Draft version March 31, 2020
Typeset using LaTeX default style in AASTeX63

Barnes-Hut Algorithm for N-Body Simulations of Galaxy Clusters

Brendan Drachler[1]

[1] *Rochester Institute of Technology*
Rochester, NY. USA.

1. INTRODUCTION

On a clear night you can see countless number of stars in the night sky. They serve to remind you that the Milky Way is a complex interplay of stars exchanging mutual gravitation on each other while they make their way around the galaxy. But this complex interplay happens on an even bigger, more unimaginable scale, where the interacting bodies aren't stars like the sun but are galaxies like the entire Milky Way. These structures, called galaxy clusters, are the largest gravitationally bound structures we've observed in the Universe. The galaxy cluster that contains the Milky Way is called the Local Group and contains an additional 35 gravitationally bound galaxies (van den Bergh (2000)).

The Local Group is undeniably massive with an estimated mass, $\mathrm{M} = (2.3 \pm 0.6) \times 10^{12}\ \mathrm{M}_\odot$, and radius, $\mathrm{R}_0 = 1.10 \pm 0.15$Mpc. But it pales in comparison to nearby galaxy clusters like the Virgo cluster which has nearly 1300 identified members (Binggeli et al. (1987)). Unfortunately, we cannot directly observe many of the interactions that take place in galaxy clusters because the timescales are far greater than a human lifetime. To address this, we can simulate a galaxy cluster into the future and address several key questions about the dynamical processes that occur within them. One way this is typically done is with the use of brute force N-body simulations where the gravitational force for a given galaxy has to be computed by summing the gravitational force from every other galaxy. For large galaxy populations, the number of computations grows rapidly to an unmanageable level if we perform N-body simulations in this way. Instead, we will use a Barnes-Hut algorithm to vastly speed up the brute force technique present in many N-body simulations.

In Section 2 we will present the theory and properties of the Barnes-Hut simulation we performed and justification for the simplifying assumptions that were made. In Section 3 we will present the results of the simulation over varying timescales and analyze the observed galactic dynamics.

2. METHODOLOGY

The Barnes-Hut algorithm uses a tree structure that resembles Fig. 1 except in our case, we are working in a 3D domain. The tree starts off as the root node. The root node will reproduce into 4 quadrants in 2D and 8 quadrants in 3D, these sub-nodes are children of their parent node. The reproduction will continue happening in each child node until only one galaxy occupies each node as shown on the left in Fig. 1. The nodes containing only 1 galaxy at the end of our tree structure are referred to as leaf nodes.

The point of the tree structure is to help us avoid computing the gravitational force of each galaxy on every other one. Instead, if a node is far enough away from a point in question, we can calculate the center of mass of the node and treat the entire node as a single mass point. Thereby, we reduce the number of computations greatly. If the points are closer than some defined distance, we will sum the forces like a classical brute force N-body code. How can we actually define that distance? In this work, it is defined as being the ratio between the length of the node, L and the distance to that node, D. If the ratio $\frac{L}{D}$ is significantly small, then the galaxy a distance D away sees that node as being a single mass point. Our limiter is defined as:

$$\frac{L}{D} \leq 0.7 \tag{1}$$

The simulated cluster will have 908 member galaxies all with the same mass, $M = 10^{12}\ \mathrm{M}_\odot$. After every iteration of the code, we will insert the galaxies into a new tree structure and calculate the forces on each. This can be done for any number of iterations but we choose to evolve the cluster until the magnitude of the position vector for a single galaxy has advanced 3 Mpc from its initial point. The time step of each iteration will be 1 Myr though this is a parameter that can be changed. 1 Myr appears to be the most efficient timescale while still maintaining accuracy. No quantitative analysis was done to validate that.

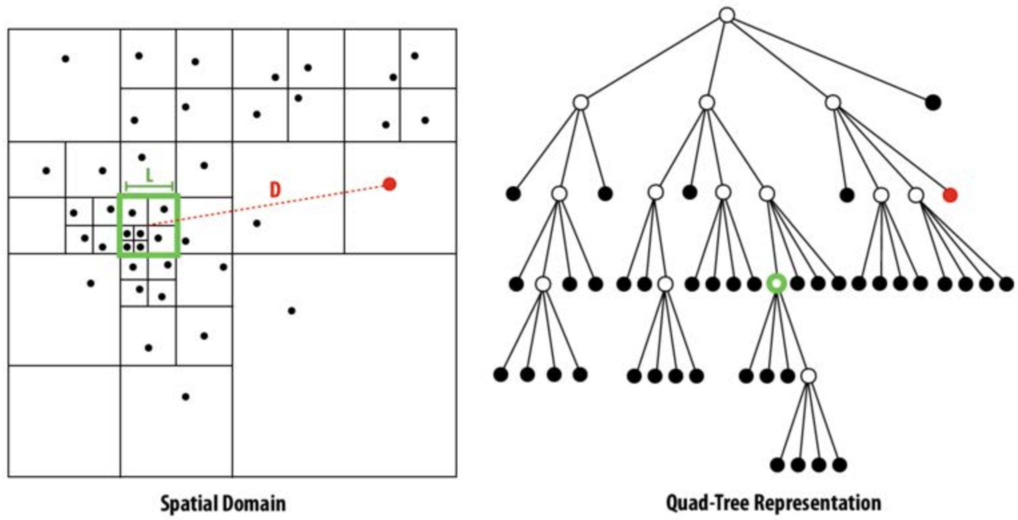

Figure 1. The tree decomposition of a 2D domain and the corresponding Quad-Tree representation.

We've also included force softening into our algorithm. This prevents any singularities or unstable points in Eq. 2 associated with galaxies that come too close ($r \to 0$) to each other.

$$\vec{a}_\alpha = \sum_{\beta \neq \alpha} G m_\beta \frac{\vec{r}_\beta - \vec{r}_\alpha}{|\vec{r}_\beta - \vec{r}_\alpha|^3} \tag{2}$$

It is apparent that this equation will diverge as $|\vec{r}_\beta - \vec{r}_\alpha| \to 0$. To address this, we will institute a force softener that will identify when $|\vec{r}_\beta - \vec{r}_\alpha| \leq \epsilon$ and treat the acceleration as in Eq. 3:

$$\vec{a}_\alpha = \sum_{\beta \neq \alpha} G m_\beta S_{\mathrm{F}} \left(|\vec{r}_\beta - \vec{r}_\alpha|\right) \frac{\vec{r}_\beta - \vec{r}_\alpha}{|\vec{r}_\beta - \vec{r}_\alpha|} \tag{3}$$

where S_{F} is defined as:

$$S(\vec{r}|\epsilon) = -\frac{1}{\sqrt{r^2 + \epsilon^2}} \tag{4}$$

We've chosen ϵ to be $1pc$ because on our global domain, given in Mpc, we will begin diverging at distances lower than $1pc$.

3. RESULTS

Snapshots of the simulated galaxy cluster are shown in Fig. 2. One galaxy is colored red in hopes the reader can notice its movement. Movement is also apparent in the upper right group of galaxies. To get a better visualization, I encourage the reader to watch the movie of the simulation uploaded to the github page associated with this code. Over the course of the total simulation time, $t_{sim} = 1Gyr$, nearly all galaxies have moved $\approx 1Mpc$.

The timescale over which an average galaxy in our simulation moves $1Mpc$ is actually a great validation test for our code. In most clusters, galaxies are observed to have velocities on the order of $\sim 1000km/s$ (Yahil & Vidal (1977)). Using this, we can calculate

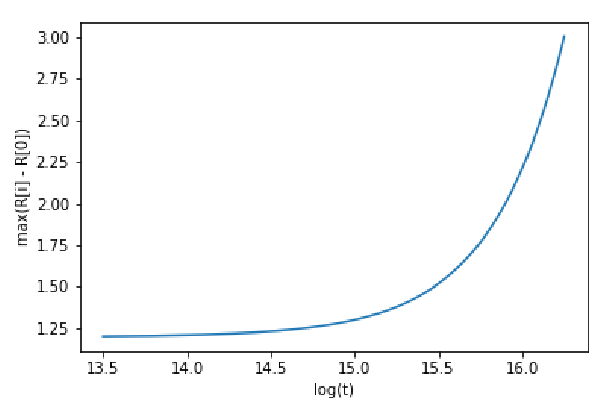

Figure 2. The time it took for galaxies to travel 3 Mpc.

the crossing time, t_{cross} - how long it takes a galaxy with $v = 1000km/s$ to move 1 Mpc.

$$t_{cross} = \frac{1Mpc}{1000km/s} \approx 1Gyr \tag{5}$$

This result validates the finding of our code that the average galaxy took $1Gyr$ to move $1Mpc$. We can trace the timescale it took for particles to move an arbitrary number of Mpc's. This is done in Fig. 2.

N-body simulations are known to exhibit chaotic behavior (Kalapotharakos (2008)). We can measure the Lyapunov exponent of galaxies in our simulation to quantitatively understand the timescale over which chaotic behavior occurs in our simulations. This will be done by inserting a new galaxy into our population with a given set of coordinates, $\boldsymbol{R_o}$. We will evolve the simulation while tracking the galaxy we've inserted. We can then insert the galaxy into another simulation at the same coordinates but with a small added offset, $\boldsymbol{R_o} + \delta$. The reason for doing this is to measure how the trajectories diverge from each other. For our small offset, we will choose $\delta = 10pc$. Ideally, we would choose a smaller offset but the simulations take on order $\sim$ 10 hours to evolve and we want to ensure that at the end of the simulation time, our trajectories differ enough to measure the Lyapunov exponent. For a separation of $\delta = 10pc$, the time to divergence may take longer than $1Gyr$.

The Lyapunov exponent will be defined as:

$$\lambda = \frac{1}{t} ln(\frac{R_\delta}{R_o}) \tag{6}$$

We can also calculate the Lyapunov timescale, which is defined to be the characteristic timescale over which the chaos manifests itself in the system.

$$t_{Lya} = \frac{1}{\lambda} \tag{7}$$

In our simulation, over a timescale of $600Myr$, the Lyapunov exponent was calculated to be, $\lambda = 7.27 \times 10^{-18} yr^{-1}$. The corresponding Lyapunov timescale is $\sim$ 10 *quadtrillion* years. It is very unlikely that it takes 10 *quadtrillion* years for chaos to manifest itself in our simulation. The more likely cause of this (wrong) number is that we have not evolved for long enough for this particular galaxy to move an appreciable amount. All of the galaxies in our simulation move with varying speeds. When our simulation detects a galaxy has moved 3 Mpc, it stops evolving. In that time, this galaxy has only moved a fraction of a Mpc. This may be addressed by giving the galaxy an initial velocity. This has not been tested.

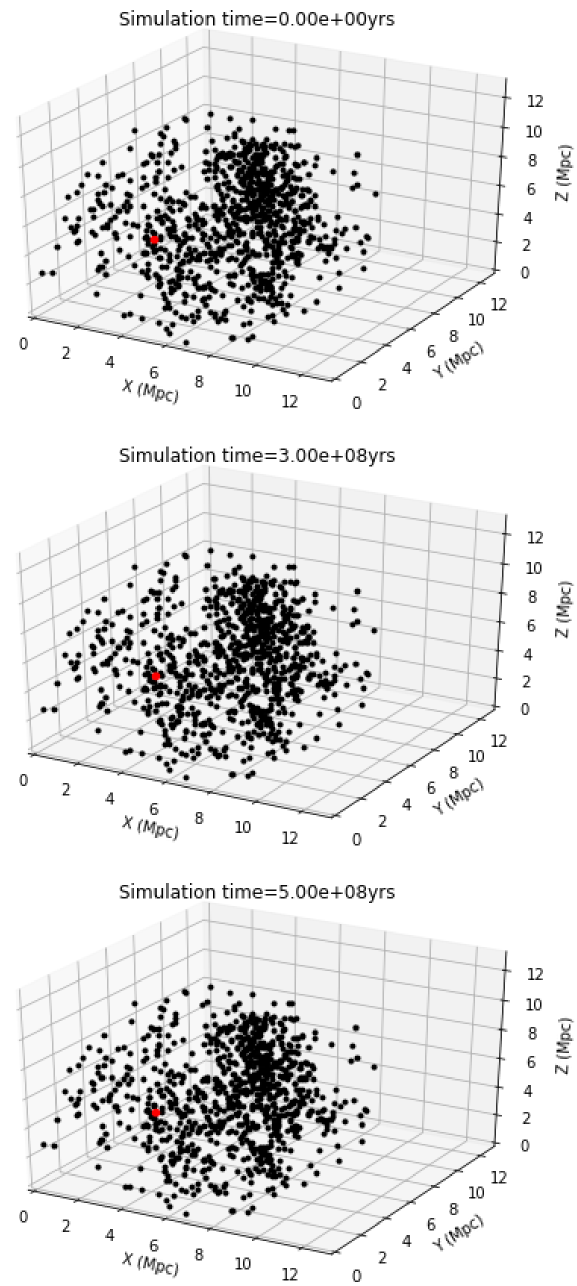

Figure 3. Evolution of the galaxy cluster.

The Barnes-Hut algorithm is an amazing upgrade from brute force N-body simulations and has been validated against observations that tell us about the motion of galaxies in clusters. Ideally, this algorithm would be parallelized and run on more powerful computing infrastructures so that we could advance the simulation much further forward. This would also allow us to bring the timestep down orders of magnitude. Furthermore, our simulation lacks many of the important features in galaxy cluster dynamics, like intergalactic gas and dust which can make up a bulk of the mass of these systems, magnetohydrodynamic effects, dark matter, etc.

REFERENCES

Binggeli, B., Tammann, G. A., & Sandage, A. 1987, AJ, 94, 251, doi: 10.1086/114467

Kalapotharakos, C. 2008, MNRAS, 389, doi: 10.1111/j.1365-2966.2008.13592.x

van den Bergh, S. 2000, PASP, 112, 529, doi: 10.1086/316548

Yahil, A., & Vidal, N. V. 1977, ApJ, 214, 347, doi: 10.1086/155257