

# FYS3150 - Project 3

Daniel Heinesen, Gunnar Lange

October 19, 2016

## Abstract

We present numerical models of our solar system with varying levels of complexity.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Theoretical model</b>	<b>2</b>
2.1	Newtonian Gravity . . . . .	2
2.2	Implementing the initial conditions . . . . .	2
2.3	Discretizing the equations . . . . .	2
2.4	Choice of units . . . . .	4
2.5	Conserved quantities in the system . . . . .	4
<b>3</b>	<b>Methods</b>	<b>4</b>
3.1	Models . . . . .	4
<b>4</b>	<b>Results</b>	<b>4</b>
<b>5</b>	<b>Discussion</b>	<b>4</b>
<b>6</b>	<b>Conclusion and outlook</b>	<b>4</b>
6.1	Conclusion . . . . .	4
6.2	Outlook . . . . .	4

# 1 Introduction

The n-body problem is a recurring theme in many physics publications (see for example **HERE**), due to its wide applicability in many different fields, such as **REFERENCE**. Analytic solutions of these problems are notoriously hard to come by and are, in most situations, unattainable. Therefore, these problems are almost always solved by use of numerical techniques. We present two different numerical methods for solving the ODE's resulting from Newton's gravitational laws - Euler's method and the velocity-Verlet method. We investigate the stability of both methods for varying complexity of the system.

## 2 Theoretical model

### 2.1 Newtonian Gravity

Newton's general law of gravitation, which describes the force of gravity between two objects, is given by:

$$\vec{F}_G = \frac{Gm_1m_2}{r^3}\vec{r} \quad (1)$$

Here  $F_G$  is the force of gravity on the first mass,  $G$  is the universal gravitational constant,  $m_i$  are the masses of the two objects, and  $\vec{r}$  is the vector pointing from  $m_1$  to  $m_2$ . This law can be combined with Newton's second law of motion, to give a set of coupled differential equations:

$$\begin{aligned} \frac{d^2x}{dt^2} &= \frac{F_{G,x}}{m_1} \\ \frac{d^2y}{dt^2} &= \frac{F_{G,y}}{m_1} \\ \frac{d^2z}{dt^2} &= \frac{F_{G,z}}{m_1} \end{aligned} \quad (2)$$

Where  $F_{G,x}$ ,  $F_{G,y}$  and  $F_{G,z}$  are the components of the gravitational force in the  $x$ ,  $y$  and  $z$  directions respectively. These equations can be reduced to a set of first-order equations by introducing the velocity components,  $v$ , such that:

$$\frac{dx}{dt} = v_x, \quad \frac{dv_x}{dt} = \frac{F_{G,x}}{m_1} = a_x(x, y, z) \quad (3)$$

And equivalently for the  $y$  and  $z$  direction. This gives six coupled, first-order, linear, differential equations.

### 2.2 Implementing the initial conditions

To find a unique solution of the set of equations in section 2.1, we require some initial conditions, i.e. some  $x(t=0)$ ,  $v_x(t=0)$ , and equivalently for the  $y$  and  $z$  direction. For our first model (described in section 3.1.1) we will, for simplicity, let the earth be 1 AU away from the sun, with an initial velocity that ensures a circular orbit. This is described in detail **LATER**. For the subsequent models, with multiple planets, we will use the initial positions and velocity provided on NASA's webpages<sup>1</sup>, letting  $t=0$  correspond to **INSERT SOMETHING HERE**.

### 2.3 Discretizing the equations

We will solve the equations in 2 numerically. Therefore we require discretized versions of these equations. We want to simulate from  $t=t_0$  to  $t=t_f$ , and choose  $N$  evenly spaced points in this

---

<sup>1</sup><http://ssd.jpl.nasa.gov/horizons.cgi#top>

interval. This gives us a timestep,  $h$ , of:

$$h = \frac{t_f - t_0}{N}$$

Let now  $t_i = a + ih$ , where  $i$  goes from 0 to  $N$ . Furthermore, let  $x_i = x(t_i)$  and  $v_{x,i} = v_x(t_i)$ , and equivalently for the  $y$  and  $z$  direction. Finally, discretize the acceleration as  $a_{x,i} = a_x(x_i, y_i, z_i)$ . This allows us to discretize the derivatives in equation 3. We will do this in two separate ways; with the forward Euler algorithm and with the Velocity-Verlet algorithm.

### 2.3.1 Euler's forward method

Euler's forward algorithm amounts to discretizing the derivatives from section 2.1 as:

$$\frac{dg}{dt} \approx \frac{x_{i+1} - x_i}{h} \quad (4)$$

This approximation is based on Taylor-expanding the function  $g(t)$  to the second order, as explained **HERE**. Inserting this approximation into equation 4, and rearranging some of the terms, gives:

$$\begin{aligned} v_{x,i+1} &= v_{x,i} + ha_{x,i} \\ x_{i+1} &= x_i + hv_{x,i} \end{aligned} \quad (5)$$

And again equivalently for the  $x$  and  $y$  direction. This is the Euler forward algorithm. Combining this with the initial conditions,  $\vec{v}(t_0) = \vec{v}_0$  and  $\vec{x}(t_0) = \vec{x}_0$ , makes it possible to solve the equations of Newtonian gravity. Closer inspection (through Taylor expansion, such as found **HERE** here), reveals that the local error is proportional to  $h^2$ , whereas the global error is proportional to  $h$ . Thus, whilst this is a straightforward algorithm, the errors quickly accumulate. It is therefore of interest to investigate a slightly improved algorithm:

### 2.3.2 Velocity-Verlet Algorithm

The Velocity-Verlet algorithm is based on the idea of Taylor-expanding the acceleration, in addition to the velocity and position. The details can be found in the appendix **APPENDIX**, but the method produces the following equations:

$$\begin{aligned} x_{i+1} &= x_i + hv_{x,i} + \frac{h^2}{2}a_{x,i} \\ v_{x,i+1} &= v_{x,i} + \frac{h}{2}(a_{x,i+1} + a_{x,i}) \end{aligned} \quad (6)$$

And similarly for the  $x$  and the  $y$  equations. Note that  $a_{i+1}$  depends on  $x_i, y_i$  and  $z_i$ , and therefore all three position coordinates must be computed before we can compute the velocity. As shown in our analysis in the **APPENDIX**, the local error is proportional to  $h^3$ , and in turn the global error is proportional to  $h^2$ , as shown **HERE**.

Notice that  $a_{xi+1}$  can be reused in the next step, so this method does not incur any additional calls to the acceleration function in the main loop, as compared to the Forward Euler algorithm. The method only costs a few more FLOPs;

- We must perform two additional multiplications for each dimension, specifically  $(h^2/2) \cdot a_{x,i}$  and  $(h/2) \cdot a_{x,i+1}$
- We must perform two further additions per dimension, namely adding  $(h^2/2)a_{x,i}$  and  $(h/2)a_{x,i+1}$  to respectively  $x_{i+1}$  and  $v_{x,i+1}$ .

This gives four additional FLOPs per timestep and dimension, i.e. 12 FLOPs per timestep. Thus if we have  $N$  points in the main-loop, this gives an additional  $12N$  FLOPs. Note that  $h^2/2$  and  $h/2$  can be precomputed, and therefore do not contribute to the number of FLOPs in the main loop.

## 2.4 Choice of units

As we will be modelling our solar system, a natural unit of length is the average distance between the earth and the sun, the astronomical unit, AU. In SI-units,  $1\text{AU} \approx 1.496 \cdot 10^{11}\text{m}$ . As the most massive object in the solar system is the sun, we choose one solar mass,  $M_{\odot}$  as our mass unit. In SI units,  $M_{\odot} \approx 1.99 \cdot 10^{30}$ . We will use the mean tropical year, which is about 365.24 days of 864000 seconds **REFERENCE HERE**. This choice of units has one considerable advantage - it simplifies the expression for the gravitational constant,  $G$ . As shown in **APPENDIX**, with this choice of units the gravitational constant is:

$$G = 4\pi^2 \text{AU}^3/\text{yr}^2$$

## 2.5 Conserved quantities in the system

To check the consistency of our results, we will investigate whether certain quantities, which we expect to be conserved, are indeed conserved.

### 2.5.1 Conservation of energy

As there are no external forces in the system, we expect energy to be conserved in the system, at least for the full system. This is not necessarily the case if we keep the sun fixed, as there is no reaction force in that case. However, in the full system, energy should be conserved. There are two kinds of energy to consider in our system: kinetic and potential energy. The kinetic energy is simply given by:

$$E_k = \frac{1}{2}mv^2$$

Where  $m$  is the mass of our celestial body and  $v$  is the absolute value of its velocity. This is the standard formula for the kinetic energy. The potential energy can be found by integrating the force, choosing infinity as a reference point, i.e.  $U(\infty) = 0$ :

$$U = - \int_r^{\infty} \frac{GMm}{r'^2} dr' = - \frac{GMm}{r}$$

## 3 Methods

### 3.1 Models

We will implement a variety of models with different complexities, to test our simulations under many circumstances.

#### 3.1.1 First model - The earth-sun system

We begin with a simplified model of our solar system, containing only the earth and the sun. We will do this in two different ways; first we will keep the sun fixed at the origin of our coordinates system, letting the earth orbit the sun. Subsequently, we will transform this into a two-body problem by letting the earth and the sun influence each other. As the sun is significantly heavier than the earth, we expect there to be little difference between these two models. This provides an important consistency check. If these solutions differ significantly, we can conclude that there is an error somewhere. We simulate these solutions

## **4 Results**

## **5 Discussion**

## **6 Conclusion and outlook**

### **6.1 Conclusion**

### **6.2 Outlook**