Some theoretical and numerical aspects of the N-body problem

Kjartan Kári Gardarsson Mýrdal

Bachelor's thesis
2013:K12

Lund University

Faculty of Science
Centre for Mathematical Sciences
Mathematics

Centrum Scientiarum Mathematicarum

Abstract

The N-body problem has been studied for many centuries and is still of interest in contemporary science. A lot of effort has gone into solving this problem but it's unlikely that a general solution will be found with the mathematical tools we have today. We review some of the progress that has been made over the centuries in solving it. We take a look at the first integrals, existence of solutions and where singularities can occur. We solve the two body problem and take a look at the special case of central configurations. We find all the possible three-body central configurations, which are known as Euler's and Lagrange's solutions. When analytic solutions are missing it is natural to use numerical methods. We implement and compare four numerical solvers for differential equations: Euler's method, Heun's method, the classical fourth-order Runge-Kutta scheme and Störmer-Verlet. Comparison of accuracy is made using the known solutions discussed in the previous parts of this thesis.

Popular scientific summary

The famous physicist and mathematician Sir Isaac Newton formulated the universal gravitation theory about 300 years ago. This theory has allowed scientists to make calculations in order to predict movement of objects under the influence of gravitational force. In astronomy the movement of planets and asteroids can be formulated as a mathematical equation based on Newton's laws of gravitation. The equation describes how a force changes the speed and direction of the moving particles in space. In order to predict a future location one needs to solve this equation. This is referred to as the N-body problem. Still today the general problem is considered unsolved. In this thesis we will consider solutions to special cases of the problem such as when there are only two bodies. We will also take a look at numerical methods that can be used when analytical solutions don't exist.

Contents

Chapter 1

Introduction

1.1 The problem

One of the oldest problems in modern physics is the study of particles as they move under gravitational force from one another, subject to Newton's laws of motion. This problem is usually referred to as the N-body problem, where N represents the number of particles in the system. By a particle or a body is meant anything that can be represented by a point mass in space. In mathematics the problem can be formulated as a second order differential equation which needs to be solved to get a location of the particles for a given time value. Even though the problem has been studied in detail, no general solution is known today [4].

1.2 History

Around the year 1687 Isaac Newton (1642-1727) is said to have solved one of the biggest questions that riddled the minds of the Royal society and other scholars of that time. He had managed to construct a mathematical model describing universal gravitation. For centuries people had tried to model and explain the movement of the planets in the sky. The oldest mathematical model known today was made by the Greek astronomer Ptolemy around the year 140. His model was however geocentric and placing the earth in the center of the model made the path of the other planets extremely complicated. Nicholas Copernicus (1473-1543) had shown that the solar system was in fact heliocentric and Johannes Kepler (1571-1630) published his laws on planetary motion based on measurements of the motion of Mars. Kepler's laws seemed to hold quite accurately so the question was no longer how the planets moved but what caused them to do so. Other attempts had been made before Newton to explain the heliocentric model but with the lack of understanding of forces, none of them succeeded. Newton's interest originally laid more in gravitation on earth but noticing similarities he managed to solve the questions of the planets and unifying the two questions with one solution. Following this discovery, he published Mathematical Principles of Natural Philosophy, usually referred to as Principia. There he described how all matter in the universe follows the universal law of gravity. In order to explain some of those phenomena, Newton had to intro-

duce infinitesimal calculus. Being the first known published text on infinitesimal calculus, Principia didn't only revolutionize physics but mathematics as well. Although Newton had given a formulation of the behavior of objects under gravitational pull, there was still a system of equations to be solved in order for the future path to be known.

Newton had not been the only one at that time interested in infinitesimal calculus. Gottfried Wilhelm Leibniz (1646 - 1716) published about it around the same time and his publications gained quickly popularity in Europe. With the new mathematical theory and Newton's laws of gravitation the analytical development of astronomy progressed quickly in the beginning of the 18th century. Even though the new theory fitted many measurements well, there were some that didn't come out as good. One of those was the earth-moon system. If it was approximated as a two body system the moon seemed to follow some other rules and many started questioning if there were some other factors at play. There was a great interest in this system due to its importance for sea navigational purposes. Some mathematicians realized that the gravitational force of the sun had to be counted for. In order to gain better insight, there were further attempts at solving the three body problem. Newton had written that the problem gave him headaches and he never published any attempts towards a solution. Both Leonhard Euler (1707-1783) and Alexis Claude de Clairaut (1713 - 1765) put quite an effort into trying to find a general solution for three particles but both reported on failure. None the less they both contributed quite a lot to the theory and managed to publish approximations to the moon problem.

In 1885 King Oscar II of Sweden and Norway pledged a substantial price and medal to whom ever could solve the N-body problem or come the closest to it. Gösta Mittag-Leffler, the editor in chief of Acta Mathematica had convinced the king to do so. One of the participants was Henri Poincaré. He focused mainly on the 3-body problem and tried to find solutions to both the general case and special cases. While Poincaré attempted to solve the problem, he soon realized that he probably wouldn't find a solution and he was right. None the less he managed to make substantial discoveries within the field which he eventually submitted to the competition. Even though he failed to solve the problem, the judges of the competition unanimously chose his work to win the price. Poincaré's discoveries allowed for many other mathematicians to take a new different approach to the problem. Some of what followed in the 20th century will be discussed in the following chapters [12, 5].

Chapter 2

The central force problem

2.1 Definition and Kepler's laws of planetary motion

Before looking at the N-body problem we consider the central force problem and Kepler's laws of planetary motion. The central force problem has an application on the N-body problem that will be discussed in detail later. These are Kepler's three laws [4]:

1. Every planet moves in an elliptical orbit with the sun in one of its foci.
2. A line between a planet and the sun sweeps out equal areas during equal intervals of time.
3. The square of the orbital period of a planet is directly proportional to the cube of the semi-major axis of the orbit.

He reached these conclusions from observations of planets orbiting the sun.

To formulate this mathematically, consider a particle in space with mass m and location vector $\mathbf{r}$. Let the particle be attracted to the origin by a force of size $mf(r)$, where $f(r)$ depends only on the distance r of the particle from the origin. According to Newton's second law the motion of the particle can be written

$$m\ddot{\mathbf{r}} = -mf(r)\hat{\mathbf{r}}, \tag{2.1}$$

where $\hat{\mathbf{r}}$ is the unit vector $\mathbf{r}/r$. The mass constant $m > 0$ can be removed from both sides and the problem is therefore independent of the mass of the particle. To reduce the problem to a first order system, we introduce the velocity denoted by $\mathbf{v}$, so that $\mathbf{v} = \dot{\mathbf{r}}$. The system can then be written as

$$\begin{aligned} \dot{\mathbf{r}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= -f(r)\hat{\mathbf{r}}. \end{aligned} \tag{2.2}$$

The set of equations (2.2) describes the motion of the particle in terms of $\mathbf{v}(t)$ and $\mathbf{r}(t)$.

This is a general formulation of the central force problem for any force. The focus here will however mostly be on particles under gravitational force. The

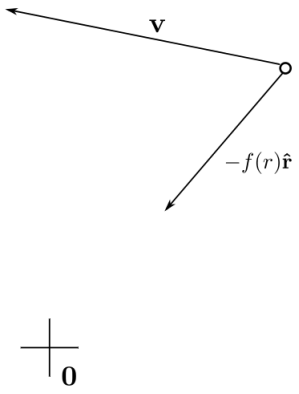

Figure 2.1: A particle in space being pulled towards the origin.

force function is then defined as $f(r) = \mu r^{-2}$, where μ is a positive constant depending on units and source of attraction. This is known as the inverse square law since the magnitude of the force is proportional to the inverse square of the distance.

We will now take a closer look at the properties of the gravitational central force problem. The presentation follows chapter one in [15].

2.2 Conservation of angular momentum

The first part of analyzing this problem is to find some of its mechanical properties.

Definition 2.1. *The angular momentum is the vector* $\mathbf{c} = m\tilde{\mathbf{c}}$*, where*

$$\tilde{\mathbf{c}} = \mathbf{r} \times \mathbf{v}. \tag{2.3}$$

Note that $\frac{d}{dt}(\mathbf{r} \times \mathbf{v}) = \mathbf{r} \times \dot{\mathbf{v}} + \dot{\mathbf{r}} \times \mathbf{v}$. Since a cross product of a vector with itself is zero, we have that $\dot{\mathbf{r}} \times \mathbf{v} = \mathbf{v} \times \mathbf{v} = 0$. According to eq. (2.2) $\mathbf{r} \times \dot{\mathbf{v}} = -f(r)(\mathbf{r} \times \hat{\mathbf{r}}) = 0$. The whole derivative is then zero and the vector $\tilde{\mathbf{c}}$ doesn't change.

Theorem 2.1. *The vector* $\tilde{\mathbf{c}}$ *is a constant of motion.*

Since $\tilde{\mathbf{c}}$ is constant over time and $\mathbf{r}$ and $\mathbf{v}$ are always perpendicular to $\tilde{\mathbf{c}}$ we get the following as well.

Theorem 2.2. *The whole motion of the particle takes place in a plane that has* $\tilde{\mathbf{c}}$ *as its normal vector and contains both* $\mathbf{r}$ *and* $\mathbf{v}$*.*

The whole motion then takes place in a plane so the vector functions $\mathbf{r}$ and $\mathbf{v}$ can be assumed to be two-dimensional. This reduces the problem from 6 dimensions in total to only 4.

Let the system be represented by polar coordinates so that $\mathbf{r} = [r\cos\theta, r\sin\theta, 0]$ and $\tilde{\mathbf{c}} = [0, 0, \tilde{c}]$. Then $\mathbf{v} = [\dot{r}\cos\theta - r\sin\theta\dot{\theta}, \dot{r}\sin\theta + r\cos\theta\dot{\theta}, 0]$ and it follows that

$$\begin{aligned}\mathbf{r}\times\mathbf{v} &= [0, 0, r\cos\theta(\dot{r}\sin\theta + r\cos\theta\dot{\theta}) - r\sin\theta(\dot{r}\cos\theta - r\sin\theta\dot{\theta})] \\ &= [0, 0, r^2\dot{\theta}].\end{aligned}$$

In general, integrating $\frac{1}{2}r^2\dot{\theta}$ gives the area described by $\dot{\theta}$ and r. The rate of the area swept by the radius is then $\tilde{c}/2$ and that's a constant. It follows that Kepler's second law is valid for the central force problem.

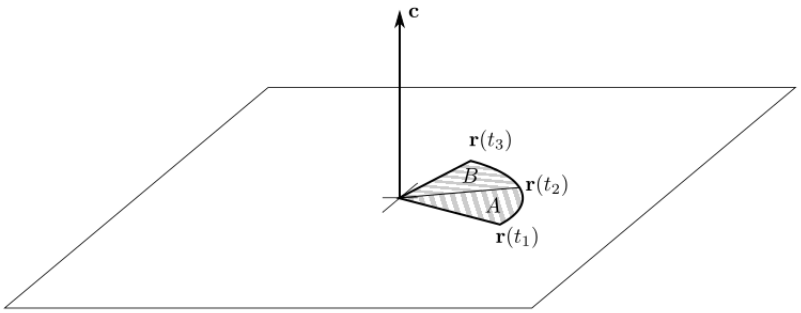

Figure 2.2: Particle moving in plane orthogonal to $\tilde{\mathbf{c}}$. If $t_2 - t_1 = t_3 - t_2$ then the areas A and B are equal.

2.3 Conservation of total energy

The system has more constants of motion. Now we shall take a look at the energy. Let's start with a definition.

Definition 2.2. *In classical mechanics $K = mv^2/2$ is known as the kinetic energy of the particle and $U = mF(r)$ as the self-potential, where $F(r)$ is the anti-derivative of the force function f. The negative of the self-potential $-U$ is known as the potential energy of the particle. The sum of kinetic and potential energy*

$$E = K - U \tag{2.4}$$

is known as the total energy.

If we take the dot product of eq. (2.2) with $\mathbf{v}$ we get

$$\begin{aligned}\dot{\mathbf{v}} \cdot \mathbf{v} &= -f(r)r^{-1}(\mathbf{r} \cdot \mathbf{v}) \\ &= -f(r)r^{-1}r\dot{r} \\ &= -f(r)\tfrac{dr}{dt}.\end{aligned}$$

Integration with respect to time yields

$$\frac{1}{2}v^2 = F(r) + h, \tag{2.5}$$

where h is a constant of integration. $F(r)$ is determined depending on the corresponding force. For gravitation it is

$$F(r) = \int_r^\infty f(x)dx = \frac{\mu}{r}.$$

Multiplication of eq. (2.5) with m gives the formula for the total energy of the system. Thus $K - U = mh$, and since h is a constant we get the following theorem.

Theorem 2.3. *The total energy E of the system is a constant of motion.*

In the case of the central force problem where there's only one mass, the constant h is often used to represent the energy instead of $E = mh$.

2.4 Existence and uniqueness

When working with differential equations it's important to consider the existence and uniqueness of a solution. Since the central force problem is time-reversible, it suffices to consider what happens forward in time. The inverse square force function $f(r) = \mu r^{-2}$ belongs to C^∞ on $\mathbb{R} \setminus \{\mathbf{0}\}$. It follows that the right hand side of (2.2) is locally Lipschitz. According to the theory of existence and uniqueness [2], we have the following result.

Theorem 2.4. *For every initial condition $(\mathbf{r}_0, \mathbf{v}_0) \in \mathbb{R}^6$ with $\mathbf{r}_0 \neq 0$ there exists a unique local solution to the central force problem.*

The above theorem doesn't guarantee that the solution is defined for all values of t and in fact this is not always the case (see the discussion at the end of section 2.5). However, one can give a useful criterion for determining whether or not the solution is global.

Theorem 2.5. *A solution to the central force problem has finite time of existence $t^* < \infty$ if and only if $r(t) \to 0$ as $t \to t^*$.*

Indeed, suppose that the maximal solution is only defined up to time $t^* < \infty$ but that $r(t) \not\to 0$ as $t \to t^*$. Then there exists a sequence $\{t_j\}$ with $t_j \to t^*$ as $j \to \infty$ and $r(t_j) > \epsilon$ for some $\epsilon > 0$. According to eq. (2.5) the corresponding speeds satisfy the bound

$$v(t_j) \leq \sqrt{2\left(h + \frac{\mu}{\epsilon}\right)}. \tag{2.6}$$

For every j, the central force problem can be solved locally with $(\mathbf{r}(t_j), \mathbf{v}(t_j))$ as initial data. The maximal time of existence of the solution can be bounded from below by a constant which depends only on μ, ϵ and h. Indeed, setting

$$R_0 = \{(t, \mathbf{r}, \mathbf{v})\colon |(\mathbf{r} - \mathbf{r}(t_j), \mathbf{v} - \mathbf{v}(\mathbf{t_j}))| \leq \epsilon/2\},$$

the solution is defined at least in the interval $|t - t_j| \leq \epsilon/(2B)$, where

$$\begin{aligned} B &= \sup_{R_0} |(\mathbf{v}, -\mu r^{-2}\hat{r})| \\ &\leq \sup_{R_0} \left(|\mathbf{v}(t_j)| + |\mathbf{v} - \mathbf{v}(t_j)| + |\mu r^{-2}\hat{r}|\right) \\ &\leq \sqrt{2\left(h + \frac{\mu}{\epsilon}\right)} + \frac{\epsilon}{2} + \frac{4\mu}{\epsilon^2} \end{aligned}$$

(see Theorem 1' in [2, p. 39]). Picking j high enough therefore allows us to continue the solution past t^* and we have a contradiction.

2.5 Eccentric axis of the orbit

Consider now the case of gravitational force, $f(r) = \mu r^{-2}$. To figure out the path of the solution we seek a differential equation for $\hat{\mathbf{r}}$. Since $r^2 = \mathbf{r} \cdot \mathbf{r}$ and $r\dot{r} = \mathbf{r} \cdot \dot{\mathbf{r}}$ it follows that

$$\begin{aligned} \frac{d}{dt}\frac{\mathbf{r}}{r} &= \frac{r\dot{\mathbf{r}} - \dot{r}\mathbf{r}}{r^2} \\ &= \frac{(\mathbf{r} \cdot \mathbf{r})\dot{\mathbf{r}} - (\mathbf{r} \cdot \dot{\mathbf{r}})\mathbf{r}}{r^3}. \end{aligned}$$

Here, the numerator is simply a triple cross product expansion and hence we get

$$\frac{d}{dt}\frac{\mathbf{r}}{r} = \frac{(\mathbf{r} \times \mathbf{v}) \times \mathbf{r}}{r^3} = \frac{\tilde{\mathbf{c}} \times \mathbf{r}}{r^3}. \tag{2.7}$$

Multiplying both sides with μ, this can be rewritten to

$$\mu\frac{d}{dt}\frac{\mathbf{r}}{r} = (-\mu r^{-2}\hat{\mathbf{r}}) \times \tilde{\mathbf{c}} = \dot{\mathbf{v}} \times \tilde{\mathbf{c}}.$$

Integration gives

$$\mu(\hat{\mathbf{r}} + \mathbf{e}) = \mathbf{v} \times \tilde{\mathbf{c}}, \tag{2.8}$$

where $\mathbf{e}$ is constant vector of the integration. It follows from eq. (2.8) that since $\hat{\mathbf{r}}$ is perpendicular to $\tilde{\mathbf{c}}$, so is $\mathbf{e}$. Therefore $\mathbf{e}$ is in the plane of motion. Now taking the dot product of $\mathbf{r}$ with eq. (2.8) gives

$$\mu(\mathbf{e} \cdot \mathbf{r} + r) = \mathbf{r} \cdot \mathbf{v} \times \tilde{\mathbf{c}} = \mathbf{r} \times \mathbf{v} \cdot \tilde{\mathbf{c}} = \tilde{\mathbf{c}} \cdot \tilde{\mathbf{c}} = \tilde{c}^2,$$

$$\mathbf{e} \cdot \mathbf{r} + r = \tilde{c}^2/\mu. \tag{2.9}$$

Go back to polar coordinates and this time define $\theta = 0$ when $\mathbf{r} \parallel \mathbf{e}$ and face the same direction. Then $\mathbf{e} \cdot \mathbf{r} = er\cos\theta$ and eq. (2.9) can be written as

$$r = \frac{\tilde{c}^2/\mu}{1 + e\cos\theta}, \tag{2.10}$$

which is the equation for a conic section.

Theorem 2.6. *The motion of the particle is in the form of a conic section.*

This should look familiar to the reader since theorem 2.6 is almost the same as Kepler's first law. Kepler only considered ellipses, which is correct for planets. In general however the other conic forms are possible solutions to the central force problem as well. For further information about conic sections, the reader may refer to appendix A. The vector $\mathbf{e}$ is therefore important in determining the motion of the particle.

Definition 2.3. *The vector* $\mathbf{e}$ *is known as the eccentric axis of the orbit and the length* e *as its eccentricity.*

A relationship between the total energy and the eccentricity can be found. Squaring eq. (2.8) gives

$$\mu^2(e^2 + \frac{2}{r}\mathbf{e}\cdot\mathbf{r} + 1) = v^2\tilde{c}^2.$$

Substituting $2h + (2\mu/r)$ for v^2 and $(\tilde{c}^2/\mu) - r$ for $\mathbf{e}\cdot\mathbf{r}$ gives

$$\mu^2(e^2 - 1) = 2h\tilde{c}^2.$$

If $h < 0$ then $e < 1$ and the orbit is elliptical, if $h = 0$ then $e = 1$ and the orbit is a parabola and finally if $h > 0$ then $e > 1$ and the orbit is a hyperbola. According to this, if an initial condition $\mathbf{r}_0$ and $\mathbf{v}_0$ of the particle at a time t_0 is known, then the path of the particle can be determined.

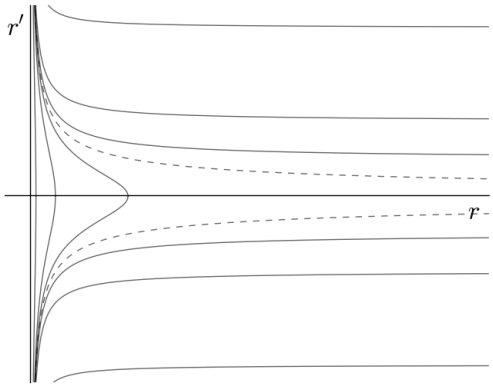

Figure 2.3: Change of position as a function of position for the 1D case. Dashed lines are for $h = 0$ where $r' \to 0$ as $r \to \infty$. When $h < 0$ the particle doesnt have escape velocity and turns back around towards zero.

There's one more case that hasn't been discussed so far and that is when $\tilde{c} = 0$. The velocity and the position vectors are then parallel. Putting this into equation (2.7), the right hand side becomes zero and so must the left hand side. Therefore $\hat{\mathbf{r}}$ is constant. This means that the problem becomes one-dimensional and the particle moves along a line. It also follows from equation (2.8) that

$\hat{\mathbf{r}} = -\mathbf{e}$ and therefore the length $e = 1$. Although it may not be obvious at first, this is a conic section as well, with the plane going through the origin and having the same slope as the line defining the cone ($e = 1$). Writing $\mathbf{r}(t) = -r(t)\mathbf{e}$, we find that r satisfies the equation

$$r''(t) = -\frac{\mu}{r(t)^2},$$

and from the conservation of energy we find that

$$\frac{(r'(t))^2}{2} = \frac{\mu}{r(t)} + h. \tag{2.11}$$

In the case $h = 0$ this equation can be solved explicitly, giving

$$r(t) = a(t - t^*)^{2/3}, \tag{2.12}$$

with $a = (18\mu/4)^{1/3}$. Note that the solution is only defined on one of the intervals $t > t^*$ and $t < t^*$, corresponding to a collision in the past or in the future. In general, the qualitative behavior of the solution can be seen by tracing out the curves defined by eq. (2.11). That can be found in figure 2.3. We find that in general the solution does not exist for all times, since either the particle comes from the origin, goes towards it or both.

2.6 Position on an elliptic orbit

So far we have the location of the particle as a function of the angle from the eccentric axis, but it would be preferable to have it as a function of time. A simple geometrical approach for the case where $h < 0$ and the path thus elliptic will be demonstrated here.

For this we will use eccentric anomaly u which is the angle to a point out from the center of the ellipse rather then from the origin. Formulating the ellipse out from u, so the the eccentric axis lines up with the x-axis, gives the equations

$$\begin{aligned} x &= a(\cos u - e) \\ y &= b \sin u. \end{aligned}$$

Differentiating both formulas with respect to time yields

$$\begin{aligned} \dot{x} &= -a\dot{u} \sin u \\ \dot{y} &= b\dot{u} \cos u. \end{aligned} \tag{2.13}$$

It's possible to use these terms to write the cross product, giving the angular momentum as before:

$$\begin{aligned} c &= x \cdot \dot{y} - y \cdot \dot{x} \\ &= a(\cos u - e) \cdot b\dot{u} \cos u - b \sin u \cdot (-a)\dot{u} \sin u \\ &= ab\dot{u}(\cos^2 u - e \cos u + \sin^2 u) \\ &= ab\dot{u}(1 - e \cos u). \end{aligned}$$

Since c is a constant, integration in time gives

$$ct = ab(u - e \sin u).$$

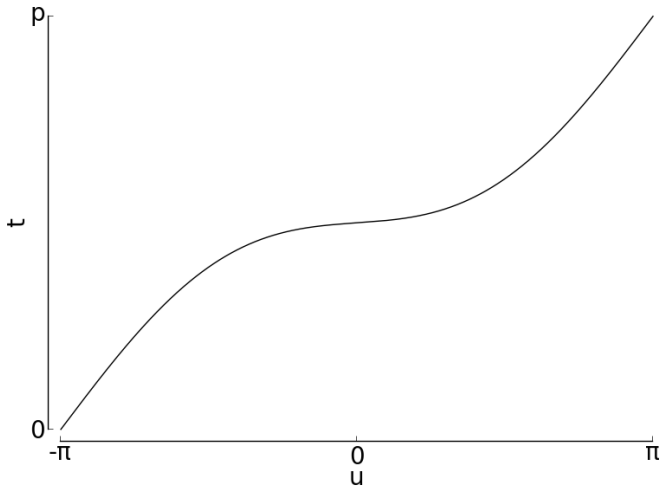

Figure 2.4: Kepler equation needs to be solved in order to find position for a time value.

and substituting $c = \sqrt{\mu p}$ and $b = \sqrt{ap}$, where $p = a(1 - e^2)$, we get

$$\sqrt{\frac{\mu}{a^3}}t = u - e\sin u.$$

Later we shall use the constant $n = \sqrt{\frac{\mu}{a^3}}$, known as the mean motion. It is useful to substitute into the equation a constant T, so that $t = 0$ doesn't have to be at perigee as defined in the formula above, but instead T will make up for the difference. That is,

$$n(t - T) = u - e\sin u.$$

Since t and u are known initially, it's straightforward to calculate T. This equation is usually known as Kepler's equation and a plot of it can be seen in figure 2.4. The main idea is to use it to find u out from a given t. This however isn't straightforward since the equation is implicit. A good way to approach this is by solving it numerically, using for example Newton's method. This is discussed further in chapter 4. Once u is known, the position on the orbit can be calculated from equations (2.13).

This method of finding Kepler's equation is simple and good for a first read. A different method of deriving the equation, that doesn't rely on the equations for an ellipse as here, can be found in [15]. As well, the cases where $h \geq 0$ can be found there.

Chapter 3

The N-body problem

3.1 Definition

Although the central force problem covered in chapter 2 can model a Newtonian system, it has limitations. It doesn't take into account that the particle also attracts it's source of attraction. For example, a planet attracts the sun even though the effect is relatively small. Also the central force problem only offers one source of attraction while in a more complex system, for example a solar system, it's necessary to account for the gravitational force from all the particles attracting each other. Here follows a definition of a more general model.

Consider N particles in space with masses m_i and location vectors $\mathbf{r}_i$ for $i = 1, ..., N$ respectively. They attract one another with a force. According to Newton's second law this system under gravitational force satisfies the set of equations

$$m_k \ddot{\mathbf{r}}_k = \sum_{\substack{j=1 \\ j \neq k}}^{N} \frac{G m_j m_k}{r_{jk}^2} \hat{\mathbf{r}}_{jk}, \quad k = 1, \ldots, n, \tag{3.1}$$

where G is a gravitational constant, $\mathbf{r}_{jk} = \mathbf{r}_j - \mathbf{r}_k$ is the relative location vector, r_{jk} its length and $\hat{\mathbf{r}}_{jk}$ the unit vector in the same direction. This is known as the N-body problem. Similar to the central force problem this system can be reduced to a first order system for the locations and velocities:

$$\begin{aligned}
\dot{\mathbf{r}}_1 &= \mathbf{v}_1 \\
&\vdots \\
\dot{\mathbf{r}}_N &= \mathbf{v}_N \\
\dot{\mathbf{v}}_1 &= \sum \frac{G m_j}{r_{j1}^2} \hat{\mathbf{r}}_{j1} \\
&\vdots \\
\dot{\mathbf{v}}_N &= \sum \frac{G m_j}{r_{jN}^2} \hat{\mathbf{r}}_{jN}.
\end{aligned}$$

Since each vector has 3 components this is a system of $6N$ equations in total.

For the central force problem the center of the system is the source of attraction. The N-body problem however doesn't have any specific center. For this we turn to a different method to define a center to locate at the origin.

Definition 3.1. *The expression*

$$M = \sum_{i=1}^{N} m_i$$

is known as the total mass and

$$\frac{1}{M} \sum_{i=1}^{N} m_i \mathbf{r}_i,$$

as the center of mass.

If all of the equations (3.1) are summed up, the right side cancels out and we therefore get

$$\sum_{i=1}^{N} m_i \ddot{\mathbf{r}}_i = 0.$$

By integrating twice we get the linear relation

$$\sum_{i=1}^{N} m_i \mathbf{r}_i = \mathbf{a}t + \mathbf{b}$$

where $\mathbf{a}$ and $\mathbf{b}$ are constant vectors.

Theorem 3.1. *The movement of the center of mass is linear.*

The linear movement of the center of mass can be removed from the system and the center of mass can be fixed at the origin of the coordinate system. This will be assumed to be the case from now on unless otherwise mentioned.

There are other methods of representing a mechanical system like this. A common one that will be used later is Hamiltonian mechanics. The Hamiltonian formulation is a description of the flow of the particles in the phase space of their locations and momenta. For Hamiltonian mechanics the momentum $m_i \mathbf{v}_i$ will be denoted by $\mathbf{p}_i$. The Hamiltonian is a function of position and momentum and is denoted by $H(\mathbf{r}, \mathbf{p})$ where $\mathbf{r}$ and $\mathbf{p}$ are the vectors of all the $\mathbf{r}_i$'s and $\mathbf{p}_i$'s. The Hamiltonian is given by the formula

$$H = \sum_{i=1}^{N} \frac{\|\mathbf{p}_i\|^2}{2m_i} - U,$$

where

$$U = \sum_{1 \le i < j \le N} \frac{G m_i m_j}{\|\mathbf{r}_i - \mathbf{r}_j\|} \tag{3.2}$$

is the self-potential. The equations of motion are then

$$\dot{\mathbf{r}}_i = \frac{\partial H}{\partial \mathbf{p}_i}, \quad \dot{\mathbf{p}}_i = -\frac{\partial H}{\partial \mathbf{r}_i}.$$

A simpler notation will be used for the partial derivatives such that

$$\frac{\partial H}{\partial \mathbf{p}_i} = H_{\mathbf{p}_i}$$

and similarly for the self-potential.

3.2 Conservation of energy, angular momentum and the moment of inertia

In the last section we already saw the conservation of linear momentum. Those six constants of **a** and **b**, the three of angular momentum and the one of total energy are usually referred to as the classical integrals of the N-body problem [13]. They are important for further insight into the problem.

Definition 2.1 is still valid except now there are multiple terms to sum up:

$$\mathbf{c} = \sum_{k=1}^{n} m_k(\mathbf{r}_k \times \mathbf{v}_k). \tag{3.3}$$

Differentiation gives

$$\frac{d}{dt}\sum_{k=1}^{n} m_k(\mathbf{r}_k \times \mathbf{v}_k) = \sum_{k=1}^{n} m_k((\mathbf{r}_k \times \ddot{\mathbf{r}}_k) + (\mathbf{v}_k \times \mathbf{v}_k)) = \sum_{k=1}^{n}\sum_{\substack{j=1 \\ j\neq k}}^{n} \frac{Gm_j m_k}{\mathbf{r}_{jk}^3}(\mathbf{r}_j \times \mathbf{r}_k).$$

The right side of the equation vanishes since each term comes twice with opposite signs.

Theorem 3.2. *The total angular momentum is a constant of motion.*

The conservation of the total energy follows directly from the Hamiltonian formulation of the system. Since we haven't seen the full properties of Hamiltonian systems here, let's show this. Knowing that

$$m_k\ddot{\mathbf{r}}_k = \frac{\partial U}{\partial \mathbf{r}_k}$$

it follows that

$$\sum_{k=1}^{n} m_k\dot{\mathbf{r}}_k \cdot \ddot{\mathbf{r}}_k = \sum_{k=1}^{n} \frac{\partial U}{\partial \mathbf{r}_k}\frac{d\mathbf{r}_k}{dt}$$

The right side is simply the time derivative of the potential and the left side can be rewritten as

$$\frac{d}{dt}\frac{1}{2}\sum_{k=1}^{n} m_k v_k^2 = \dot{U}.$$

The left side is then the time derivative of the kinetic energy $\dot{K}$. Integration yields the expected result $K = U + E$ where E is a constant of integration.

Theorem 3.3. *The total energy of the system, $E = K - U$, is a constant of motion.*

The moment of inertia is going to be used later. Dealing with point masses makes it easy to define.

Definition 3.2. *The moment of inertia for the system is*

$$I = \sum_{k=1}^{n} m_k r_k^2 = \sum_{k=1}^{n} m_k(\mathbf{r}_k \cdot \mathbf{r}_k). \tag{3.4}$$

Using the assumption that the center of mass is at the origin, one can rewrite the moment of inertia as

$$I = \frac{1}{M}\sum_{k<j} m_j m_k |\mathbf{r}_j - \mathbf{r}_k|^2 = \frac{1}{M}\sum_{k<j} m_j m_k r_{jk}^2, \tag{3.5}$$

thus allowing the moment of inertia to be determined by the mutual distances of the particles instead of the individual location vectors.

Differentiating eq. (3.4) gives

$$\begin{array}{rcl} \frac{1}{2}\dot{I} & = & \sum_{k=1}^n m_k r_k \dot{r}_k, \\ \frac{1}{2}\ddot{I} & = & \sum_{k=1}^n m_k(\dot{r}_k^2 + r_k\ddot{r}_k) = 2K + \sum_{k=1}^n r_k \frac{\partial U}{\partial r_k}. \end{array}$$

But U is homogeneous of degree -1 and therefore

$$\sum_{k=1}^n r_k \frac{\partial U}{\partial r_k} = -U$$

which gives us

$$\frac{1}{2}\ddot{I} = 2K - U.$$

Using the definition of the total energy this can be rewritten as

$$\frac{1}{2}\ddot{I} = K + E = U + 2E. \tag{3.6}$$

3.3 Existence and singularities

In chapter 2 the singularity of the inverse square force was considered. Unlike the central force problem however, the N-body problem doesn't necessarily have a singularity at the origin. Since the terms of the force function (3.1) each depend on the mutual distances between the particles, they have singularities when $r_{ij} = 0$. The simplest kind of singularity is a collision, defined as follows.

Definition 3.3. *A collision occurs at t^* when every position vector $\mathbf{r}_i$ converges to a value as $t \to t^*$ and some distinct $\mathbf{r}_i$ and $\mathbf{r}_j$ converge to the same value. A total collapse occurs when all the particles collide at the same time.*

To show the existence of a solution we wish to exclude collisions. Consider the smallest of the mutual distances

$$r_{\min} = \min(r_{12}, r_{13}, ..., r_{(n-1)n})$$

and let M be the sum of all the masses as before and m_0 the smallest of all the masses. Bounds for the equations of motion can then be obtained from the force equation (3.1) and the conservation of the total energy:

$$|H_{\mathbf{r}_i}| < G\left(\frac{M}{r_{\min}}\right)^2, |H_{\mathbf{p}_i}| < \sqrt{\frac{2}{m_0}\left(|E| + \frac{GM^2}{r_{\min}}\right)}.$$

Just as for the central force problem before, these bounds guarantee existence and uniqueness for the problem as long as $r_{\min} \not\to 0$.

Theorem 3.4. *A solution to the N-body problem has finite time of existence $t^* < \infty$ if and only if $r_{\min}(t) \to 0$ as $t \to t^*$.*

What makes this much more complicated than the central force problem is that $r_{\min} \to 0$ as $t \to t^*$ does not necessarily mean a collision. An oscillation between the particles defining $r_{\min}$ can cause a singularity without convergence, or a particle could shoot off to infinity in finite time. These scenarios are known as non-collision singularities or simply pseudo-collisions. We will get back to these later on, but first we consider the 2-body problem in more detail.

3.4 The 2-body problem

Start from the pair of equations given by (3.1)

$$\begin{aligned} m_1\ddot{\mathbf{r}}_\mathbf{1} &= -\frac{Gm_1m_2}{r_{12}^2}\hat{\mathbf{r}}_\mathbf{12} \\ m_2\ddot{\mathbf{r}}_\mathbf{2} &= -\frac{Gm_2m_1}{r_{21}^2}\hat{\mathbf{r}}_\mathbf{21} \end{aligned} \tag{3.7}$$

and let the center of mass be at the origin, as described above. This means that we have

$$m_1r_1 = m_2r_2, \quad m_1\mathbf{r}_1 + m_2\mathbf{r}_2 = 0,$$

which can be used to rewrite equations (3.7) in the form

$$\begin{aligned} \ddot{\mathbf{r}}_1 &= -\frac{Gm_2^3}{M^2r_1^2}\hat{\mathbf{r}}_1 \\ \ddot{\mathbf{r}}_2 &= -\frac{Gm_1^3}{M^2r_2^2}\hat{\mathbf{r}}_2 \end{aligned} \tag{3.8}$$

where $M = m_1 + m_2$. This makes the equations independent of one another.

Theorem 3.5. *The 2-body problem can be reduced to two central force problems.*

As we have already seen in chapter 2 this allows for the problem to be fully solved. Methods for solving the two-body problem will be discussed further in chapter 4. Note in particular that finite-time singularities in the two-body problem are always due to collisions [15].

3.5 Singularities for $N \geq 3$

As mentioned above, singularities can be more complicated when $N \geq 3$. We begin by considering the simplest type of singularities, a total collapse. All the particles must then converge to the origin, since that's where the center of mass is fixed. From the definition of I we see that $I \to 0$ as $t \to t^*$ is a necessary and sufficient condition for a total collapse.

We begin by investigating if a total collapse can happen when $t \to \infty$. Looking at the self-potential (3.2) we see that $U \to \infty$ when the distances between the particles go to zero. But then eq. (3.6) shows that $\ddot{I} \to \infty$ as well. According to this, from some time to infinity $\ddot{I} \geq 1$ and integrating gives $I \geq \frac{1}{2}t^2 + At + B$, where A and B are constants. Therefore $I \to \infty$ when $t \to \infty$. This contradicts that a total collapse will happen as $t \to \infty$.

Theorem 3.6. *A total collapse can only happen in finite time. That is, $I \to 0$ as $t \to \infty$ is impossible.*

The next thing to consider is if there are any restrictions on the system which prevents a collapse from occurring. From eq. (3.3) for the angular momentum we have

$$\begin{aligned} c &\leq \textstyle\sum_{k=1}^{n} m_k |\mathbf{r}_k \times \mathbf{v}_k| \\ &\leq \textstyle\sum_{k=1}^{n} m_k r_k v_k \\ &= \textstyle\sum_{k=1}^{n} (\sqrt{m_k} r_k)(\sqrt{m_k} v_k) \end{aligned}$$

By Cauchy's inequality

$$c^2 \leq \sum_{k=1}^{n} m_k r_k^2 \sum_{k=1}^{n} m_k v_k^2 = I(2K). \tag{3.9}$$

This is known as Sundman's inequality [15]. Now that we have the inequality, let's assume a collapse. Then $I \to 0$ as $t \to t_1$ for some finite t_1, since it can't happen when $t \to \infty$. Since $\ddot{I} \to \infty$ in finite time, there exists a time interval $t_2 \leq t \leq t_1$ so that $\ddot{I} > 0$. That means $\dot{I}$ is monotone on the time interval $[t_2, t_1]$ and since $I > 0$ as well and $I \to 0$ we know that $\dot{I} < 0$ for $t_2 \leq t \leq t_1$. Going back to Sundman's inequality (3.9), use eq. (3.6) to rewrite it and then multiply by $-\dot{I}$ gives

$$\frac{1}{2} c^2 (-\dot{I} I^{-1}) \leq E\dot{I} - \frac{1}{2} \dot{I}\ddot{I}.$$

Integration yields

$$\frac{1}{2} c^2 \log(I^{-1}) \leq EI - \frac{1}{4} \dot{I}^2 + A \leq EI + A$$

where A is constant of integration. Dividing through with $\log(I^{-1})$ gives

$$\frac{1}{2} c^2 \leq \frac{EI + A}{\log(I^{-1})}.$$

But if $I \to 0$ so does the right side of the equation above. So for a collapse to occur we must have $c = 0$. This is known as Sundman's theorem of total collapse [18].

Theorem 3.7. *For a total collapse to occur the angular momentum must be zero.*

We now specialize to the case of three particles with a singularity at time $t = t^*$ so that $r_{\min} \to 0$ as $t \to t^*$. According to eq. (3.6) $\ddot{I} \to \infty$ as $t \to t^*$ since $U \to \infty$ as $r_{\min} \to 0$. For a time sufficiently close to t^* we have that for the rest of the time to the singularity $\ddot{I} > 0$ and I must approach a limit. If $I \to 0$, then the singularity is a total collapse. If however $I \to D$ for some value $D > 0$ (including ∞), then from eq. (3.5) a lower bound for the longest mutual distances $r_{\max}$ can be found,

$$I \leq \frac{1}{M} \sum_{k<j} m_j m_k r_{\max}^2,$$

ensuring the bound $r_{\max} > A$ for some constant A depending on the masses. For a time relatively close to t^* so that $r_{\min} < A/2$, the triangle inequality

contradicts that two of the distances $r_{jk} < A/2$ and therefore guarantees that $r_{\min}$ stays defined by the same pair of particles. Consequently, the distance between these particles converges to 0 as $t \to t^*$. In order to show that there is a collision, we have to prove that the particles approach finite limits as $t \to t^*$. Note that one of the particles, say the third, must eventually be bounded away from the other two. The equations of motion then guarantee that

$$|\mathbf{r}_3''| \leq B$$

as $t \to t^*$ for some constant $B > 0$. Integration from t_1 to t_2 yields

$$|\mathbf{v}_3(t_2) - \mathbf{v}_3(t_1)| \leq B|t_2 - t_1|$$

as $t_1, t_2 \to t^*$. It follows from Cauchy's criterion that $\mathbf{v}_3(t)$ has a finite limit as $t \to t^*$ and by integration that $\mathbf{r}_3(t)$ has a finite limit $\mathbf{r}_3^*$. The normalisation

$$m_1\mathbf{r}_1 + m_2\mathbf{r}_2 + m_3\mathbf{r}_3 = \mathbf{0}$$

and the fact that $r_{12} \to 0$ now imply that

$$\mathbf{r}_1(t), \mathbf{r}_2(t) \to -\frac{m_3}{m_1 + m_2}\mathbf{r}_3^*$$

as $t \to t^*$.

Theorem 3.8. *All singularities of the three-body problem are collisions.*

Knowing this, the Finnish mathematician and astronomer Karl F. Sundman was able to solve the 3-body problem in terms of power series expansions. He assumed that the angular momentum was non-zero, leaving only the possibility of binary collisions. These binary collisions can be regularized by a change of variables. To explain this change of variables, consider the explicit collision solution $r(t) = a(t-t^*)^{2/3}$ in the central force problem discussed in section 2.5. This solution is completely regular in the new variable $s = (t-t^*)^{1/3}$, and $r(s) = as^2$, showing that the solution 'bounces off the origin'. Sundman's transformation has the same effect for any binary collision in the 3-body problem, making the particles bounce off one another at a collision. Sundman's power series solution was later generalized to $N > 3$ and $N = 3$ with $c = 0$ by Q. Wang [21, 22]. He had to avoid the question of singularities altogether, and simply attained a convergent power series valid over the maximal time of existence of the solution. Sundman's and Wang's power series solutions are not used in practice since they converge very slowly. Numerical methods have proved to be superior.

The method proving theorem 3.8 does not hold for more particles than 3. In 1992 Z. Xia found a solution that had a pseudo-collision for the case $N = 5$ [23] and this was later extended to $N > 5$. The case where $N = 4$ is however still unsolved [5]. The theory behind this goes beyond what is covered here.

3.6 Central configurations

Central configurations are considered important since they e.g. appear asymptotically in connection with collisions and unbounded growth as time goes to infinity [17]. Let's start by defining what is meant by a central configuration.

Definition 3.4. *The particles of the N-body problem are said to form a central configuration if the force acting on each particle is in proportion to its mass and position vector relative to the center of mass. A formulation of this in mathematical terms is*

$$\frac{\partial U}{\partial \mathbf{r}_i} = -\lambda m_i \mathbf{r}_i \tag{3.10}$$

where λ is a scalar that is independent of i.

Note that the right side of eq. (3.10) is $-\lambda/2$ times the gradient of the moment of inertia I. It follows that $(\mathbf{r}_1, \ldots, \mathbf{r}_N)$ is a central configuration if and only if it is a critical point of U subject to the constraint of fixed I. Since U is homogeneous of order -1 it follows that

$$-U = \sum_{i=1}^{n} \mathbf{r}_i \cdot \frac{\partial U}{\partial \mathbf{r}_i} = -\lambda \sum_{i=1}^{n} m_i r_i^2 = -\lambda I.$$

and therefore

$$\lambda = U/I. \tag{3.11}$$

Furthermore, since $I_{\mathbf{r}_i} = 2m_i\mathbf{r}_i$ eq. (3.10) can be rewritten as

$$IU_{\mathbf{r}_i} = -\frac{1}{2} U I_{\mathbf{r}_i},$$

or better yet

$$(IU^2)_{\mathbf{r}_i} = 0. \tag{3.12}$$

Since I is homogeneous of order 2 and U of order -1 it follows that the equation above is invariant under dilations. The moment of inertia can be represented by eq. (3.5), so it and the self-potential only depend on the mutual distances of the particles. Therefore a rotation of a central configuration is also a central configuration.

Theorem 3.9. *A rotation and scaling of a central configuration is a central configuration.*

We will count two central configurations which only differ by a rotation and a scaling as one and the same. Be aware that some authors don't distinguish between two different configurations which are related by a reflection.

To start out with a simple example, one can think of N bodies, all of equal weight, uniformly distributed on a circle around the origin. It's easy to see for symmetric reasons that the particles form a central configuration. It's also possible to find non-flat central configurations. For example there are the platonic solids. The simplest one is a regular tetrahedron, again with equal masses. Every particle is above an equilateral triangle right in the middle and attracted by equal force from all the corners of it. Again simple geometrical vector math shows that we have a central configuration.

Another simple source of central configurations is the two-body problem. Note from equations (3.7) and (3.8) that

$$\frac{\partial U}{\partial \mathbf{r}_i} = -\lambda m_i \mathbf{r}_i, \quad i = 1, 2,$$

with

$$\lambda = \frac{Gm_2^3}{M^2 r_1^3} = \frac{Gm_1^3}{M^2 r_2^3} = \frac{GM}{r_{12}^3}.$$

Thus a solution of the two body problem forms a central configuration at any point in time. According to the counting method we discussed before, the two-body problem has only one central configuration since all configurations are simply a rotation or scaling of every other.

This is not the case in general for the N-body problem since a central configuration can occur in one moment but be gone the next. In order for a solution to remain within the same equivalence class of central configurations, the system is only allowed to rotate or scale. Consider first the case of a pure scaling. The positions are then given by $\mathbf{r}_i(t) = \phi(t)\mathbf{a}_i$, where the $\mathbf{a}_i$ are fixed vectors and ϕ is a real-valued function. We obtain the equation

$$\ddot{\phi} = -\frac{\lambda\phi}{|\phi|^3}, \tag{3.13}$$

for ϕ, while the $\mathbf{a}_i$ have to solve the equations (3.10). Equation (3.13) is the one-dimensional central force problem discussed in section 2.5. In particular, this kind of solution leads to a total collapse either in the future or in the past (or both).

Assume next that the $\mathbf{a}_i$ are coplanar and identify them with complex numbers. There is nothing which prevents us from allowing ϕ to be complex-valued. Since multiplication with a complex number geometrically corresponds to a rotation followed by a scaling, the solution will again remain within the same equivalence class of central configurations. This method allows us to construct a solution of the N-body problem for every solution of the central force problem (recall that these are all coplanar) and every central configuration. The special case when ϕ is a pure rotation is known as relative equilibrium. The particles then move uniformly in circles. The name is due to the fact that in a rotating coordinate system the solution becomes an equilibrium point.

Although arranging few particles so they form a central configuration can be easy, the general question of finding all relative equilibria for the N-body problem and whether or not they are finitely many has not been solved. The mathematician Steve Smale, who was awarded the Fields medal in 1966, considered this a problem important enough to be included among many famous problems like Poincaré's conjecture and the Riemann hypothesis on his list of problems worthy for this century [19]. This list was his response to a request from the International Mathematical Union in the late 20th century. The case $N = 3$ has however been solved and those solutions will be covered in the next two sections.

3.7 Euler's solutions

In a search for central configurations to the three-body problem, Euler found the first one. We seek three particles of any finite masses, in a collinear configuration. Call the positions of the particles along the line ξ_1, ξ_2 and ξ_3. Equation

(3.10) then reduces to

$$G\sum_{j\neq k}^{n}\frac{m_j}{r_{jk}^3}(\xi_j-\xi_k)=-\lambda\xi_k, \tag{3.14}$$

with $r_{jk}=|\xi_j-\xi_k|$. Arranging the numbers of the particles so they are ordered, that is $\xi_1<\xi_2<\xi_3$, let us write out the distances between the particles as $\xi_2-\xi_1=a$ and $\xi_3-\xi_2=a\rho$ and then $\xi_3-\xi_1=a(1+\rho)$. The fact that the center of mass is at the origin gives us the equations

$$\begin{array}{rcl} m_2a+m_3a(1+\rho) & = & -M\xi_1 \\ m_1a(1+\rho)+m_2a\rho & = & M\xi_3 \end{array} \tag{3.15}$$

where M is the total mass as before. Combining equations (3.14) and (3.15) for ξ_1 gives

$$a^3=M\frac{G}{\lambda}\frac{m_2+m_3(1+\rho)^{-2}}{m_2+m_3(1+\rho)}. \tag{3.16}$$

This gives a if ρ is known. To find ρ we calculate $-\xi_1/\xi_3$ with (3.14) and (3.15). This gives

$$\frac{m_2+m_3(1+\rho)}{m_1(1+\rho)+m_2\rho}=\frac{m_2+m_3(1+\rho)^{-2}}{m_1(1+\rho)^{-2}+m_2\rho^{-2}}$$

which can be written out as the polynomial equation

$$\begin{aligned}(m_2+m_3)+(2m_2+3m_3)\rho+(3m_3+m_2)\rho^2 \\ -(3m_1+m_2)\rho^3-(3m_1+2m_2)\rho^4-(m_1+m_2)\rho^5=0.\end{aligned} \tag{3.17}$$

Putting $\rho=0$ gives a positive left side and $\rho\to\infty$ gives $-\infty$ so there must be a root on the positive half axis. Furthermore, according to Descartes' rule of signs there is at most one root [15]. Hence there is exactly one central configuration with this ordering of the masses. Interchanging the roles of m_1 and m_3 will simply produce a configuration which is rotated 180° around the origin. The only way to produce a non-equivalent configuration with the same masses is to change which mass is in the middle. We have thus proved the following result.

Theorem 3.10. *Given three masses, there are exactly three collinear central configurations of the three body problem.*

Euler's solution has been extended for all values of N. According to Moulton [14] it is possible to find $N!/2$ collinear central configurations for any set of N (distinct) masses.

3.8 The Lagrange solutions

So far we have found all collinear central configurations of the three-body problem. What other solutions can exist? Lagrange found the answer to this question.

One way of tackling this question would be to formulate the problem in a similar way as the collinear case in last section. Note that three points in space always lie in a common plane, so that the configuration has to be coplanar.

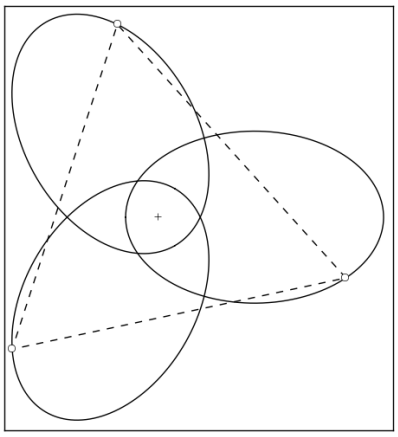

Figure 3.1: Triangular central configuration with all distances equal.

We would therefore have to solve a complex-valued version of eq. (3.14). We refer to [15] for this approach. Instead we shall follow [13] and use the mutual distances r_{12}, r_{13}, r_{23} as new variables. These might seem like too few independent variables, since after fixing the center of mass at the origin there are four independent variables left. However, the mutual distances are in fact local coordinates near any non-collinear configuration if we identify two configurations which differ by a rotation. Indeed, after a rotation and a reflection we can assume that m_1 and m_2 lie on a horizontal line in the lower half plane and that m_1 lies to the left of m_2. Given the position of m_1, the distance r_{12} determines the position of m_2 and the distances r_{13} and r_{23} the position of m_3. Together the points determine a triangle, which is so far only unique up to translations (the position of m_1). Fixing the center of mass at the origin however determines the triangle and the positions of the masses uniquely. Eq. (3.5) will therefore be used for the momentum of inertia, while the self-potential is already expressed in terms of the mutual distances. Differentiating U and I with respect to the three r_{ij}'s gives

$$G\frac{m_i m_j}{r_{ij}^2} = 2\lambda\frac{m_i m_j r_{ij}}{M}$$

and solving for r_{ij} gives

$$r_{ij} = \sqrt[3]{\frac{GM}{2\lambda}}$$

Since λ is independent of particle pair and G and M are constants, all the mutual distances must be equal.

Theorem 3.11. *There exist exactly two non-collinear central configurations in the three body problem and those are the two possible equilateral triangles.*

These being the only possible non-collinear configurations, we have found all possible central configurations of the three-body problem.

This theory has been tested in astronomy. If one side of a triangle is defined between the Sun and Jupiter, then asteroids can be found in the vertices of triangles forming in both directions. This was first discovered in 1906 by the astronomer Max Wolf (1863–1932) and by now thousands of 'trojan' asteroids have been found in these locations. Similar cases have been found elsewhere in our solar system, e.g. in relation with the Sun and Mars [12].

Chapter 4

Numerical approximations

4.1 Introduction

As we have seen so far the range of known analytical solutions to the N-body problem is quite limited. For many applied problems in science, no such solutions exist. For those problems we turn to numerical approximations. Instead of finding a function or infinite series of functions that satisfy the system of equations (3.1), the idea is to estimate the numerical value of the system at a future time for a given initial value. For this we use the first order system

$$\mathbf{y}'(t) = \mathbf{g}(t, \mathbf{y}), \tag{4.1}$$

where

$$\mathbf{y}(t) = \begin{bmatrix} \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_N \\ \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_N \end{bmatrix} \quad \mathbf{g}(t, \mathbf{y}) = \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_N \\ \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_N \end{bmatrix} \tag{4.2}$$

and

$$\mathbf{f}_k = \sum_{\substack{j=1 \\ j \neq k}}^{n} \frac{Gm_j}{r_{jk}^2} \hat{\mathbf{r}}_{jk}, \quad k = 1, ..., n. \tag{4.3}$$

Starting from a initial value problem $\mathbf{y}_0$, t_0 the derivative is evaluated and the solution is moved forward in time. But time is continuous so that between any two given values t_k and t_{k+1} are infinitely many other values and it's impossible to evaluate them all. For this reason the time variable must be discretized so that at time t_k the values are moved to $t_{k+1} = t_k + h$ for some given time step h. This means that information is lost in between the time steps and a discretization error is introduced into the calculation. A numerical estimation at a time value t_k is denoted with $\mathbf{y}_k$. Another limitation of numerical computations on computers is the representation of numbers. In a typical floating point representation only a finite number of digits are considered. The error that this causes is referred to as rounding error.

An important condition for a numerical method to be acceptable, is that it converge towards the correct solution. That is, $\mathbf{y}_i \to \mathbf{y}(t_i)$ as $h \to 0$. If

the method doesn't converge then it's completely unpredictable and useless. A wide range of methods for solving ordinary differential equations exist. Some are general purpose while other are specifically constructed for certain set of problems. One way to measure the accuracy of a method is to consider its rate of convergence.

Definition 4.1. *A method is said to be of order p if*

$$\mathbf{y}_k - \mathbf{y}(t_k) = \mathcal{O}(h^{p+1})$$

for every analytic right hand side $\mathbf{f}$ *and every k [11].*

As the order of the method is higher, the order of the remaining error is higher. Be aware though that higher order method does not guarantee lower error.

In the following sections are implementations of four methods and some analysis using the theoretical knowledge of the problem we have seen so far. All coding is done in the Python programming language and executed using Python 2.7.3. The code can be found in appendix B. Numerical values are represented as standard 64-bit floating point numbers. Since the gravitational constant only depends on the units chosen we will use $G = 1$ in all calculations.

4.2 Euler's method

The simplest method to solve a set of differential equations is Euler's method (EM). The idea is to move the values in the direction of the derivative by the time step h. Or to put it into a formula

$$\mathbf{y}_{k+1} = \mathbf{y}_k + h\mathbf{g}(t_k, \mathbf{y}_k). \tag{4.4}$$

We shall denote the error of a method with $\mathbf{e}_{k,h} = \mathbf{y}_k - \mathbf{y}(t_k)$. Assuming that $\mathbf{g}$ is analytic, from equation (4.4) and by Taylor's theorem it's possible to get for the error $\mathbf{e}_{k+1,h}$ at time step $k+1$:

$$\mathbf{e}_{k+1,h} = \mathbf{e}_{k,h} + h[\mathbf{g}(t_k, \mathbf{y}(t_k) + \mathbf{e}_{k,h}) - \mathbf{g}(t_k, \mathbf{y}(t_k))] + \mathcal{O}(h^2)$$

with $k = 0, 1, ..., \lfloor t^*/h \rfloor$ for a given $t^* > t_0$. It follows that

$$|\mathbf{e}_{k+1,h}| \leq (1 + hL)|\mathbf{e}_{k,h}| + ch^2$$

where L is a Lipschitz constant. By induction we reach that

$$|\mathbf{e}_{k,h}| \leq \frac{c}{L}h[(1 + hL)^k - 1].$$

Since hL is positive it holds that $(1 + hL)^k < e^{khL}$. Taking $k = \lfloor t^*/h \rfloor$ gives then $e^{\lfloor t^*/h \rfloor hL} \leq e^{t^* L}$ and thus

$$|\mathbf{e}_{k,h}| \leq \frac{c}{L}(e^{t^* L} - 1)h$$

It follows that $\max_k |\mathbf{e}_{k,h}| \to 0$ as $h \to 0$ for k such that $0 \leq kh \leq t^*$ [11]. We therefore conclude that Euler's method is convergent.

The above comparison of Euler's method with Taylor series tells us that the method is of order one.

Theorem 4.1. *Euler's method is of order one.*

This is the lowest possible rate at which a method can converge to a solution and therefore not much can be expected from it. The main purpose of including this method is that if another method does even worse, then it's probably not worth considering further.

4.3 Heun's method

The Runge-Kutta methods are based on numerical integral procedure know as Gaussian quadrature [11]. This is a large family of both implicit and explicit methods. Two explicit methods of this family will be discussed here.

The first one is a two step Runge-Kutta method known as Heun's method (HM) [10]. The formulation is as follows:

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \frac{h}{2}(\mathbf{Y}'_1 + \mathbf{Y}'_2), \tag{4.5}$$

with the stage derivatives

$$\begin{array}{rcl} \mathbf{Y}'_1 & = & \mathbf{g}(t_k, \mathbf{y}_k) \\ \mathbf{Y}'_2 & = & \mathbf{g}(t_k + h, \mathbf{y}_k + h\mathbf{Y}'_1). \end{array} \tag{4.6}$$

A detailed discussion of Runge-Kutta methods can be found in [8]. In particular we have following result.

Theorem 4.2. *Heun's method is of order two.*

Using this one also obtains the global estimate

$$|\mathbf{e}_{k,h}| \leq \frac{c}{L}(e^{t^* L} - 1)h^2.$$

See [8].

4.4 Classical fourth-order scheme

The second Runge-Kutta method that will be use here is a four step Runge-Kutta method which is often referred to as the classical fourth-order scheme (CFS) [10]. The method is:

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \frac{h_k}{6}(\mathbf{Y}'_1 + 2\mathbf{Y}'_2 + 2\mathbf{Y}'_3 + \mathbf{Y}'_4), \tag{4.7}$$

with

$$\begin{array}{rcl} \mathbf{Y}'_1 & = & \mathbf{g}(t_k, \mathbf{y}_k) \\ \mathbf{Y}'_2 & = & \mathbf{g}(t_k + \frac{h_k}{2}, \mathbf{y}_k + h_k \frac{\mathbf{Y}'_1}{2}) \\ \mathbf{Y}'_3 & = & \mathbf{g}(t_k + \frac{h_k}{2}, \mathbf{y}_k + h_k \frac{\mathbf{Y}'_2}{2}) \\ \mathbf{Y}'_4 & = & \mathbf{g}(t_k + h_k, \mathbf{y}_k + h_k \mathbf{Y}'_3). \end{array} \tag{4.8}$$

The following theorem can be obtained using the theory in [8]. As before this leads to a global bound of order $\mathcal{O}(h^4)$.

Theorem 4.3. *The classical fourth-order scheme method is of order four.*

It has been shown that this method is the highest order of Runge-Kutta method where the order is equal to the number of stage derivatives. All higher order methods need a number of stage derivatives beyond the order [11].

4.5 Störmer-Verlet

The methods we have seen so far are general ODE solvers that can be applied to most problems. The method we shall take a look at now however is specifically constructed for solving force problems. Here it will be referred to as Störmer-Verlet method (SV), but it's known under many other names as well. The reason for so many names is that it has been rediscovered over and over again and introduced by many under different names. As a matter of fact, it can be found in Newton's Principia where he used it for integration to show that Kepler's second law holds for the central force problem [7].

Going back to the notation of locations and velocities by $\mathbf{y}$, we write out the method as follows

$$\begin{aligned} \mathbf{v}_{k+\frac{1}{2}} &= \mathbf{v}_{k-\frac{1}{2}} + h\mathbf{f}(\mathbf{r}_k) \\ \mathbf{r}_{k+1} &= \mathbf{r}_k + h\mathbf{v}_{k+\frac{1}{2}}. \end{aligned} \tag{4.9}$$

An initial half step is needed to get started

$$\mathbf{v}_{k+\frac{1}{2}} = \mathbf{v}_k + \frac{h}{2}\mathbf{f}(\mathbf{r}_k) \tag{4.10}$$

which can as well be used along the calculation in order to align the position and velocity for the time variable. To find the error we use Taylor's theorem as before

$$\mathbf{r}(t_{k+1}) - \mathbf{r}_{k+1} = \mathbf{r}(t_k) - \mathbf{r}_k + h(\mathbf{v}(t_k) - \mathbf{v}_{k+\frac{1}{2}}) + \frac{h^2}{2}\mathbf{f}(\mathbf{r}(t_k)) + \mathcal{O}(h^3).$$

By eq. (4.10) we can rearrange it to

$$\mathbf{r}(t_{k+1}) - \mathbf{r}_{k+1} = \mathbf{r}(t_k) - \mathbf{r}_k + h(\mathbf{v}(t_k) - \mathbf{v}_k) + \frac{h^2}{2}(\mathbf{f}(\mathbf{r}(t_k)) - \mathbf{f}(\mathbf{r}_k)) + \mathcal{O}(h^3)$$

and as before, a global error bound can be found which leads to the following conclusion:

Theorem 4.4. *Störmer-Verlet is of order two.*

This does however not hold in general and applying this method to other problems than force problems can mean that it won't converge to the correct solution [7].

It's remarkable to have method of order two that only needs a single evaluation of the right hand side for each step, but that's not all. Störmer-Verlet has more characteristics that makes it feasible for problems like the N-body problem. Those are symmetry, reversibility, symplecticity and volume preservation [7].

The method is said to be symmetric, since changing the direction of time will result in the exact same method. A property to the symmetry is that reversing the velocity will not change the solution trajectory. So by turning time around the method will go back to the exact same values as it came from.

The flow φ_t of a Hamiltonian system is said to be symplectic because it satisfies

$$\varphi_t'^T J \varphi_t' = J, \quad J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix} \tag{4.11}$$

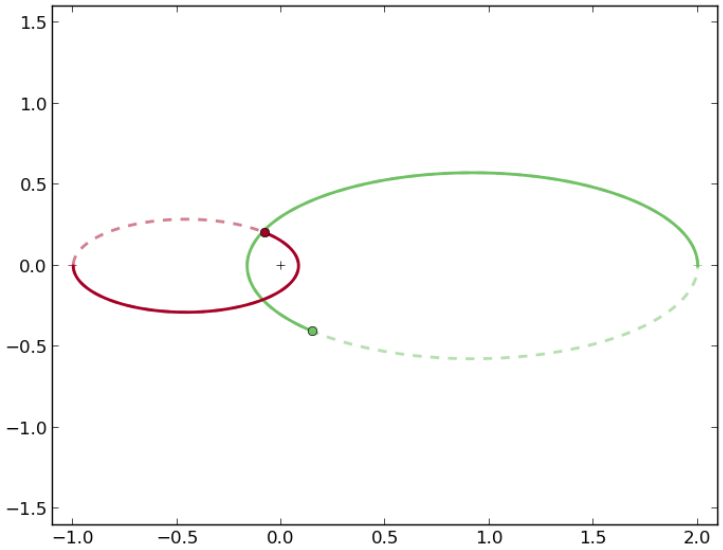

Figure 4.1: Short time interval for the two-body problem. The plot shows the system from its start to end value.

where I is the identity matrix. Störmer-Verlet is said to be a symplectic integrator since when applied to a symplectic system, property 4.11 holds (see theorem 2.2 in [7]). From the symplecticity it follows that Störmer-Verlet is volume preserving in phase space when applied to a Hamiltonian system.

One more important property of the Störmer-Verlet method is that it preserves linear first integrals [7]. A first integral is a non-constant function $I(\mathbf{y})$ for a differential equation, if $I(\mathbf{y}(t))$ is constant for every solution $\mathbf{y}(t)$. As we have seen in section 3.2 the N-body problem has a few. A big change of the classical integrals can mean big change for the system, for example if the energy changes sign or the angular momentum goes to zero. It can therefore be assumed that preserving those constants through out a calculation can be important.

4.6 The 2-body problem

Numerical methods are normally used where exact solutions aren't known or the numerical method can reach high enough accuracy with less effort than calculating the actual solution. We will however start here by studying problems whose solutions we already know exactly and compare the approximations to the exact solution. This will allow us to investigate the quality of the methods and make sure they produce numerical solutions as expected. The first system will be the two-body problem.

As we already seen in section 3.4, the two-body problem can be solved as two central force problems. Starting with the masses and initial values for the

positions and velocities, the first step is to fix the center of mass at the origin. The position $\mathbf{r}_c$ and velocity $\mathbf{v}_c$ of the center of mass can be calculated from

$$\begin{aligned}\mathbf{r}_c &= \tfrac{1}{M}(m_1\mathbf{r}_1 + m_2\mathbf{r}_2),\\ \mathbf{v}_c &= \tfrac{1}{M}(m_1\mathbf{v}_1 + m_2\mathbf{v}_2).\end{aligned}$$

By reducing those from the corresponding components the system stays the same but is fixed at the origin. The angular momentum can be calculated directly from equation (2.3), the μ's are

$$\mu_1 = \frac{Gm_2^3}{M^2}, \quad \mu_2 = \frac{Gm_1^3}{M^2}$$

and $\mathbf{e}$ can then be found from equation (2.8). That is all one needs to calculate a position on the orbit given a value of the eccentric anomaly. If one however wishes to find a corresponding time value or solve Kepler's equation, the mean motion $n = \sqrt{\frac{\mu}{a^3}}$, the angle of the true anomaly for the initial values $u_0 = \arcsin(\sqrt{\mu a}e\mathbf{r}_0 \cdot \mathbf{v}_0)$ and the time offset constant $T = -n^{-1}(u_0 - e\sin u_0)$ are needed as well. The calculation of u_0 can have a large numerical error due to the arcsin. To avoid that, the system considered here will have initial data on the x axis with perpendicular velocities. The eccentric axis and initial positions will then be parallel and the initial angle u_0 will either be 0 or $\pm\pi$.

That's all the information needed to solve for the ellipses. For the ellipses the major axes are

$$a_i = \frac{\tilde{c}_i^2}{\mu_i|e_i^2 - 1|}, \quad i = 1, 2$$

and the centers are

$$\mathbf{C}_i = -\mathbf{e}_i a_i, \quad i = 1, 2.$$

The minor axes are then

$$b_i = \sqrt{a_i^2 - |\mathbf{C}_i|^2}$$

and the ellipses can be calculated from a given u.

The system that will be used here for comparison has the following initial values

body nr.	m	r_x	r_y	v_x	v_y
1	2	-1	0	0	-0.13
2	1	2	0	0	0.26

that will give the following constants

body nr.	$\tilde{c}$	e_x	μ
1	0.13	0.8479	$0.\overline{11}$
2	0.52	-0.8479	$0.\overline{88}$

furthermore $u_0 = -\pi$.

The first calculation is a bit more than half a rotation to the value $u = 0.25\pi$. The expected trajectories can be seen in figure 4.1. The step size is varied to see how the methods converges towards the correct solution of the position as the step size gets smaller. In figure 4.2 we see how the methods converge at different rates due to their different orders. At a certain point the limit of the

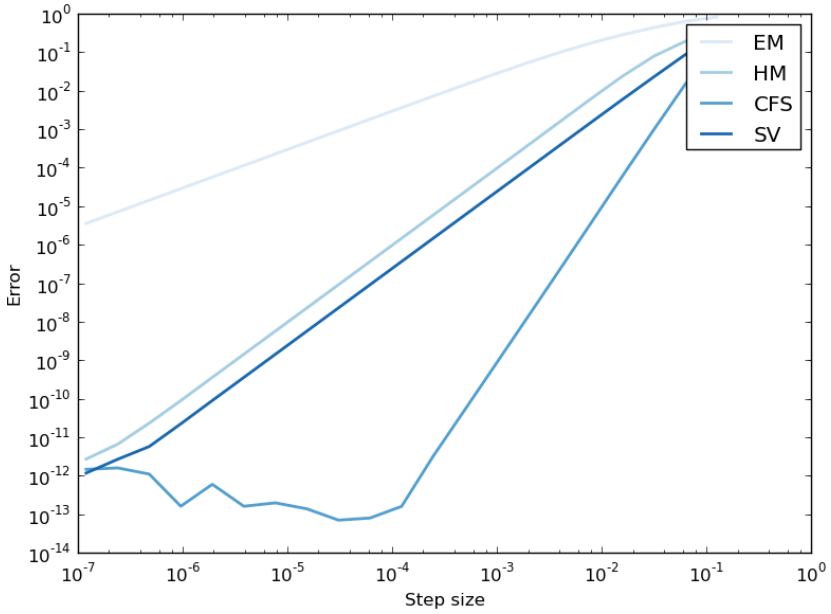

Figure 4.2: Short time interval for the two-body problem. The error for each corresponding method for different step sizes.

number representation on the computer is hit, the error is completely dominated by the rounding error. A further reduction in step size h will not result in a more precise solution.

As expected, Euler's method is the least accurate and converges slower than the other methods. Störmer-Verlet does a bit better than Heun's method and they converge at the same rate while CFS is the most accurate. Comparing algorithms based on step size alone however is not enough. A single step has different computational cost depending on what method is used. The most expensive part of the computation is the evaluation of the force in the right hand side of the equation. Since each step for Störmer-Verlet takes one evaluation but Heun's method takes two and CFS takes four it's to be expected that they take a longer time by the same factor. Timing the calculations confirms this. A calculation with fixed step size that took Störmer-Verlet 1 hour, took Heun's method 1 hour and 54 minutes and CFS 3 hours and 42 minutes. Therefore it's possible to do a calculation with Störmer-Verlet with about twice as many steps as Heun's method and about four times as many steps as CFS in approximately the same calculation time. Considering the error data in figure 4.2 shows that Störmer-Verlet would do better than CFS up to about the error 10^{-3} for the same calculation time. If higher accuracy than 10^{-3} is required CFS turns out to be the most efficient method of the ones we considered.

To better see how well the methods hold the accuracy, another calculation is done for a longer distance. The numerical solution is now evolved to 60 whole periods and then to $u = 0.8\pi$ so almost 61 circles ($t = 456.388..$). This time Euler's method fails to converge what so ever. The Runge-Kutta methods

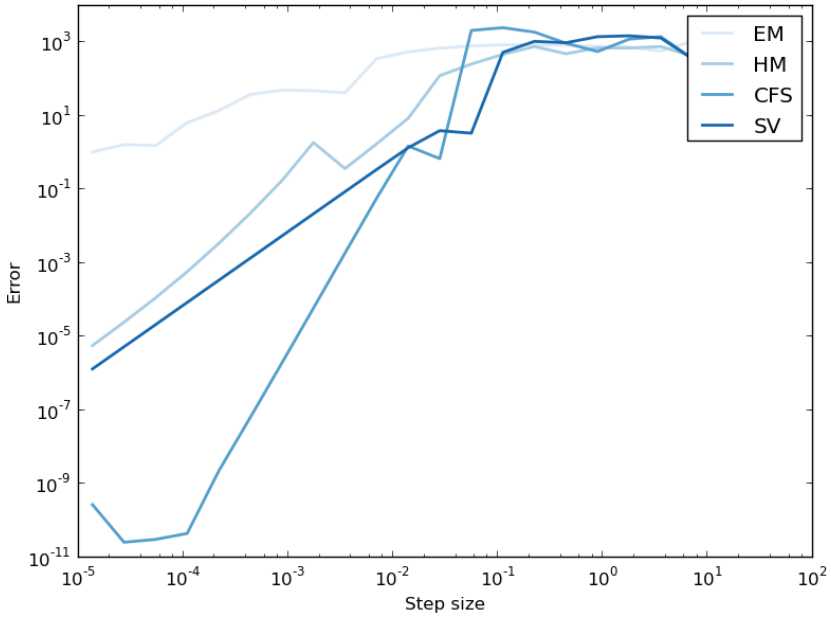

Figure 4.3: Error in the long term calculation of the two-body problem for different step sizes.

contain more noise than before and converge at an unexpected faster rate. Once the time step is small enough Störmer-Verlet however behaves exactly as before. For the step sizes tested only CFS reached rounding error. Comparing the error from the short approximation and the longer one, for a fixed step size the error grows faster for CFS than it does for Störmer-Verlet. Even so, for this example CFS eventually results in a higher accuracy for the same computational cost.

To investigate further we take a look at the total energy of the system throughout a run. From this point forward Euler's method will be excluded. The step size for Störmer-Verlet is 0.0015, for Heun's method it's 0.003 and for classical fourth-order scheme it's 0.006 so that the calculations take about the same time. As seen in figure 4.4 the energy of the system grows when Heun's method is used, shrinks for classical fourth-order scheme but is steady for Störmer-Verlet although it fluctuates for every period the system makes. This can have effect on the final outcome as the simulation gets longer.

4.7 Collinear configuration

The second system we consider is a three-body collinear configuration. In order to construct a collinear configuration, equation (3.17) needs to be solved for one of the variables. With three masses given one can solve the equations with a numerical method. Since we are not interested in any specific given masses it's easier to take two masses and ρ, then solve explicitly for the third mass.

With the masses and ρ known, the particles can be arranged on the x-axis

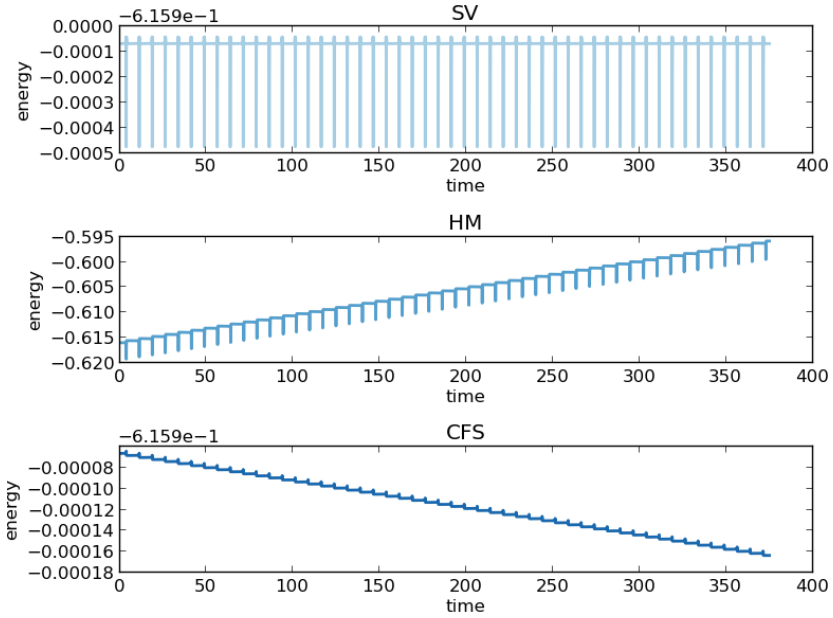

Figure 4.4: Behavior of the energy for three different methods. SV takes jumps at close encounters but returns to the same value, HM slowly grows while for CFS the energy reduces.

and with a given initial angular velocity, the ellipses can be found with the special cases of μ's. From equation (3.16), we get

$$\lambda = M\frac{G}{a^3}\frac{m_2 + m_3(1+\rho)^{-2}}{m_2 + m_3(1+\rho)}, \tag{4.12}$$

where a is as defined in section 3.7. Using equations (3.15) and that the center of mass is at the origin, a can be swapped for the corresponding distances from the center

$$\begin{aligned} a &= r_1 M/(m_2 + m_3(1+\rho)) \\ &= r_2 M/(m_1 - m_3\rho) \\ &= r_3 M/(m_1(1+\rho) + m_2\rho). \end{aligned}$$

Replacing a with a certain r_i in equation (4.12) gives the corresponding μ_i. Once the μ's are known, the problem can be solved for each particle in the same way as the two-body problem in the previous section.

In many ways the three-body collinear configuration is similar to the two-body problem. They are both coplanar central configurations and have periodic solutions. In figure 4.6 are the trajectories of particles aligned in a collinear configuration. The method used is Heun's method with time step $\frac{1}{20}$. As we can see the trajectories at first hold reasonable resemblance to the correct solution but then looses configuration and the particles shoot of into separate directions.

Unlike the measurements of error in last section, our interest here will be to see how long an algorithm can hold the form of a collinear configuration. The

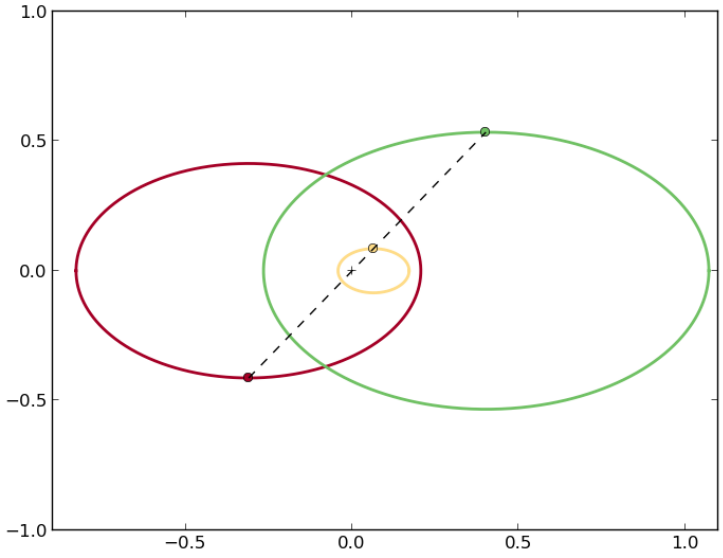

Figure 4.5: An example of a solution to the collinear configuration.

experiment will be conducted by running an approximation with different step sizes until an error of 0.5% has entered ρ. The calculation will be stopped and the current time registered. This means we only monitor the central configuration and not if it scales up or down or even slows down or speeds up. The values used to construct the system are $m_1 = 1$, $m_2 = 2$, $\rho = .9$, scaling of the system just 1 and the angular velocity 1.2. The third mass is then $m_3 = 0.45500326264274077$. A single period of the solution is about 1.62982149..

The result can be seen in figure 4.7. It might look a bit different from what one expects. Decreasing the step size doesn't seem to make any difference and if something it's making it worse. To gain insight into what's going on there one might consider figure 4.8 where a big time step is used. Even though the system grows and goes away from the actual trajectories the central configuration is held and the system keeps going in a rotating motion.

4.8 A system of 5 particles

The final test is a random system of five bodies in three dimensions. A few systems were created and the most interesting one was picked. The step size for Runge-Kutta 2 is 2^{-7}, for Runge-Kutta 4 2^{-6} and for Störmer-Verlet 2^{-8}. The trajectories of the particles in the calculation done with the classical fourth-order scheme can be seen in figure 4.9. Since we don't really know what the actual solution looks like we'll consider the total energy and see how the methods behave compared to what we've seen so far.

In figure 4.10 we can see that all of the methods eventually have a big

energy jump caused by a close encounter of two particles. The added on energy results with one or more particle reaching escape velocity when it shouldn't and shooting away from the others. Since the energy, as we already know should be constant this behavior should not be expected from the actual system and therefore should be unacceptable to take place in the calculations.

Before anything like that happens it is possible to observe a similar pattern as we've seen before. In the zoomed in plot in figure 4.10, we can see the energy behavior at a close encounter. For Heun's method the energy has already grown and the close encounter doesn't have any visual effect on the energy. Classical fourth-order scheme however dives down and looses energy while Störmer-Verlet takes a dip but then returns to the same level. This appears to be similar to the behavior we saw in figure 4.4 for the two body problem.

One way to avoid a blow up of the system is to make the step size smaller and get a more accurate approximation. In figure 4.11 the energy is plotted for three different step sizes with Heun's method. The step sizes are 2^{-8}, 2^{-9} and 2^{-10}. All of them clearly get further without a blow up than the original one in figure 4.10. A close encounter occurs around $t = 17.5$ that blows up for the step size 2^{-9}. In the case of the bigger step size 2^{-8}, the system has already had some minor growth that leads to the close encounter not being as close. This leads to the bigger step size actually getting further without a blow up. With the smallest step size 2^{-10}, the method manages to approximate all the close encounters that it fails on with the other step sizes and get the furthest before the system blows up at about $t = 31$.

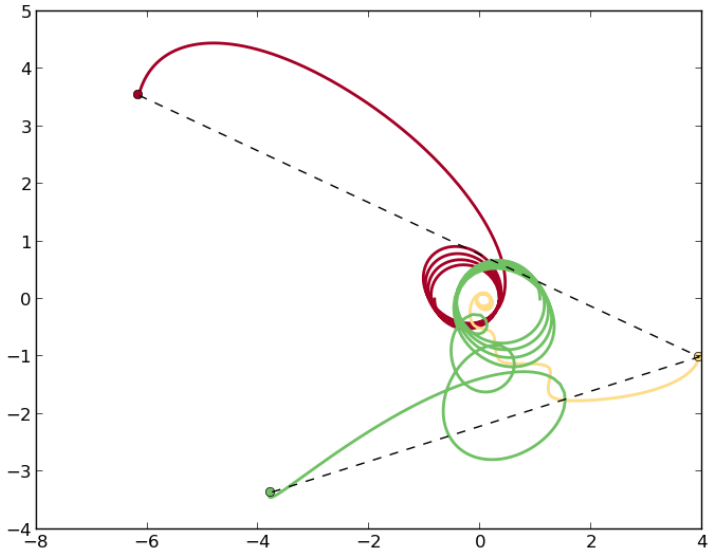

Figure 4.6: Collinear configuration, calculation over long period. Once error has entered the ratio of the distances between the particles, the original configuration is lost and the particles go of in different directions.

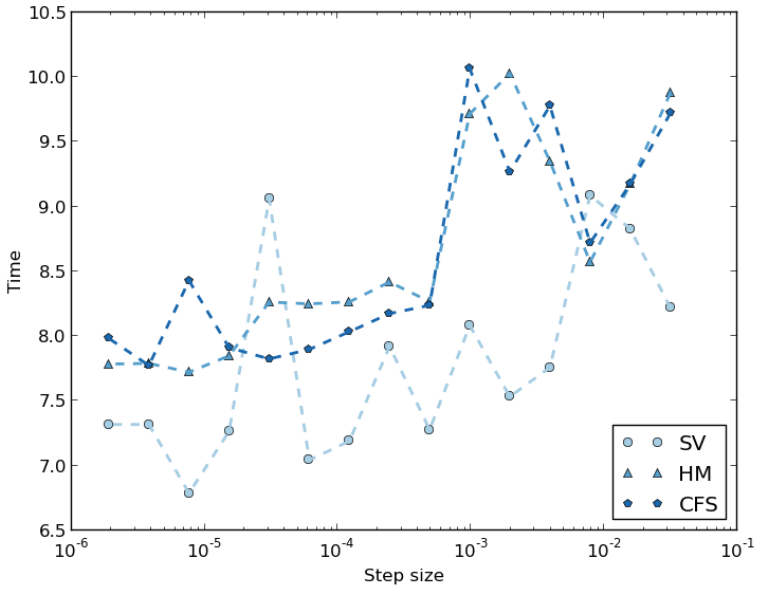

Figure 4.7: The time until the calculation methods loose the central configuration of the system for different step sizes (log scale).

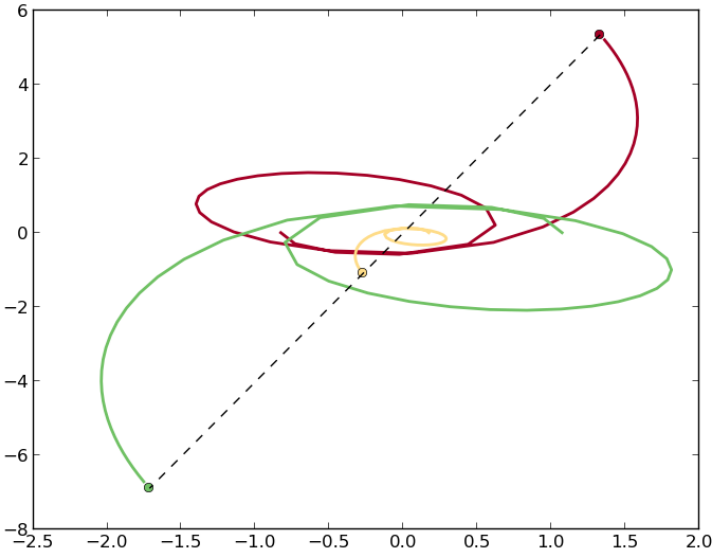

Figure 4.8: A big step size with Heun's method on the central configuration. The system grows and the trajectories spiral outwards but still the central configuration is held.

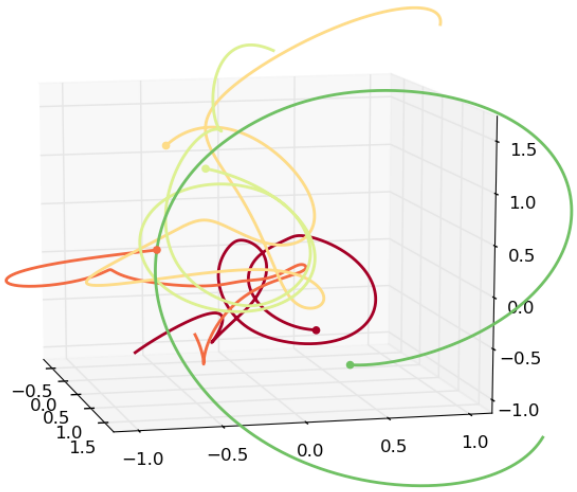

Figure 4.9: The five body system as approximated by the classical fourth-order scheme.

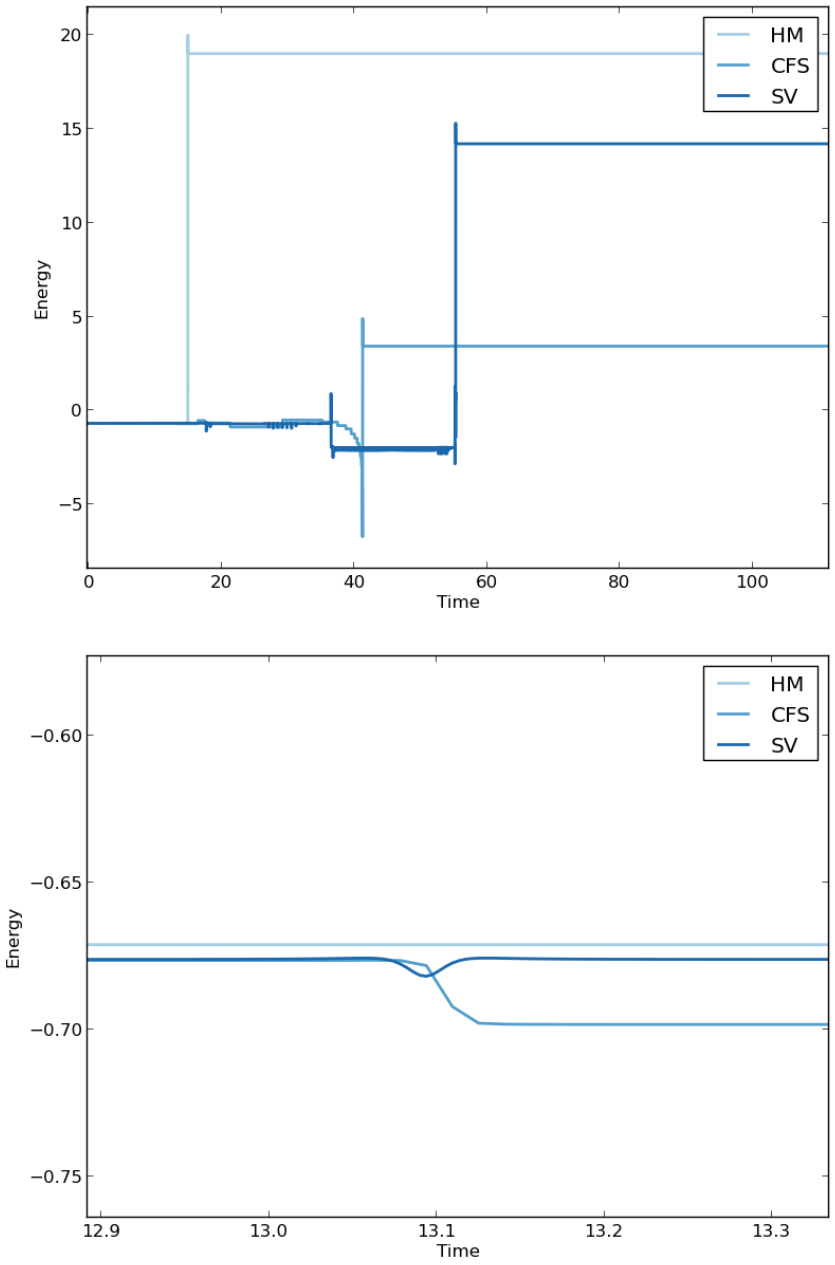

Figure 4.10: Behavior of the energy for five particles. First one for long term showing that all of the methods end up eventually with positive energy and the second one a zoom in on a close encounter.

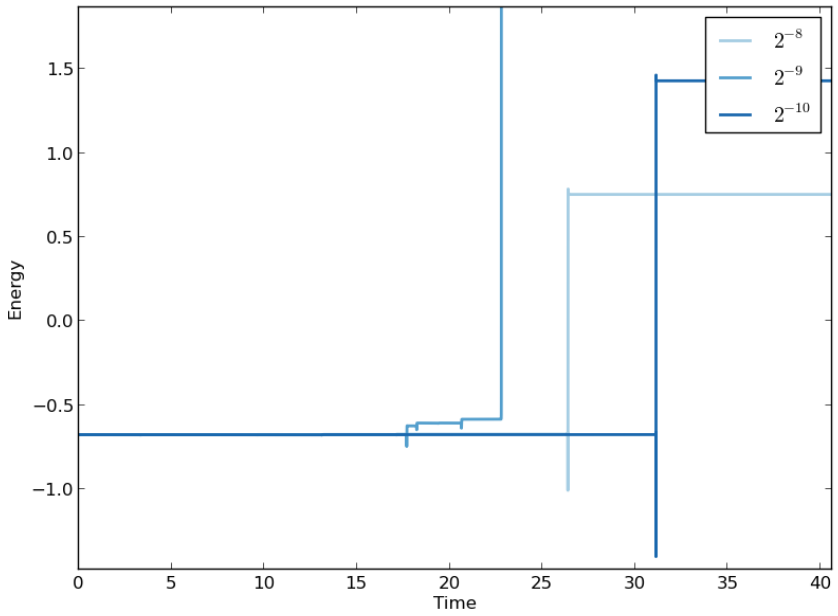

Figure 4.11: Energy of five particles for three different step sizes of Heun's method. A blow up happens at different times for different step sizes.

Chapter 5

Conclusion

5.1 What we've seen and what we haven't

So far we have covered the foundation for the study of the N-body problem. We've seen the simpler system, the central force problem and some solutions. The range of other known solutions is limited but there are some that haven't been discussed here. The study of moons traveling around planets resulted in the restricted three body problem. In that case it's assumed that one of the three particles (the moon) has so little weight that it does not affect the other particles. The two bigger ones then act as a two body problem while the third one spins around one of them [15]. Another three body system that has been studied in more recent times is where all three bodies travel by the same path in the form of a figure eight. Quite recently a categorized list was published that contains some periodic three body problems that have been discovered [20]. As was mentioned in chapter 3 there has been some interest in central configurations. A detailed study can be found on periodic solutions [18] and on the general case of central configurations [17]. As Saari points out [16], the nature of the central configurations makes it possible for a non-expert in the field to bring new insights towards a solution.

The system as it was considered here was only in the form of gravitation as Newton introduced it. Other form of it exist, for example in molecular dynamics with the Lennard-Jones potential [6], where the particles bounce away from one another when they get too close. As well when it comes to astronomy we don't have to go further away than inside our own solar system to observe affects of Einstein's theory of relativity. Measurements of Mercury's position does not agree with calculations [12]. In modern time it has become standard practice for some mathematicians to tackle the problems not only for the inverse square force but rather the general case of inverse force. That is, solve the problem for the force $\frac{1}{r^k}$ for some $k > 0$ [17].

When solving the equations numerically the decision on method and step size usually comes down to the accuracy needed against the time that the calculation will take. All the systems that were tested here are fairly small and simple but with every added particle the calculation time grows fast. Depending on the type of system there are possibilities to improve the methods used. One observation that can be made from the tests above is that at close encounters

the step size needs to be relatively small for the system not to blow up. At the same time when the particles are far apart there's not much going on and the step size can be bigger. For this it's possible to use methods with adaptive step size. For the Runge-Kutta 4 method discussed in last chapter there is a method known as Runge-Kutta 34 that with one extra stage derivative evaluation for each step can compare the solution by a third order Runge-Kutta and the fourth order method and from that estimate an appropriate step size for the next step [8]. Similar approach for a geometrical integrator like Störmer-Verlet is difficult since the current step size in Runge-Kutta 34 is evaluated on past data and that would break the properties of Störmer-Verlet. An approach to make geometrical integrators adaptive has been introduced by Hairer & Söderlind [9] that applies to Störmer-Verlet.

Adaptive step size is not always enough. Take for example a solar system with a sun at the center with planets rotating and moons going around the planets. Compared to the sun-planet rotation the moons are always at close encounters with the planets. In order for the moons not to shoot of into the distance the step size needs to be small enough but then a lot of calculation time is wasted on the movement of the planets that not necessarily improves the accuracy that much. A solution to this problem is discussed by Aarseth [1] by using predictor-corrector methods allowing for independent step sizes.

Appendix A

Conic sections

Due to connection of the theory in this paper with conic section this appendix is included as an overview. This text will not go into detail in proofs and the theory but more just go over some of it's properties. This material has been studied in detail for a long time and it's possible to find a more detailed description in most calculus text books.

A *cone* can be defined as a line going through the origin at an angle α and turned around an axis and forming a surface. A *conic section* is then intersection between a plane and a cone forming a curve. There are three different conic sections depending on the slope of the plane. If the slope is less than of the line defining the cone, then the section forms an ellipse, if the slope is equal it's parabola and if it's more, then the section is hyperbola. Note that a circle is a special case of an ellipse where the plane is parallel to the axis of rotation discussed before and therefore it's slope zero.

There are many ways of representing the conic sections, for example can most cases be reduced down to a quadratic formula of two variables [3]. Another definition uses points known as foci. Defining two points F_1 and F_2, then the set of all points in a plane where the sum of the distances d_1 and d_2 from these points is constant forms an ellipse. Similarly a set where the absolute value of the difference of d_1 and d_2 is constant forms a hyperbola. The parabola has only one focal point and a line called directrix. Then the parabola is a set of all points that have the same distance from the point and the shortest distance from the line. The relationship between the conic sections, foci points and cones is shown in the ice-cream-cone proof. It is omitted here. It can be found on page 499 of Apostol, Tom M. Calculus: volume I.

When working with angular momentum and a point of attraction as it is in the theory of this paper, it's good to use polar coordinates out from a focal point. To construct the formula for that we start by considering a single focal point and a directrix line as before. Then a conic section is a constant ratio between the distance from the focal point and the closes point of the directrix. This constant ratio is called eccentricity and is noted by the letter e. Lets call the location of a point X the directrix L and the focal point F. Then this can be formulated as

$$\|X - F\| = ed(X, L) \tag{A.1}$$

where $d(X, L)$ is the shortest distance form a point X to a line L. We need a way to calculate $d(X, L)$. That can be done by taking a vector from X to any

point P on the directrix and take the dot product with a unit vector normal N to the directrix. That is,

$$d(X, L) = |(X - P) \cdot N|$$

and then equation (A.1) becomes

$$\|X - F\| = e|(X - P) \cdot N|$$

Now fix P so that it's the point on the directrix closest to the focal point and the distance between them d. Then $F - P = dN$ and equation A can be rewritten as

$$\|X - F\| = e|(X - F) \cdot N - d|$$

and to simplify further, F can be defined at the origin and removed from the equation and $X = \mathbf{r}$ as used in the paper and $\|X\| = r$ so it becomes

$$r = e|\mathbf{r} \cdot N - d|$$

Further it's possible to remove the dot product $\mathbf{r} \cdot N$ and get

$$r = e|r \cos\theta - d|$$

Solving for r gives then

$$r = \frac{ed}{e \cos\theta + 1} \tag{A.2}$$

and a polar coordinate equation has been found for the conic sections.

Well aware reader might have noticed that e appears in the numerator of equation A.2 but doesn't in equation 2.10. This means for the case that the conic section is a circle ($e = 0$) equation A.2 becomes 0. It is thus easy to see that when $e \to 0$ and $d \to \infty$ the section becomes circular. It's however obvious that eq. A.2 is unnecessarily complicated to find a circle since the radius of a circle is constant, but that fits eq. 2.10, that is when $e = 0$ it becomes

$$r = \frac{c^2/\mu}{1 + 0 \cos\theta} = c^2/\mu$$

that is, a constant over time.

The ellipse

So far the discussion has been generally about the conic sections but now the elliptical case will get a special attention.

The ellipse is the only conic section that is a closed curve. That can be seen from the eccentricity in equation A.2. The ellipse can also be represented by a quadratic formula

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$

where a is the semi-major axis and b is semi-minor axis. That is, a is the furthest distance from the center c of the ellipse and b the shortest. In this representation this happens on the coordinate axes so that the longest distance is at $(\pm a, 0)$ and shortest at $(0, \pm b)$ in cartesian coordinates given that c is at the origin. A

relationship between the eccentricity of the polar coordinate and the minor and major axes of the cartesian coordinates exists

$$\begin{aligned} a &= ed/(1-e^2) \\ b &= \sqrt{a^2(1-e^2)} \end{aligned}$$

and the focal points are at a distance $f = ae$ from the center along the x-axis. The cartesian coordinates of a point X on the ellipse can then be found with

$$\begin{aligned} x &= a\cos E \\ y &= b\sin E \end{aligned}$$

where E is an angle from the center with the x-axis. If the origin is at a focal point and not the center then the equation for the x coordinate changes to $x = a(\cos E \pm e)$.

Appendix B

Code

two_body.py

```
import numpy as np
import scipy.optimize

class Tbp:

    def __init__(self, name, r1, r2, v1, v2, m1, m2):
        self.r = np.vstack((r1,r2))
        self.v = np.vstack((v1,v2))
        self.m = np.array([m1,m2],dtype='float')
        self.M = self.m.sum()
        self.name = name

        self.reduce_r()
        self.reduce_v()
        self.c = np.cross(self.r,self.v)
        self.totC = self.c[0] * self.m[0] + self.c[1] * self.m[1]

        self.mu = np.array([self.m[1]**3 / (self.M**2),
                            self.m[0]**3 / (self.M**2)])
        self.e = np.empty((2,3))
        for i in xrange(2):
            self.e[i] = (np.cross(self.v[i],self.c[i]) / self.mu[i]
                         - self.r[i] / np.linalg.norm(self.r[i]))

        self.a = np.empty(2)
        self.f = np.empty(2)
        self.b = np.empty(2)
        self.center = np.empty((2,3))
        for i in xrange(2):
            self.a[i] = np.sign(self.e[i,0]) * \
                        np.linalg.norm(self.c[i])**2 / \
                        (self.mu[i] *
```

```
                    abs(np.linalg.norm(self.e[i])**2 - 1))
            self.f[i] = np.linalg.norm(self.e[i]) * self.a[i]
            self.b[i] = np.sign(self.e[i,0])*np.sign(self.c[i,2]) * \
                        np.sqrt(self.a[i]**2 - self.f[i]**2)
            self.center[i] = -self.e[i] * abs(self.a[i])

        self.p = 2 * np.pi * np.sqrt(abs(self.a[0]**3)) / \
                 np.sqrt(self.mu[0])
        self.n = np.sqrt(self.mu[0]) / np.sqrt(abs(self.a[0]**3))
        if np.sign(self.r[0,0]) == np.sign(self.e[0,0]):
            self.E = 0
        else:
            self.E = np.pi

        self.T = (self.E - np.linalg.norm(self.e[0]) *
                  np.sin(self.E)) / self.n
        if np.dot(self.v[0], self.e[0]) > 0:
            self.T *= -1

        self.t = np.array([.5 * self.m[0] * (norm(self.v[0])**2),
                           .5 * self.m[1] * (norm(self.v[1])**2)])
        self.u = np.array([self.m[1] * self.m[0]**3 * (self.M**-2) /
                           norm(self.r[0]),
                           self.m[0] * self.m[1]**3 * (self.M**-2) /
                           norm(self.r[1])])
        self.h = self.t - self.u
        self.tot_h = self.h.sum()

    def get_y(self,dim = 2):
        return np.hstack((self.r[:,:dim].flat,self.v[:,:dim].flat))

    def reduce_r(self):
        rc = (self.m[0] * self.r[0] + self.m[1] * self.r[1]) / self.M
        self.r -= rc

    def reduce_v(self):
        vc = (self.m[0] * self.v[0] + self.m[1] * self.v[1]) / self.M
        self.v -= vc

    def sol(self, u, i):
        return np.array([self.a[i] * np.cos(u) + self.center[i,0],
                         self.b[i] * np.sin(u) + self.center[i,1]]).T

    def get_r(self,u):
        return np.hstack((self.sol(u,0),self.sol(u,1)))

    def solTime(self, t):
        return scipy.optimize.newton(lambda u: self.n * (t - self.T) -
                                     u + np.linalg.norm(self.e[0]) *
                                     np.sin(u),
```

```
                                        self.n * (t - self.T) ,
                                        lambda u:
                                        np.linalg.norm(self.e[0]) *
                                        np.cos(u) - 1,
                                        tol = 1.48e-12, maxiter = 1000)

    def tFromU(self, u):
        return (u - norm(self.e[0]) * np.sin(u)) / self.n + self.T
```

collinear.py

```
#!/usr/bin/env python
#coding: utf8

import numpy as np
import numpy.linalg as la
import scipy.optimize

class Collinear:

    """
    Create a collinear three-body system out of initial data and
    masses. Make sure that the center of mass is fixed at the origin.
    """

    def __init__(self,name, m1,m2,rho,scale,angVel):
        """
        Initial data is two masses and rho, third mass is then
        calculated to fit.
        """
        self.name = name
        self.scale = scale
        self.rho = rho

        self.m = np.array([m1,m2,0])
        self._find_m3()
        self.M = self.m.sum()

        self.r = np.zeros((3,3))
        self.r[0,0] = - (self.m[1] + self.m[2] * (1 + self.rho))
        self.r[1,0] = self.m[0] - self.m[2] * self.rho
        self.r[2,0] = self.m[0] * (1 + self.rho) + self.m[1] * \
                      self.rho
        self.r = self.r / self.M * scale

        self.v = np.zeros((3,3))
        self.v[:,1] = self.r[:,0]* angVel * scale
        self.c = np.cross(self.r,self.v)
```

```
        self.mu = self._findMu()
        self.e = np.empty((3,3))
        for i in xrange(3):
            self.e[i] = (np.cross(self.v[i],self.c[i]) / self.mu[i] -
                         self.r[i] / np.linalg.norm(self.r[i]))

        self.a = np.empty(3)
        self.f = np.empty(3)
        self.b = np.empty(3)
        self.center = np.empty((3,3))
        for i in xrange(3):
            self.a[i] = np.sign(self.e[i,0]) * \
                        np.linalg.norm(self.c[i])**2 / \
                        (self.mu[i] *
                         abs(np.linalg.norm(self.e[i])**2 - 1))
            self.f[i] = np.linalg.norm(self.e[i]) * self.a[i]
            self.b[i] = np.sign(self.e[i,0]) * np.sign(self.c[i,2]) \
                        * np.sqrt(self.a[i]**2 - self.f[i]**2)
            self.center[i] = -self.e[i] * abs(self.a[i])

        self.p = 2 * np.pi * np.sqrt(abs(self.a[0]**3)) / \
                 np.sqrt(self.mu[0])
        self.n = np.sqrt(self.mu[0]) / np.sqrt(abs(self.a[0]**3))
        if np.sign(self.r[0,0]) == np.sign(self.e[0,0]):
            self.E = 0
        else:
            self.E = np.pi

        self.T = (self.E - np.linalg.norm(self.e[0]) *
                  np.sin(self.E)) / self.n
        if np.dot(self.v[0], self.e[0]) > 0:
            self.T *= -1

    def _find_m3(self):
        m = self.m
        rho = self.rho
        num = (3*m[0] + m[1])*rho**3 + (3*m[0] + 2*m[1])*rho**4 + \
              (m[0] + m[1])*rho**5 - m[1] - 2*m[1]*rho - m[1]*rho**2

        den = (1 + 3*rho + 3*rho**2)

        self.m[2] = num / den

    def get_y(self):
        return np.hstack(self.r.flat,self.v.flat)

    def _findMu(self):
        mu = np.empty(3)

        m1m2 = self.m[0] * self.m[1]
```

```
        m2m3 = self.m[1] * self.m[2]
        m1m3 = self.m[0] * self.m[2]
        lambdConst = (self.M * (m1m2 + m2m3/self.rho +
                      m1m3 / (1+ self.rho)) /
                     (m1m2 + m2m3*self.rho**2 + m1m3 *
                      (1 * self.rho)**2))

        a2r = self.M / np.array([self.m[1] + self.m[2] *
                                 (1 + self.rho),
                                 self.m[0] - self.m[2] * self.rho,
                                 self.m[0]*(1+self.rho) +
                                 self.m[1]*self.rho])

        for i in xrange(3):
            mu[i] = lambdConst / a2r[i]**3

        return mu

    def sol(self, u, i):
        """
        For a given particle i (0. 1 or 2) and
        an angle u, return the solution.
        """
        return np.array([self.a[i] * np.cos(u) + self.center[i,0],
                         self.b[i] * np.sin(u) + self.center[i,1]]).T

    def solTime(self, t):
        """Approximate angle u for a given time t."""
        return scipy.optimize.newton(lambda u: self.n *
                                               (t - self.T) - u +
                                               np.linalg.norm(
                                               self.e[0]) *
                                               np.sin(u),
                                     self.n * (t - self.T) ,
                                     lambda u: np.linalg.norm(
                                               self.e[0]) *
                                               np.cos(u) - 1,
                                     tol = 1.48e-12, maxiter = 1000)

    def solTimeVec(self, t):
        """Vectorize solTime(t)."""
        u = np.empty_like(t)
        for i in xrange(len(t)):
            u[i] = self.solTime(t[i])
        return u
```

n_body_problem.py

```
#!/usr/bin/env python
#coding: utf8

import numpy as np
import numpy.linalg as la
import scipy.optimize
import matplotlib.cm as cm
import matplotlib.pyplot as plt

class N_body_problem:

    """
    Create a random N-body system out of given number of particles.
    Make sure that the center of mass is fixed at the origin.
    """

    def __init__(self, nrBodies, m=None):

        self.nrBodies = nrBodies
        if m and len(m) == nrBodies:
            self.m = m
        else:
            self.m = np.random.uniform(1,2,nrBodies)
        self.r = self._construct(nrBodies,self.m)
        self.v = .1 * self._construct(nrBodies,self.m)

    def _construct(self, nrb, m):
        cns = np.random.uniform(-2,2,nrb*3).reshape((nrb,3))
        return self._reduce(cns, m)


    def _reduce(self,z,m):
        return z - np.sum(z*m.reshape(-1,1),axis=0)/m.sum()

    def get_y(self):
        return np.hstack((self.r.flat,self.v.flat))

    def get_m(self):
        return m.copy()
```

n_body_approx.py

```
#!/usr/bin/env python
#coding: utf8

import numpy as np
import numpy.linalg as la
```

```
import time, os

class NBodyApprox:

    """
    An instance of approximation for the N-body problem.
    """

    def __init__(self, name, y, dt, m = np.empty(0), G=1.,
                 isStore=True, stTime=False):

        self.methods = {'rk2': self.rk2,
                        'euler': self.euler,
                        'rk4': self.rk4,
                        'stVer':self.stVer}

        self.G = G
        self.dt = dt
        self.dim = len(y)/ (2 * len(m))
        self.N = len(m)
        self.y = y.copy()
        self.t = 0.
        self.m = m.copy()
        self.stTime = stTime
        self.isStore = isStore
        self.name = name

        self._storePath = ''
        self.file = None


    def steps(self, mthd, nrSteps):
        method = self.methods[mthd]

        if mthd == 'stVer':
            self.y = self.initStVer(self.y, self.dt)

        if self.isStore:
            self._initStoring(mthd)

        for i in xrange(nrSteps):
            self.y = method(self.y, self.dt)
            self.t += self.dt

            if self.isStore:
                self._store(self.y, self.t)

        if self.isStore:
            self._endStoring()
        return self.y, self.t
```

```
def setStorePath(self, path):
    self._storePath = path

def _initStoring(self, mthd):
    """
    Create storage file in store location with current 'Unix
    time' as file name in a folder with approximations name.
    """
    fileName = "%i.dat" % (int(time.time()))
    path = os.path.join(self._storePath, self.name)
    if not os.path.exists(path):
        os.makedirs(path)
    self.file = open(os.path.join(path, fileName), 'w')
    self._insertHeader(mthd)
    self._store(self.y, self.t)

def _insertHeader(self, mthd):
    """
    Create header for storage file with all relative
    data to this approximation. First two bytes
    descripe endianness (value should be 1 for 16
    bit integer).
    """
    np.ones(1,dtype='int16').tofile(self.file)
    np.array([self.dim, self.N,
              self.stTime], dtype = 'int32').tofile(self.file)
    np.array([self.G, self.dt],
             dtype = 'float64').tofile(self.file)
    self.m.tofile(self.file)
    np.array([mthd],dtype='|S5').tofile(self.file)

def _store(self, y, t):
    """Stor current state, and time if specified."""
    y.tofile(self.file)
    if self.stTime:
        np.array((t,)).tofile(self.file)

def _endStoring(self):
    self.file.close()

def rhs(self, y):
    """Calculate the right hand side of the N-body problem."""
    a = self.force(y)
    return hstack((y[self.N*self.dim:], a))

def force(self, y):
    a = np.zeros((self.N,self.dim))
    for j in xrange(self.N):
        for k in xrange(j):
```

```
                r = y[k*self.dim:(k+1)*self.dim] - \
                    y[j*self.dim:(j+1)*self.dim]
                f = self.G * r/(la.norm(r)**3)

                a[j] += f*self.m[k]
                a[k]-= f*self.m[j]
        return a.flatten()

    def euler(self, y, h):
        return y + self.rhs(y) * h

    def rk2(self, y, h):
        k1 = self.rhs(y)
        k2 = self.rhs(y + h * k1)
        return y + h / 2. * (k1 + k2)

    def rk4(self, y, h):
        h2 = h / 2.
        k1 = self.rhs(y)
        k2 = self.rhs(y + h2 * k1)
        k3 = self.rhs(y + h2 * k2)
        k4 = self.rhs(y + h * k3)
        return y + (h / 6.) * (k1 + 2. * k2 + 2. * k3 + k4)

    def stVer(self, y, h):
        r = y[:self.N * self.dim] + h * y[self.N * self.dim:]
        v = y[self.N * self.dim:] + h * self.force(r)
        return np.hstack((r, v))

    def initStVer(self, y, h):
        """Take initial half step for stVer with rk4."""
        v = self.euler(y, h/2.)[self.N * self.dim:]
        return np.hstack((y[:self.N * self.dim], v))
```

misc.py

```
#!/usr/bin/env python
#coding: utf8

import numpy as np

def loadData(filePath):
    """
    Load the data from a file as it's saved in NBodyApprox. Return
    a dictionary containing the approximation data and the values
    from the header. Remember to check for endiannes.
    """
    res = {}
```

```
    with open(filePath) as f:
        res['endian'] = np.fromfile(f,dtype='int16',count=1)
        tmp = np.fromfile(f,dtype='int32',count=3)
        res['dim'] = tmp[0]
        res['N'] = tmp[1]
        res['stTime'] = tmp[2]
        tmp = np.fromfile(f,dtype='float64',count=2)
        res['G'] = tmp[0]
        res['dt'] = tmp[1]
        res['m'] = np.fromfile(f,dtype='float64',count=res['N'])
        res['mthd'] = np.fromfile(f,dtype='|S5',count=1)
        res['y'] = np.fromfile(f,dtype='float64')
    col = 2 * res['N'] * res['dim']
    if res['stTime']:
        col += 1
    rows = res['y'].size / col
    res['y'] = res['y'].reshape((rows,col))
    return res

def tot_en(y,m,d=3,G=1.):
    """Calculate the total energy of states y of a system."""
    n = len(m)

    t = np.zeros(len(y))
    for i in xrange(n):
        t += np.sum(y[:,(n+i)*d:(n+i+1)*d]**2,axis=-1) * m[i]
    t *= .5

    u = np.zeros(len(y))
    for i in xrange(n):
        for j in xrange(i):
            r = np.sum((y[:,j*d:(j+1)*d] -
                        y[:,i*d:(i+1)*d])**2,axis=-1)**.5
            u += (m[i] + m[j]) / r
    u *= G

    return t - u
```

Bibliography

[1] S. J. Aarseth, *Gravitational N -Body Simulations*, Cambridge University Press, Cambridge, UK, 2003.

[2] K. G. Andersson and L.-C. Böiers, *Ordinära differentialekvationer*, Studentlitteratur AB, Lund, 1989, 1992.

[3] T. M. Apostol, *Calculus Volume I, second edition*, Blaisdell Publishing Company, Waltham, MA, 1967.

[4] J. M. A. Danby, *Fundamentals of Celestial Mechanics*, The Machmillan Company, New York, NY, 1970.

[5] F. Diacu and P. Holmes, *Celestial encounters: the origins of chaos and stability*, Princeton University Press, Princeton, NJ, 1996.

[6] E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration: Structure-Preserving Algorithms*, Springer-Verlag, Berlin, Heidelberg, New York, 2002.

[7] ——, *Geometric numerical integration illustrated by the Störmer-Verlet method*, Acta Numerica, 12 (2003), pp. 399–450.

[8] E. Hairer, S. P. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations I: Nonstiff Problems*, Springer-Verlag, Berlin, Heidelberg, New York, 1987.

[9] E. Hairer and G. Söderlind, *Explicit, time reversible, adaptive step size control*, SIAM J. SCI. COMPUT, 26 (2005), pp. 1838–1851.

[10] M. T. Heath, *Scientific computing: an introductory survey – 2nd. ed.*, McGraw-Hill, New York, NY 10020, 2002.

[11] A. Iserles, *A First Course in the Numerical Analysis of Differential Equations*, Cambridge University Press, Cambridge, UK, 2009.

[12] C. M. Linton, *From Eudoxus to Einstein A History of Mathematical Astronomy*, Cambridge University Press, Cambridge, 2004.

[13] K. R. Meyer and G. R. Hall, *Introduction to Hamiltonian Dynamical Systems and the N-Body Problem*, Springer-Verlag, New York, NY, 1991.

[14] F. R. Moulton, *The straight line solutions of the problem of N bodies*, Ann. Math., 12 (1910), pp. 1–17.

[15] H. Pollard, *Mathematical introduction to celestial mechanics*, Blaisdell Publishing Company, Waltham, MA, 1966.

[16] D. G. Saari, *Central Configurations — A Problem for the Twenty-first Century.*

[17] ——, *Collisions, Rings and Other Newtonian N-Body Problems*, vol. 104 of CBMS Regional Conference Series in Mathematics, AMS, Providence, RI, 2005.

[18] C. L. Siegel and J. K. Moser, *Lectures on Celestial Mechanics*, Springer-Verlag, Berlin, 1971.

[19] S. Smale, *Mathematical Problems for the Next Century*, Math. Intelligencer, 20 (1998), pp. 7–15.

[20] M. Šuvakov and V. Dmitrašinović, *Three Classes of Newtonian Three-Body Planar Periodic Orbits*, Physical Review Letters, 110 (2013), p. 114301.

[21] Q. D. Wang, *The existence of global solution of the N-body problem*, Chinese Astronom. Astrophys., 10 (1986), pp. 135–142.

[22] ——, *The global solution of the N-body problem*, Celestial Mech. Dynam. Astronom., 50 (1991), pp. 73–88.

[23] Z. Xia, *The existence of non-collision singularities in Newtonian systems*, Ann. Math., 135 (1992), pp. 411–468.

Bachelor's Theses in Mathematical Sciences 2013:K12
ISSN 1654-6229

LUNFMA-4025-2013

Mathematics
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, Sweden

http://www.maths.lth.se/