Peter Craig
ASTP 720
HW 3
02/15/2020

1 Introduction

Galaxy clusters are some of the largest structures in the universe. Basically these are clusters of many galaxies which are all gravitationally bound. Clusters tend to have a large overall mass, and can be comprised of a variety of different types of galaxies. Many clusters have on the order of hundreds or thousands of galaxies. Galaxy clusters are interesting for the purposes of studying dark matter as they tend to have a very large amount of it. In fact, something like 90% of the mass in a typical cluster is comprised of dark matter. They also contain large amounts of hot gas inbetween the galaxies in the cluster, which emits a large number of x-rays.

What I did was to simulate how the galaxies inside a a galaxy cluster move. We start out with a collection of 908 galaxies inside of a cluster, then use an nbody code in order to study how the cluster evolves over time. This simply computes the gravitational interactions of galaxies in the cluster and uses that to determine future positions for each of the galaxies. This nbody code uses a barnes - hut tree algorithm with a Verletet method to update the positions.

2 Methods

The N-body code operates using a Barnes-Hut tree algorith. Basically the way that this works is to split up the array of particle into a tree structure. This can be accomplished recursively, and the way that we do this is as followes. We start with a single empty node in the tree. Then we iterate through our list of particles. The first particle gets added to this empty node. What we want to do is to keep splitting things up until every particle has its own node. So when we add the second particle, we create 8 child nodes, one for each octant of the region of our simulation, and place the two particles into the appropriate nodes based on where they are.

We then continue this process, where each particle c=gets added to the appropriate existing nodes, and if it is not alone in any of these nodes then we create 8 new nodes from the smallest node it is in, and place it into one of those. If we continue this process through every particle in the simulation, then we end up with a tree where the end nodes, or leaves, each contain at most one particle, and every particle in the simulation is present in the tree. This allows us to approximate the gravitational forces by directly evaluating the force from nearby particles, and grouping together particles at large distances.

So we continue by computing the center of mass of each node in the tree, and then we use this to actually evaluate the forces. To do this, you take a particle, and evaluate each node starting from the top (big nodes). For each node, we compute $\theta = L/D$, where L is the side length of the node, and D is the distance to the node. If theta is less than some cutoff, we use the center of mass of the node. Otherwise we move on and repeat with that node's children until either there is only one particle, or $\theta < 1$. 1 is chosen for θ because that tells you that the size of the cell is the same as the distance, so if we are larger than this it is likely that we need to use smaller nodes for an approximation, and if we are smaller than this then we should have a decent approximation.

The last thing to discuss here is how we actually compute individual accelerations. We do this using force softening, so basically it is just a normal gravitational force at large distances, but modified at small distances to avoid problems with very large accelerations. In particular, we multiply the force by the softening kernel shown in equation **??**. In my case I have chosen $\epsilon = 30$ kpc, which is roughly the size of a typical galaxy, which is typically a good way to set this parameter. Additionally, a smaller parameter tends to make aa few of the particles get flung out of the cluster, which probably means that two gaalaxies had a close interaction, and experienced a very large change in velocity due to a large acceleration over a large time step.

To run the simulation, we have to actually use these accelerations to update the positions. For the first timestep we use a slightly modified Verletet method in order to change the size of the time step. Then we move on to a standard Verletet method for the remainder of the simulation. The update methods for these two methods are shown in equations 1 and 2.

$$r_{i+1} = r_i + (r_i - r_{i-1})\frac{\Delta t_i}{\Delta t_{i-1}} + a\frac{\Delta t_i + \Delta t_{i-1}}{2}\Delta t_i \tag{1}$$

$$r_{i+1} = 2r_i - r_{i-1} + ah^2 \tag{2}$$

3 Results

First, I ran a simulation with a total length of 2 Gyr. Shown below in Figure 1 is the end configuration of the galaxies in this simulation. I ran this code with a time step of 10 Myr.

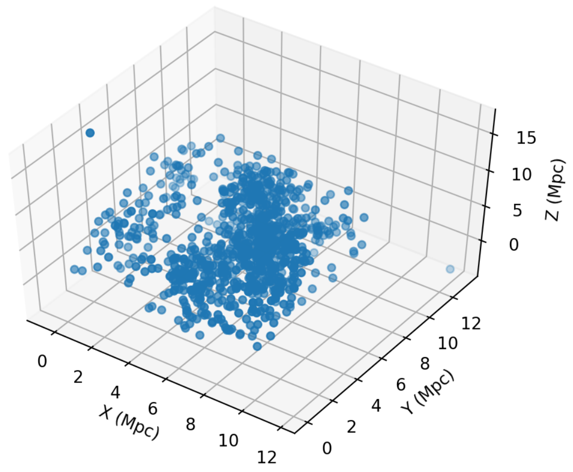

Figure 1: End configuration of the galaxies in my simulation.

Next we wanted to estimate the Lyapunov exponent, which we can do by placing a particle at some location (we use (1 , 5 , 5) in Mpc), and then introducing some small error and running a few short simulations from here. Basically we want to see how different the trajectory of this particle is when we slightly change the initial conditions, which is an indicator of chaos. This yields $\lambda = 0.05685157536$ 1/Gyr, which was calculated using the relationship shown in equation 3. This gives us a timescale of 17.5 Gyr.

$$d_f = e^{\lambda t} d_i \tag{3}$$

The final item was to make a plot for a hypothetical potential for this galaxy cluster. This plot is shown below in figure 3. The potential is that of a uniform sphere, so outside the sphere the potential is the same as it would be for a point source, shown in equation 4. On the inside of the sphere, we only include the enclosed mass, so the potential can be given by equation 5, where M is the total mass of the cluster, and R is the size of the cluster. Once I had this potential, I just made a contour plot for a slice at z = 5 Mpc.

$$\Phi = \frac{-GM}{r^2} \tag{4}$$

$$\Phi = \frac{-GM(3R^2 - r^2)}{2R^3} \tag{5}$$

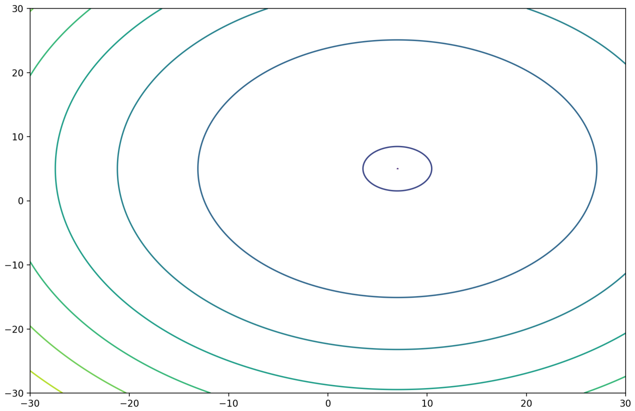

Figure 2: Slice of the potential of our distribution