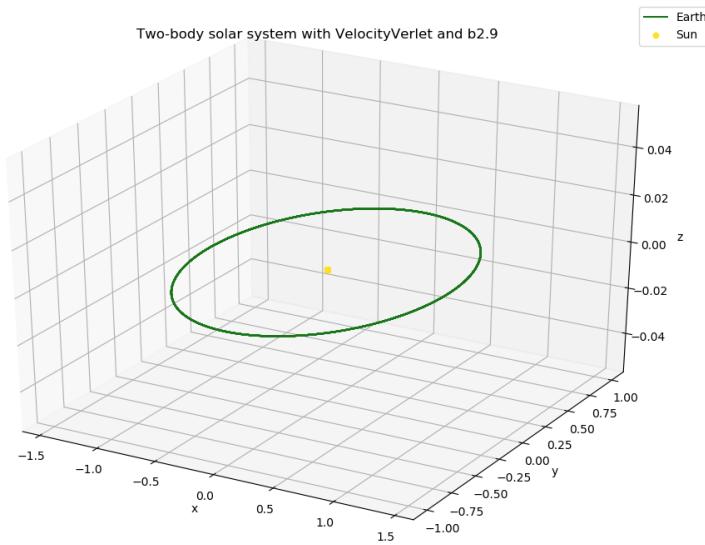


Figure 39: Plot of the Sun-Earth solar system with $\beta = 2.9$, made with Velocity Verlet.

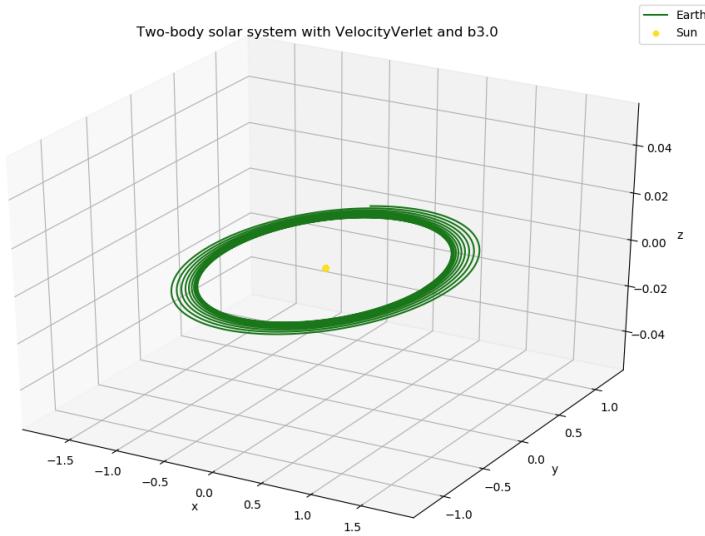


Figure 40: Plot of the Sun-Earth solar system with $\beta = 3.0$, made with Velocity Verlet.

4.2 Sun-Mercury solar system

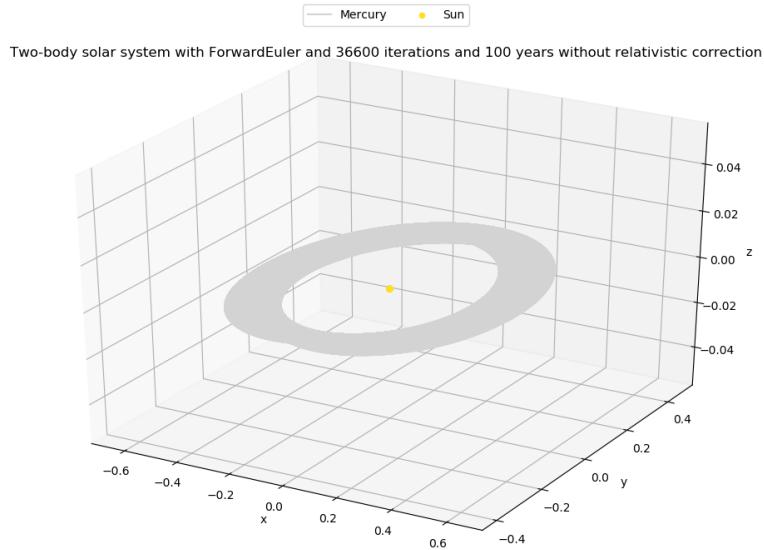


Figure 41: Plot of the Sun-Mercury solar system with 36600 iterations over 100 years without the relativistic correction, made with Euler-Cromer.

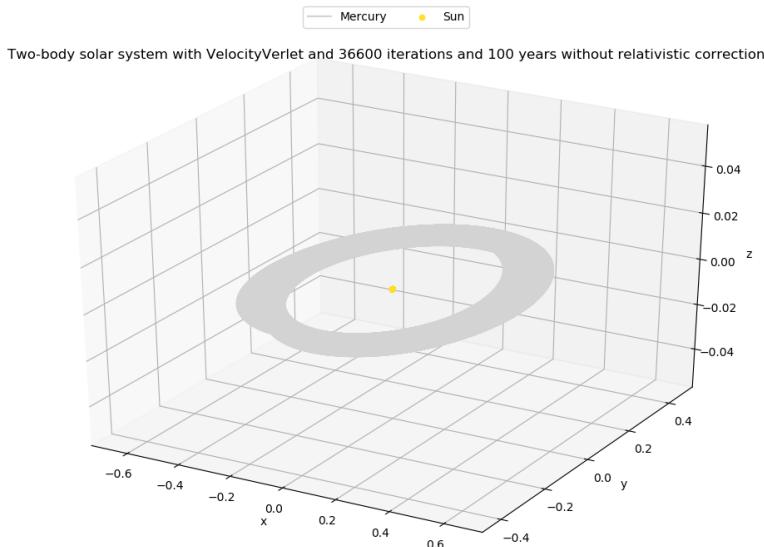


Figure 42: Plot of the Sun-Mercury solar system with 36600 iterations over 100 years without the relativistic correction, made with Velocity Verlet.

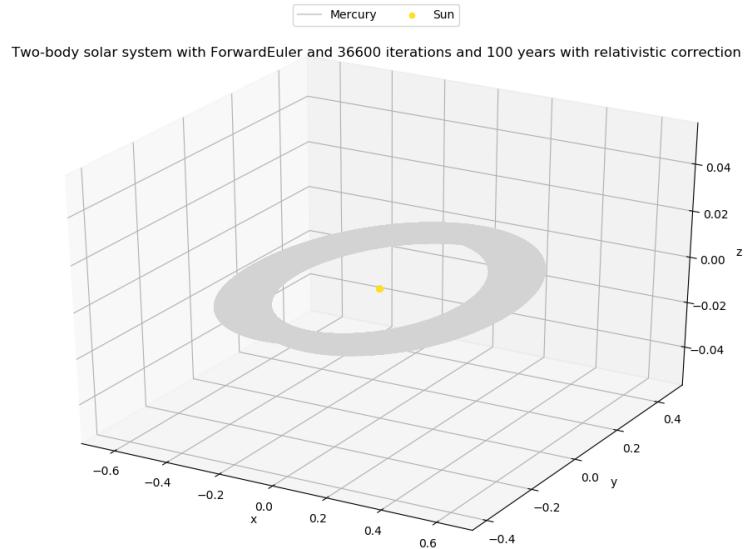


Figure 43: Plot of the Sun-Mercury solar system with 36600 iterations over 100 years with the relativistic correction, made with Euler-Cromer.

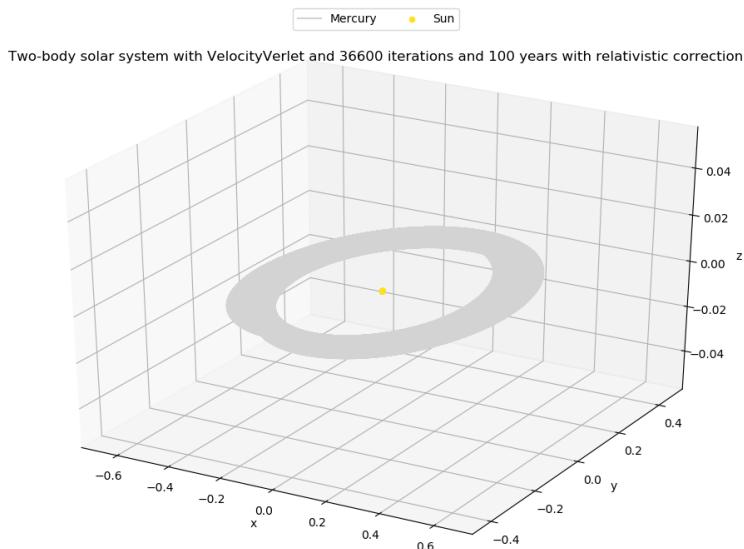


Figure 44: Plot of the Sun-Mercury solar system with 36600 iterations over 100 years with the relativistic correction, made with Velocity Verlet

4.3 Sun-Earth-Jupiter solar system

4.3.1 Massfactor 1

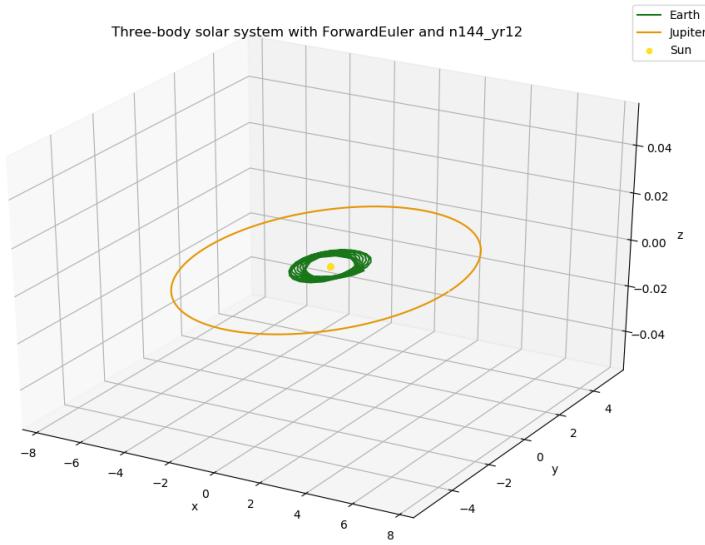


Figure 45: Plot of the Sun-Earth-Jupiter solar system with 144 iterations over 12 years and a massfactor of 1 for Jupiter, made with Euler-Cromer.

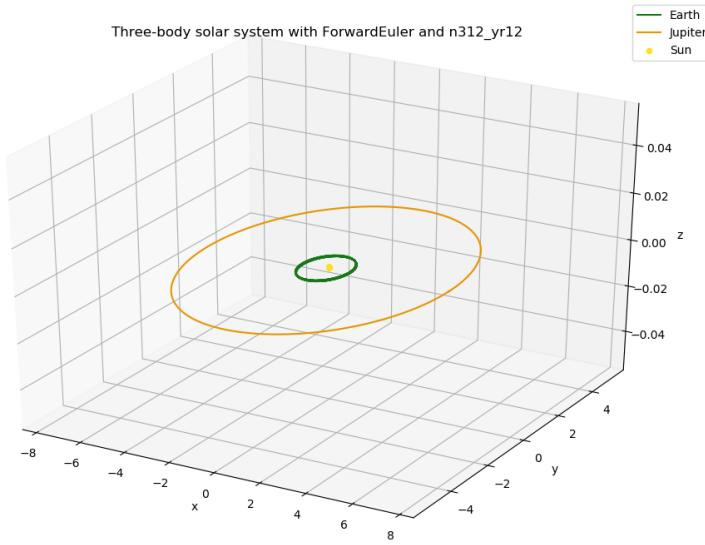


Figure 46: Plot of the Sun-Earth-Jupiter solar system with 312 iterations over 12 years and a massfactor of 1 for Jupiter, made with Euler-Cromer.

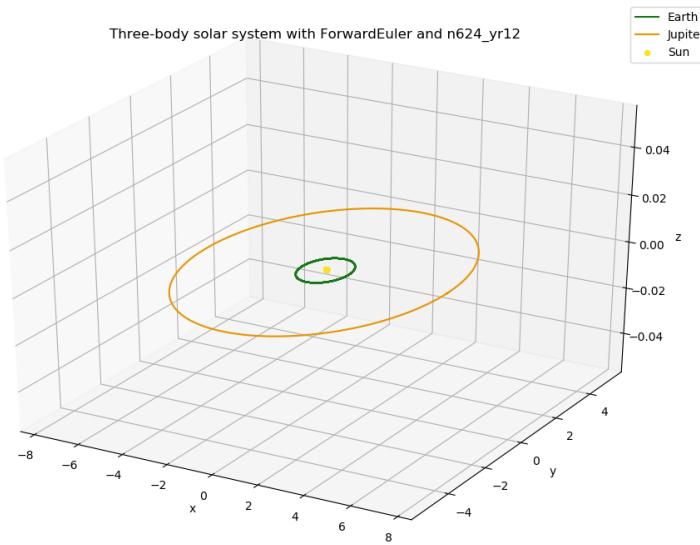


Figure 47: Plot of the Sun-Earth-Jupiter solar system with 624 iterations over 12 years and a massfactor of 1 for Jupiter, made with Euler-Cromer.

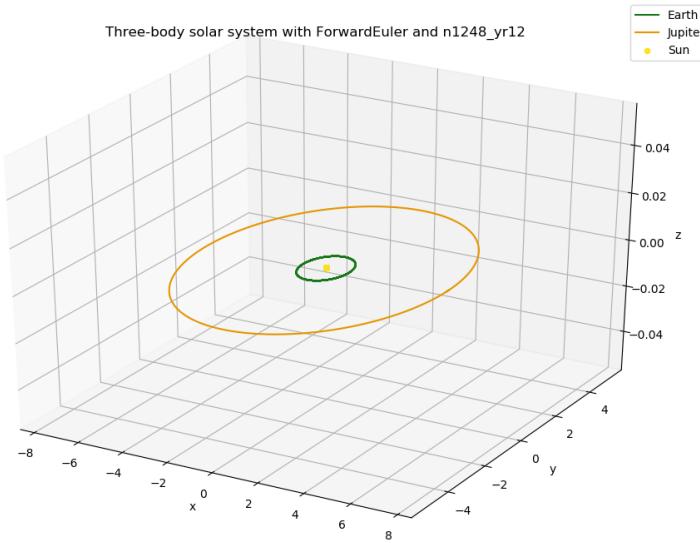


Figure 48: Plot of the Sun-Earth-Jupiter solar system with 1248 iterations over 12 years and a massfactor of 1 for Jupiter, made with Euler-Cromer.

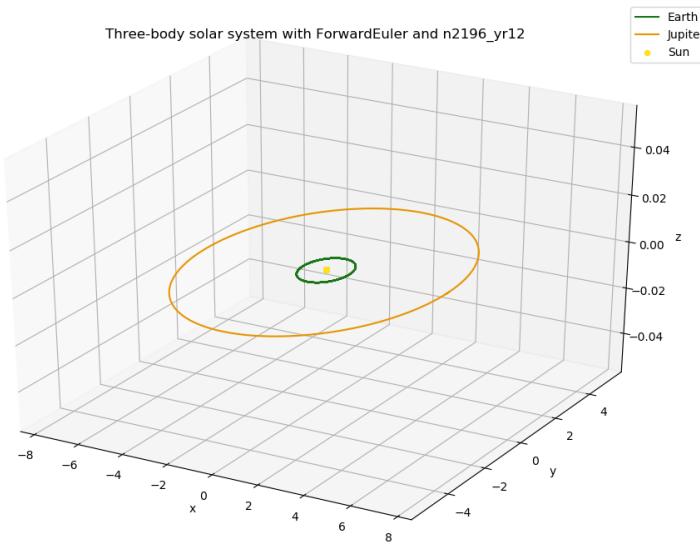


Figure 49: Plot of the Sun-Earth-Jupiter solar system with 2196 iterations over 12 years and a massfactor of 1 for Jupiter, made with Euler-Cromer.

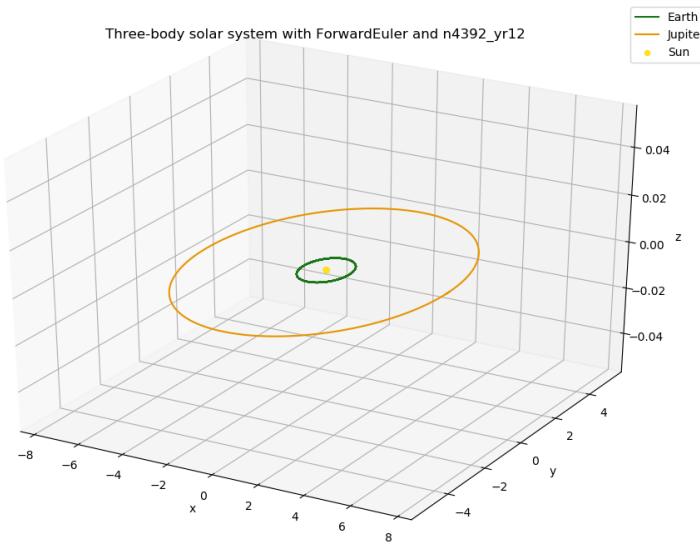


Figure 50: Plot of the Sun-Earth-Jupiter solar system with 4392 iterations over 12 years and a massfactor of 1 for Jupiter, made with Euler-Cromer.

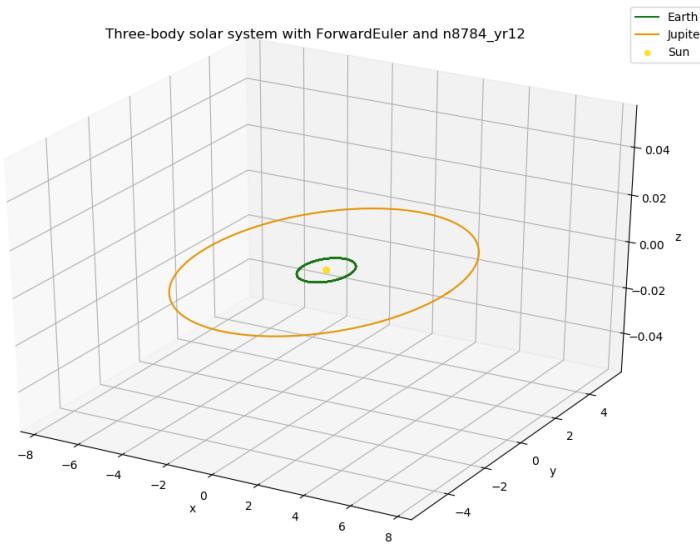


Figure 51: Plot of the Sun-Earth-Jupiter solar system with 8784 iterations over 12 years and a massfactor of 1 for Jupiter, made with Euler-Cromer.

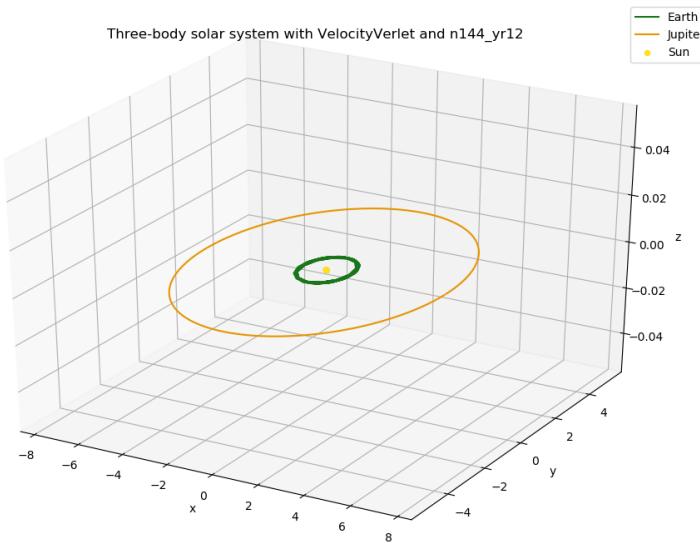


Figure 52: Plot of the Sun-Earth-Jupiter solar system with 144 iterations over 12 years and a massfactor of 1 for Jupiter, made with Velocity Verlet.

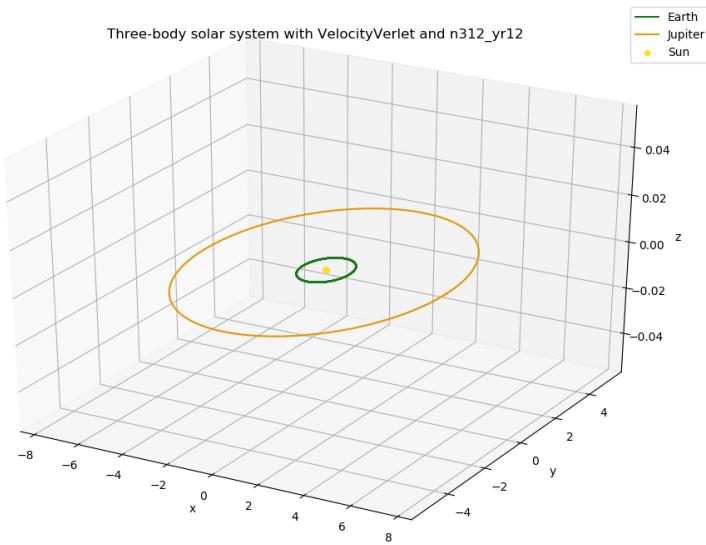


Figure 53: Plot of the Sun-Earth-Jupiter solar system with 312 iterations over 12 years and a massfactor of 1 for Jupiter, made with Velocity Verlet.

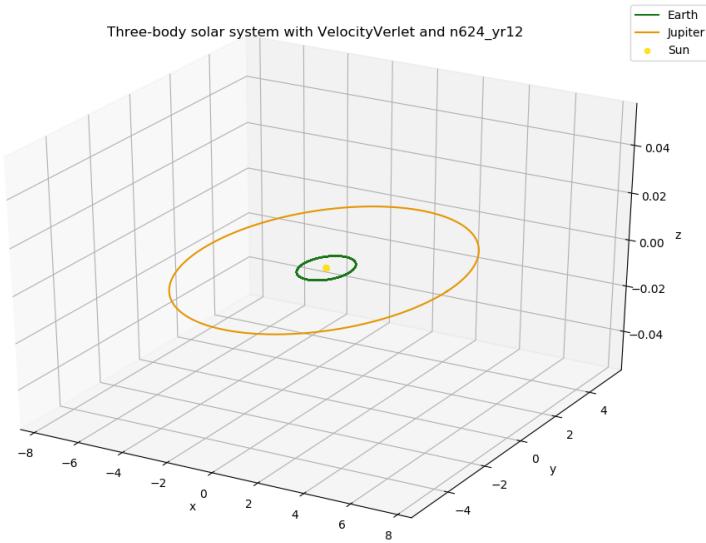


Figure 54: Plot of the Sun-Earth-Jupiter solar system with 624 iterations over 12 years and a massfactor of 1 for Jupiter, made with Velocity Verlet.

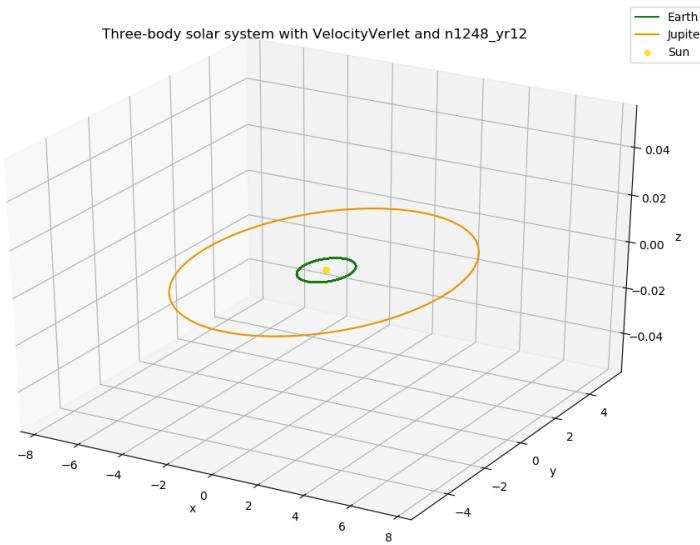


Figure 55: Plot of the Sun-Earth-Jupiter solar system with 1248 iterations over 12 years and a massfactor of 1 for Jupiter, made with Velocity Verlet.

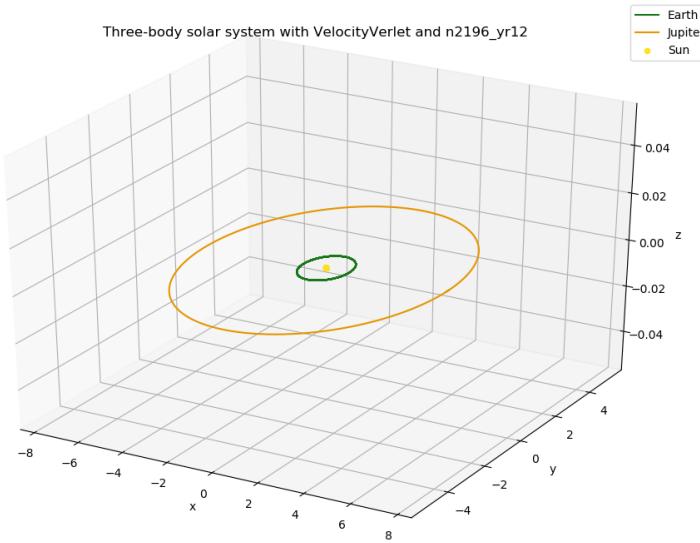


Figure 56: Plot of the Sun-Earth-Jupiter solar system with 2196 iterations over 12 years and a massfactor of 1 for Jupiter, made with Velocity Verlet.

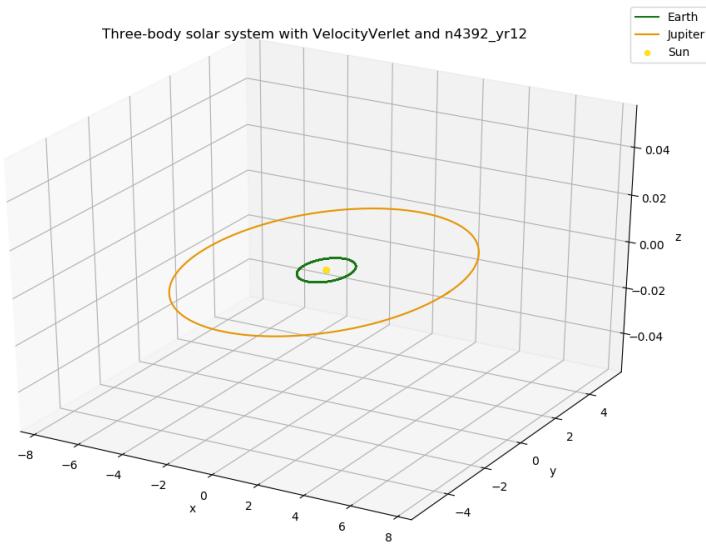


Figure 57: Plot of the Sun-Earth-Jupiter solar system with 4392 iterations over 12 years and a massfactor of 1 for Jupiter, made with Velocity Verlet.

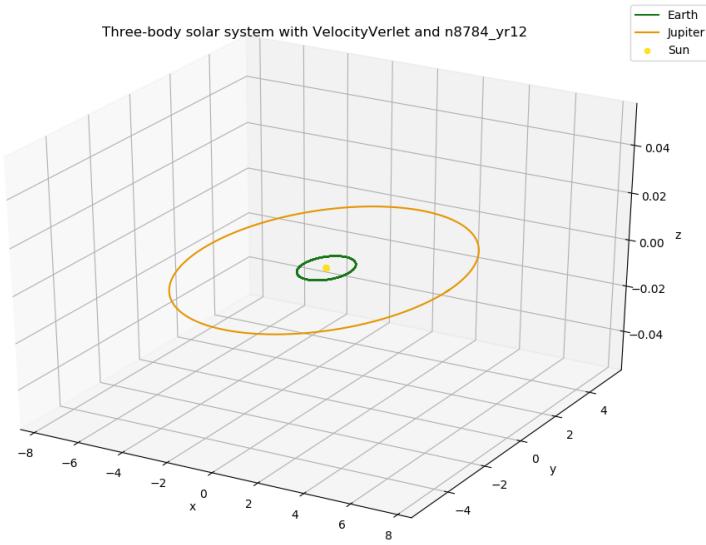


Figure 58: Plot of the Sun-Earth-Jupiter solar system with 8784 iterations over 12 years and a massfactor of 1 for Jupiter, made with Velocity Verlet.

4.3.2 Massfactor 10

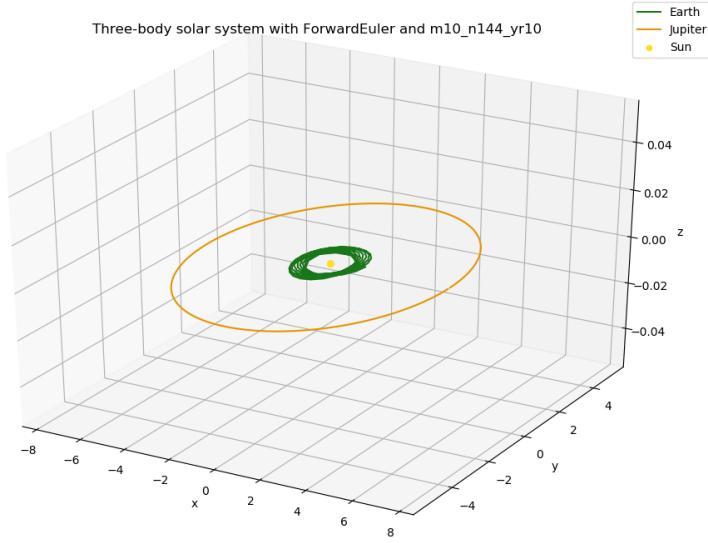


Figure 59: Plot of the Sun-Earth-Jupiter solar system with 144 iterations over 10 years and a massfactor of 10 for Jupiter, made with Euler-Cromer.

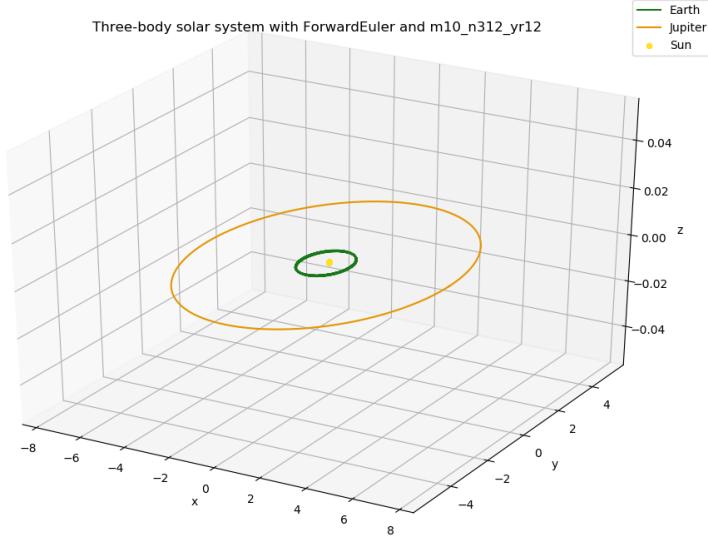


Figure 60: Plot of the Sun-Earth-Jupiter solar system with 312 iterations over 12 years and a massfactor of 10 for Jupiter, made with Euler-Cromer.

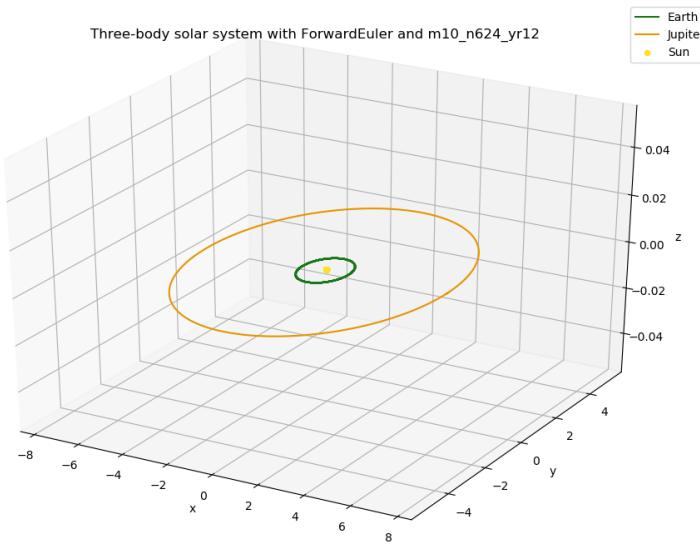


Figure 61: Plot of the Sun-Earth-Jupiter solar system with 624 iterations over 12 years and a massfactor of 10 for Jupiter, made with Euler-Cromer.

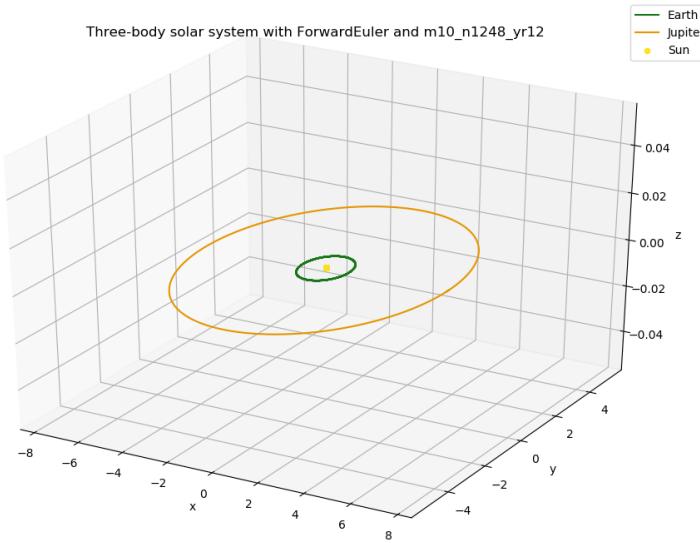


Figure 62: Plot of the Sun-Earth-Jupiter solar system with 1248 iterations over 12 years and a massfactor of 10 for Jupiter, made with Euler-Cromer.

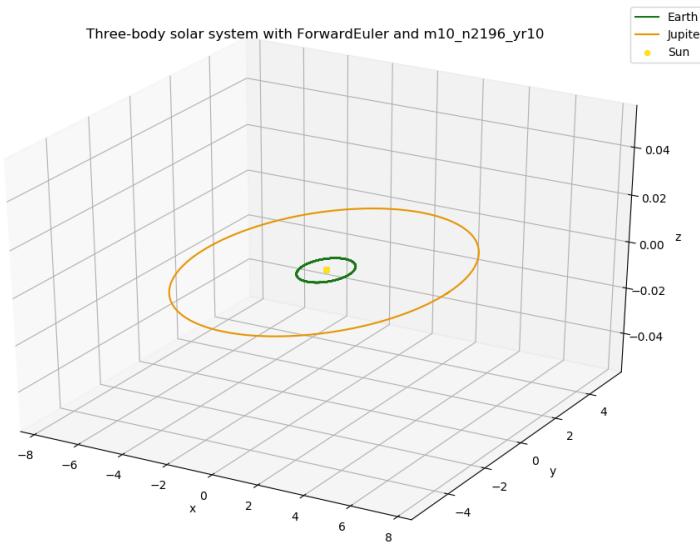


Figure 63: Plot of the Sun-Earth-Jupiter solar system with 2196 iterations over 10 years and a massfactor of 10 for Jupiter, made with Euler-Cromer.

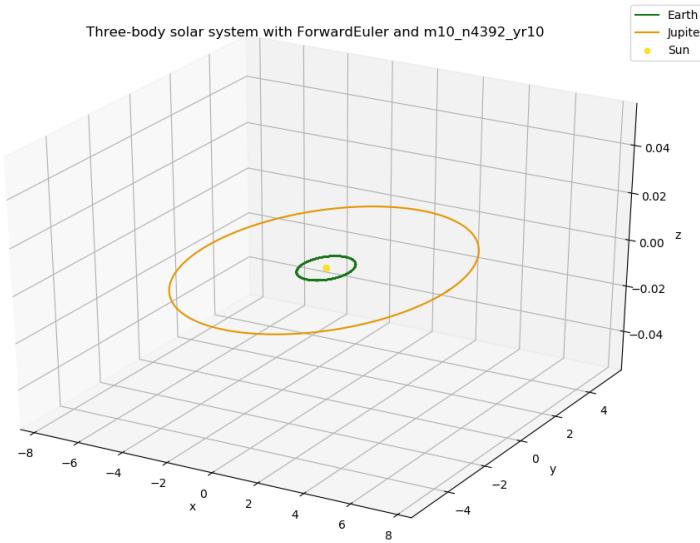


Figure 64: Plot of the Sun-Earth-Jupiter solar system with 4392 iterations over 10 years and a massfactor of 10 for Jupiter, made with Euler-Cromer.

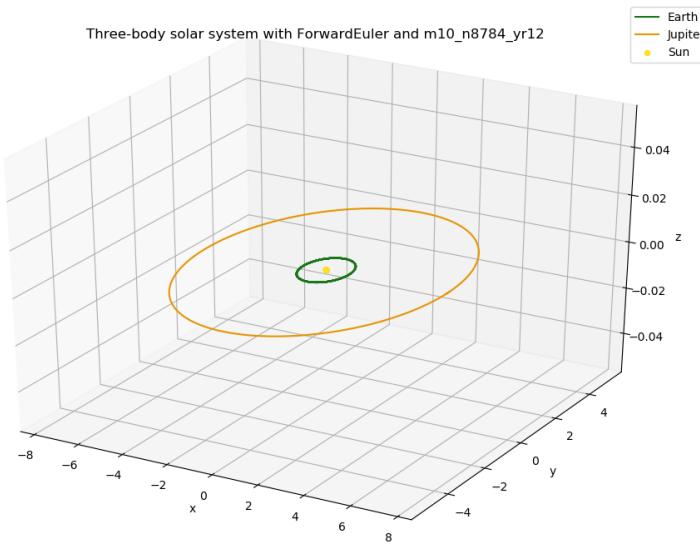


Figure 65: Plot of the Sun-Earth-Jupiter solar system with 8784 iterations over 12 years and a massfactor of 10 for Jupiter, made with Euler-Cromer.

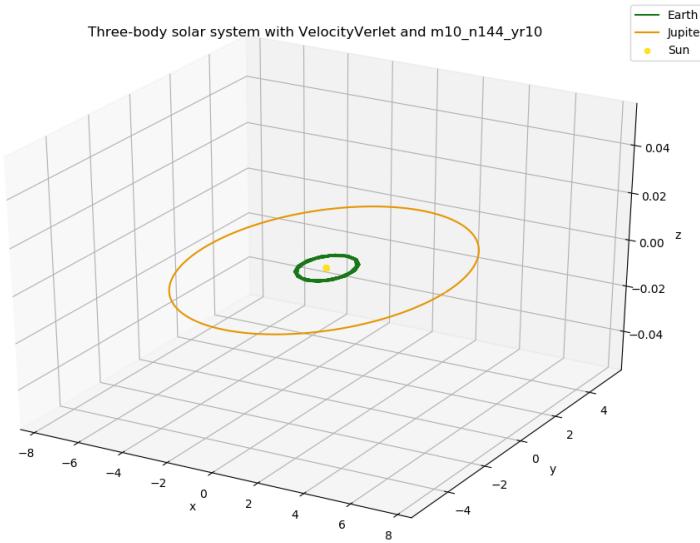


Figure 66: Plot of the Sun-Earth-Jupiter solar system with 144 iterations over 10 years and a massfactor of 10 for Jupiter, made with Velocity Verlet.

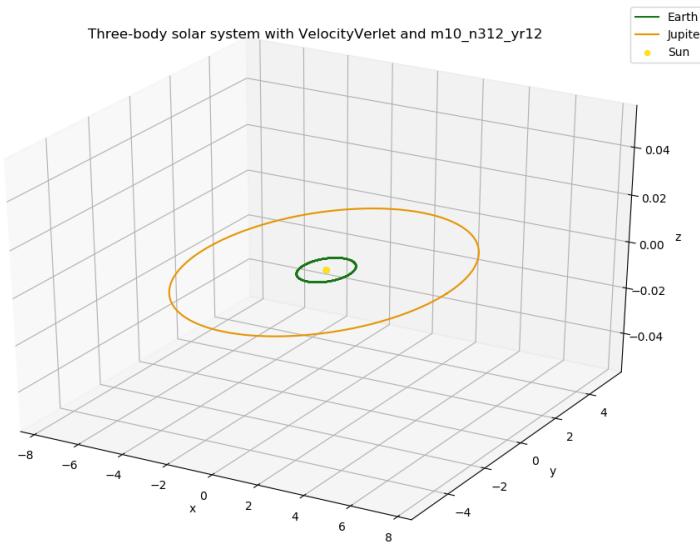


Figure 67: Plot of the Sun-Earth-Jupiter solar system with 312 iterations over 12 years and a massfactor of 10 for Jupiter, made with Velocity Verlet.

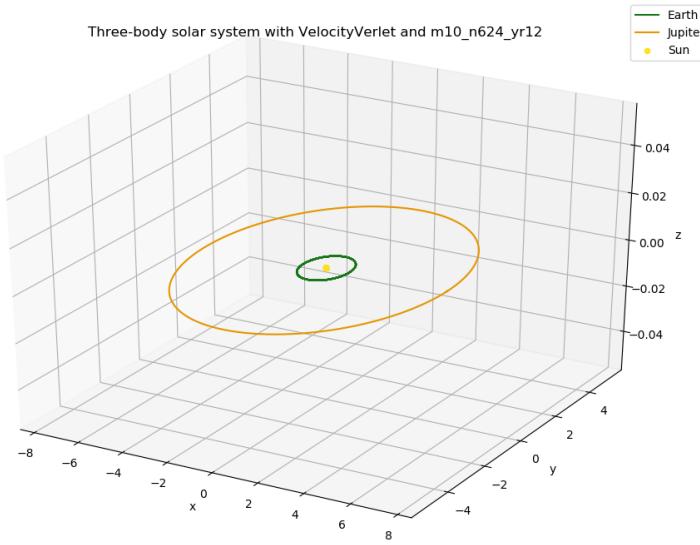


Figure 68: Plot of the Sun-Earth-Jupiter solar system with 624 iterations over 12 years and a massfactor of 10 for Jupiter, made with Velocity Verlet.

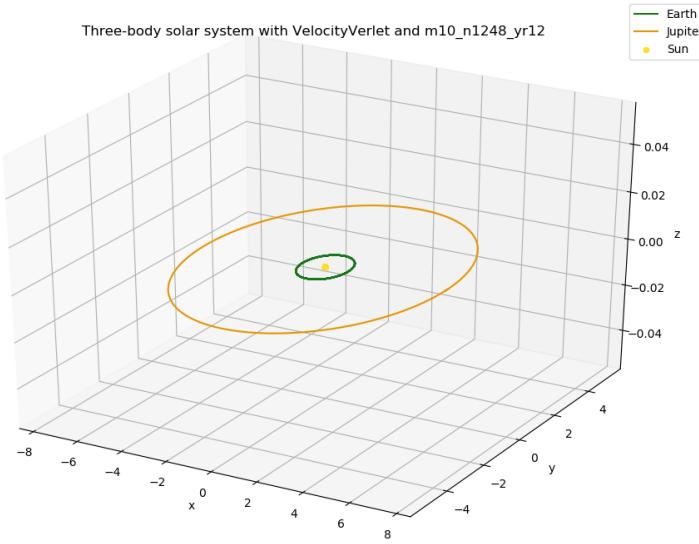


Figure 69: Plot of the Sun-Earth-Jupiter solar system with 1248 iterations over 12 years and a massfactor of 10 for Jupiter, made with Velocity Verlet.

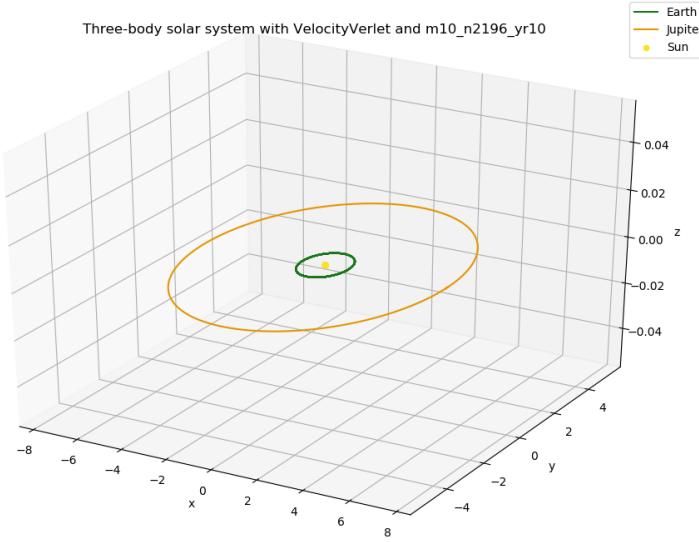


Figure 70: Plot of the Sun-Earth-Jupiter solar system with 2196 iterations over 10 years and a massfactor of 10 for Jupiter, made with Velocity Verlet.

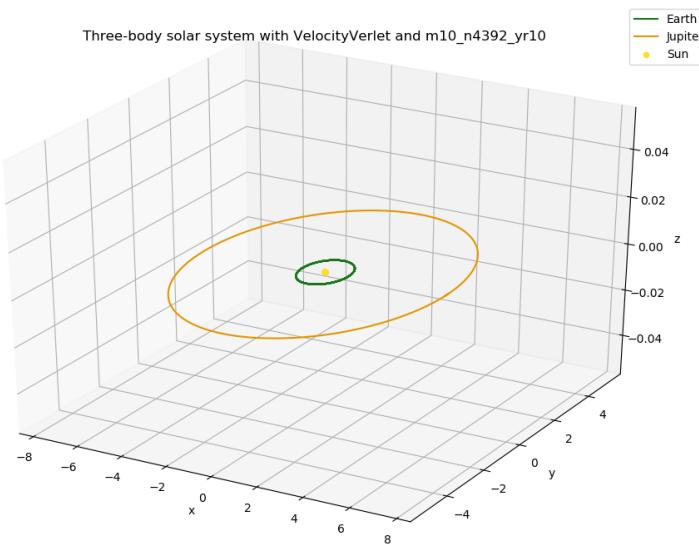


Figure 71: Plot of the Sun-Earth-Jupiter solar system with 4392 iterations over 10 years and a massfactor of 10 for Jupiter, made with Velocity Verlet.

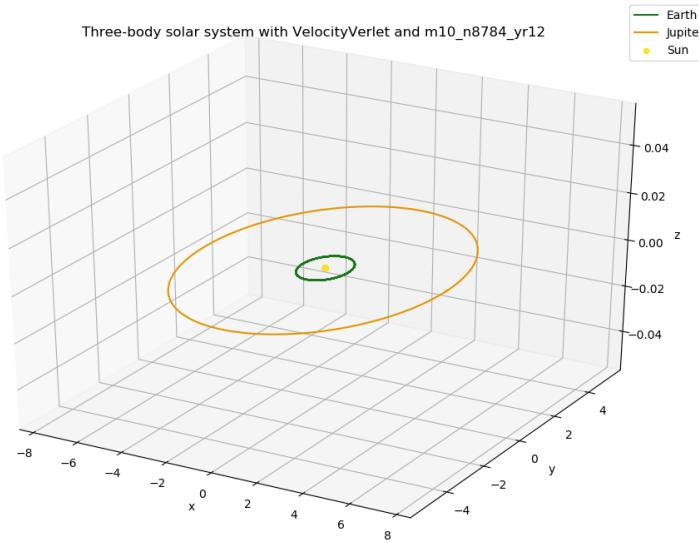


Figure 72: Plot of the Sun-Earth-Jupiter solar system with 8784 iterations over 12 years and a massfactor of 10 for Jupiter, made with Velocity Verlet.

4.3.3 Massfactor 1000

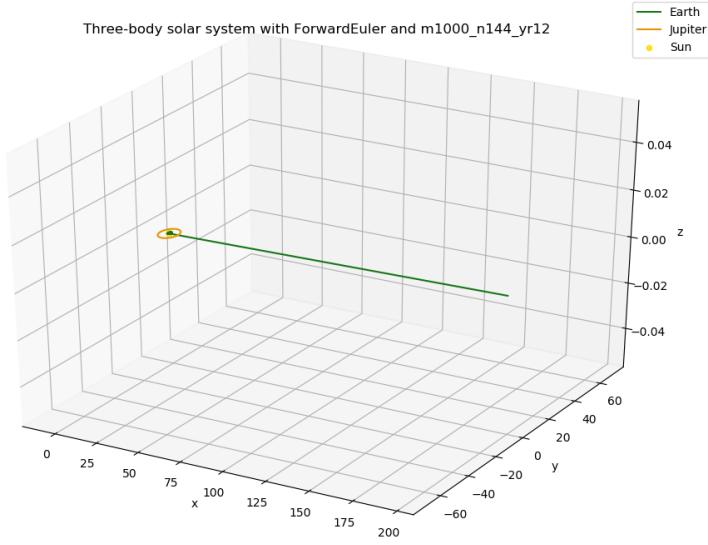


Figure 73: Plot of the Sun-Earth-Jupiter solar system with 144 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Euler-Cromer.

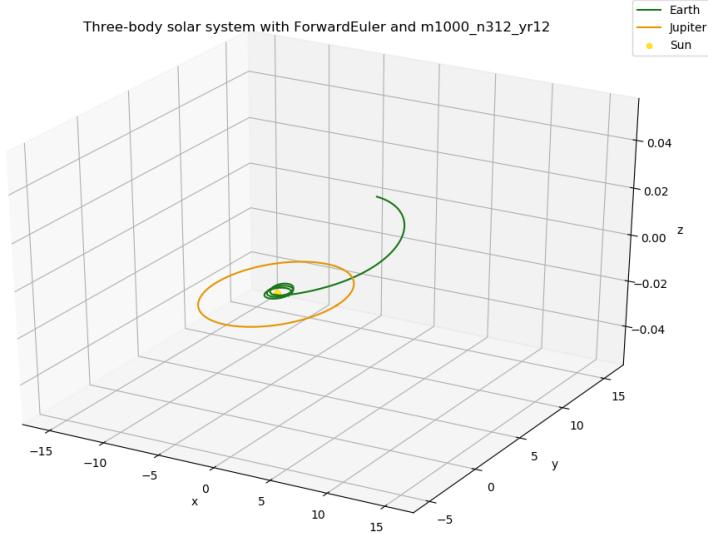


Figure 74: Plot of the Sun-Earth-Jupiter solar system with 312 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Euler-Cromer.

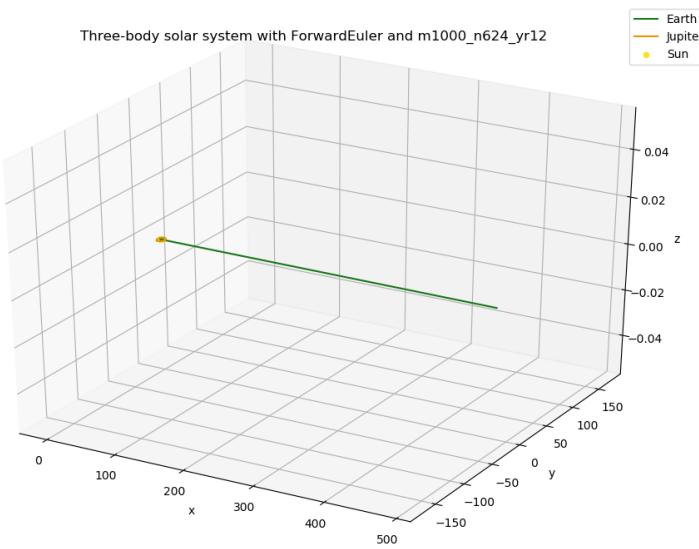


Figure 75: Plot of the Sun-Earth-Jupiter solar system with 624 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Euler-Cromer.

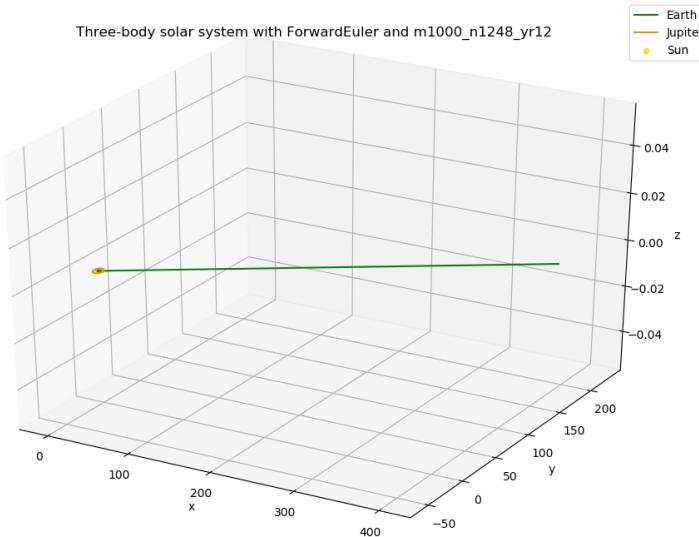


Figure 76: Plot of the Sun-Earth-Jupiter solar system with 1248 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Euler-Cromer.

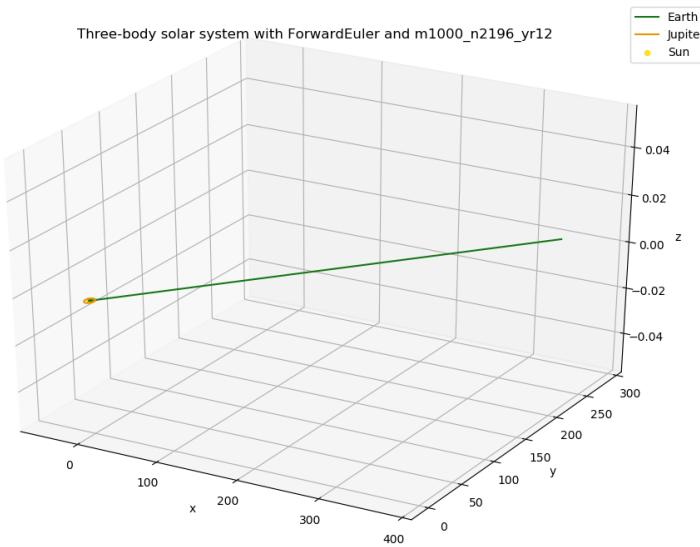


Figure 77: Plot of the Sun-Earth-Jupiter solar system with 2196 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Euler-Cromer.

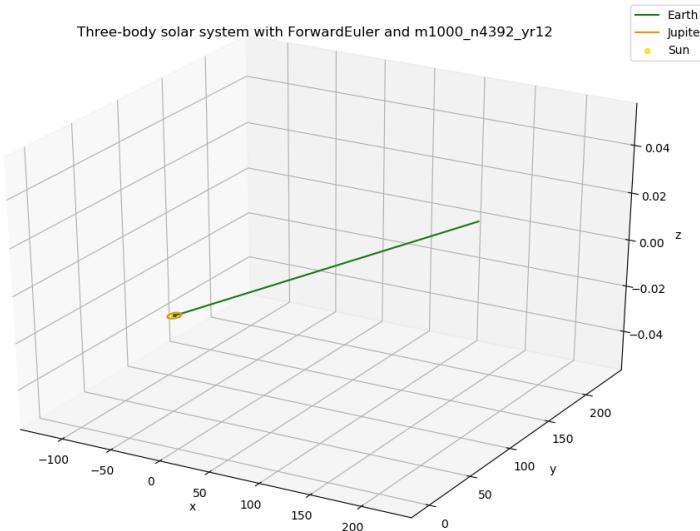


Figure 78: Plot of the Sun-Earth-Jupiter solar system with 4392 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Euler-Cromer.

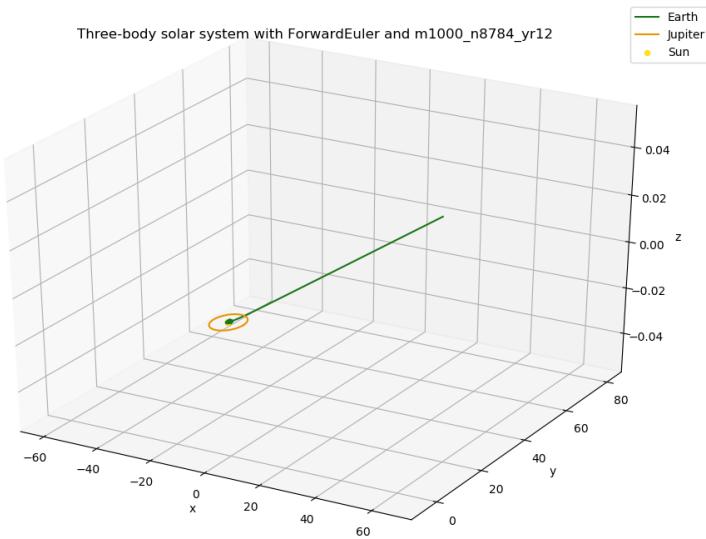


Figure 79: Plot of the Sun-Earth-Jupiter solar system with 8784 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Euler-Cromer.

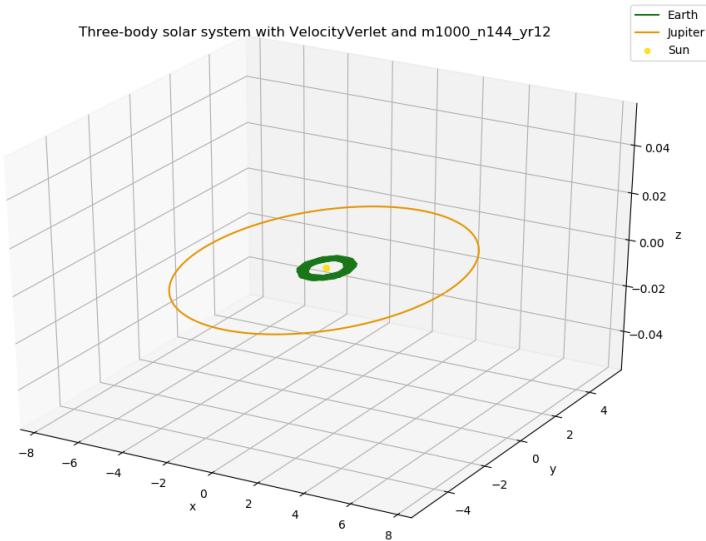


Figure 80: Plot of the Sun-Earth-Jupiter solar system with 144 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Velocity Verlet.

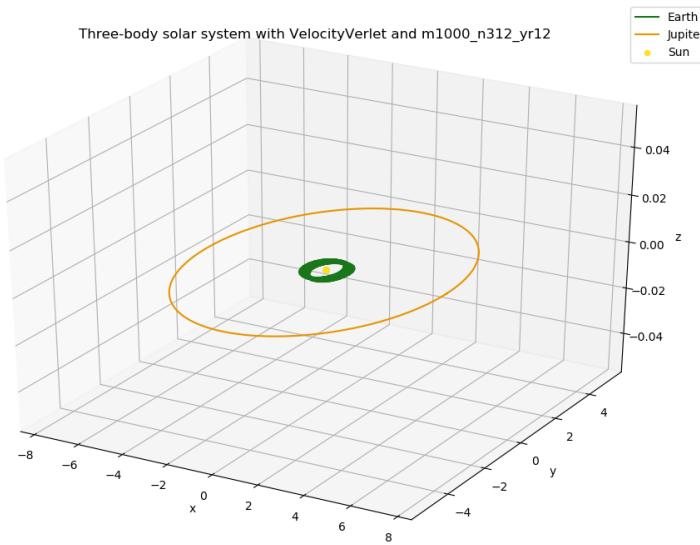


Figure 81: Plot of the Sun-Earth-Jupiter solar system with 312 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Velocity Verlet.

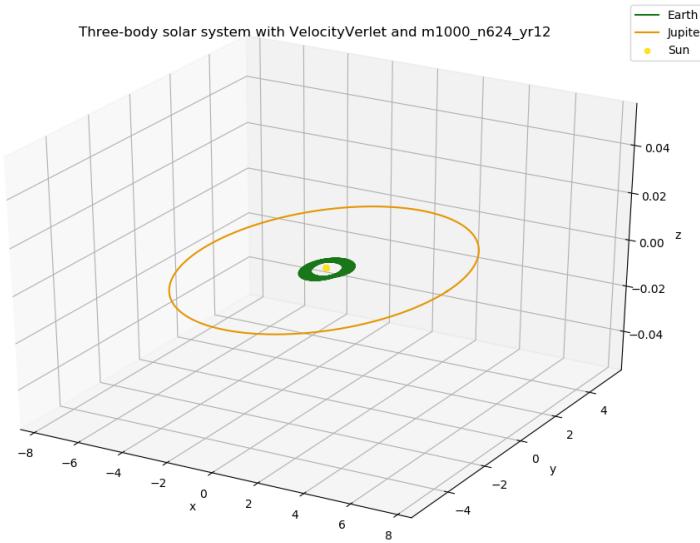


Figure 82: Plot of the Sun-Earth-Jupiter solar system with 624 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Velocity Verlet.

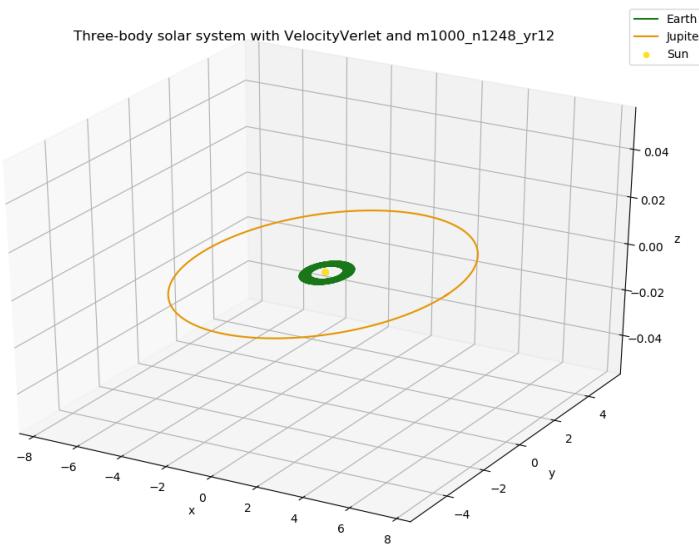


Figure 83: Plot of the Sun-Earth-Jupiter solar system with 1248 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Velocity Verlet.

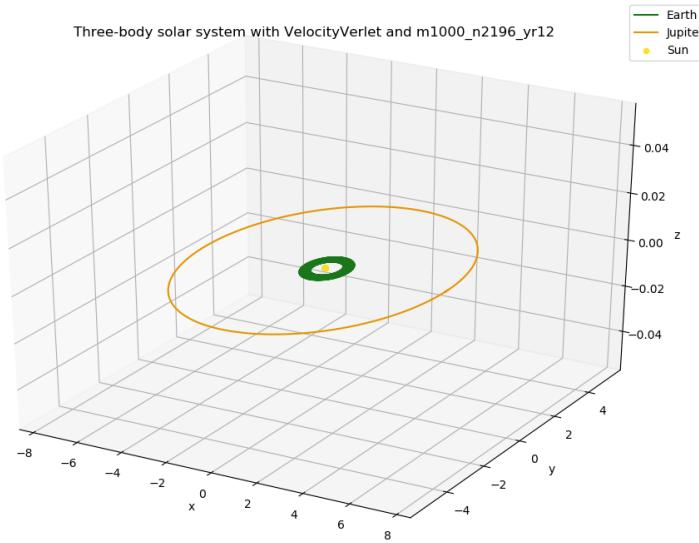


Figure 84: Plot of the Sun-Earth-Jupiter solar system with 2196 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Velocity Verlet.

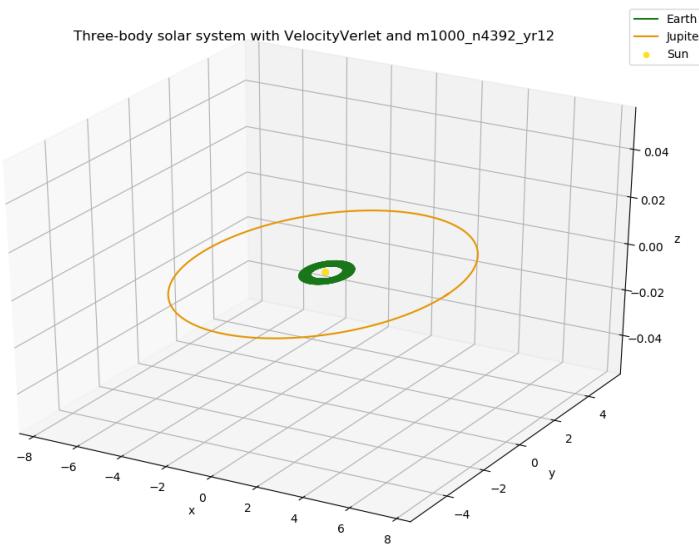


Figure 85: Plot of the Sun-Earth-Jupiter solar system with 4392 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Velocity Verlet.

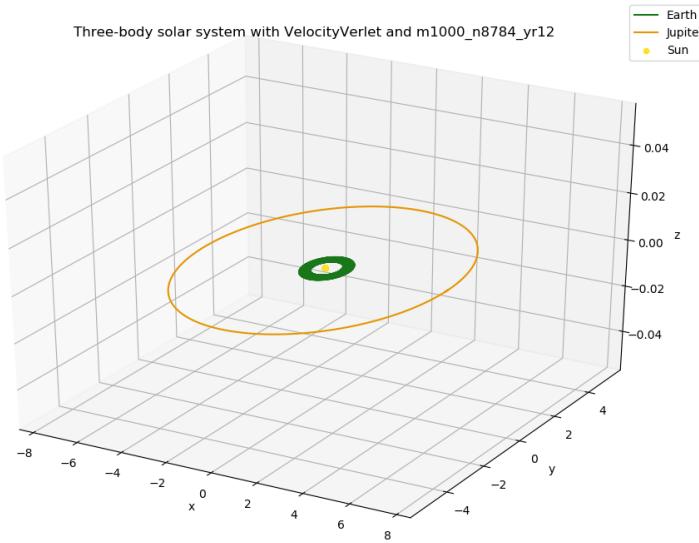


Figure 86: Plot of the Sun-Earth-Jupiter solar system with 8784 iterations over 12 years and a massfactor of 1000 for Jupiter, made with Velocity Verlet.

4.4 Dynamic sun

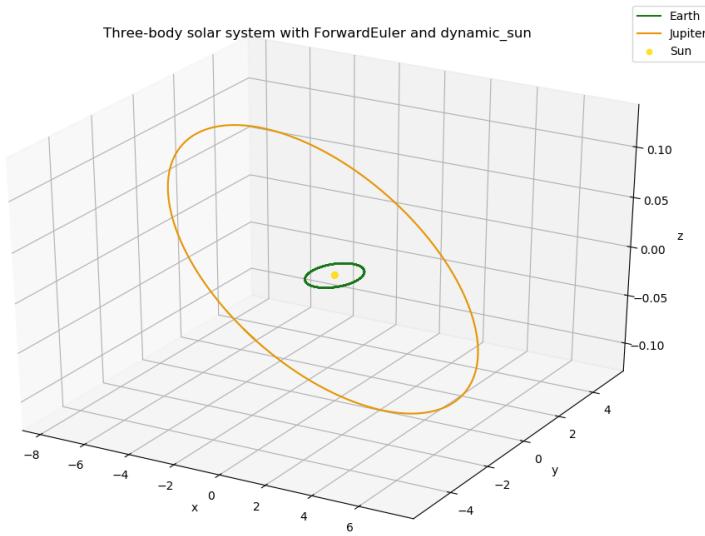


Figure 87: Plot of the Sun-Earth-Jupiter solar system with a dynamic sun, made with Euler-Cromer.

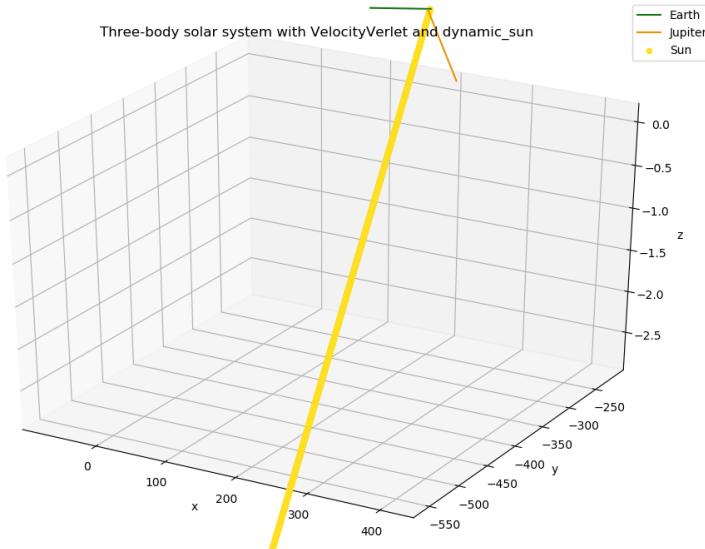


Figure 88: Plot of the Sun-Earth-Jupiter solar system with a dynamic sun, made with Velocity Verlet.

4.5 Ten-body solar system

4.5.1 Static sun

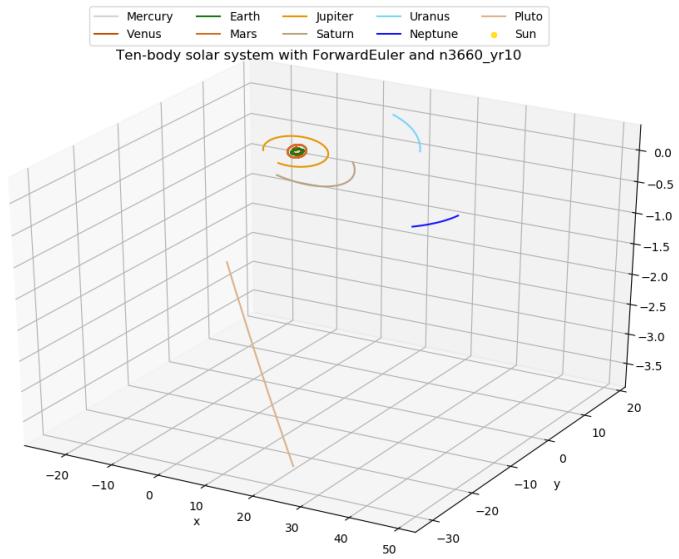


Figure 89: Plot of the ten-body solar system with 3660 iterations over 10 years with a static sun, made with Euler-Cromer.

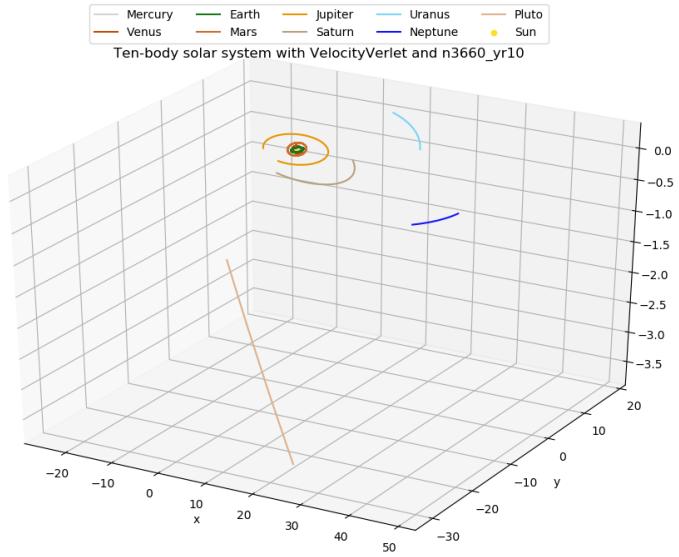


Figure 90: Plot of the ten-body solar system with 3660 iterations over 10 years with a static sun, made with Velocity Verlet.

4.5.2 Dynamic sun

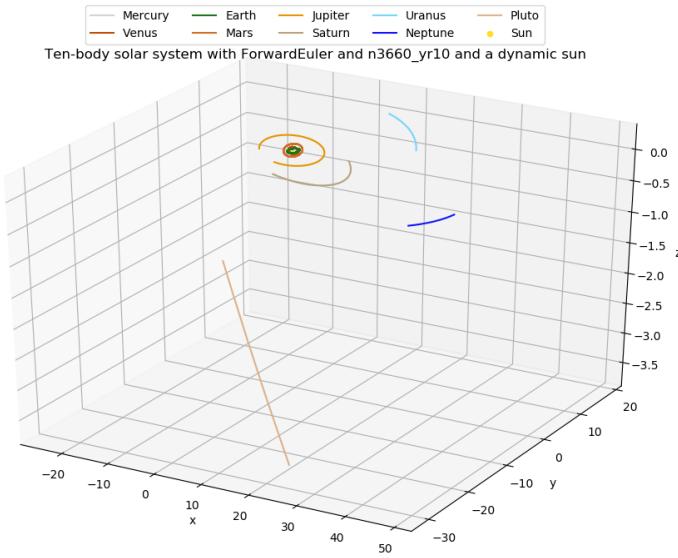


Figure 91: Plot of the ten-body solar system with 3660 iterations over 10 years with a dynamic sun, made with Euler-Cromer.

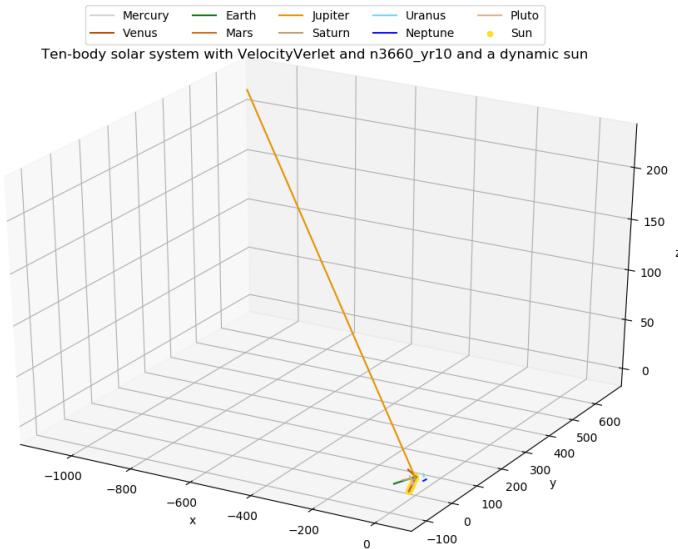


Figure 92: Plot of the ten-body solar system with 3660 iterations over 10 years with a dynamic sun, made with Velocity Verlet.

5 Discussion

5.1 Stability of the Sun-Earth solar system given by the Euler-Cromer method and the Velocity Verlet method

Figures (1) through (6) plots the Sun-Earth solar system for different iterations. Here it is possible to see that an increasing number of iterations will improve the stability of Earth's orbit around the Sun. For figure (1) and (4) the Earth's orbit is not stable, given by that the orbit changes position drastically. The Velocity Verlet method demonstrates how it is a better approximation than Euler-Cromer given by that the orbit changes less and is therefore more stable. By increasing the number of iterations to $n = 260$ the orbit has stabilized by a lot for both methods, though mostly for Velocity Verlet which is almost indistinguishable from the next iteration step with $n = 520$. With further increase to $n = 520$ both methods have stabilized compared to a higher number of iterations. Figures (3) and (6) illustrates this. The plot given by the Euler-Cromer method may have a small increase in stability by a higher number of iterations, but given by the figures for the Velocity Verlet method this should be very small. Originally this test of the two methods have data for other larger numbers of iterations, but seeing as the orbits have seemingly stabilized we deemed the plotting of these datas as unnecessary. The stability should increase by an increasing number of iterations because a larger number of iterations means that more changes in the bodies in the solar system can be accounted for, giving a more accurate representation of the solar system. The stabilizing of the Earth's orbit therefore implies the stability of the Euler-Cromer method and the Velocity Verlet method.

The number of FLOPs for one iterations is for Euler-Cromer $(2n+8)_\gamma + 4$ and for Velocity Verlet $(2n+8)_\gamma + 6$, where n is the number of iterations and γ is the number of planets. Therefore, the Velocity Verlet should use longer time to compute, because it has more operations to compute. The timings is listed in the table (1) and supports the theory of FLOPs. The table shows that Euler-Cromer used 26 s whereas Velocity Verlet used 34 s. Euler-Cromer used fewer seconds because it had fewer FLOPs. However, the approximation of Velocity Verlet is better than Euler-Cromer, hence the time difference is easy to overlook to instead have better accuracy of the solar system. When looking at all iterations performed, the FLOPs above should be multiplied with n for the number of iterations.

When looking at the conservation of energy for the two different methods, it is possible to see that Euler-Cromer does not conserve energy, whereas Velocity Verlet does. This is because plots of the Sun-Earth system with Euler-Cromer has an orbit of the Earth which moves around the Sun, while the orbit for the Earth with Velocity Verlet just rotates around the Sun. Figures (1) and (4) illustrates this the best. The conservation of energy is important in simulations of the solar system because energy has an important effect on the orbit of the planets. This is shown by the plots with the Velocity Verlet method, which conserves energy, and how the orbits are rounder and more consistent.

5.2 The escape velocity of the Earth in the Sun-Earth solar system

The plots of the escape velocity is given in figures (13) through (18). From the theory and appendix, equation (1) and section A.2, the theoretical escape velocity of the Earth is $v_p = 8.886$ m/s, which is a little smaller than 2.8π . Therefore, it is expected that the Earth will not escape its orbit around the Sun until the initial velocity of the planet is around 2.8π . In figure (13) the Earth keeps its orbit, which is expected as the velocity is 1.0π . The same applies for figures (14) and (15), with respectively an initial velocity of 2.0π and 2.5π . For figure (16) it is possible to see that Earth escapes. This is not in agreement with the theoretical escape velocity. However, around $x = -15$ Earth starts to move back to the Sun. The Earth does not escape successfully, and with more iterations, which gives more positions, the Earth should return to its orbit around the Sun. For figure (17), with an initial velocity of 2.8π , the Earth has successfully escaped, as expected from the theoretical escape velocity. Figure (18) further illustrates the escape with initial velocity over the theoretical escape velocity, where the initial velocity is 3.0π .

5.3 Varying β for the Sun-Earth solar system

The plots of the varying of β is given in figures (30) through (40). β is the number which denotes the exponent of the distance between the Earth and the Sun, given in the equation (8) below.

$$F_G = \frac{GM_{\odot}M_{earth}}{r^{\beta}} \quad (8)$$

By increasing the value of β the gravitational force should become smaller. This means that the Sun should have less effect on the orbit of the Earth, and the Earth could possibly escape the solar system. When $\beta = 2$, which gives the regular gravitational force when substituted in equation (8), the Earth behaves as normal. Figure (30) shows this. With increasing β from 2.0 to 2.9 there is no apparent visible difference. However, when $\beta = 3.0$ the Earth slowly goes in a larger orbit around the Sun. This means that the gravitational force which holds Earth in its orbit is too small, and the Earth escapes. This is seen in figure (40).

5.4 Stability of the Sun-Earth-Jupiter solar system

As the program went from Sun-Earth to Sun-Earth-Jupiter it was interesting to watch how the system would change, and to see how the program handled a three-body problem. Looking at figure (50) and (57) we can see that Euler and Verlet is more or less the same.

Looking at stability one can see that the system looks pretty stable already at $n = 312$ and $yr = 12$ (measure every other week). The system looks really stable and has visually no difference after $n = 1248$ (measure twice per week). To see more about the measuring frequencies, look at section (A.5).

5.5 Varying mass for Jupiter in the Sun-Earth-Jupiter solar system

For the normal mass of Jupiter, it can bee seen that the movement of Earth does not change noticeably from the Sun-Earth case. Looking at the case where Jupiter is 10x as heavy there does not seem to be a difference, and when Jupiter is 1000x as heavy the orbit just seems thicker.

Looking at the gifs for these runs the differences is much more noticeable. [Normal mass of Jupiter](#), [10x mass of Jupiter](#) and [1000x mass of Jupiter](#). Though these gifs have a low iteration-count it is easy to see how Jupiter slightly pulls the orbit of the Earth towards the position of Jupiter. For larger masses this is much more noticeable.

5.6 Moving Sun in the in the Sun-Earth-Jupiter solar system

For Velocity Verlet the Sun spirals of out of the solar system. See figure (57). It might be that there was too few iterations, and so something went wrong. At the same time, since Velocity Verlet conserves energy, it should not happen. This could mean there is an error in the program that has gone unnoticed.

Looking at The Euler-Cromer method the Sun stays in a stable orbit. See figure (50). Jupiter's orbit stars going skew. The static Sun-case simulated everything in the xy-plane, whilst the dynamic Sun-case took NASA's data [4]. It could be that the real data has a more skew motion of the planets' orbits. Nevertheless, the data sseems to show a stable system. This is good, and an indication that a static sun is a good approximation. Most likely, had the static Sun-simulation used real data, the orbit would be more or less identical to the ones in the system with a dynamic Sun.

5.7 Model for all planets of the solar system

Extend your program to include all planets in the solar system (if you have time, you can also include the various moons, but it is not required) and discuss your results.

Figures (89)-(92) shows the plots of the complete solar system. One interesting thing to note right off the bat is that for system with a dynamic sun, the Velocity Verlet algorithm makes Venus slingshot out into outer space. A possible reason for this can be to few iterations. Looking closer and trying with $n = 7320$, Venus flew off faster and longer. $n = 7320$ and $yr = 10$ would be two measuring points per day. One would think this was enough iterations for the problem to disappear, yet it did not. It might be that the problem disappeared if we had ran even more iterations. This seems very strange considering the fact that Velocity Verlet has worked flawlessly for all other runs and that the data is fetched from NASA [4]. This does not happen when using a static sun.

When comparing the Euler-Cromer plots for static, figure (89), and dynamic sun, figure (91), it is possible to see that there is practically no difference. This is excellent and means that a static Sun is a good approximation for the solar system! It is quite interesting to see how the solar system evolves over ten years. For the inner solar system (Mercury to Mars) the bodies completes a full revolution, or more, around the Sun. Looking at the outer solar system (Jovian planets + Pluto) a full revolution is not reached. This is due to the immense distance they have to travel, and the velocity they have. Had the simulation been for ≈ 250 years we would see even Pluto complete its revolution as its orbital period is ≈ 248 years.

5.8 Perihelion precession of Mercury

Looking at the perihelion precession of Mercury an iteration-count of 36600 was used over 100 years, as this was more than enough for all other runs. The data is shown in figure (42) and (44). The 1st of the two is using non-relativistic gravity, whilst the 2nd is with relativistic gravity. The difference in the plots is not noticeable, which means that the relativistic correction might have been incorrect. The likeness can also be attributed to the fact that the amount of iterations was too low. Having looked around a bit, it seemed other people doing the same simulations had used a lot more iterations (around 10^8 or more).

Evaluating the data in `perihelion.py` one can see that something is quite wrong. This could be from a faulty conversion from radians to arc-seconds, or to the crude method of picking out data-points. Nevertheless, the last value (which should be 43 arc-seconds) turns out to be ≈ -195000 which is WAY too much. Even in radians the value is ≈ -0.942 , even though it should be multiple magnitudes lower.

Sadly there was not enough time to bug fix the issue. However, the issue was looked into and attempted to be solved. It can be assumed that the relativistic addition to the equation of gravitational force made a better prediction of the perihelion precession of Mercury.

6 Conclusion and perspective

In this numerical study the solar system was simulated with two different numerical methods. The solar systems studied was the Sun-Earth solar system, the Sun-Mercury solar system, the Sun-Earth-Jupiter solar system and the whole ten-body solar system including the dwarf planet Pluto. The two methods used were Euler-Cromer, which does not conserve energy, and Velocity Verlet, which conserves energy. Conservation of energy proved to have an important effect on the stability of the solar systems. See sections (5.1) and (5.4) for the discussion of the stability of the Sun-Earth solar system and the Sun-Earth-Jupiter solar system. Here the superiourness of the Velocity Verlet was proven, even though it used a longer computation time. For the Sun-Earth solar system we tested the escape velocity of Earth in comparison to the theoretical value calculated in the appendix, see section A.2 and equation (1). The discussion is in the section 5.2. Here it was found that the escape velocity from our data is almost exactly that of the theoretical. Further for the Sun-Earth solar system we tested the gravitational force from the Sun on the Earth with differing exponents of the distance between the two objects, see section 5.3. This also behaved as expected. When looking at the Sun-Earth-Jupiter we can see that increasing the mass of Jupiter by factors of 1, 10 and 1000 with differing number of iterations, will affect Earth's orbit in the solar system, see section 5.5. For the ten-body solar

system it is interesting to see how vastly different the distances between the planets to the Sun are and their orbital velocity. This is most visible by the fact that most planets does not make a full revolution around the Sun by ten years. See the section 5.7. This study has in conclusion shown the difference between the two numerical methods by their simulation of the solar system, and given interesting and cool simulations of the solar system.

7 References

- [1] Morten H. Jensen (2019), [Project 5](#), Departement of Physics, University of Oslo, Norway
- [2] Erik B. Grammeltvedt, Alexandra Jahr Kolstad, Erlend T. North (2019), [GitHub](#), Students of Department of Physics, University of Oslo, Norway
- [3] Morten H. Jensen (2015), [Lecture slides for FYS3150](#), Department of Physics, University of Oslo, Norway
- [4] Jon. D. Giorgini (2019), [Physical data of the planets and the sun.](#), Solar System Dynamics Group, Horizons On-Line Ephemeris System, Jet Propulsion Laboratory, Pasadena, California, USA

A Appendix

A.1 Mass-conversion

Celestial Body	Mass [kg]	Mass [M_{\odot}]	Distance to sun [AU]
Sun	$2 \cdot 10^{30}$	1	0
Mercury	$3.3 \cdot 10^{23}$	$1.65 \cdot 10^{-7}$	0.39
Venus	$4.9 \cdot 10^{24}$	$2.45 \cdot 10^{-6}$	0.72
Earth	$6 \cdot 10^{24}$	$3.0 \cdot 10^{-6}$	1
Mars	$6.6 \cdot 10^{23}$	$3.3 \cdot 10^{-7}$	1.52
Jupiter	$1.9 \cdot 10^{27}$	$9.5 \cdot 10^{-4}$	5.20
Saturn	$5.5 \cdot 10^{26}$	$2.75 \cdot 10^{-4}$	9.54
Uranus	$8.8 \cdot 10^{25}$	$4.4 \cdot 10^{-5}$	19.19
Neptune	$1.03 \cdot 10^{26}$	$5.15 \cdot 10^{-5}$	30.06
Pluto	$1.31 \cdot 10^{22}$	$6.55 \cdot 10^{-9}$	39.53

Table 2: Astronomical data retrieved from [4].

A.2 Explaining the calculation of a planets escape velocity

As some of these problems are solvable by hand there are some calculations that can be done prior to the simulations in order to have something to compare results with. For instance one can quite easily calculate a planets escape velocity from the Sun. One can use that the escape velocity would be the lowest energy required to move the planet out of the gravitational pull from the Sun. In that case the planet would be

left with zero kinetic energy. Because the planet is moved out of the gravitational field the potential energy would also be 0. Therefore the expression for the system will be given as equation (9). This is with the condition that there is no other forces acting on the system except for the gravitational force.

$$U_i + K_i = U_f + K_f \quad (9)$$

Due to the criteria we have set for the model and the need for conservation of energy the expression becomes:

$$U_i + K_i = 0 + 0 \quad (10)$$

The potential energy is given by the gravitational potential which is:

$$U = -\frac{GMm}{r} \quad (11)$$

The kinetic energy is given by the planets kinetic energy:

$$K = \frac{1}{2}mv_p^2 \quad (12)$$

Therefore the escape velocity can be calculated as the kinetic energy being equal to the potential energy.

$$v_p = \sqrt{\frac{2GM}{r}} \quad (13)$$

A.3 Rewriting Newtons second law of motion

Newton's second law of motion is stated in equation (14).

$$F = ma \quad (14)$$

Equation (14) can be written as a differential equation that will give the force as a function of position (15).

$$F(x, t) = m \frac{d^2x}{dt^2} \quad (15)$$

Since we know that the velocity, v , is given as a function of change in position over change in time. This gives us equation (16).

$$v(x, t) = \frac{dx}{dt} \quad (16)$$

Using equation (14), (15) and (16), one can now create two coupled differential equations. The first one is (16) and the second one is (17). (17) is a combination of (14), (15) and (16).

$$\frac{dv}{dt} = F(x, t)/m = a(x, t) \quad (17)$$

Solving this system for x with a Taylor expansion one gets (18).

$$x(t + h) = x(t) + hx^1(t) + \frac{h^2}{2}x^2(t) + O(h^4) \quad (18)$$

From equation (16) and (17) we have that $a(x, t) = x^2(t)$. Using equation (16), $x^1(t) = v(x, t)$ can now be substituted in to equation (18). This gives equation (19).

$$x(t + h) = x(t) + hv(x, t) + \frac{h^2}{2}a(x, t) + (O)h^4 \quad (19)$$

As one can see from the equation the system has a truncation error of $(O)h^4$.

Using a Tayler expansion for the velocity as well as for the position gives equation (20).

$$v(t + h) = v(t) + hv^1(x, t) + \frac{h^2}{2}v(t) \quad (20)$$

It is important to discretize the expressions as they are the ones that will be used in the program. This changes $x(t + h)$ to $x(i + 1)$.

A.4 Adjusting Newtons method for relativity

Einstiens theory of relativity helped answer several unknown ideas. One of them is a relativistic correction to Newton's law of universal gravity. This correction led to a better understanding of planetary orbitals. In this project the simulation of Newton's law of universal gravity with a relativistic correction has been done on Mercury. This is based on equation (21) below.

$$F = \frac{GMm}{r^2} \left(1 + \frac{3l^2}{r^2c^2}\right) \quad (21)$$

l is the angular momentum of Mercury's orbital $l = |\vec{v} \times \vec{r}|$ and c is the speed of light in vacuum.

A.5 Measuring frequencies

Stability was checked with the following measuring frequencies:

Frequency	$\frac{n}{yr}$
Once per month	12 per year
Every other week	26 per year
Every week	52 per year
Twice per week	104 per year
Every other day	183 per year
Every day	366 per year
Twice per day	732 per year

Table 3: Measuring frequencies and their given n/yr

To see what frequency was used, look at the description and divide n by yr to find measuring points per year. Then look that number up the list above.