

Simulating the Universe

Najla Alhamadi, Michael Cock, Rebecca Halloran, Brady Metherall, Hector Robinson

April 23, 2017

1 Introduction

In this project, our goal was to create our own realistic simulation of the universe in a cosmological scale. Our simulation starts from a redshift equal to 50 until a redshift of 0 which is the redshift at our current time. In addition, our simulation only considers matter and the cosmological constant as component of the universe.

2 A Very Big Box

Our first step was to find some parameters of our current universe, those parameters were . The current Hubble constant values is 73.8 km/s/Mpc, the density parameter of the cosmological constant is 0.73, the density parameter for matter is 0.27 and the total density parameter is 1.02. The critical density of the universe was found to be 9.277e-25 kg/m³ and the value of the matter density of the universe is 2.505e-25 kg/m³. The current age of the universe was found to be 13923.769 million years. Finally, the age of the universe at z = 50 was found to be 45.813 million years. We took a cubic section of the universe with dimensions of 50x50x50 and calculated the mass of matter inside that cube to be 1.0146e46.

3 Particle-Mesh Code

(write about the writing of the code)

In this step, we built our own Particle-Mesh Code, the input is an initial conditions N-body file and the output is a snapshot of the system at regular intervals. The only parameter that we could manipulate is the time step. After some testing, the value of the time step was chosen to be 0.0025. The code was tested by creating two systems; the first one was to create a perfect grid of particles, evenly spaced. Since theres no difference in density from the average value, the grid should persist throughout the simulations and that is what is observed in our results in Figure 1. The second system was to create a box of randomly placed particles. This system should form small structures where initially overdense regions are located, and void where initially underdense regions are. Figure 2 shows the results of the second system which agrees with the expected results.

4 A (somewhat) Realistic Simulation

(add how the scaling was done)

In this step, we ran our code using the initial conditions provided, after scaling the box size the results are presented in Figure 3.

5 FoF

One question to investigate in our simulations is the structures formed, how much of the mass do they contain? It is difficult to define exactly where a structure ends, so to do this we implemented the "friend-of-a-friend" method (FoF). The idea with this method, is that groups are made up of particles within a "linking

length” of each other. If two particles have one or more common neighboring particles, they are merged into a group. It checks every particle in the system and links them together into large groups. If the linking length is chosen properly, these groups will correspond to the structures seen in the simulations.

To implement this we used a software package called FoF, created by Volker Springel. We used instructions on how to implement the code from LASTRO, which can be found here. This software provided us with information about the groups’ mass, center of mass, and size. We plotted the groups to make sure they matched up. It is seen quite obviously that the groups match up with those in the simulation. The largest two groups had masses of 29

6 Comparisons

7 answer questions here

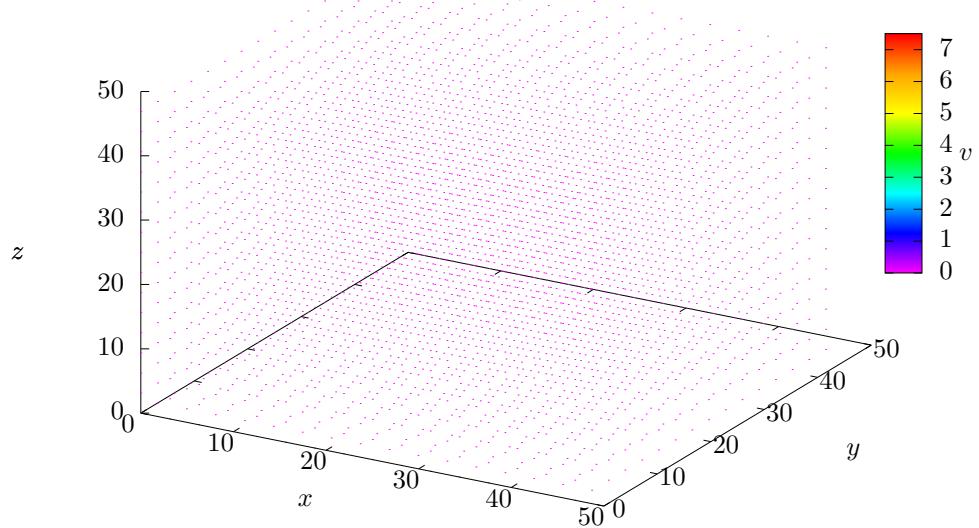


Figure 1: Final state of system with ordered initial conditions. Nothing happened here; all particles remained motionless

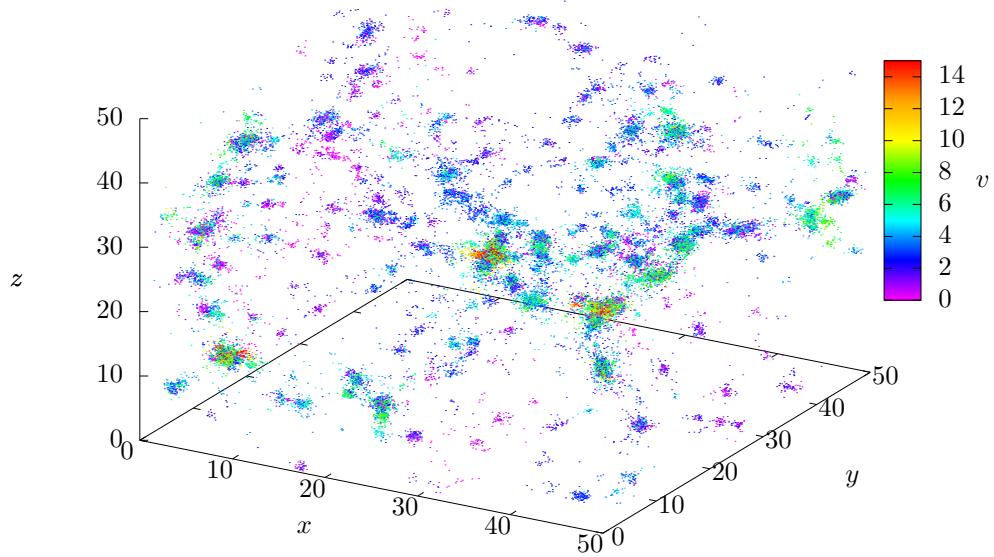
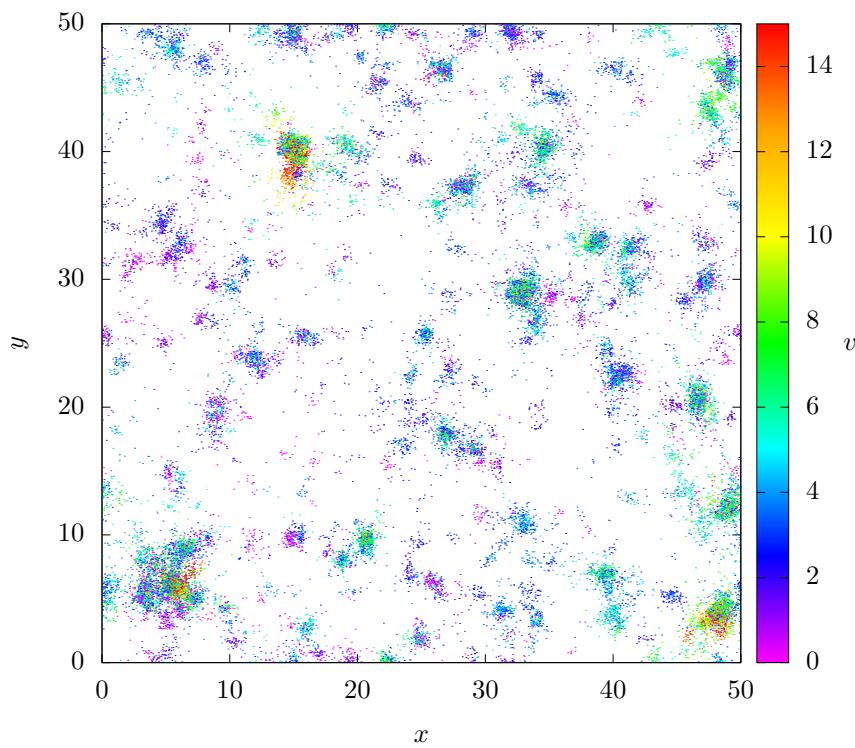
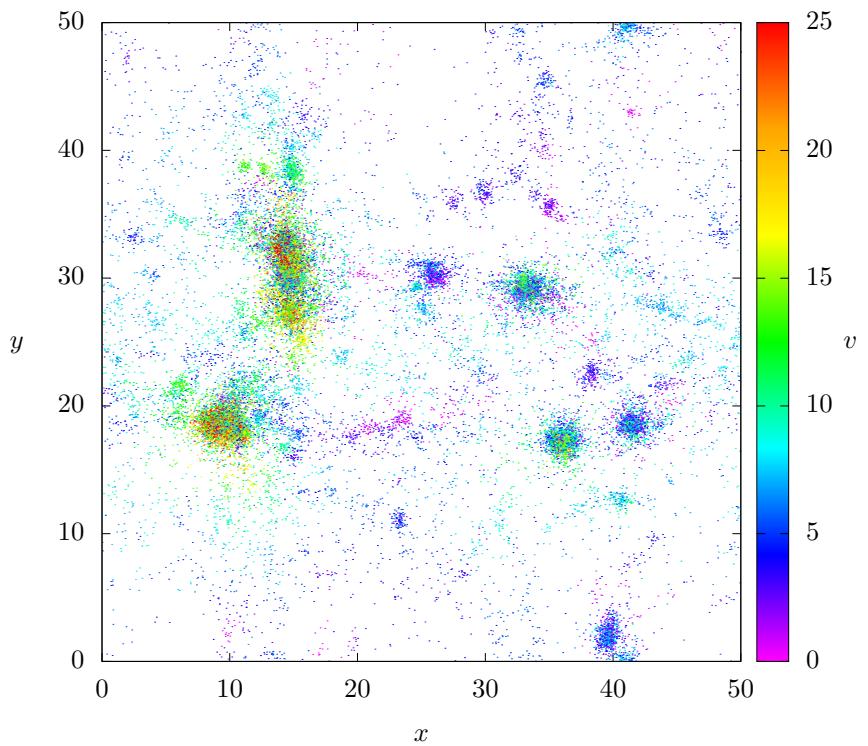
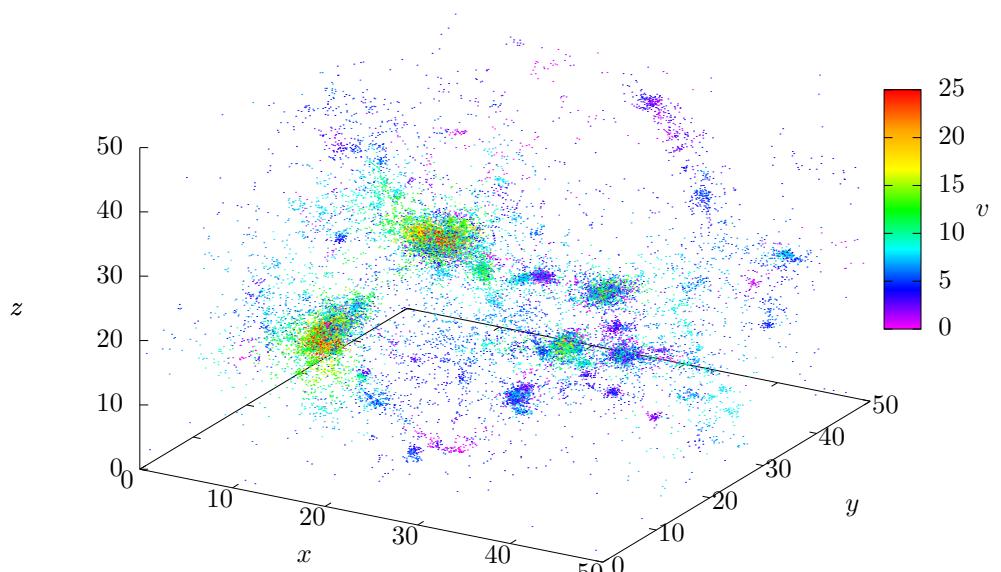


Figure 2: Final state of system with random initial conditions, shown in 2D and 3D. Color corresponds to particle velocity.



(a) Top-down view.



(b) 3D view.

Figure 3: Control simulation.

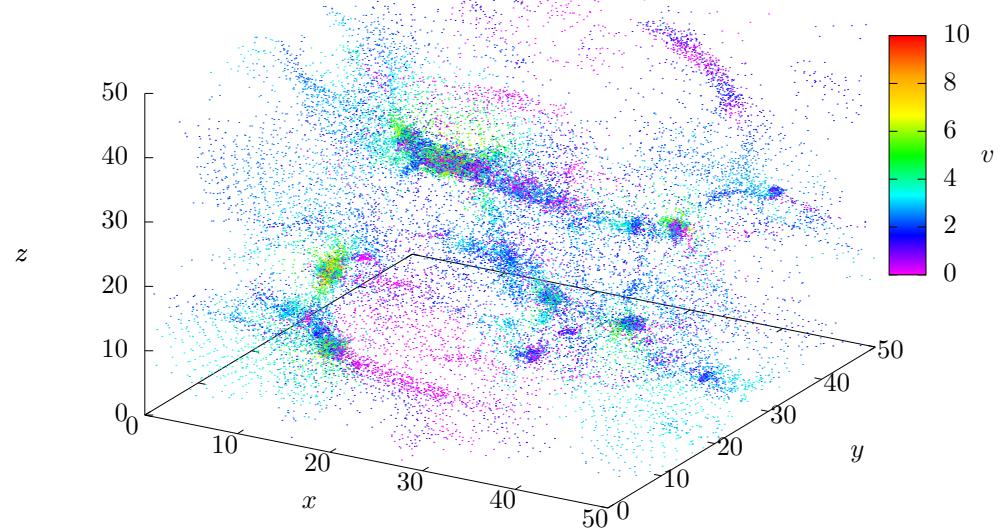
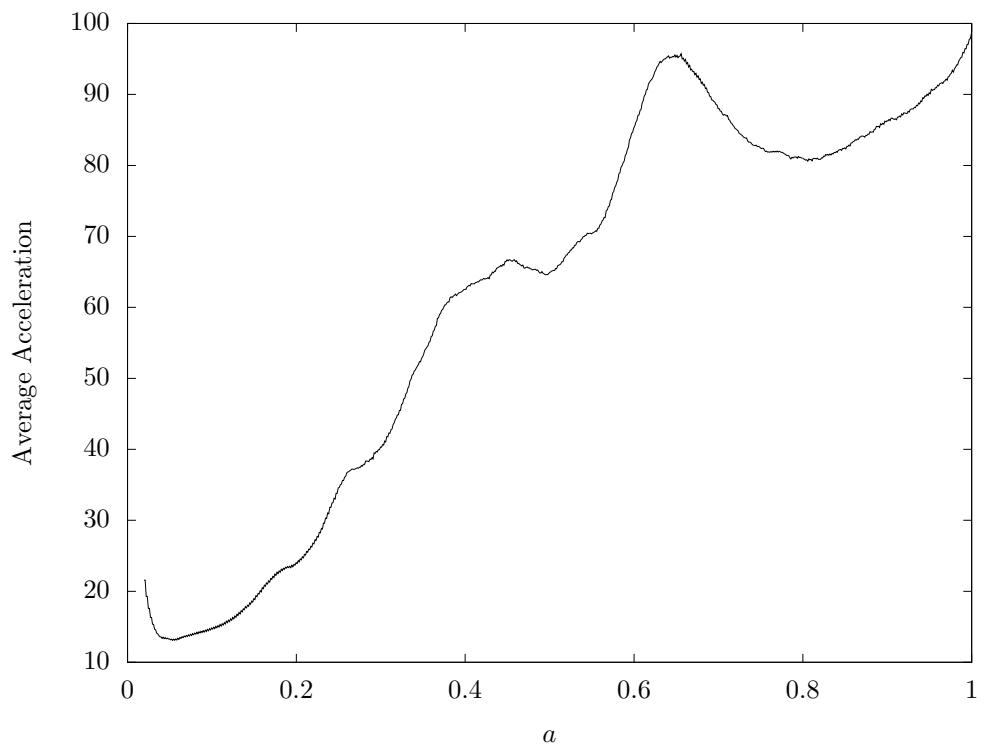


Figure 4: $N_c = n^{1/3}$

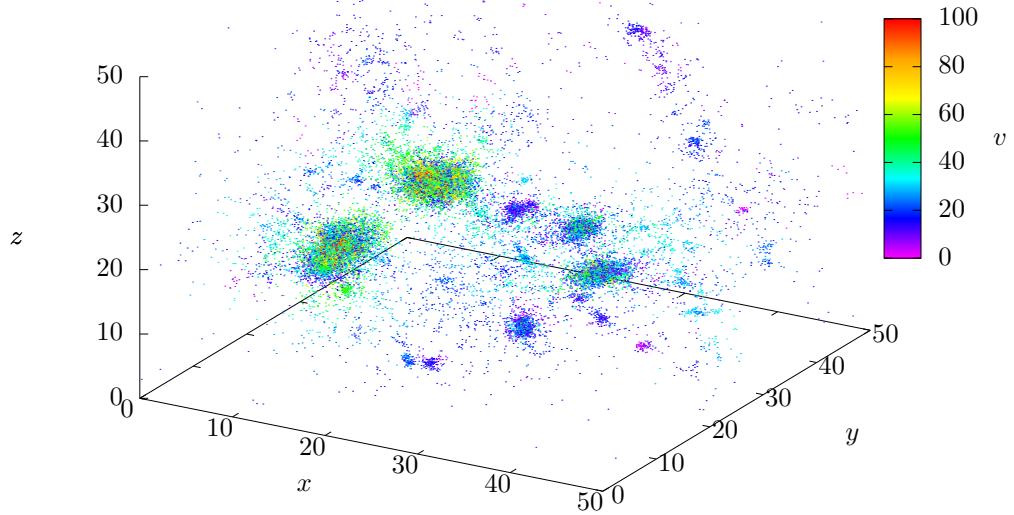


Figure 5: $\Omega_m = 3.0$, $\Omega_k = \Omega_\Lambda = 0$

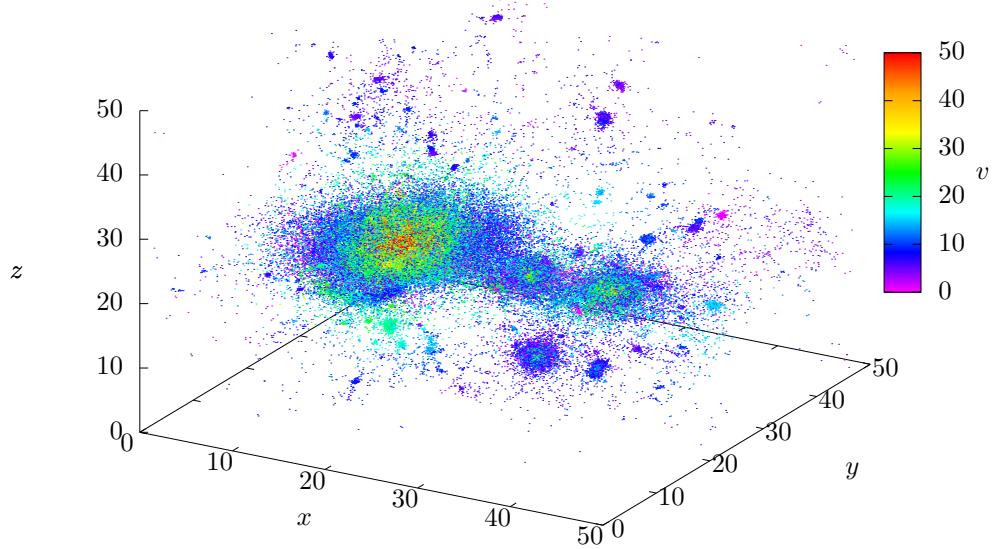


Figure 6: $n = 64^3$

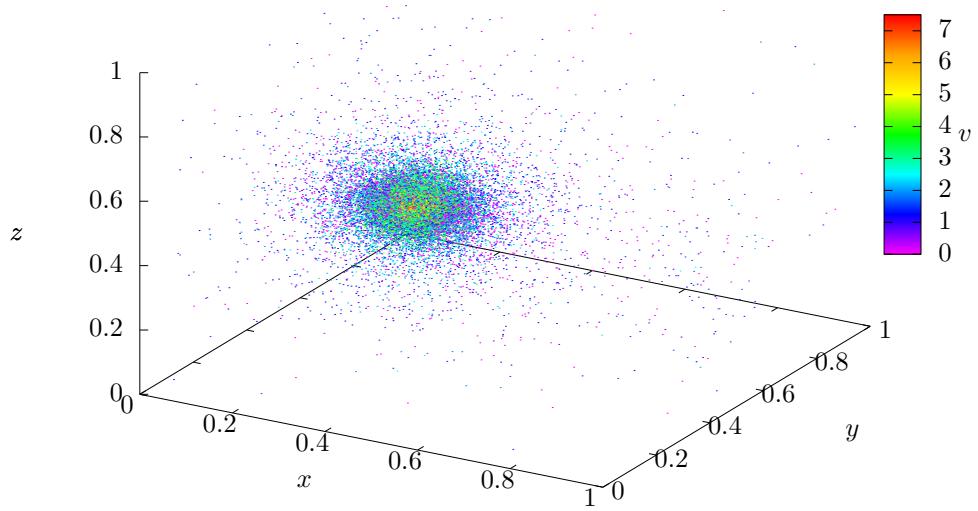


Figure 7: boxsize = 1

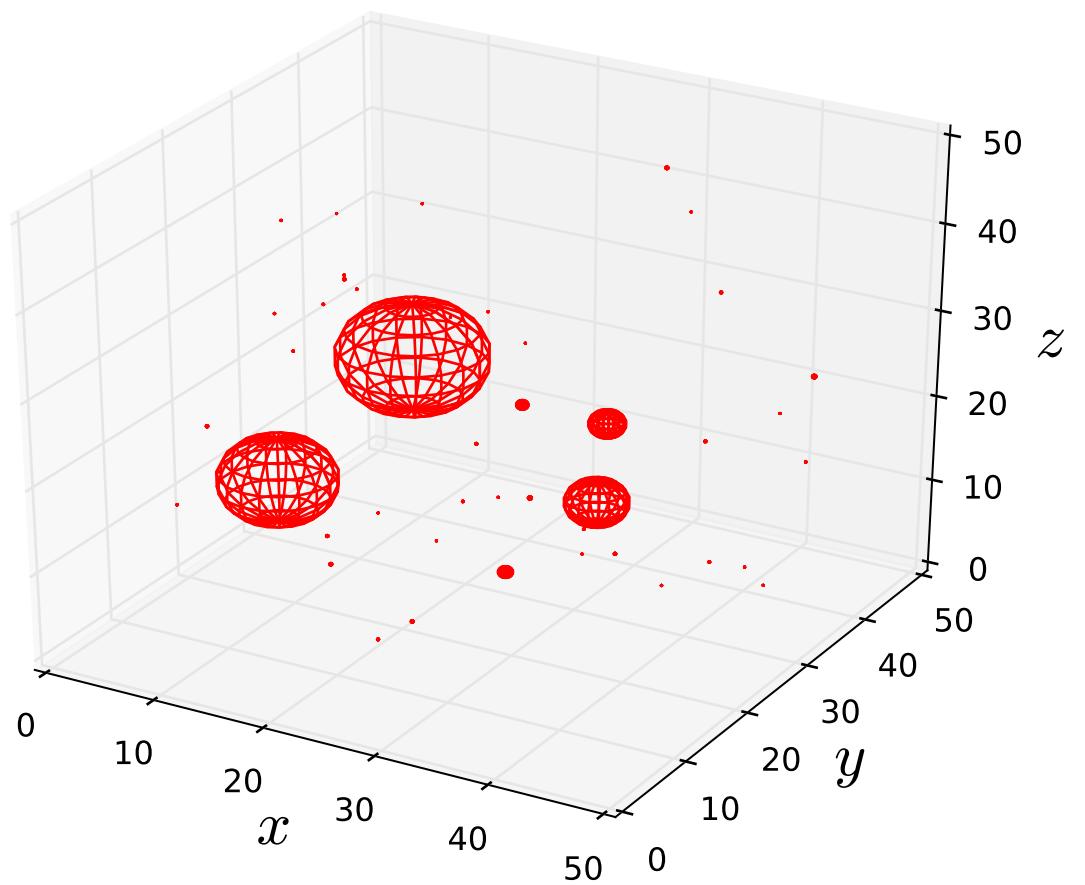


Figure 8: Groups