FYS4150 - Project 5
N-body simulation of an open galactic cluster

Simen Nyhus Bastnes & Eirik Ramsli Hauge

9. December 2016

Abstract

The aim of this project is to simulate a cold collapse with gravitational N-body simulations as described by M. Joyce and co-workers in their article [4]. By using the collapse time as the time unit, we will simulate N particles, in our case stars, starting with negative total energy and no velocity. Using only the gravitational force we will find that our simulation is never stabilized, thus we will implement a smoothing factor ε which must be found by trial and error. Once the system is stabilized we will look at energy conservation, fraction of particles escaping (i.e. achieves a positive total energy) and the kinetic energy of the escaped particles. We will also see if the virial theorem is valid for our system and plot the radial density and try to fit equation (12) to the data. Lastly we will find the mean radius and the standard deviation of the radius for different number of particles.

The results showed us that our simulation was stable once the smoothing factor was found and added. This values was found to be 0.2 for 200 particles, but assumed to be 0.1 for 1000. The energy was stable for these smoothing factors. The fraction of particles escaping seemed to increase with increasing N, this was however not the case for 1000 particles. The kinetic energy of these particles was found to decrease with increasing N.

By testing the virial theorem we found that the energies fitted better the larger number of particles we simulated. This is because we take an average of a large system and thus it is dependent on N.

Both radial density and mean radius were found to decrease with increasing N.

Almost all our result were coinciding with those of M.Joyce and his co-workers as mentioned above.

Introduction

In this project we will attempt to develop code that can be used for gravitational N-body simulations. Taking inspiration from an article by M. Joyce and co-workers [4], we will build a simple model of an open galactic cluster.

An open cluster is a group consisting of up to a few thousand gravitationally bound stars, created from the collapse of a molecular cloud, and are usually found in the arms of spiral galaxies or in in irregular galaxies. The collapse of the cloud leads to a burst of star formation. As the stars of the cluster are of similar age and chemical composition, they are very important objects in the study of stellar evolution. Once formed, they gradually dissipate as members are ejected due to random interactions with other clusters, clouds of gas, or internal collisions.

Since open clusters are of such importance, it is of great astrophysical use to be able to simulate this model. This is also a N-body problem, which for a physicist, the ability to be able to simulate N bodies is enormously useful as it is applicable within many different fields of physics. Thus we have two main reasons to do this project. Generally it is useful to learn how to perform N-body simulations, and more specifically, in astrophysics it is useful to look at this specific N-body problem. While increasing the number of particles beyond a few thousand particles the interpretation of one particle as one or a few stars will not be useful, the simulation can be applied to discussions about dark matter halos.

First, we look at some of the physics behind our model, making sure to find ideal units to fit our timescale. Running our model for $N = 100$ particles, we check the stability of our system, and then find an ideal smoothing function for the gravitational force to remove some of the numerical instabilities. Having done that, we check how long the system takes to stabilize, conservation of energy, and checking the fraction of particles ejected from the system. Looking at only the bound particles, we can check if our results are consistent with the virial theorem.

Finally, we increase the number of particles while keeping the total mass constant, to see how the radial density looks like. This can then be compared to well known profiles such as the Navarro-Frenk-White profile.

Theory

N-body simulation

In order to make our model of an open cluster, we will first look at some of the assumptions we make.

- The cluster consists of N separate particles only interacting with each other via the Newtonian gravitational force. The force between two particles are then

$$F_G = -\frac{GM_1, M_2}{r^2} \tag{1}$$

  where G is the gravitational constant, r is the distance between the particles, and M_1, M_2 is the mass of the particles.

- Since the only interaction is the gravitational force, the system is collisionless. If the probability of interactions between particles is low, then collisions have no significant effect on the results. For very large N, this might not hold as well.

- The particles start with little or no initial velocity, so-called *cold collapse.*

- The particles are uniformly distributed within a sphere of radius $R_0 = 20$ ly. Masses are randomly distributed by a Gaussian distribution around $10\,M_\odot$ with a deviation of one solar mass.

Earlier this fall, we made a model of our solar system. [1] We recall that for a system only affected by Newtonian gravity, Newton's second law of motion gives us the following differential equations for particle i

$$\frac{d^2\mathbf{r_i}}{dt^2} = \sum_{j\neq i} \frac{F_G(\mathbf{r}_{ij}, M_i, M_j)}{M_i} \tag{2}$$

where $\mathbf{r}$ is a three-dimensional vector in (x, y, z), and we sum over the gravitational contribution of the other particles. We note that the length and timescale should be adjusted to units more fit for the dynamics of the system.

For the length scale, we can use lightyears, as the particles are initially uniformly distributed inside a sphere of radius $R_0 = 20$ ly. In order to find units of time that fit the timescale however, we will take a short detour into non-linear perturbation theory and cosmology.

The spherical top-hat model

While solving non-linear perturbation theory analytically can be difficult (or impossible), there does exists an analytical solution for the so-called "*spherical top-hat*" model [3]. In this model, we look at a spherical perturbation of radius R, with a uniform density inside. It can be shown that this model has a parametrised solution

$$\begin{aligned} R &= A(1-\cos\theta) \\ t &= B(\theta - \sin\theta) \\ A^3 &= GMB^2 \end{aligned} \tag{3}$$

From equation (3), we see that the sphere reaches a maximum at $\theta = \pi$, at time $t = \pi B$ (also referred to as turn-around). The sphere collapses completely at $\theta = 2\pi$, at time $t = 2\pi B$. At $\theta = \frac{3\pi}{2}$, the model virializes at radius $1/2R_0$. For the peak radius

$$\begin{aligned} R(\theta = \pi) &= 2A = R_0 \\ t(\theta = \pi) &= \pi B \end{aligned}$$

Solving for B

$$B = \frac{A^{3/2}}{\sqrt{GM}}$$

Using that the density of the perturbation is given by $\rho_0 = M/V = M/(4/3\pi R_0^3)$

$$\begin{aligned} B &= \frac{(R_0/2)^{3/2}}{\sqrt{G\frac{4\pi R_0^3\rho_0}{3}}} \\ B &= \sqrt{\frac{3}{32\pi G\rho_0}} \end{aligned}$$

Since our simulation starts with cold collapse (at turn-around $\theta = \pi$), the time it takes to collapse is

$$\begin{aligned} t_{\text{coll}} &= t_{2\pi} - t_\pi \\ t_{\text{coll}} &= 2\pi B - \pi B = \pi B \end{aligned}$$

inserting the expression we found for B, we get the collapse time

$$t_{\rm coll} = \sqrt{\frac{3\pi}{32G\rho_0}} \tag{4}$$

As this time, $t_{\rm coll}$ is an analytical expression for collapse in the *spherical top-hat model*, this is a much more fitting timescale to probe our system with compared to seconds or years.

In order to make use of this, we need to scale equation (1) in these units, specifically G. Rewriting equation (4) gives us

$$G = \frac{3\pi}{32\rho_0 t_{\rm coll}^2}$$

Scaling so that we are looking at timescales of order $t_{\rm coll}$, and inserting $\rho_0 = 4/3\pi R_0^3$

$$G = \frac{\pi^2 R_0^3}{8M_{tot}} = \frac{\pi^2 R_0^3}{8N\mu} \tag{5}$$

where N is the number of particles, and μ is the average mass (in solar masses) per particle.

Inserting equation (5) into the differential equation given by (2) gives us the units $[{\rm ly}/t_{\rm coll}]$, which is what we wanted.

Numerical instabilities

Since we will be solving the system numerically, we have to be aware of some of the problems of doing so. In this project, we will focus on the numerical instability that arises when two particles come very close to each other. When two particles get close to each other, the accelerations get large, and unless we have very small timesteps, the trajectory of a particle is not calculated accurately. To deal with this, we will introduce a very simple smoothing function by modifying the Newtonian force (1)

$$F_{\rm mod} = \frac{GM_1M_2}{r^2 + \varepsilon^2} \tag{6}$$

where ε is a small real constant that causes equation (6) to be finite on short length scales. While adding this smoothing factor smooths out the instabilities, it also means that on length scales smaller than ε, the trajectories are not integrated accurately. This means that we should make sure to set the smoothing factor ε to a length so that 1) instabilities are smoothed out, and 2) the inaccuracy in the trajectory does not play a significant part of the macroscopic evolution of the system.

As we discussed earlier, we assume the system to be collisionless, and that collisions (or close encounters) should not affect the system much. Since our particles does not represent point particles, but mass distributions with some finite radius, a really close encounter between two particles might not be fully collisionless. The smoothing factor does then in a sense represent the fact that our system is not fully collisionless, but the magnitude of it is most likely considerably larger than the scale at which collisions become relevant.

Finding an ideal size for ε can be difficult. A simple way to evaluate this is to first find the mean interparticle distance. [5]

$$l \equiv \left(\frac{3V}{4\pi N}\right)^{1/3} = \frac{R_0}{N^{1/3}} \tag{7}$$

where R_0 is the initial radius of the sphere, and N is the number of particles. Since this is the average distance between the particles at the start of the simulation, we should make sure that our choice of ε is significantly less than l, as it limits how small the structure is allowed to collapse. For example, for the situation $R_0 = 20$ and $N = 100$, we have $l \approx 4.3$.

Particle ejection

Due to looking at a *cold collapse*, all particles start with a negative energy, but as the system evolves, some of the particles can end up with a positive energy, and completely escape from the system. We can therefore identify these ejected particles by comparing the kinetic and potential energy per particle, and identifying which particles have a total energy greater than zero

$$E_{i,\mathrm{tot}} = K_i + V_i > 0 \tag{8}$$

where K_i and V_i is the kinetic and potential energy for particle i.

The energy taken away from the system by particle ejection can be found by energy conservation and that once the system stabilizes, the potential energy of the ejected particles and the interaction between bound and ejected particles is negligible

$$E_0 = V_b + K_b + K_f \tag{9}$$

where E_0 is the initial energy of the system, V_b and K_b is the kinetic and potential energy of the particles that are gravitationally bound, and K_f is the kinetic energy of the ejected particles. K_f can be found as discussed earlier in equation (8).

Virial theorem

For a bound gravitational system in equilibrium, the virial theorem says that

$$2\langle K\rangle = -\langle V\rangle \tag{10}$$

where $\langle K\rangle$ is the time-average kinetic energy of the system, and $\langle V\rangle$ is the time-average potential energy. By the ergodic hypothesis, we can assume that an average over a large enough system should give the same result as the time average. Thus we have

$$2K_b = -V_b \tag{11}$$

where K_b and V_b are the energies for the bound particles, and can be found from equation (9)

Radial density of particles

For an approximately spherical symmetric collapsed halo (virialized structure), the radial density profile can often be fit very well with the simple expression

$$n(r) = \frac{n_0}{\Big(1 + (\frac{r}{r_0})^4\Big)} \tag{12}$$

where n_0 and the scale radius r_0 has some dependency on the number of particles N.

The well-known Navarro-Frenk-White profile can also be compared to the radial density distribution.

$$\rho(r) = \frac{\rho_0}{\frac{r}{r_0}\Big(1 + \frac{r}{r_0}\Big)^2} \tag{13}$$

where ρ_0 and r_0 depends on N in some way. ρ_0 sets the normalization of the profile, and r_0 represents a scale at which the profile changes from $\rho \propto r^{-1}$ to $\rho \propto r^{-3}$.

Numerical solving

In order to solve the system numerically, we will employ the Velocity Verlet method. A full derivation of the Velocity Verlet method can be found in [1], where the following algorithm is also found.

Algorithm 1 Velocity Verlet

```
for $i = 0, n$ do
    $r_{i+1} = r_i + v_i h + \frac{1}{2} a_i h^2$
    $a_{i+1} = F_{G,r_{i+1}}/m$
    $v_{i+1} = v_i + \frac{1}{2} h (a_i + a_{i+1})$
end for
```

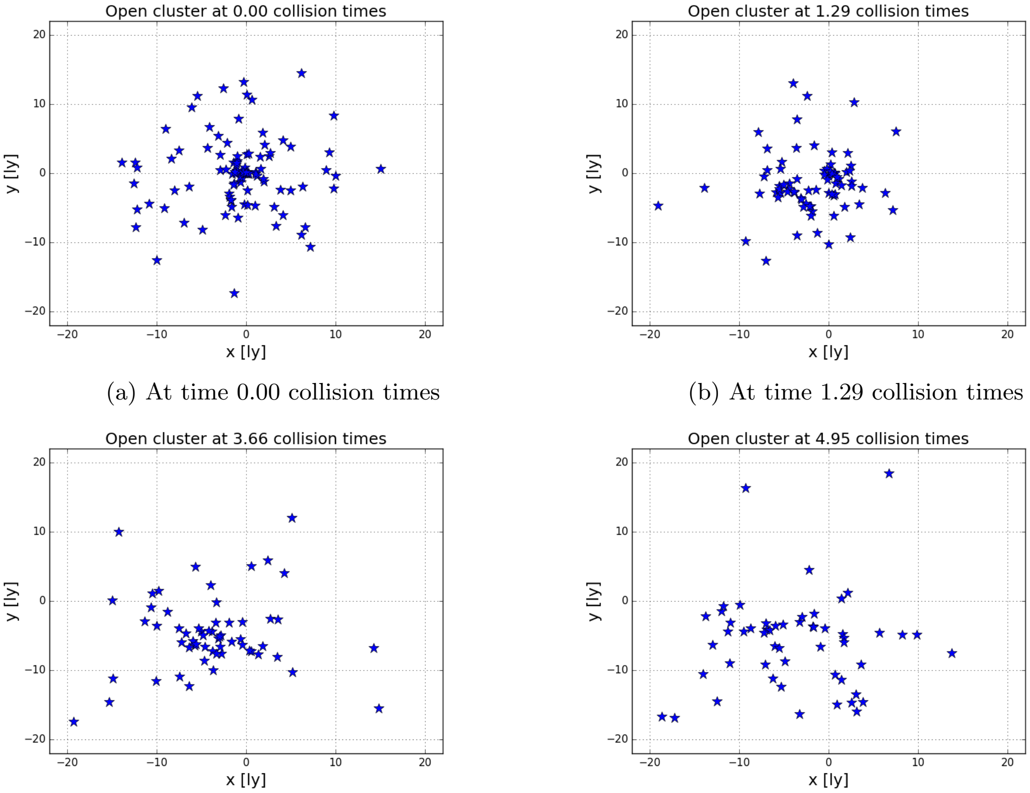
Experimental

The programs used in this project can be found in the GitHub repository [2], in the `/src/` folder. When running the program it takes 4 command line arguments: `N`, `t_coll`, `dt` and `eps`. `N` is the number of stars in our cluster, `t_coll` is our time unit, based on the collision time, `dt` is the time step and `eps` is the smoothing factor used in the force calculation. All data written to file by the main program are stored in `/benchmarks/` where most of the data analysis were done by python programs stored in `/python/` and the figures and gifs can be found in `/figures/`.

The simulation starts by randomly positioning different stars inside a sphere with radius R = 20 light years. Each star will have an individual mass where the total mass is forced to be 1000 solar masses and an initial velocity set to be 0 in all directions.
Once all the positions and masses have been set, we can find the forces acting between them. While using velocity verlet to simulate the behaviour we can also find the individual kinetic and potential energy of each star as well as the total kinetic and potential energy of the system.

Results

A simulation for N = 100 particles over a time of 5 collision times with a time step of 0.001 and no smoothing factor is plotted in figure 1 for different times during the simulation. From these figures and the figure 2 we can see that without a smoothing factor we will never reach an equilibrium.



(a) At time 0.00 collision times

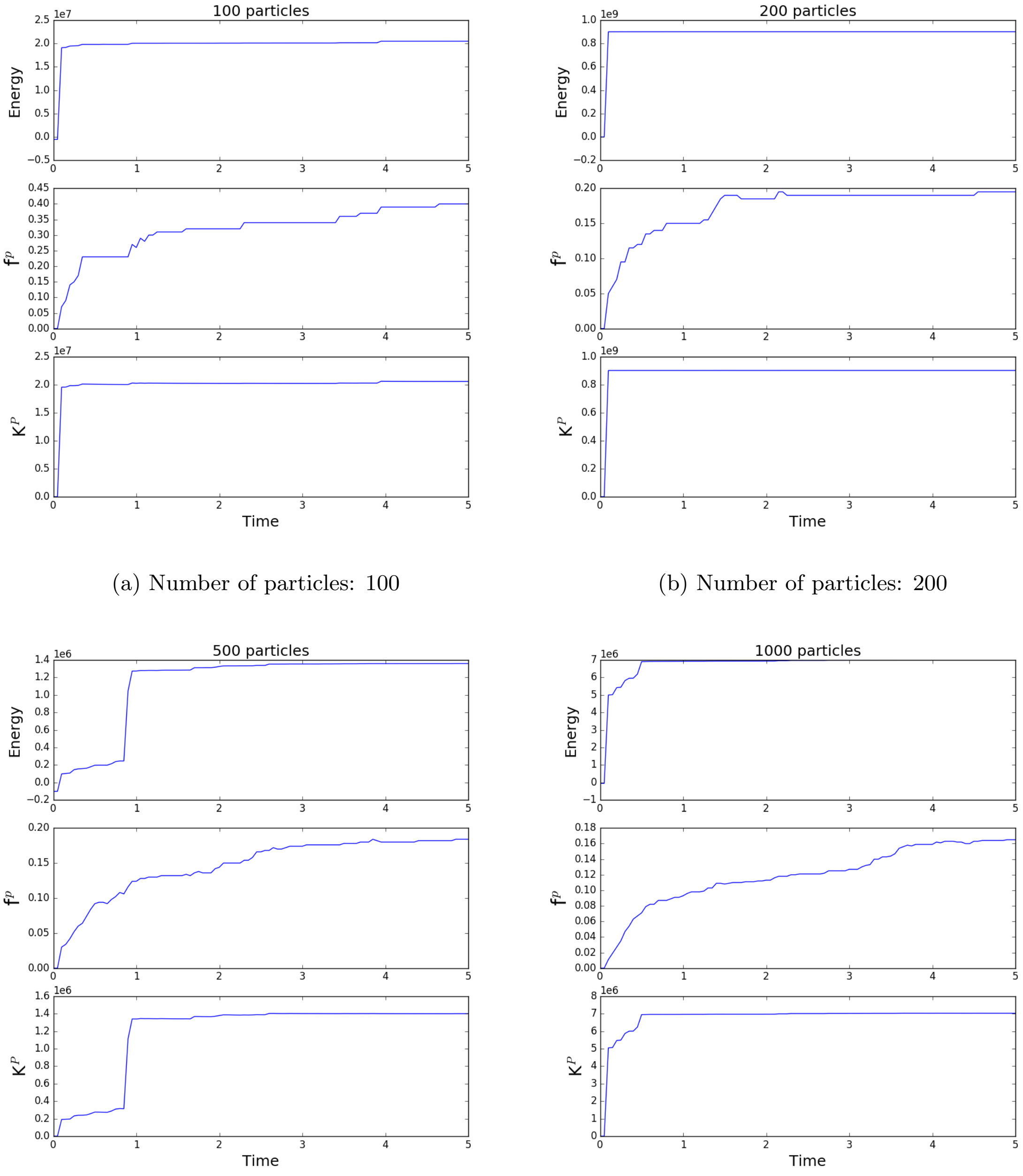
(b) At time 1.29 collision times

(c) At time 3.66 collision times

(d) At time 4.95 collision times

Figure 1: Position of each star in 2D as a function of time.

Letting our program run for different number of particles we found the total energy, the fraction of particles with positive total energy and the kinetic energy of the escaped particles. These results are shown below in figure 2.



(a) Number of particles: 100

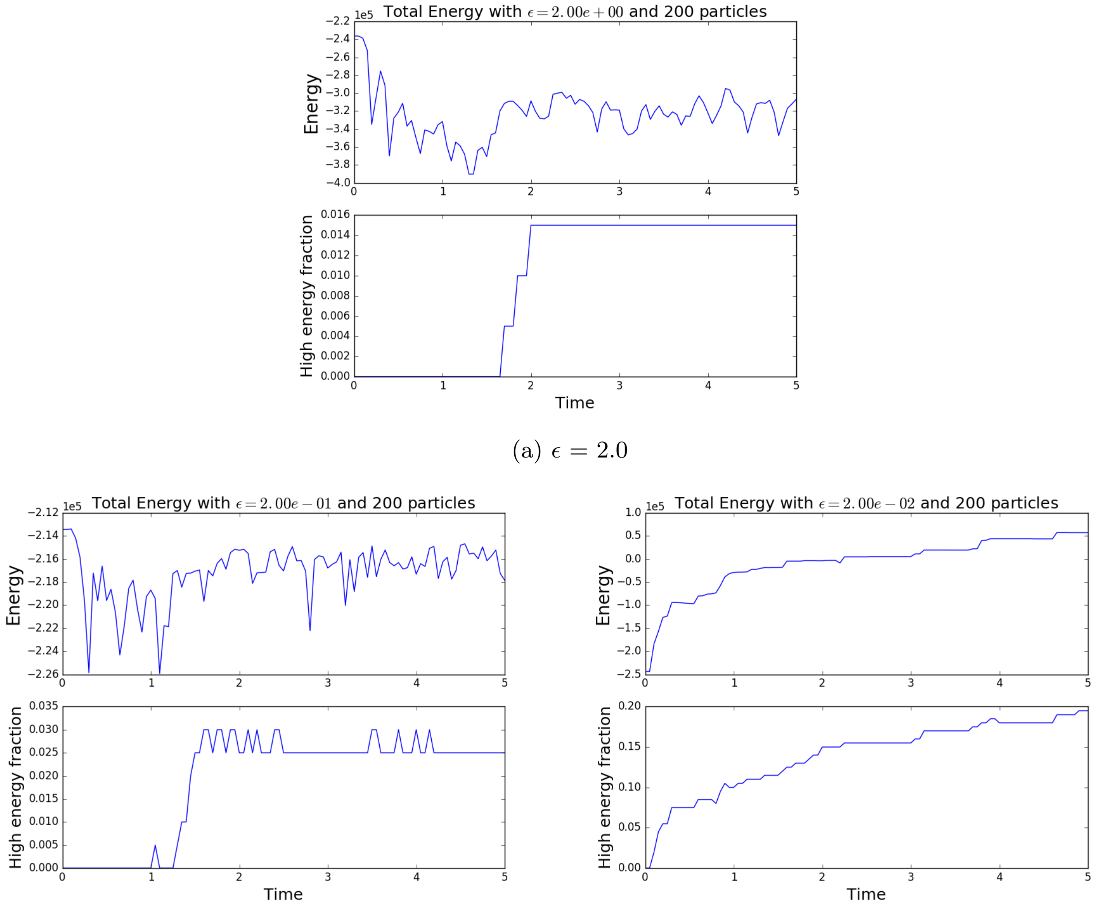
(b) Number of particles: 200

(c) Number of particles: 500

(d) Number of particles: 1000

Figure 2: Total energy (top), fraction of escaped particels (middle) and the kinetic energy of the escaped partivles (bottom) for different number of particles without a smoothing factor.

To find the best smoothing factor discussed above in the theory we used the trial and error method. For 200 stars with a time step of 0.001 and a time from 0 to 5 collision times we found the total energy as presented in figure 3



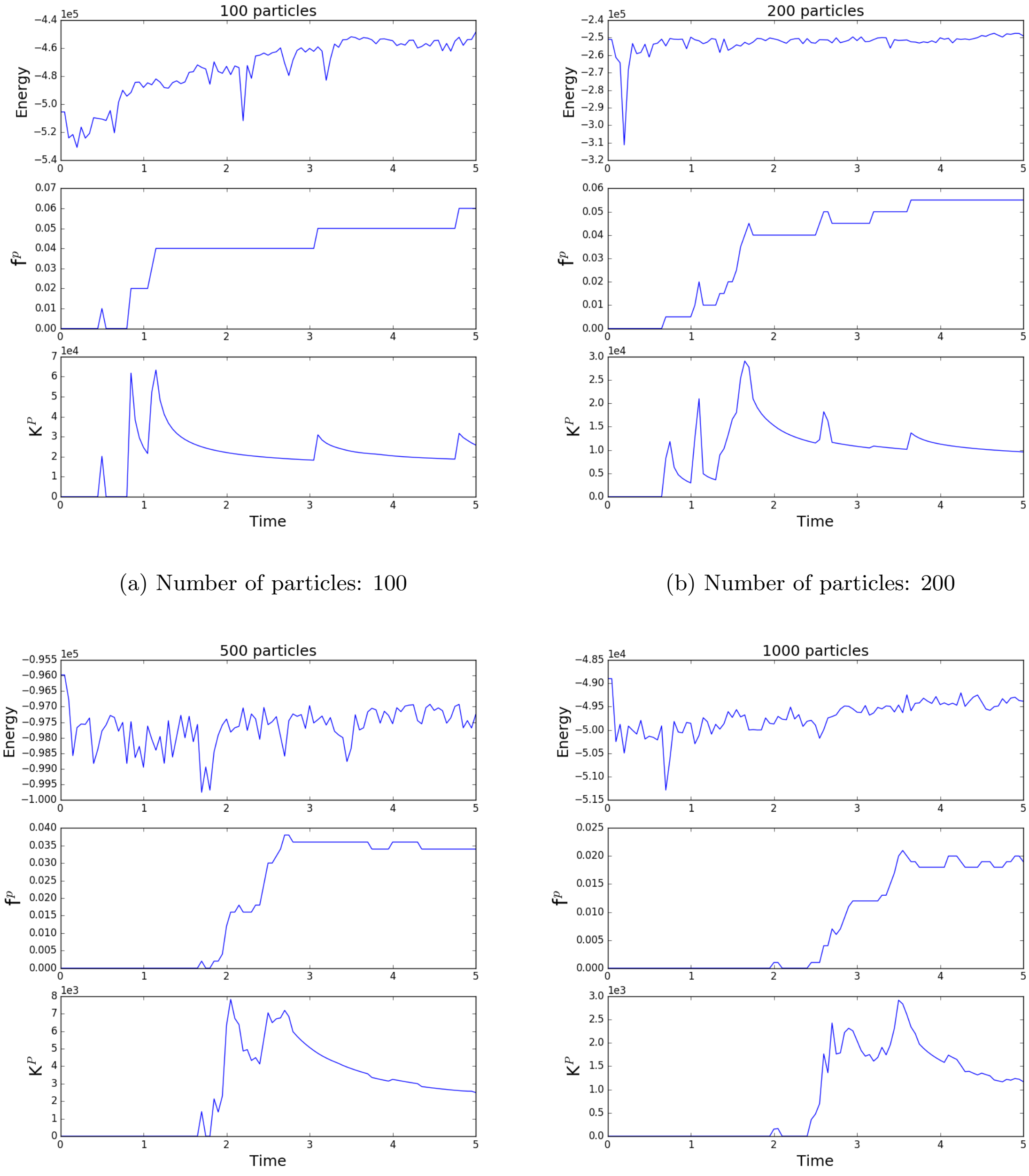
(a) $\epsilon = 2.0$

(b) $\epsilon = 0.2$

(c) $\epsilon = 0.02$

Figure 3: Total energy (top) and fraction escaped particles (bottom) for different smoothing factors for a simulation of 200 particles.

As we now have found a better smoothing factor we can once again simulate for different number of particles and see the difference. By setting a smoothing factor of 0.1 and simulating for different number of particles with time step 0.001 and over a time period of 5 collisions times the graphs in figure 4:



(a) Number of particles: 100

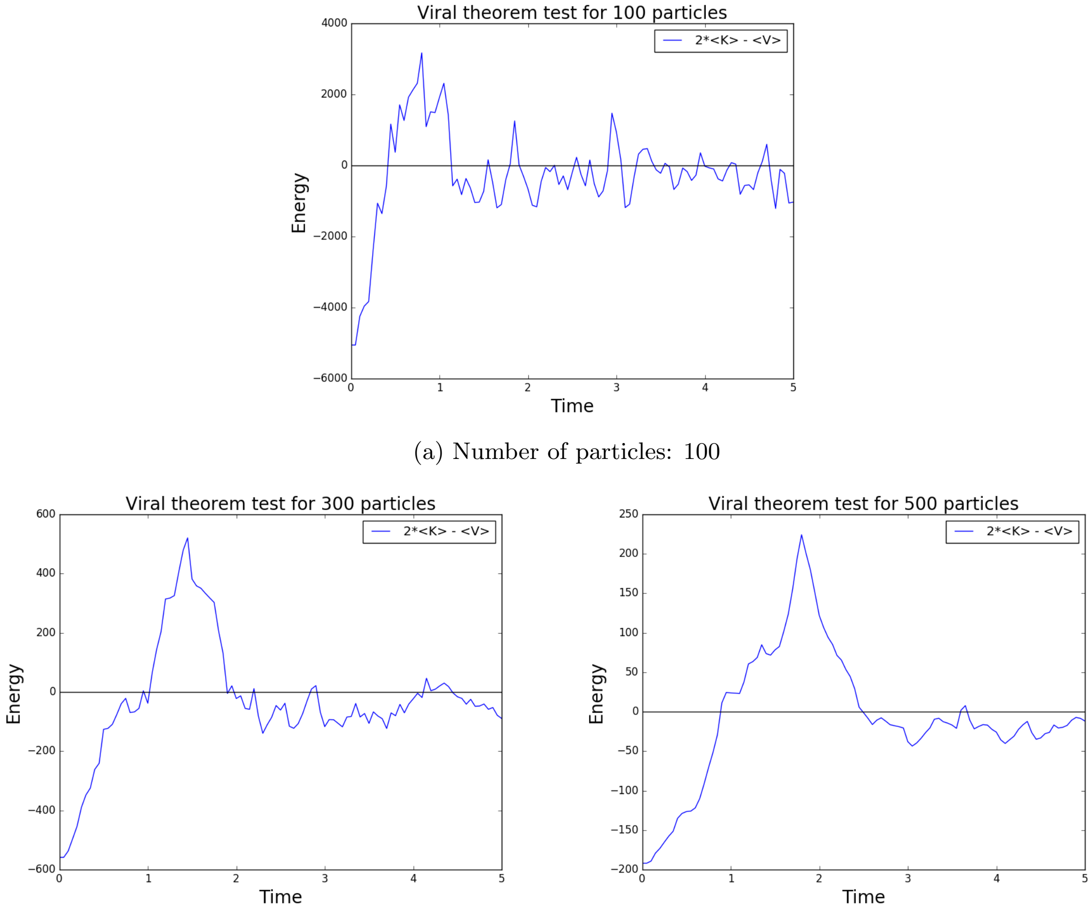
(b) Number of particles: 200

(c) Number of particles: 500

(d) Number of particles: 1000

Figure 4: Total energy (top), fraction of escaped particels (middle) and the kinetic energy of the escaped partivles (bottom) for different number of particles with a smoothing factor.

To see if we our results were consisten with the viral theorem, we checked if equation (11) held for our system. The result are plotted in figure 5



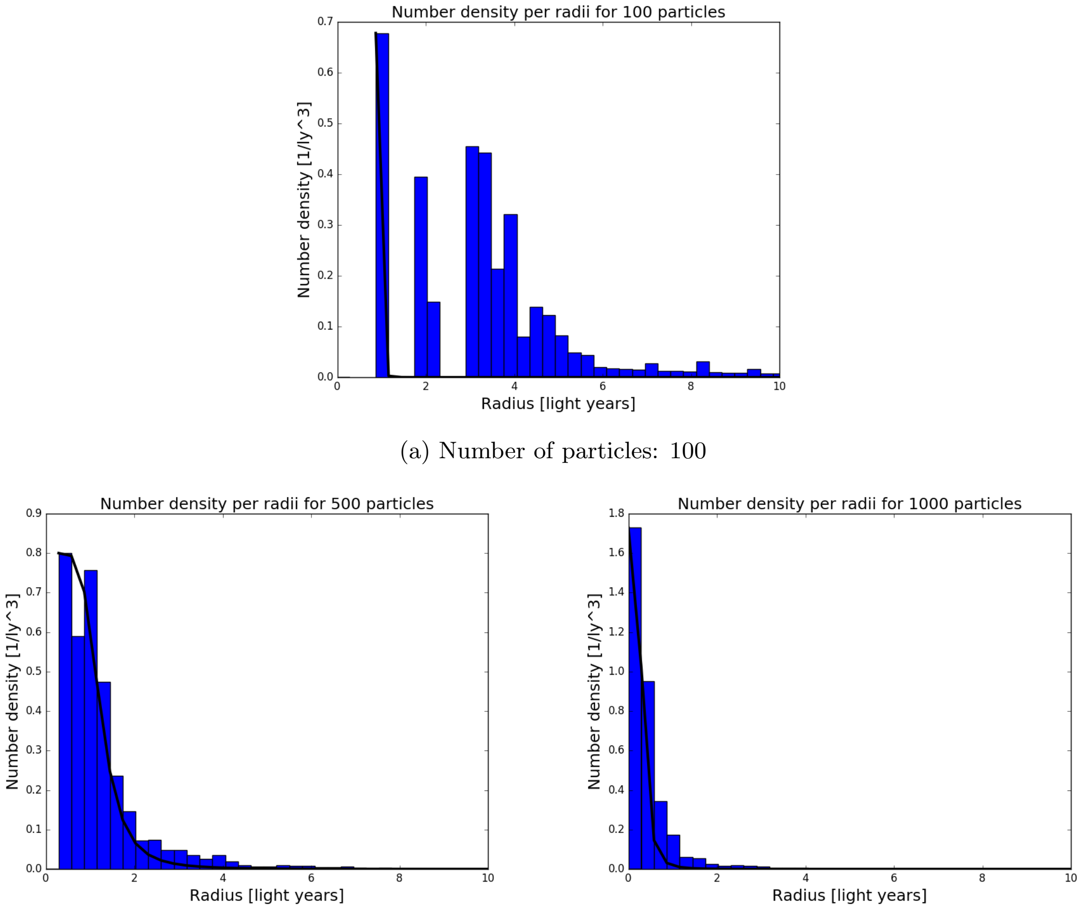
(a) Number of particles: 100

(b) Number of particles: 300

(c) Number of particles: 500

Figure 5: Test of the viral theorem. When the blue and black line crosses, we have fullfilled the viral theorem. The energies are the average of a large system.

By simulating different number of particles with the same total mass we were able to find the radial density, mean distance from origo and the standard derivation of distance from origo (see figure 6 and 7)



(a) Number of particles: 100

(b) Number of particles: 500

(c) Number of particles: 1000

Figure 6: Radial density for different number of particles.

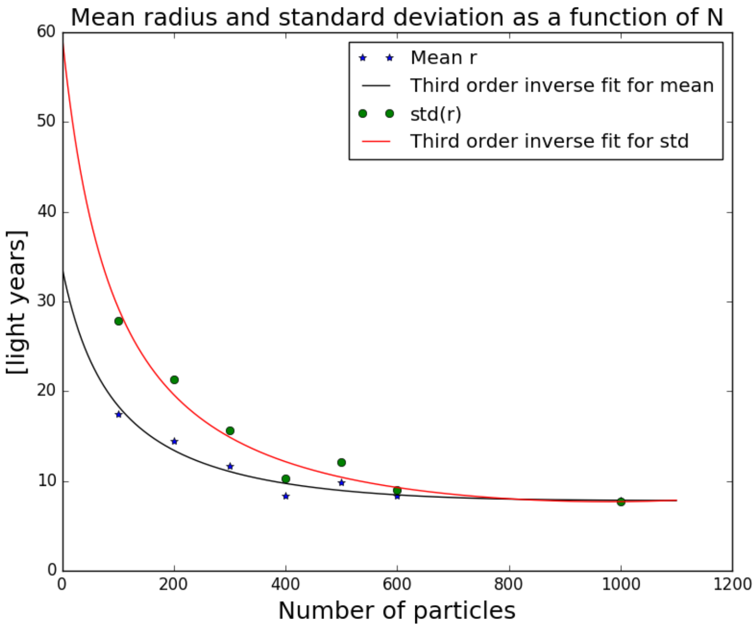


Figure 7: Standard deviation and average distance from origo for all particles

Discussion

The choice of time step was done by trial and error. From the extended report by Joyce [5], we found that they used a time step of 5.0e-05 during the collapse phase around t_{coll}, while 5.0e-4 during the other phases. We tried different values and found that a dt of 1.0e-03 was a good time step, which both gave us a good resolution while also keeping the time usage of the program low.

In the limit $N \rightarrow \infty$ with a constant ρ we get a continuous fluid. The system should then collapse into a singularity after one t_{coll}. However, we do not observe this collapse in our simulation. This is due to the fact that while we distribute our particles uniformly, it will not be fully isotropic for finite N, and the small deviations cause particles to miss the center, and shoot outwards again high velocity. The end result of this is that the system "*virializes*" in an equilibrium state at a finite radius.

As we can see from figure 2 where $\varepsilon = 0$, the energy is never conserved nor is it stable and there is no point in discussing these plots in detail, as the numerical instabilities discussed earlier causes some particles to gain a lot of energy. Even with testing for time steps as small as 1.0e-6, the instabilities remain.

The plots in figure 4, where we have added a smoothing factor are much more interesting and we are especially interested in the middle and bottom ones for each simulation. Number of particles escaping and the kinetic energy of these particles are found using equation (8) and we notice two trends. As the number of particles increases there seem to be a larger fraction of escaping particles.

This is true up until 500 particles, but the fraction is a bit lower at 1000. The cause of this might either be that there are abnormalities in our measurements, this could easily be corrected by doing the simulation many times and the take an average, or it could be because the fraction increases for low values of N and then it is more constant for larger values of N. By looking at the energy of the escaped particles we see a different story. Max values of each simulation at the equilibrium time (see below) seem to decrease as there are more particles added to the system. This means that with more particles, even though more are escaping, the ones that do escape have a lower kinetic energy per particle. As we keep the total mass of the system constant, each particle has less mass, and thus the difference could be attributed to this.

Once the system reaches equilibrium there are still some particles being ejected, as seen from the fraction of particles escaping, but this fraction becomes more and more stable for larger values of N. The bottom part of the three plots for each N-values shows us the energy of the escaped particles. We notice that once equilibrium is reached the escaped kinetic energy slowly decreases over time. This is due to the fact that when the particle is defined as ejected ($E > 0$), it is still affected gravitationally by the other particles. As time progresses, the ejected particles lose some kinetic energy as they work their way out of the gravitational potential.

The system seems in general to be more stable for larger values of N. If we were to do this experiment again we would likely use larger N-values, and do multiple simulations for each N-value. This time around however, we had exams in a lot of other courses and had only time for the simulations we have run.

Due to the numerical instabilities that arises when two particles come very close, we implement a smoothing factor to our force calculations as shown in (6). The ideal value for the smoothing factor is a factor that is difficult to pinpoint. However, we would like it to be a couple of factors lower than the mean interparticle distance given by equation (7), but also large enough to smooth the instabilities enough. As we can see from the figures in 2 there were several values tried out for $N = 200$ which would have a mean interparticle distance of 3.42 light years. The value $\epsilon = 0.02$ was clearly the worst. With no stabilization in the escaped particles and a total energy showing a increase rather than being stable, the value could not be used. $\epsilon = 0.2$ and $\epsilon = 2.0$ both show a stabilizing effect. The oscillation of the fraction of high energy particles for $\epsilon = 0.2$ could be a result of particles on the border of escaping, going back and forth. If we did this project again I would like to do this for even more values of N, finding a better way to determine ϵ and find the reason for the oscillating.
With a known value of ϵ for 200 N we redid the results from earlier tasks. The challenge here was to estimate a value of ϵ for every N. As the mean interparticle distance decreases for higher N, it should become smaller with larger N and that for 0.2 was good for 200, we set the value to 0.1 before simulating 100, 200, 500 and 1000 particles. Ideally every values of N should have its own ϵ, but since we only know that it should be less than the mean interparticle distance and 0.2 worked for 200, we thought that 0.1 would be a good estimation for all these N-values, even though a more thorough evaluation of ε should have been done. If we were to do this project again, it would be wise to find individual ϵ by comparing different values as in figure 2, but as our results are not used any further, our estimation seems to be a good one.
These figures shows us that with a smoothing factor we can reach an equilibrium which differs in time for the different values of N that are placed randomly within the same volume and with the same total mass. By looking at the fraction of particles with positive energy, there seems to be an

equilibrium at around 1 t_{coll} for 100 particles, 1.6 t_{coll} for 200, 2.5 t_{coll} for 500 and 3.5 t_{coll} for 1000 particles. This increase in equilibrium time is a bit weird as one would expect the equilibrium time to be approximately t_{coll} for all N. The fact that we observe such a large difference tells us that something seems to be wrong, but we have not been able to see what. Once again we realize that if we were to do this again, the increasing equilibrium time is definitely something that should be looked at closer.
The addition of the smoothing factor can be justified to some extent, as discussed in the theory, by the fact that we do not look at point particles but rathermass distributions of some finite extent, and that we do not simulate collisions between the particles. This shows that we will not get an accurate representation of the trajectories at small length scales, but for our purposes in this project the simple smoothing correction is a good estimation as long as the system remains mostly collisionless.

From figure 5 the virial theorem fits better and better for higher values of N. This makes sense as a larger N means that kinetic and potential energy is closer to the time averaged energies, as discussed earlier. The difference from zero is however to large to ignore, at least for $N = 100$. This means that the virial theorem is mostly consistent with our results, but not quite and especially not for a small number particles. It might be the way we extract the energy for bound particles, but as we discussed earlier, once the system reaches equilibrium the ejected particles should not affect the gravitational energy of the system. It might be that we have a fault in our simulation, or that the smoothing factor is not ideal. Some variation around zero is to be expected, but if we were to do the experiment again, attempting to improve these results should be done. But as the other parts of our program makes sense and our virial theorem test improves for larger values of N, it is reasonable to argue that we have something that works fairly well, but it is not ideal for small N or that our energy is either extracted wrong or manipulated wrong by our calculations.

As we can see from figure 6 the cluster is densest near origo. This is as expected, as most of the particles should form a structure at some finite radius. The solid line shows our best fit of the number density by equation (12), shifted to start at the maximum value on the histogram as our simulation does not allow for many particles to occupy the small volume near $r = 0$, or if the anisotropy of the initial conditions makes the center-of-mass move. We see that the profile does fit our results fairly well when $r_0 \propto N^{-1/3}$, while n_0 is set to the maximum value of the histogram due to the fact that it is normalized. Finding an N dependence on n_0 could have been done.

The mass density plotted versus radius is not included, as they were close to indistinguishable from the number density (figure 6). This is not surprising, given the fact that we only have one type of particle, where the mass is given by a gaussian distribution. We attempted to fit the NFW profile (13) to the mass density, without success due to time constraints.

Figure 7 shows that the mean radius decreases with increasing N. This is consistent with the results from [4]. The spherical top-hat model discussed earlier has a virialization radius at $1/2R_0$, which in our case would be $R_{vir} = 10$ ly. Once again, testing for higher values of N would most likely allow us to improve the accuracy of our results.

Conclusion

In the end we found that a time step of $dt = 0.001$ was a good time step. Theoretically we could have simulated extremely large numbers of particles, and even higher, but restrictions regarding time, computational powers and the fact that we directly sum all the forces limits the amount we can simulate. Although there are better ways to compute the gravitational potential. [4] Our simulation was not stable before we added a smoothing factor and this factor was found through trial and error to be good as 0.2 for N = 200 and assumed to be 0.1 for N = 1000. With a smoothing factor we could look at total energy stability, fraction of particles escaping and the kinetic energy of these particles. We found that the energy was stable for all our values of N, that the fraction of particles escaping to some degree increased with increasing N although it went down again for N = 1000. The kinetic energy of the escaped particles decreased with increasing N, meaning that even though a bigger fraction escaped, they had less energy with them.

The virial theorem was seen to be increasingly better fullfilled for a system in equilibrium for increasing N, this is because we used the average over a large system, whose validity increases with increasing N, instead of time average.

We see that the radial density decreases mostly as the function in equation (12).

The mean radius decreases with increasing N as expected from the article [4]. In general, our results were found to mostly be coinciding with those presented by M. Joyce and his co-workers [4], [5].

References

[1] Github repository, project 3. `https://github.com/scuper42/fys4150-project-3`.

[2] Github repository, project 5. `https://github.com/simennb/fys4150-project-5`.

[3] Lecture notes ast4320. `https://dl.dropboxusercontent.com/u/16515147/AST4320/LecturenotesAST4320_Nov30.pdf`. Pages 19-21 and 57.

[4] Joyce et. Al. Cold uniform spherical collapse revisited. Hand out.

[5] M. Joyce, B. Marcos, and F. Sylos Labini. Energy ejection in the collapse of a cold spherical self-gravitating cloud. *Monthly Notices of the Royal Astronomical Society*, 397:775–792, August 2009.