Project 3 FYS4150

Kjetil Karlsen and Vilde Mari Reinertsen

October 27, 2017

Abstract

The program used in this project can be found at Github.

Egen klasse for å kjøre algoritmer i (tidsteg, skrive fil)

PLAN: Torsdag: Kjetil: Teori, metode, (intro?), Fikse angulærmoment Vilde: Plots, andre resultater (CPU-tid...), (skrive resultater)

Test: Energy conservation, modulus "position" (lengde vektor) bevart Alle vectorer samme str.

Printe + plotte energi stabilitet mellom euler og verlet.

To do: OBS Unit tests

3b: Forklare objektorientering, hvorfor deler kan generaliseres.

3c: - Find out which initial velocity that gives a circular motion (plot) - Test stability (energy-stability) as function of dt (both Verlet and Euler) - Plot the earth orbiting the sun - Check (for the circular orbit) that the energy is conserved (plot - both kin and pot separated and together?) - Check that angular moment is conserved

- Discuss differences between Euler and Verlet - number of FLOPS + CPU time

* Plotte ulike dt-er * Plott energi som funk av ulike dt * Referere til convergens - funksjonen. * Vise at angulærmoment bevart

3d: - Find escape velocity (plot) - Compare with numerical results(Result? or Discussion?) - Find exact escape velocity (theory?) - Changing beta (plot) - Comment result + What happens when beta -> 3 ?

Exact løsning escape vel Plots ulike init.hastigheter Bytte gravitasjonskrefter...

3e: - How much does Jupiter alter Earth's orbit? - Position of Jupiter and Earth (plot) - Plot Earth's motion for increased mass of Jupiter (3 masses) - - Discuss stability of velocity verlet (3 body)

* 3 ulike masser * Plotte alle banene * Stabilitet: Energi-plot

3f: - Find center off mass - use as origin - Give sun initial velocity so momentum is zero (origin is fixed) - Compare with 3e) - Extend to all planets (plot) - Discuss difference 3e) and 3f) (3 body) - Discuss result of all planets

3g: - Find perihelion for both relativistic and non-relativisitc (table) - Relativistic - should be a few magnitudes smaller. - Can the observed perihelion precession of Mercury be explained by the general theory of relativity?

FLOPS euler/Verlet Result: - Find out which initial velocity that gives a circular motion (plot) - Test stability (energy-stability) as function of dt (both Verlet and Euler) - Plot the earth orbiting the sun - Check (for the circular orbit) that the energy is conserved (plot - both kin and pot separated and together?) - Check that angular moment is conserved

- Find escape velocity (plot) - Compare exact and numerical results - Find exact escape velocity (theory?) - Changing beta (plot) - Comment result + What happens when beta -> 3 ? (last part in discussion? - How much does Jupiter alter Earth's orbit? - Position of Jupiter and Earth (plot) - Plot Earth's motion for increased mass of Jupiter (3 masses) - Find center off mass - use as origin - Give sun initial velocity so momentum is zero (origin is fixed) - Compare with 3e) - Extend to all planets (plot) - Find perihelion for both relativistic and non-relativisitc (table) - Relativistic - should be a few magnitudes smaller. - Can the observed perihelion precession of Mercury be explained by the general theory of relativity?

Discussion: - Discuss differences between Euler and Verlet - number of FLOPS + CPU time - Discuss stability of velocity verlet (3 body) - Discuss difference 3e) and 3f) (3 body) - Discuss result of all planets

Contents

1 Introduction

There are many physical problems that deals with differential equations. A simple example of this is the ordinary differential equations that governs how objects move in relations to each other, for example the solar system. The aim of this project is to model the our solar system using a relatively simple model, relativistic model for the movement of the planets and the sun. In order to do this efficiently to different integration methods will be explored, the Euler method and the Velocity Verlet method.

This report starts by the explaining basic theoretical physics that governs the movement of the objects in the solar system. These equations can be scaled and implemented in a general algorithm for solving 2. order differential equations. The two different integration models will then be discussed alongside how we chose to implement these in our experiment.

The results of benchmark and stability tests of the different methods will be discussed together with more general results of how the planets are moving. This is all wrapped up in a conclusion at the end.

2 Theory

2.1 Classical Solar system

In the classical description of the solar system, there is only a single force working:

$$F_G = -G\frac{M_1 M_2}{r^2} \tag{1}$$

By applying Newtons 2.law on component form we achieve two more equations, in the case of a two dimensional system.

$$\frac{\mathrm{d}^2\vec{r}}{\mathrm{d}t^2} = \frac{\vec{F}_G}{M_2} \tag{2}$$

Equation 2 is in reality two seperate, independent equations, one for the x-direction and the other for the y-direction.

2.2 Units and scaling

A computer has a limited bit-resolution and the distances and timescale are large when computing the solar system. This means that it is important to use appropriate units. The distance between the sun and the earth is defined as 1 Astronomical unit (1 Au) and the timescale used in this project will be in units of 1 year.

For simplicity we will define a new unite of energy: $J_{ast} = \frac{m_{planet}}{|M_{sun}|}\left(\frac{Au}{year}\right)^2$, where $|M_{sun}|$. In reallity this is simply Joule with a prefactor. This prefactor can be obtained by inserting the values into J_{ast}:

$$J_{ast} = \frac{m_{planet}}{|M_{sun}|}\left(\frac{Au}{year}\right)^2 \tag{3}$$

$$= \frac{m_{planet}}{2\cdot 10^{30}}\left(|v|\frac{1.5\cdot 10^{11}m}{360*24*60*60s}\right)^2 \tag{4}$$

$$\simeq 1.163\cdot 10^{-23}J \tag{5}$$

The dimensionality of the variables are as follows: $[v] = m/s$ and $[m_{planet}] = kg$.

We know that the earth needs one year to orbit the sun, meaning that $v = \frac{2\pi r}{1\text{ year}}$. This can be rewritten with $v = \tilde{v}v_0$ and $r = \tilde{r}r_0$. The units of r and v are contained in $v_0 = \frac{1\text{Au}}{1\text{ year}}$ and $r_0 = 1$ Au, giving that $\tilde{v}^2\tilde{r} = 4\pi^2$. In the same way $t = \tilde{t}t_0$, with $t_0 = 1$ year.

For the case of the earth - sun system one can assume that the sun is stationary, as $M_{sun} >> M_{earth}$. The force experienced by the earth is thus centrifugal, which means that $a = \frac{v^2}{r}$, with $v = 2\pi r/1year$. Combining this with equations 1 and 2 it is possible to scale the equations in the following manner:

$$a_E = \frac{F_E}{M_E} = G\frac{M_{sun}}{r^2} = \frac{v^2}{r} \tag{6}$$

$$GM_{sun} = v^2 r = 4\pi^2\frac{(1\text{ Au})^3}{(1\text{ year})^2} \tag{7}$$

$$\tag{8}$$

This gives that the dimensionless expression can be stated as:

$$\frac{\mathrm{d}\tilde{v}}{\mathrm{d}\tilde{t}} = \frac{4\pi^2}{\tilde{r}^2} \tag{9}$$

In the two dimensional system $r = (x, y) = (r\cos\theta, r\sin\theta)$ Using the notation $\dot{p} = \frac{\mathrm{d}p}{\mathrm{d}t}$, this gives the following coupled differential equations:

$$\dot{v}_x = \frac{4\pi^2 r\cos\theta}{r^3} = \frac{4\pi^2 x}{r^3} \tag{10}$$

$$\dot{v}_y = -\frac{4\pi^2 r\sin\theta}{r^3} = \frac{4\pi^2 y}{r^3} \tag{11}$$

$$\dot{x} = v_x \tag{12}$$

$$\dot{y} = v_y \tag{13}$$

$$\tag{14}$$

2.3 Energy considerations

As with any physical system, the total energy has to be conserved. The potential energy $E_p = \int_{r'}^{\infty} \vec{F}(r') \cdot d\vec{r}' = -\frac{GM_1M_2}{r}$, while the kinetic energy is $E_K = \frac{1}{2}mv^2$. This gives a total energy of:

$$E_{tot} = \frac{1}{2}mv^2 - \frac{GM_1M_2}{r} \tag{15}$$

In addition, the angular momentum $(\vec{L})$ of the system has to be preserved. This is because there are no additional sources of torque $(\vec{\tau})$ once the system has been initialized and $\vec{\tau} = \frac{\mathrm{d}\vec{L}}{\mathrm{d}t} = 0$. As a result all the absolute value of $\vec{L}$ hast to be constant.

In order to maintain in the gravitational field of another object, the distance between them needs to be smaller than ∞. However, with a large enough velocity it is possible to escape. By setting equation 15 equal to 0 one obtains the lowest escape velocity v_{esc}:

$$\frac{1}{2}mv_{esc}^2 - \frac{GM_1M_2}{r} = 0 \tag{16}$$

$$v_{esc} = \sqrt{\frac{2GM_1M_2}{mr}} \tag{17}$$

Nevne approksimasjon med barrycentre v sol i sentrum v pos sol @ t=0: Vi har ingen beregninger med sol i sentrum, men bruker sol ved t=0 som posisjon.

Hvorfor bør egentlig være barrycentre til univers?

Can the observed perihelion precession of Mercury be explained by the general theory of relativity?

3 Method

In this project we tested two numerical integration techniques, the Euler method and the velocity Verlet method. This section is largely based on Hjort-Jensen [1]

3.1 The Euler method

When evaluating the function $x(t)$ in the interval between t and $t + h$ it is natural to do a Taylor expansion:

$$x(t+h) = x(t) + \sum_{n=1}^{\infty} \frac{h^n}{n!} \frac{\mathrm{d}^n x(t)}{\mathrm{d}t^n} \tag{18}$$

By choosing a small h, it is sufficient to truncate the sum after n=1, giving $x(t+h) = x(t) + h\frac{\mathrm{d}x}{\mathrm{d}t} + O(h^2)$. The term$O(h^2)$ contains the rest of the infinite sum, often called the truncation error. In order for a computer to use this method it is necessary to discretise the expression, substituting $x(t) \to x(t_i) \to x_i$. The Euler method can thus be expressed as

$$x_{i+1} = x_i + h\dot{x}_i + O(h^2) \tag{19}$$

Combining this with equations 10 and 12 we get an algorithm for doing a 1 dimentional integration for the solar system:

$$a_i = F(t_i) \tag{20}$$

$$v_{i+1} = v_i + ha_i \tag{21}$$

$$x_{i+1} = x_i + hx_i \tag{22}$$

This gives in total 4 floating points operations per iteration, per dimension. For our two dimentional system this results in a total of 8N FLOPS per iteration, with $N = \frac{1}{h}$.

3.2 The Velocity Verlet method

The velocity Verlet can be derived from the same Taylor expansion as the Euler method, equation 18. As with the Euler method, we will truncate the taylor series for position after n=1. However, for the velocity there exist a similar Taylor series: $v_{i+1} = v_i + h\dot{v}_i + \frac{h}{2}\ddot{v}_i + O(h^3)$. Unfortunately, there is no expression for $\ddot{v}_i$, but it can be approximated by $h\ddot{v}_i \simeq \dot{v}_{i+1} - \dot{v}_i$. Adding this to the expression for v_{i+1} and doing some simple algebra one get the final expression for the velocity:

$$v_{i+1} = v_i + \frac{h}{2}\left(\dot{v}_{i+1} + \dot{v}_i\right) + O(h^3) \qquad (23)$$

Looking a bit closer at equation 23, there is a immediate problem. In order to calculate v_{i+1} one need to already know v_{i+1}. This problem can be solved by updating the velocity in two steps, which gives the following algorithm:

$$v_{i+\frac{1}{2}} = v_i + \frac{h}{2}a_i \qquad (24)$$

$$x_{i+1} = x_i + v_{i+\frac{1}{2}} \qquad (25)$$

$$a_{i+1} = \frac{F_{i+1}}{m} \qquad (26)$$

$$v_{i+1} = v_{i+\frac{1}{2}}. + \frac{h}{2}a_{i+1} \qquad (27)$$

$$(28)$$

The numerical cost of the algorithm can be understood when analysing the number of floating points operations. In the case of the velocity verlet method there are 8 FLOPS per iteration and dimension. As the value $\frac{h}{2}$ a constant, it can be calculated in advance of the loop, thus reducing the FLOPS to 6. This gives a total of 12 FLOPS per iteration for the two dimensional system.

3.3 Choice of origin

As every planet in the solar system is moving, choosing a point of origin is not straight forwards. For the smaller systems, ie. the "sun-earth"-system, we chose to select the suns position at the start as the origin. This allows the sun to move. Another choice of origin is the solar system barycentre, which we utilized when calculating the entire solar system. Using the barrycentre can give a prettier picture of the physics which is (nummerically) unfolding, as every object will rotate around this point.

3.4 Object orientation

In order to simplify the calculation of an ensemble of planets, each with velocities, positions, energies and angular momentum, it is useful to generalize the code in an object oriented way. We chose to create one class for the planets, where all the internal dynamics (energies, position, velocity, ...) were stored. The forces experienced by the planets are specific for this project and we kept this in the Planet class, so that the second class, the Solver class, could be more general and easier reused.

This Solver class is were all the technicalities are located, including the different integration methods. For each time step one need the location of every planet in the system and Solver includes therefore a function that for each time step loop over all the planets. In order to update their position it is necessary to again loop over all the different planets in order to find the total gravitational force exerted on the current planet.

When all the calculations are taken care of by the solver class and all the properties of the different planets are stored in each planet object. This means that the main part of the program only needs to initialize the different planetary instances , adding these to the a instance of the Solver class which is initialized according to what output we want to achieve, see the snippet below for how this looks in 'main'.

```
    Planet earth("Initializing inputs");
            Planet sun("Initializing
inputs");
    Solver verlet("Initializing inputs")
;
    verlet.add(earth);
    verlet.add(sun);
    verlet.add(mercury);
    verlet.algorithm("input variables");
```

4 Result

with steps per year: 7*3600*360 Running velocity verlet Perihelion position after 100 years: 0.307498, -0.000933806 Perihelion angle after 100 years: -626.38 arc seconds CPU time: 3248.07

Result: - Find out which initial velocity that gives a circular motion (plot)

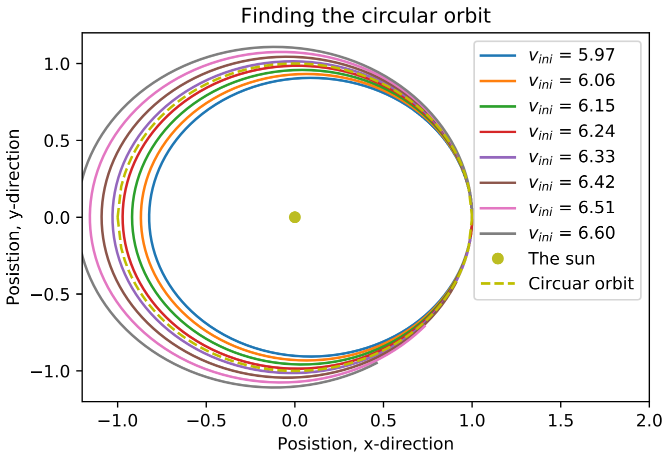

Figure 4.1: This is a plt of the orbit with different initial velocities. The circular orbit has a velocity between 6.24 and 6.33. $2\pi \approx 6.28$ and that is the initial velocity that gives a circular orbit.

- Test stability (energy-stability) as function of dt (both Verlet and Euler)

Weird results - I think we should look at this tomorrow...

- Plot the earth orbiting the sun 3 YEARS: Running velocity verlet CPU time: 0.013 Running Euler CPU time: 0.17

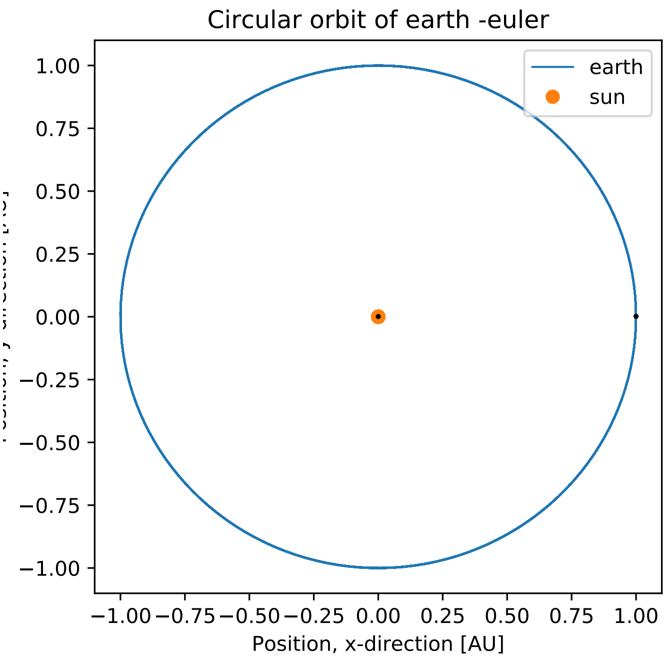

Figure 4.2: This is a plot of the earth's orbit for 100 years using Euler's method to simulate the motion.

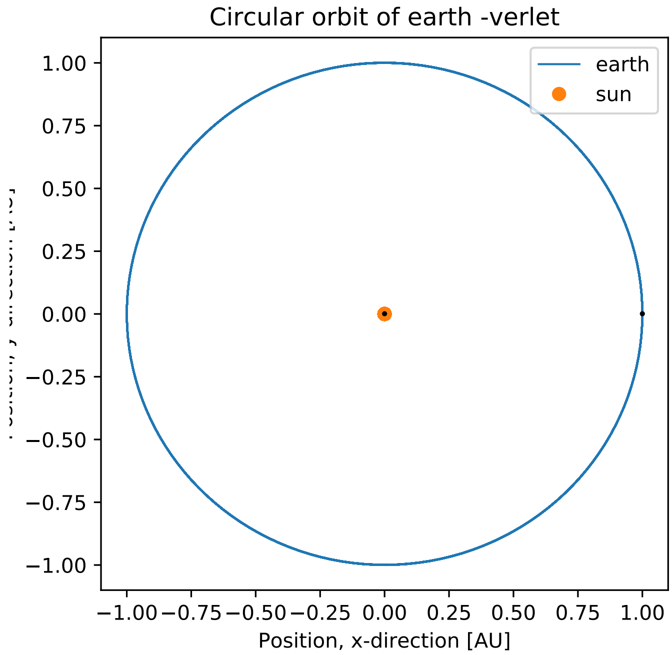

Figure 4.3: This is a plot of the earth's orbit for 100 years using the velocity Verlet method to simulate the motion.

- Check that angular moment is conserved

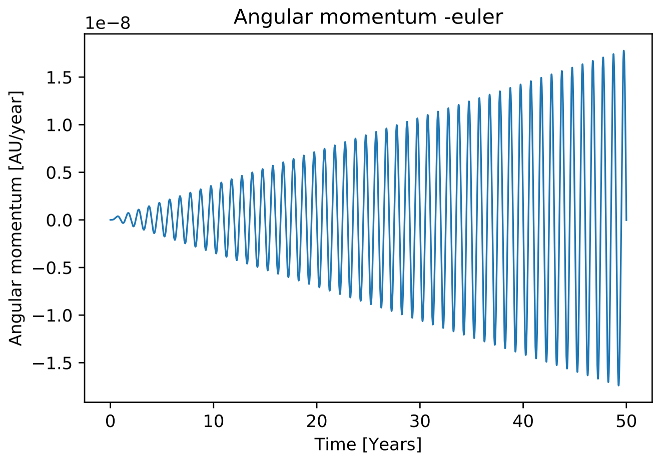

Figure 4.4: This is a plot of the angular momentum of the sun-earth system for 100 years using Euler's method.

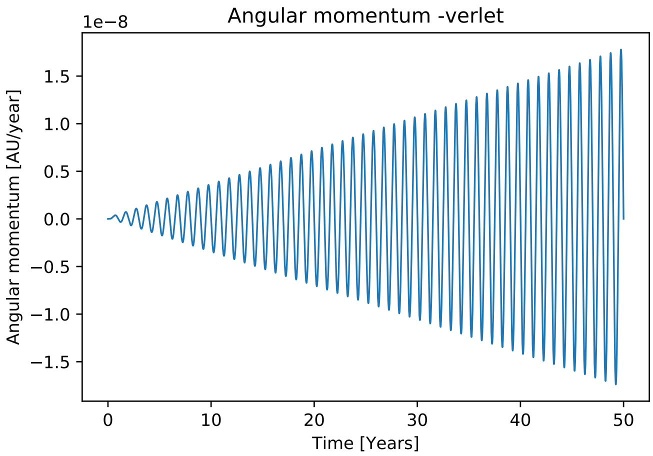

Figure 4.5: This is a plot of the angular momentum of the sun-earth system for 100 years using the velocity Verlet method.

- Check (for the circular orbit) that the energy is conserved (plot - both kin and pot separated and together?)

NEED SOME NUMBERS HERE:

Table 4.1: This is a table that list how the energy oscillates in the two different method (plot in appendix).

	Euler's Method	Ve
Kinetic energy oscillation		
Potential energy oscillation		
Total energy oscillation		

Put these in the appendix?

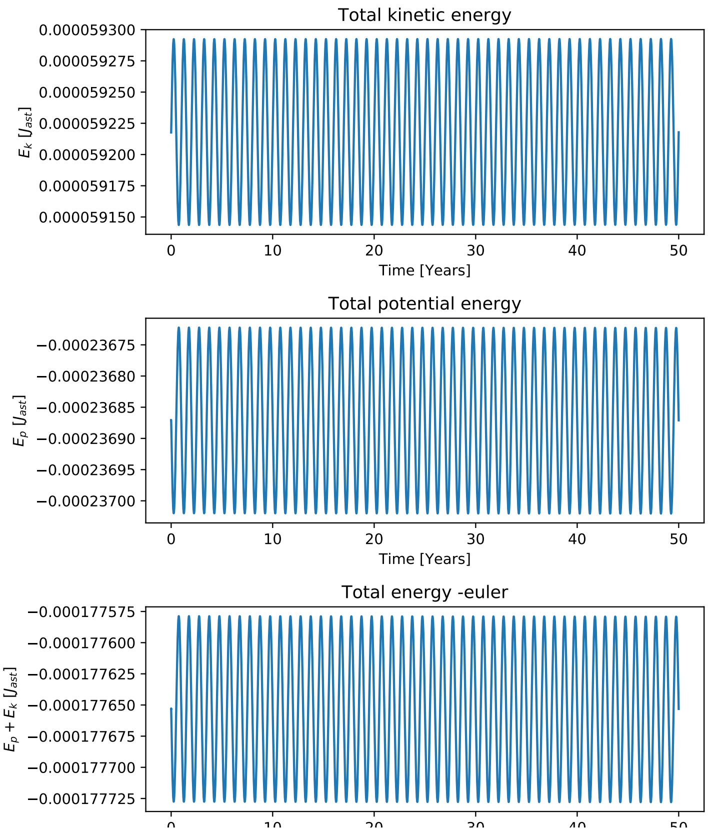

Figure 4.6: This is a plot of the total energy of the sun-earth system for 100 years using Euler's method.

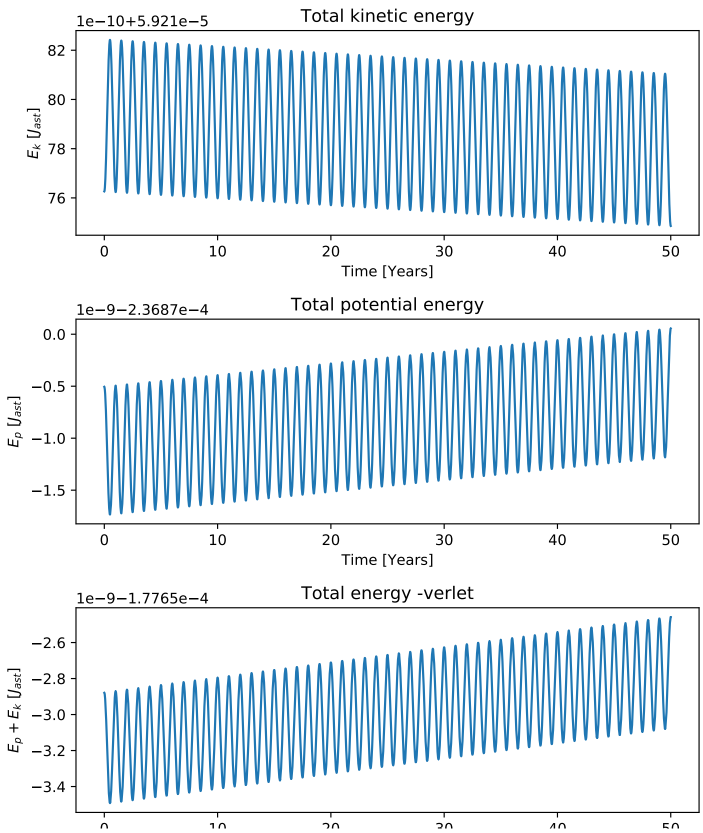

Figure 4.7: This is a plot of the total energy of the sun-earth system for 100 years using the velocity Verlet method.

- Find escape velocity (plot) - Compare exact and numerical results

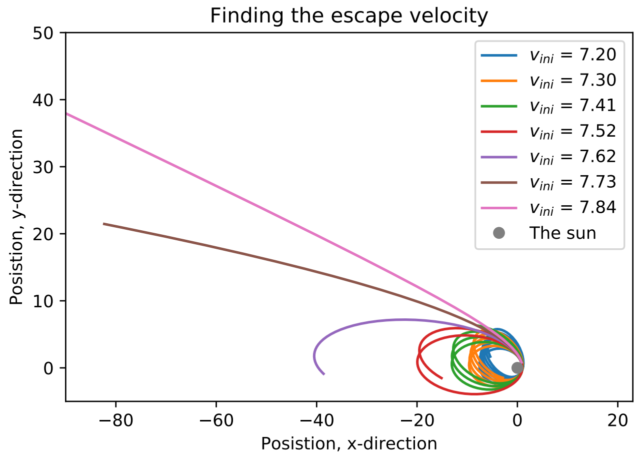

Figure 4.8: This is a plot of the sun-earth system for different initial velocities. The plot shows the necesary inital velocity to escape the sun.

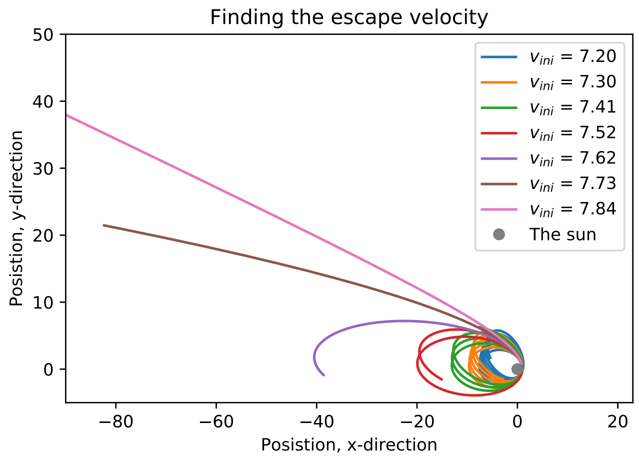

Figure 4.9: This is a plot of the sun-earth system for different initial velocities. The plot shows initial velocities near the exact escapevelocity, $\sqrt{2}2\pi$.

- Find exact escape velocity (theory?)

Exact escape velocity:

$$E_p + E_k = 0 + 0 \implies \frac{gM_E M_S}{r^2} = \frac{1}{2}M_E v^2$$

$$v^2 = \frac{4\pi^2 M_E}{M_E 1^2} AU^2 \implies v = \sqrt{2}2\pi$$

- Changing beta (plot) - Comment result + What happens when beta -> 3 ? (last part in discussion?

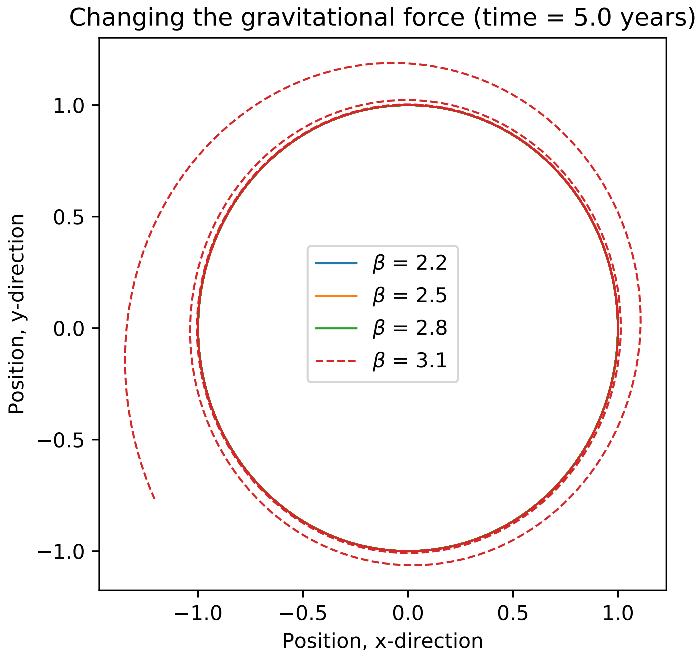

Figure 4.10: This is a plot of the sun-earth system for different gravitational forces. The gravitational force is given by $F = \frac{-GM_E M_S}{r^\beta}$ and in the plot β is changed from it's normal value $\beta = 2$ to $\beta = 3.1$.

- How much does Jupiter alter Earth's orbit? - Position of Jupiter and Earth (plot)

Ok - if initial values are correct.

- Plot Earth's motion for increased mass of Jupiter (3 masses)

same.

- Find center off mass - use as origin

Found and incorporated.

- Give sun initial velocity so momentum is zero (origin is fixed)

How? What momentum? velocity?

- Compare with 3e) - Extend to all planets (plot)

Table 4.2: This is a table with the Perihelion information about Mercury after one century, 100 years.

	Newtonian	Relativistic
Position [AU]:	(,)	(,)
Angle [arcseconds]:		

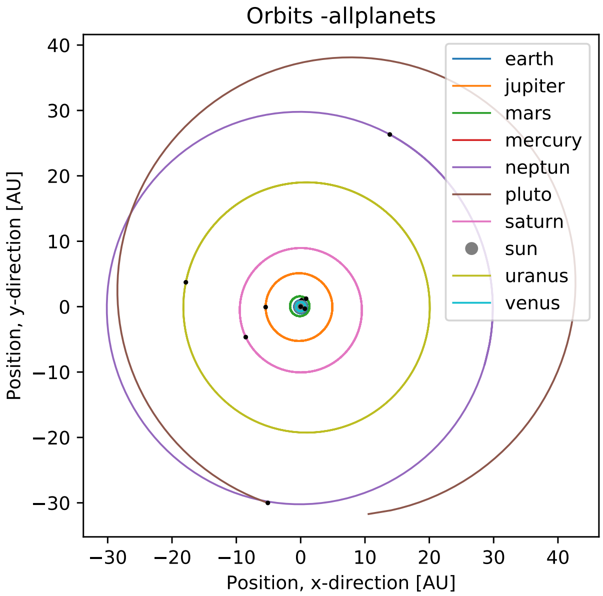

Figure 4.11: This is a plot of the Solar System after 200 years of motion.

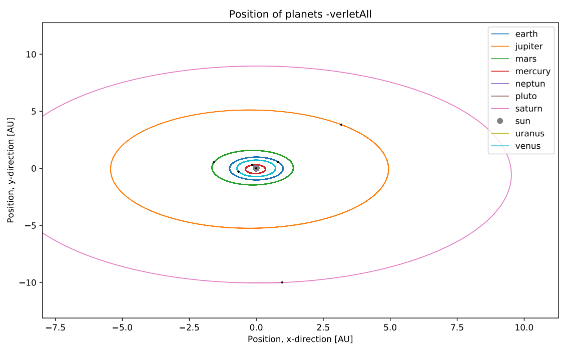

Figure 4.12: This is a plot of planets closest to the sun in the Solar System after 30 years of motion.

- Find perihelion for both relativistic and non-relativistic (table)

- Relativistic - should be a few magnitudes smaller.

- Can the observed perihelion precession of Mercury be explained by the general theory of relativity?

5 Discussion

6 Conclusion

References

[1] Morten Hjorth-Jensen. Computational physics: Lecture notes fall 2015. Department of Physics, University of Oslo, 8 2015. Chapter 2 and 6.