Draft version April 26, 2021
Typeset using LATEX **preprint** style in AASTeX63

A575: Project 2

Brooke Kimsey-Miller[1]

[1] *Indiana University*
727 E 3rd St
Bloomington, IN 47405 USA

ABSTRACT

The results of four sets of initial conditions are presented. The results essentially show that homogeneous models are not good models for forming spiral galaxies. Two types of Hernquist model approximations are presented. When a disk forms in isolation, it generally expands more than if a halo were present. It also allows for parts of the disk to collapse away from the center, forming tendrils. It's possible that the final model presented may have yielded promising results to form spiral arms had a rotational velocity been added to the system. The final model, as is, forms a peanut structure.

1. INTRODUCTION

Hernquist (1993) (Hern93) had beautiful simulations of spiral galaxies that formed spiral arms. The overall goal of the project was to recreate the results from Hern93 using `Gadget2` (Springel (2005),Springel et al. (2001)). I could then plot the positions, density profiles, and velocity dispersion profiles in `Python3` utilizing the output files of `Gadget2`.

Unfortunately the equations from Hern93 proved too difficult for me to solve for positions. He explicitly states 'particle coordinates can be initialized *trivially* using a variety of techniques (e.g., Press et al. 1986)'. Well, I did not find it trivial and I also have the text he mentions. My attempts at solving these equations are found in Section 4.

I therefore switched to a homogeneous model, found in Section 3. It allowed me to build the machinery required to attempt Hern93 again.

The figures from both models are found in Section 5. The code used to plot the models, as well as the initial conditions, are found in Section 2. My failures and attempts far exceed my successes, so I therefore added Section 7. I conclude the paper with Section 6.

2. CODE

The hierarchy is structured as shown in Figure 1. The code has been uploaded to the GitHub repository https://github.com/brookeKimsey/A570_Project.

I find back importing functions difficult, so I chose to put all the codes into one folder (`src`) and subsequently tell the codes where to find the input and where to place output.

As a result, the *attempt* folder is set up for results from `Gadget2` for a halo, disk and bulge simulation *(all_results)*, an isolated disk simulation *(d_results)*, a halo and disk simulation *(hd_results)*, and a disk and bulge simulation *(db_results)*. Each **_results* folder has a *plots* folder where the created plots and .gif files are sent. Each *plots* folder has a .gif file made of the plots created.

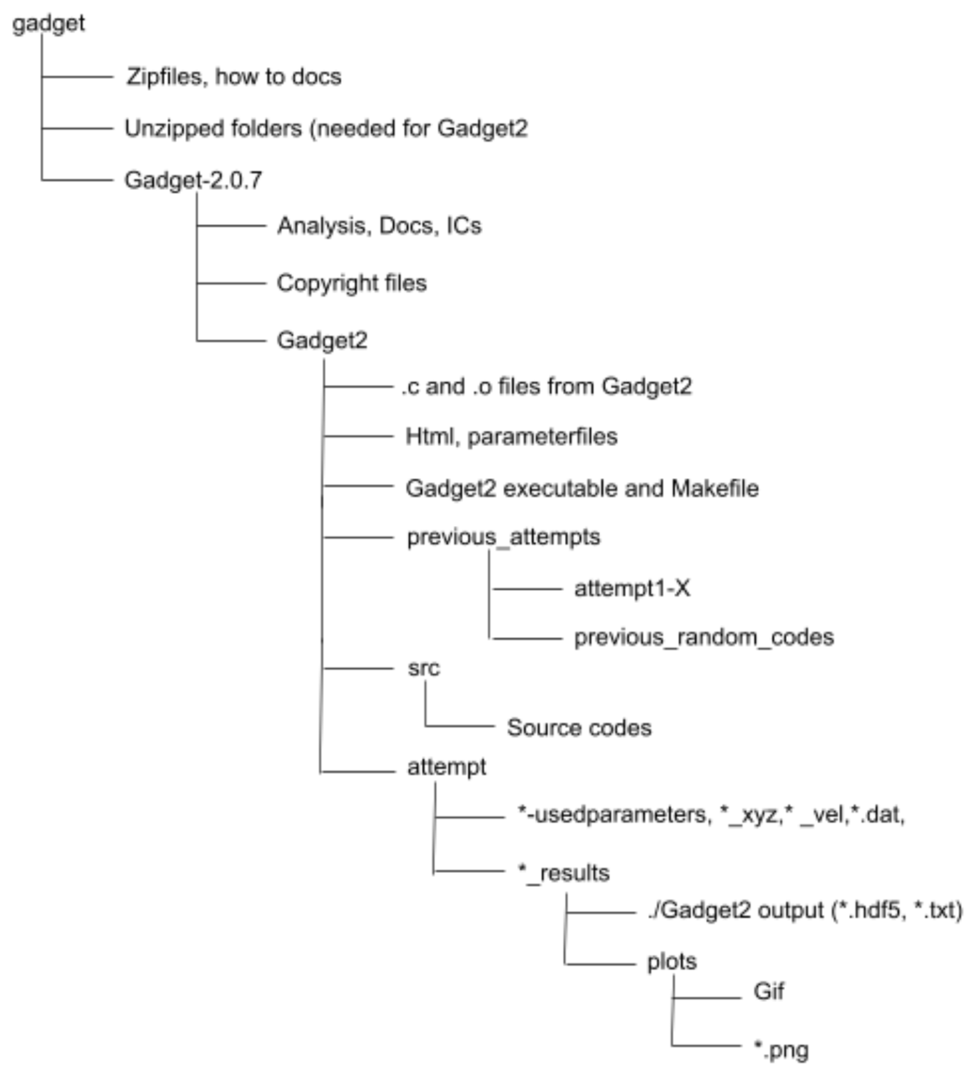

Figure 1. The structure of the project folders. The only code I've written are under *previous_attempts* and *src*. I edited the `Gadget2 Makefile`. The results for the current run are sent to *attempt*. When the attempt is complete, I move *attempt* to the *previous_attempts* folder. I rename the *attempt* folder based on the attempt number I am on. By attempt 16, I stopped running halo+disk and disk+bulge simulations.

There are two main issues from this format. One, it is difficult to write a pipeline. Two, I have to move the -*usedvalues* files to the *attempt* folder. When I forget to do this, the next run overwrites the older version. I then no longer have a reference for the values used for the simulation because I have changed the parameters files.

3. HOMOGENEOUS MODELS

The functions used for both models are found in `functions.py`, called `sph_homo_r()` and `disk_homo_r()`. They use the standard method to find random points in a homogeneous volume. The initial velocities are randomly chosen from a Gaussian. In `Python3`, the seed is based on the time clock. Therefore, each call of the built in function has a different seed and will subsequently have different random values. The figures are presented in Section 5.

3.1. *Model I*

The halo is three times more massive than the disk, the bulge is 10 times less massive than the disk (e.g, N_halo = 3 N_disk = 30 N_bulge). These ratios come from the approximate mass values in Binney & Tremaine (2008) (BT2008), page 113, Table 2.3. The halo was scaled to in between 60 and 3 provided in the table. This implies the bulge and disk simulation is approximately the disk simulation. This also implies the halo and disk simulation is approximately the same as the halo, disk, and bulge simulation. The initial velocities were found using `random.gauss()` with a mean of 1.0 and a standard deviation of 0.1.

All the created files during this run are found under */previous_attempts/attempt13*. The simulation was ran for 10 time units with a snapshot every 0.01 time units. The simulation stopped doing anything interesting around snapshot 250, or about time unit 2.5. It simply started to move out of the frame. I suppose this is equilibrium. No arm structures form, but I didn't expect them to anyways considering it's the wrong initial conditions. It wasn't feasible to put all the plots in this paper, so only a few for each have been presented. The original .gif created was over 200 Mb and would not upload to GitHub. A scaled down version of the .gif has been uploaded to the repository. The created .gif from the plots can be found in the same repository as the code, in each *plots* folder.

The results when comparing the halo+disk+bulge versus an isolated disk seem to agree well with Ostriker & Peebles (1973). The galaxies are more stable with a halo and tend to fly apart without it. The same can be said for the other simulations as well.

3.2. *Model II*

The halo is one times more massive than the disk, the bulge is 10 times less massive than the disk (e.g, N_halo = N_disk = 10 N_bulge). These ratios come from the approximate mass values in BT2008, page 113, Table 2.3. The halo was scaled to within 10 kpc. The initial velocities were found using `random.gauss()` with a mean of 10^{-3} and a standard deviation of 10^{-5}. The results are all under */previous_attempts/attempt14*.

4. HERNQUIST MODEL

The Hernquist models and approximations are derived in the following subsections.

4.1. *Derivations*

4.1.1. *Exponential Disk*

I attempted to solve the basic exponential disk in a similar fashion to the homework. For the function `exp_r()` in `functions.py` under the *src* folder, I add in the z component as a random Gaussian.

$$\Sigma(R) = \Sigma_0 e^{-R/R_d} \tag{1}$$

$$M(R) = 2\pi \int_0^R R\,\Sigma(R)\,dR \tag{2}$$

$$M(R) = 2\pi\, R_d^2\, \Sigma_0 \int_0^R \frac{R}{R_d} e^{-R/R_d} \frac{dR}{R_d} \tag{3}$$

$$u = \frac{R}{R_d}\,;du = \frac{dR}{R_d} \tag{4}$$

$$M(R) = 2\pi\, R_d^2\, \Sigma_0 \int_0^{R/R_d} u\, e^{-u}\, du \tag{5}$$

$$f = u\,;df = du\,;dg = e^{-u}\, du\,;g = -e^{-u} \tag{6}$$

$$M(R) = 2\pi\, R_d^2\, \Sigma_0 \left(-u\, e^{-u}|_0^{R/R_d} - \int_0^{R/R_d} du\,(-e^{-u}) \right) \tag{7}$$

$$M(R) = 2\pi\, R_d^2\, \Sigma_0 \left(-\frac{R}{R_d} e^{-R/R_d} - e^{-u}|_0^{R/R_d} \right) \tag{8}$$

$$M(R) = 2\pi\, R_d^2\, \Sigma_0 \left(1 - e^{-R/R_d}(R/R_d + 1)\right) \tag{9}$$

$$\frac{M(R)}{M_{tot}} = q_r = \frac{2\pi\, R_d^2\, \Sigma_0}{2\pi\, R_d^2\, \Sigma_0} \left(1 - e^{-R/R_d}(R/R_d + 1)\right) \tag{10}$$

$$1 - q_r = e^{-R/R_d}(R/R_d + 1) \tag{11}$$

Using Equation 11, I attempt to find the initial conditions in the x-y plane using an interpolation function. It yielded strange results. It would cluster in a perfect circle near the outer edges.

I therefore made a second version of an exponential disk, `exp_r_v2()`, that uses a Gaussian function to find a random radius. It is an exponential, so it was the closest approximation I could think to do. I also change the z component to a uniform distribution.

I attempted to solve the exponential disk density profile provided in Hern93 using cylindrical coordinates. I used similar steps as above which yielded

$$M(R) = M_d \left(1 - e^{-r/h}\left(1 + \frac{r}{h}\right)\right) \left(\tanh(\frac{z}{z_0})\right) \tag{12}$$

$$e^{-r/h}\left(1 + \frac{r}{h}\right) = 1 - \frac{2\pi\, q_r\, \rho_0\, h^2}{M_d \tanh(z/z_0)} \tag{13}$$

I therefore would have had the same problem with function `exp_r()`, coupled with the fact I don't know how to obtain the z coordinate either.

I also attempted to solve the exponential disk on page 112 of BT2008. Again, the math was similar to the derivation of the first disk. This yielded the following equation:

$$M(R) = 4\pi \Sigma_d R_d^2 \left(1 - e^{-R/R_d}\left(1 + \frac{R}{R_d}\right)\right) \left(\frac{1}{2}\alpha_0(1 - e^{-z/z_0}) + \frac{1}{2}\alpha_1(1 - e^{-z/z_1})\right). \tag{14}$$

I didn't solve for this r because I didn't know how to find the z component.

4.1.2. *Halo*

I decided to used the Hernquist Model from BT2008. We derived this in Homework 2, but I have rewritten here for easier reference.

$$q_r = \frac{M(R)}{M_{tot}} = \frac{2\pi\,\rho_0\,a^3}{2\pi\,\rho_0\,a^3}\frac{(r/a)^2}{(1+(r/a)^2} \tag{15}$$

$$q_r\,(1+(r/a)^2 = (r/a)^2 \tag{16}$$

$$r = a\left(\frac{\sqrt{q_r}}{1-\sqrt{q_r}}\right) \tag{17}$$

The halo seemed to be reasonable, with the exception it has very large radii. This function is simply called `halo_r()` in the `functions.py` file. I think it would have been a fun exercise to try to replicate Dubinski et al. (2009) work on the NFW halo and bar instabilities. However, I'm not sure if any of my results would constitute a bar. It would be interesting to see how the dark matter halo density profile would change the simulation.

4.1.3. *Bulge*

I attempted to solve the bulge density profile in Hern93. I had hoped to eventually turn it into a tri-axial system.

$$\rho_b = \frac{M_b}{2\pi ac^2}\frac{1}{m(1+m)^3},\;\; m = \sqrt{\frac{x^2+y^2}{a^2}+\frac{z^2}{c^2}} \tag{18}$$

By assuming $a = c$, I reduced m to $m = r/a$, and could solve for r in the standard way.

$$M(R) = 4\pi\int_0^r dr\,r^2\,\rho(r) = \frac{4\pi M_b}{2\pi a^3}\int_0^r \frac{r^2\,dr}{\frac{r}{a}\left(1+(r/a)\right)^3} \tag{19}$$

$$M(R) = 2\,M_b\int_0^r \frac{\frac{r\,dr}{a^2}}{(1+(r/a))^3} \tag{20}$$

$$u = r/a;\; du = \frac{dr}{a} \tag{21}$$

$$M(R) = 2\,M_b\int_0^{r/a}\frac{u\,du}{(1+u)^3} = 2\,M_b\left(\frac{-2u-1}{2(u+1)^2}\right)_0^{r/a} \tag{22}$$

$$M(R) = M_b\left(1-\frac{2r/a+1}{(1+r/a)^2}\right) \tag{23}$$

Here, it is obvious the units are in mass, so I was confident it was reasonable. Further, as $r\to 0$, $M(R)\to 0$; as $r\to\infty$, the denominator goes to ∞ faster than the numerator, and therefore, $M(R)\to M_b$.

$$q_r = \frac{M(R)}{M_{tot}} = \frac{M_b\left(1-\frac{2r/a+1}{(1+r/a)^2}\right)}{2\pi\rho_0 a^3} \tag{24}$$

$$\frac{q_r 2\pi a^3 \rho_0}{M_b} = c = \left(1 - \frac{2r/a + 1}{(1 + r/a)^2}\right) \tag{25}$$

I realize (now as I type this) that $M_{tot} = M_b$. However, this is not what the script is and therefore, the full derivation for the script function is shown. I wrote a version 2 of the same code correcting this error. The correction is simply replacing $c = q_r$. After lots of tedious algebra, I yielded

$$r = \pm\frac{\sqrt{c} - c}{c - 1}. \tag{26}$$

Since $\sqrt{c} < c$ and $r > 0$, then

$$r = \frac{\sqrt{c} - c}{c - 1}. \tag{27}$$

5. FIGURES

I had a hard time deciding which figures I wanted to present. I created multiple versions. In Figure 1, the **_results* includes different versions of the plots. However, *plots_v2* and *plots_v3* are only found under *all_results* and *d_results*. By attempt16, I stopped running simulations of the versions of the disk+bulge and halo+disk. I did this because from the other .gifs and plots revealed that, essentially, the disk+halo+bulge ≈ disk+halo and disk = disk+bulge. This is mainly (I think) due to the ratio of the particles. All the figures and .gifs can be found on the GitHub cite under the different attempts and different plot versions. Each version has a .gif created, which makes viewing the plots efficient. All figures can be found on the GitHub cite.

Most of the papers I looked at on .arxiv seemed interested in the disk component and how it changes in different simulations. I therefore chose to only present one of each version of the plots, but to focus on the disk component. The details about each figure are found in the Figures themselves. The discussion between the different results are left for Section 6.

5.1. *Replicate Hern93 Figures*

I wanted to recreate similar plots presented in Hern93. The script to create these files is located in *src* titled `hern_plots.py`.

5.2. *First Version of Plots*

The first version of the plots are found in the *plots* folder of each attempt. It plots the total system, halo, disk, and/or bulge. The velocity dispersion and density distribution are plotted as well. Each combination (halo+disk+bulge,halo+disk,disk, and disk+bulge) had plots created and a .gif created. The scripts that created these files are located in *src* titled `read_snapshots.py` and `create_gif.py`. It was difficult to pick a scaling for the x,y,z, the radius, and the different distributions. If too high or too low, different features were lost. If the scale was allowed to vary, it makes the .gif look odd and it makes it difficult to visually see what the system is doing. A version was made for each combination run for the simulation.

5.3. *Second Version of Plots*

This version shuts off all tick marks in the plots. This makes is easier to see the whole simulation, but the halo expands much farther out than the disk. It makes it difficult to tell how the disk changes

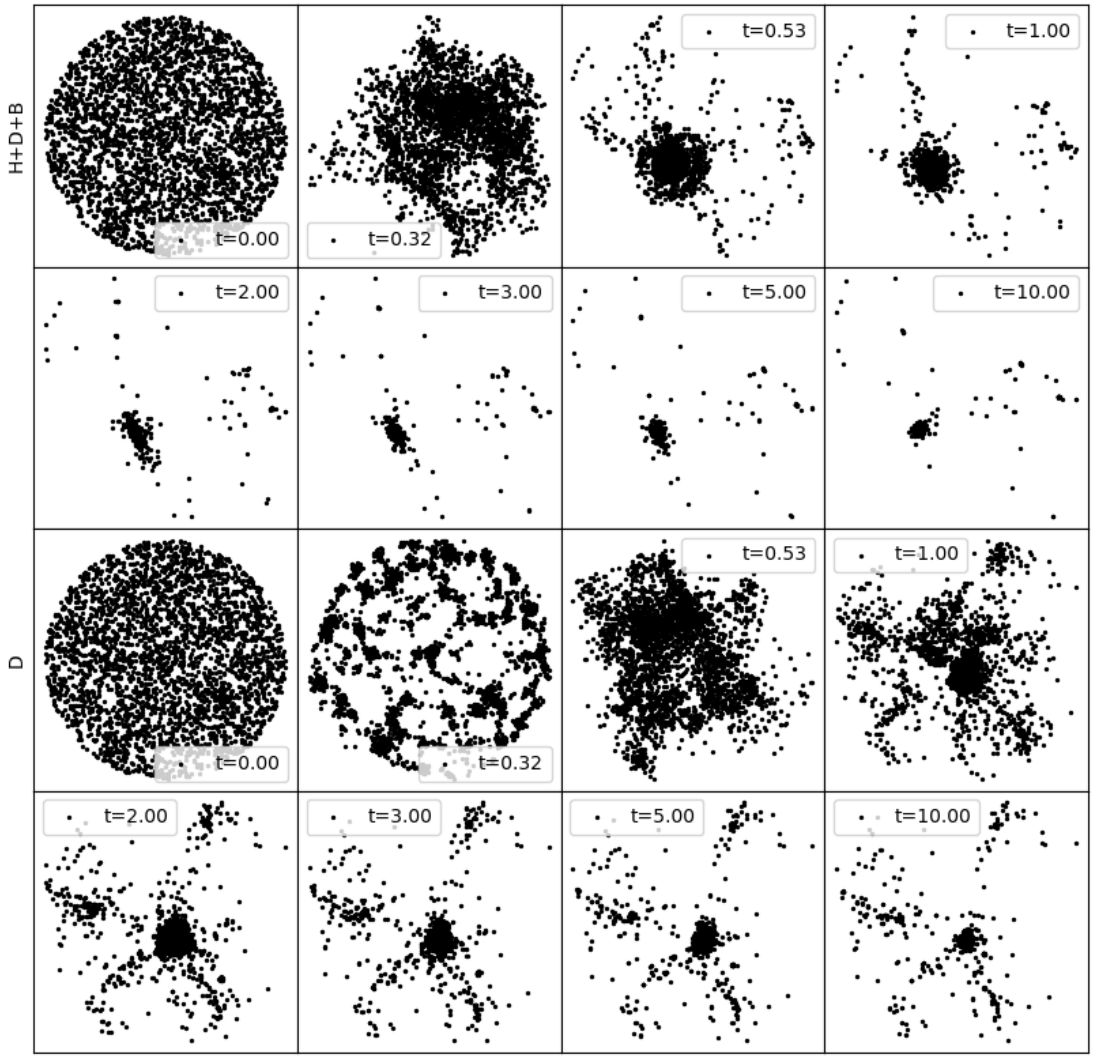

Figure 2. From Section 3.1, the evolution of the disk with a halo and bulge versus an isolated disk. The top two rows are with the halo and bulge, while the bottom two rows are from an isolated disk. The initial conditions are exactly the same. The disk with a halo and bulge collapses into a ball. A disk in isolation allows particles to collapse into different sections and form structures.

over time. The plots and .gif for each simulation with halo+disk+bulge and an isolated disk is found are the *plots_v2*. The scripts that created these files are located in *src* titled `read_snapshots_v2.py` and `create_gif_v2.py`.

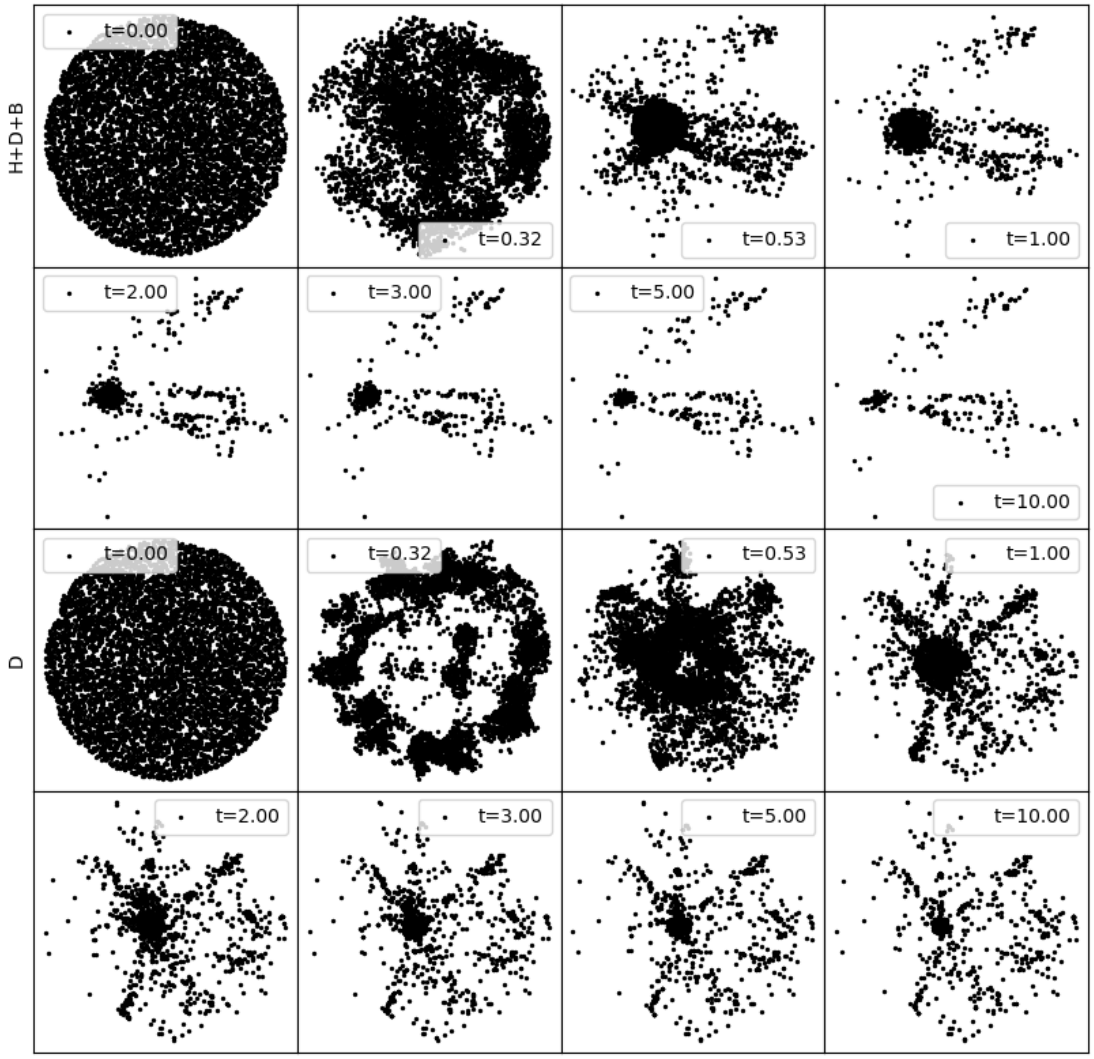

Figure 3. From Section 3.2, the evolution of the disk with a halo and bulge versus an isolated disk. The top two rows are with the halo and bulge, while the bottom two rows are from an isolated disk. A disk in isolation is more spread out than one with a halo and bulge. Recall the difference between this simulation and the one in Figure 2 are velocities and N-body ratios. The velocities are much smaller here while N_halo=N_disk=N_bulge/10.

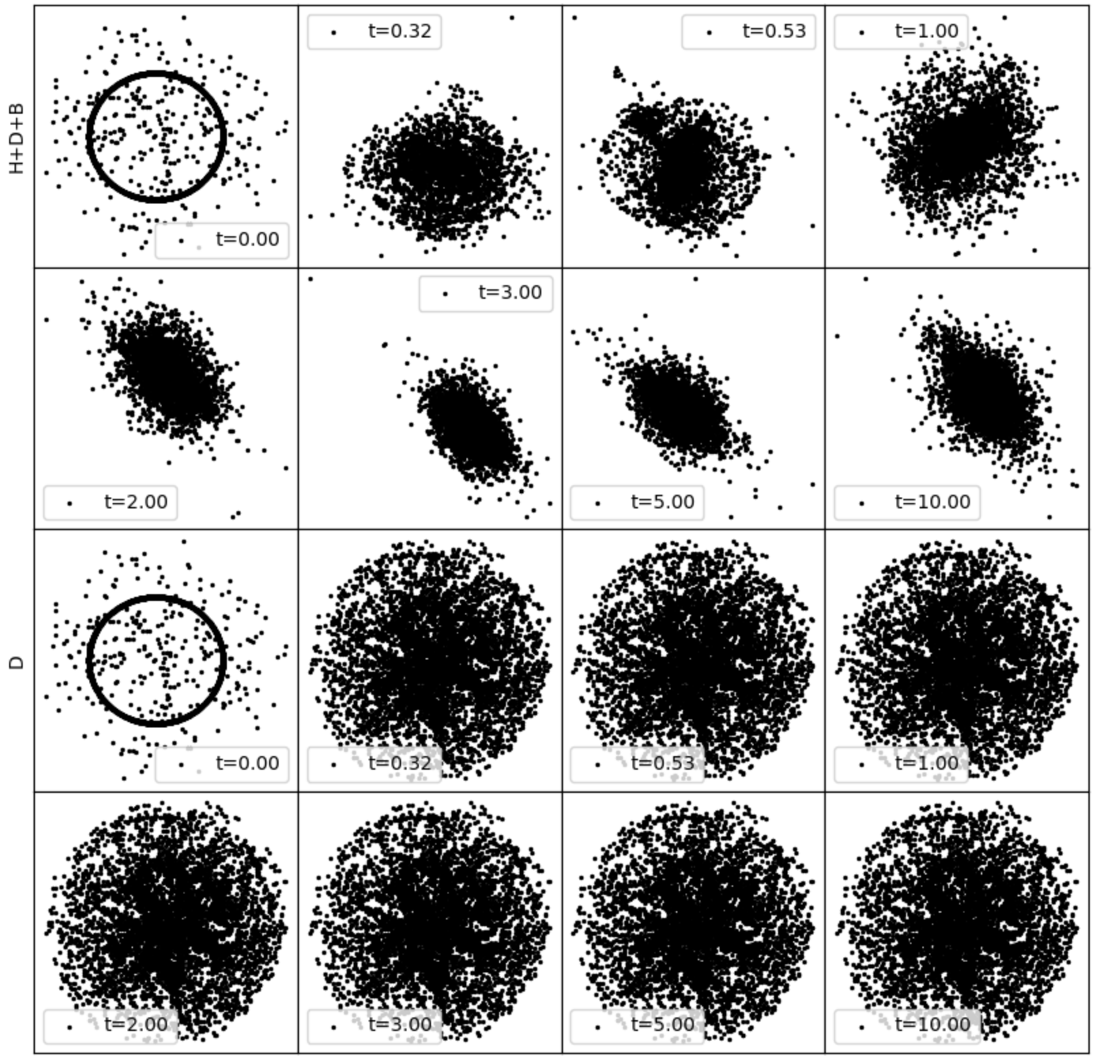

Figure 4. From Section 4, the evolution of the disk with a halo and bulge versus an isolated disk. The top two rows are with the halo and bulge, while the bottom two rows are from an isolated disk. There's clearly issues with the initial conditions. A disk with a halo and bulge with collapse, but different from the homogeneous models. It appears to be less symmetric. A disk without a halo and bulge doesn't evolve much in structure for most of its lifetime.

5.4. *Third Version of Plots*

These illustrate how the disk structure changes, which appears to be the most studied feature in the literature, I created the third version of plots. The plots and .gif for each simulation with

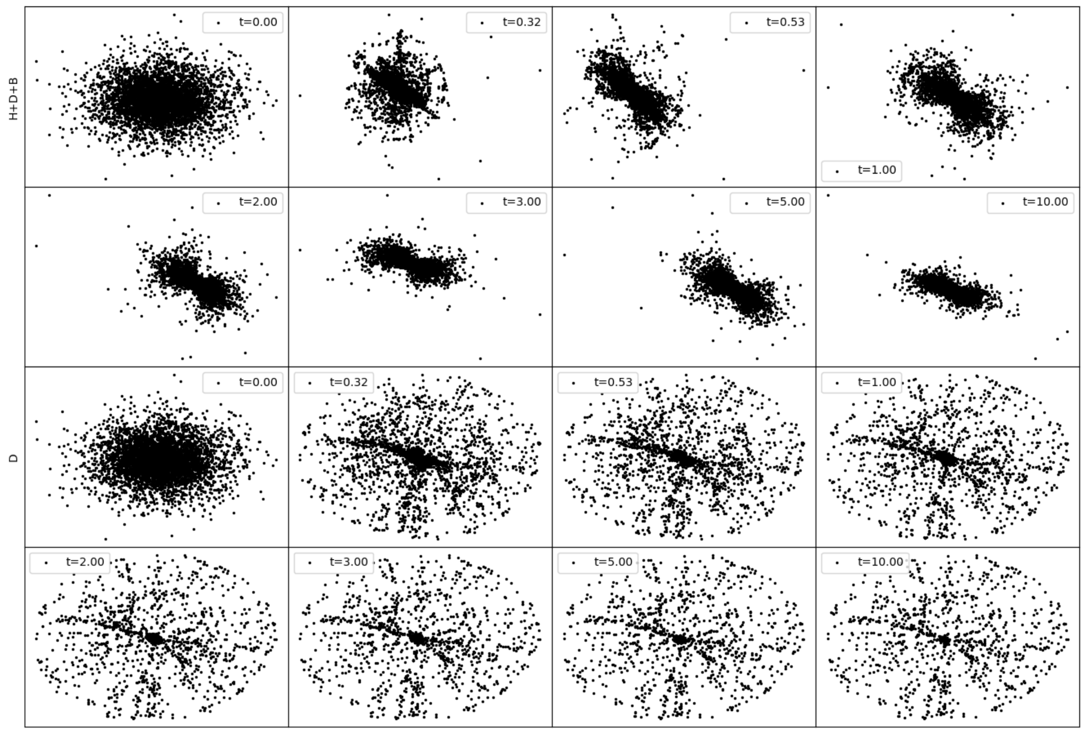

Figure 5. From Section 4, the evolution of the disk with a halo and bulge versus an isolated disk. The top two rows are with the halo and bulge, while the bottom two rows are from an isolated disk. Recall the difference between this simulation and the simulation in Figure 4 are the disk initial conditions. A disk with a bulge and halo appears to form a peanut structure, which actually is a real structure seen in some galaxies. With out a bulge and halo, the galaxy doesn't evolve much in structure for most of it's lifetime.

halo+disk+bulge and an isolated disk is found are the *plots_v3*. The scripts that created these files are located in *src* titled `read_snapshots_v3.py` and `create_gif_v3.py`.

5.5. *Interesting Features*

The peanut structure that forms is interesting. Particularly, Figures 5, 14, and 16 show the peanut feature. I think this may imply I was on the right track to get results similar to Athanassoula (2013), which specifically discusses peanut/boxy systems. Another cool feature in the Hernquist model with an approximated exponential disk forms waves as the particles bounce off each other.

6. DISCUSSION & CONCLUSION

The most egregious error I think comes from the velocity dispersion. I only gave a radial velocity dispersion and never gave it a rotational velocity. Of course, Hern93 has complicated equations to find the velocity dispersion. Perhaps if I had been able to solve those the rotation would have allowed for the arms to form. It appears that spiral arms tend to sweep up material and stabilize these arms

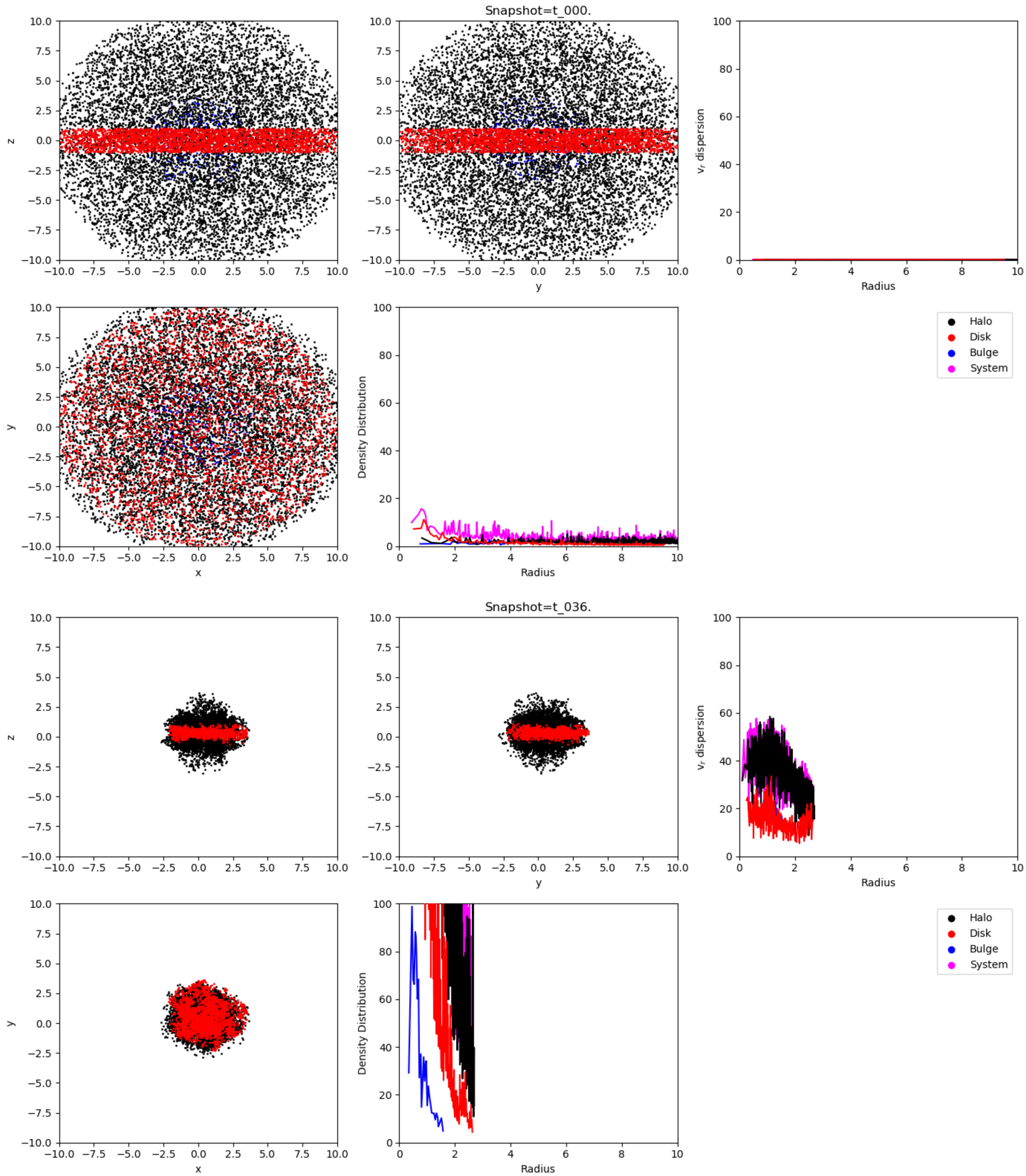

Figure 6. The initial conditions and the point of densest collapse (t=0.36) for the homogeneous model I. The particles subsequently bounce off each other and many particles are lost. This is version 1 of the plots for all models. The bulge should be place in front of the disk and halo, but unfortunately, the error wasn't resolved until later models.

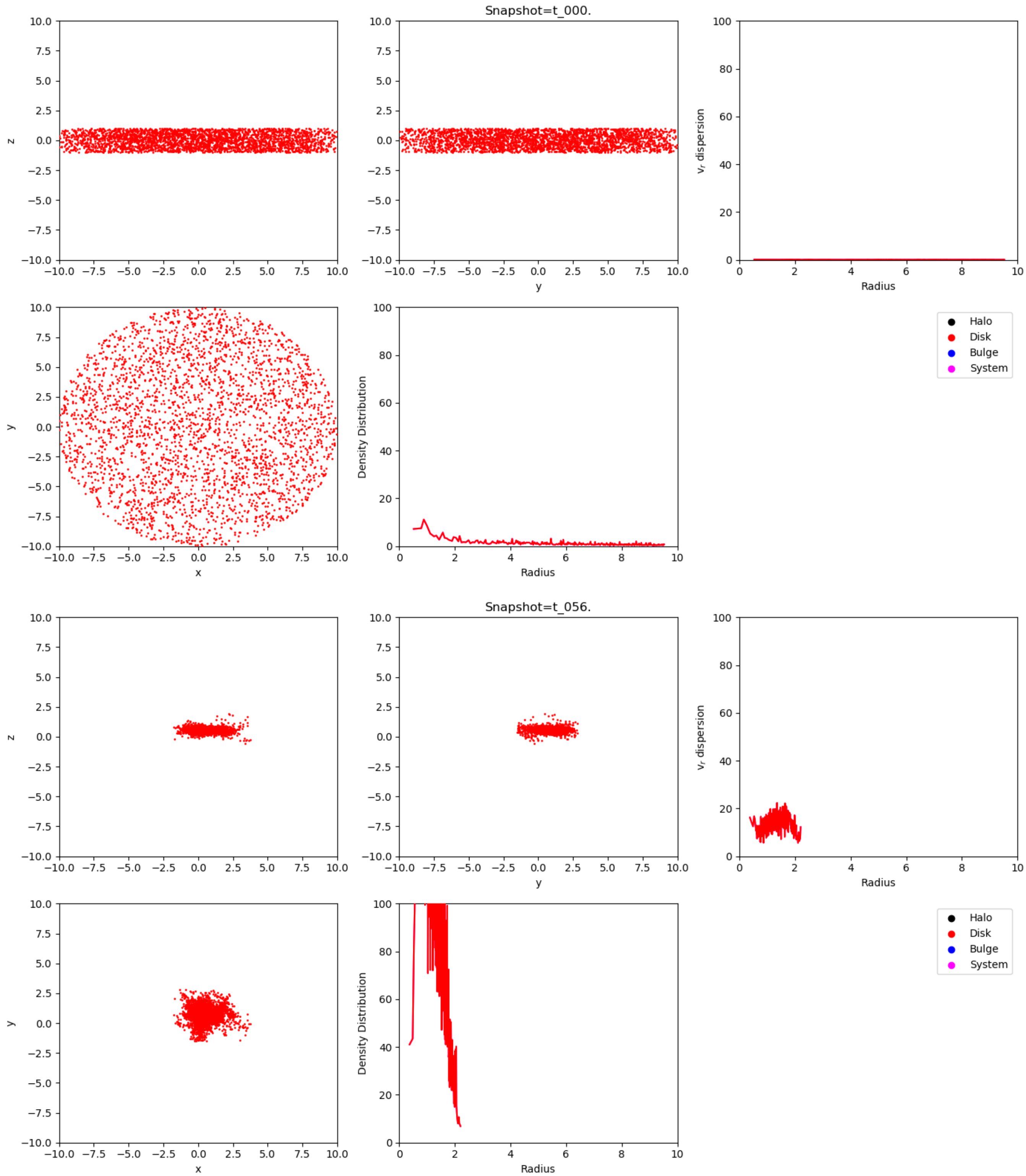

Figure 7. The initial conditions and at t=0.56 for the homogeneous model I, but with an isolated disk. The particles are at the closest point of collapse later than one with more particles.

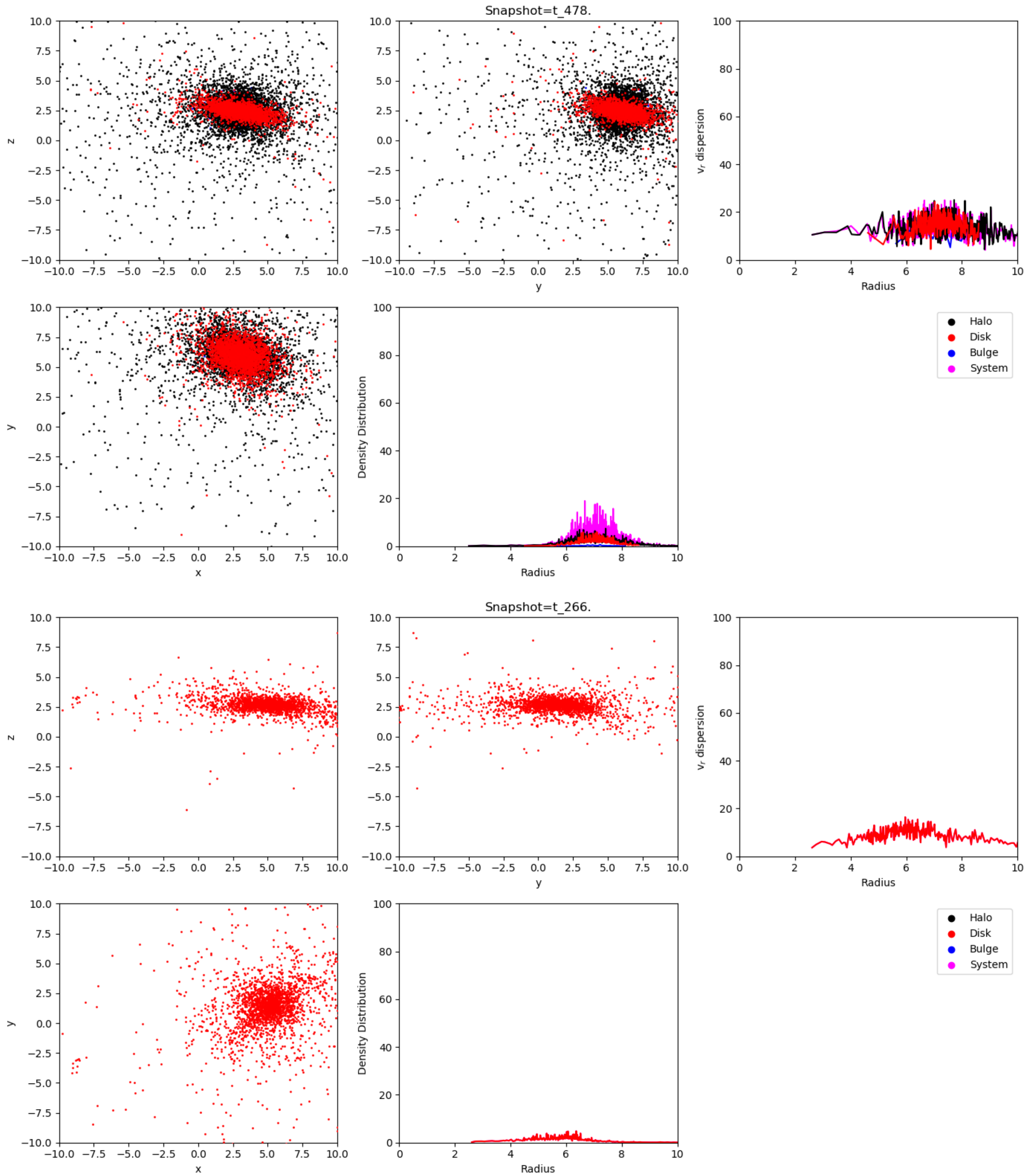

Figure 8. The first version of the plots leaves the last snapshot with the galaxies out of view. Therefore, the last snapshot with the galaxy in view is given for the disk with a halo an bulge and a disk in isolation. This is for the homogeneous model I.

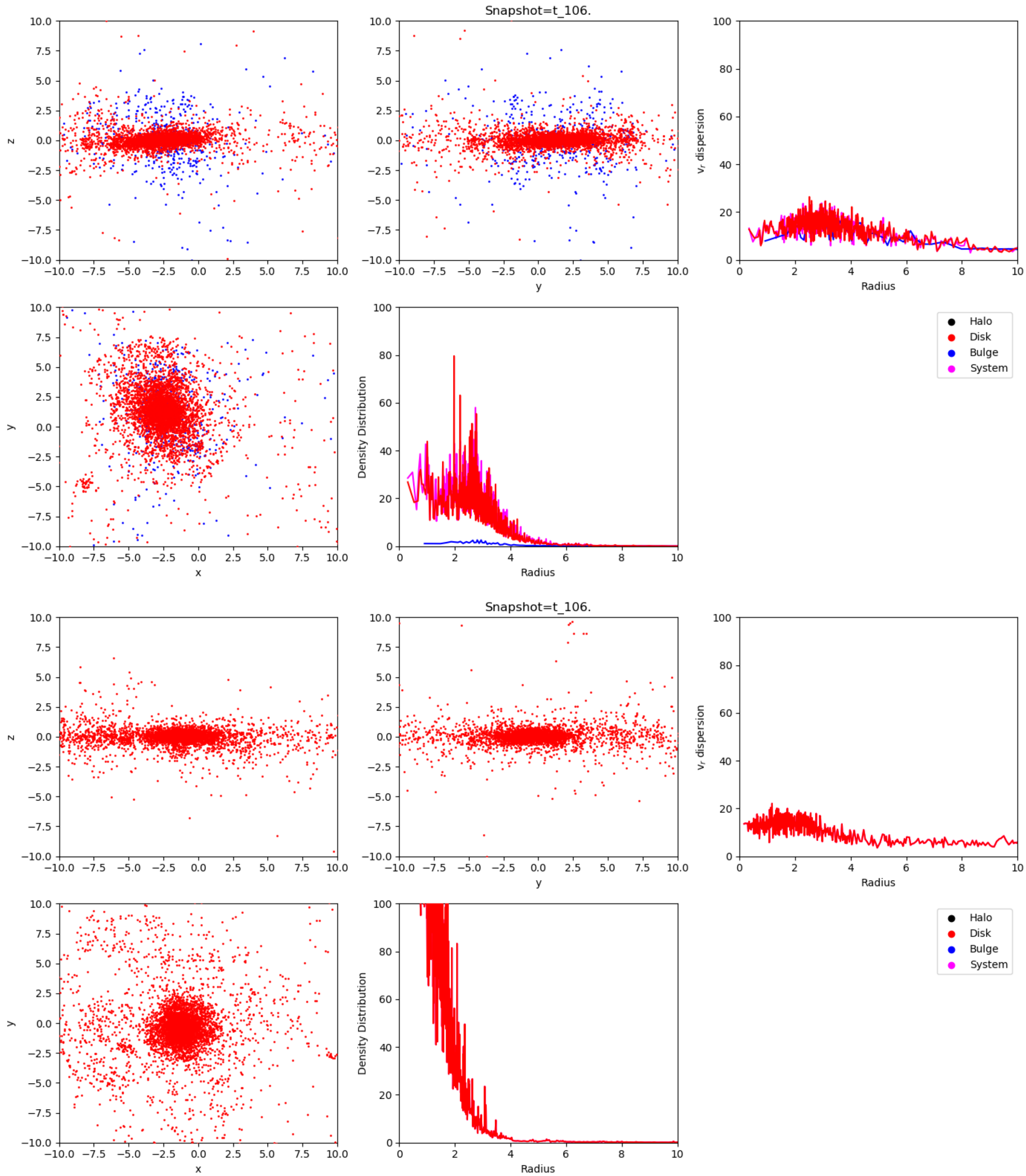

Figure 9. The bulge+disk model and the isolated disk model are shown here to illustrate with it is acceptable to only model an isolated disk. The timestamps are the same and yield similar simulations for the homogeneous model II (as well as other simulations). The difference lies in the density profile. The increase in particles (disk+bulge) appears to only change the time when the particles collapse to the highest density.

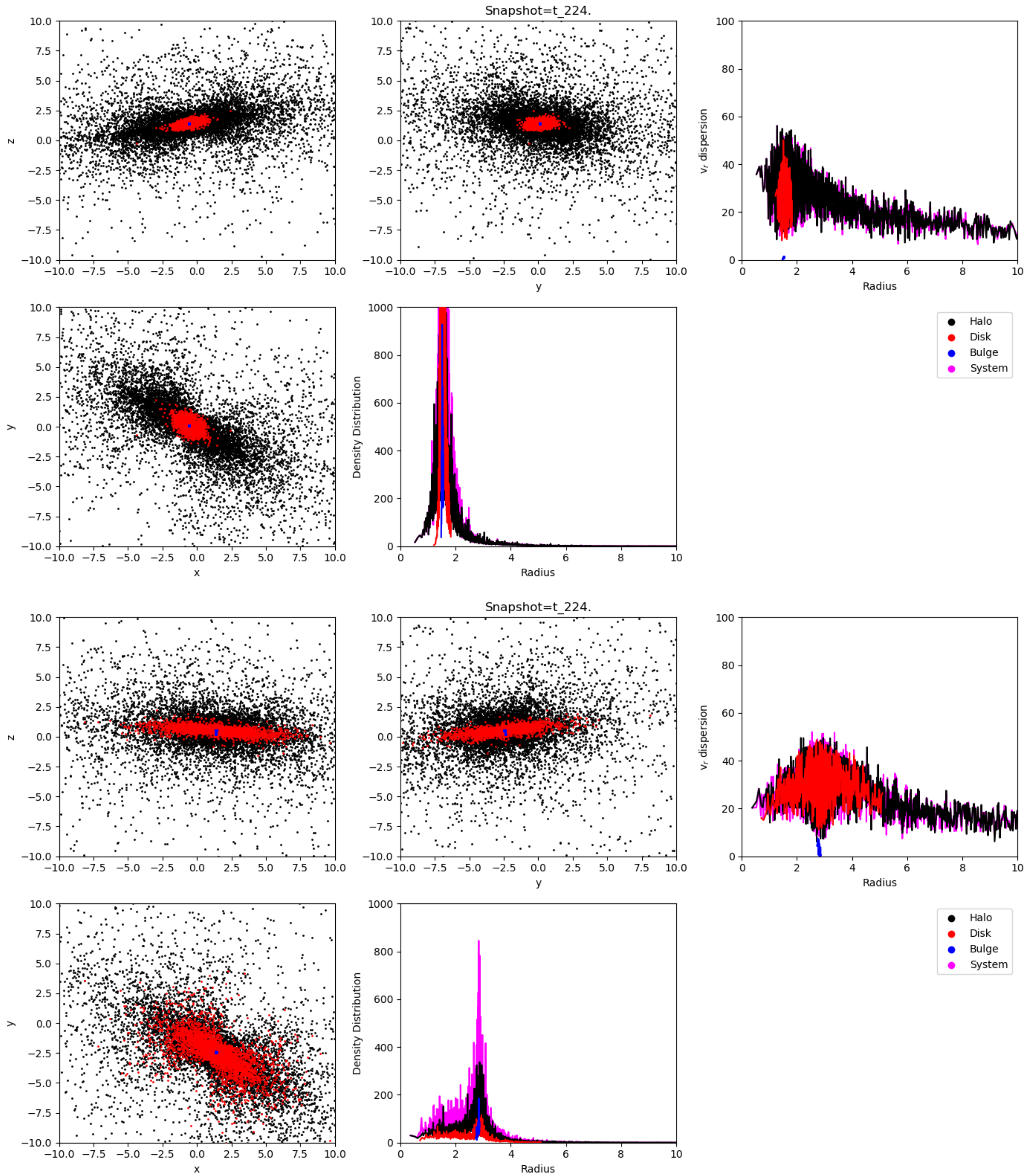

Figure 10. To compare the different Hernquist models, the Hernquist model with the interpolated radius is presented on top, while the Hernquist model with the approximated radius is presented on the bottom. When the disk is relatively exponential, it keeps its structure more and appears to form a peanut like structure. The density distribution does not appear as high as the first model because it doesn't collapse as easily.

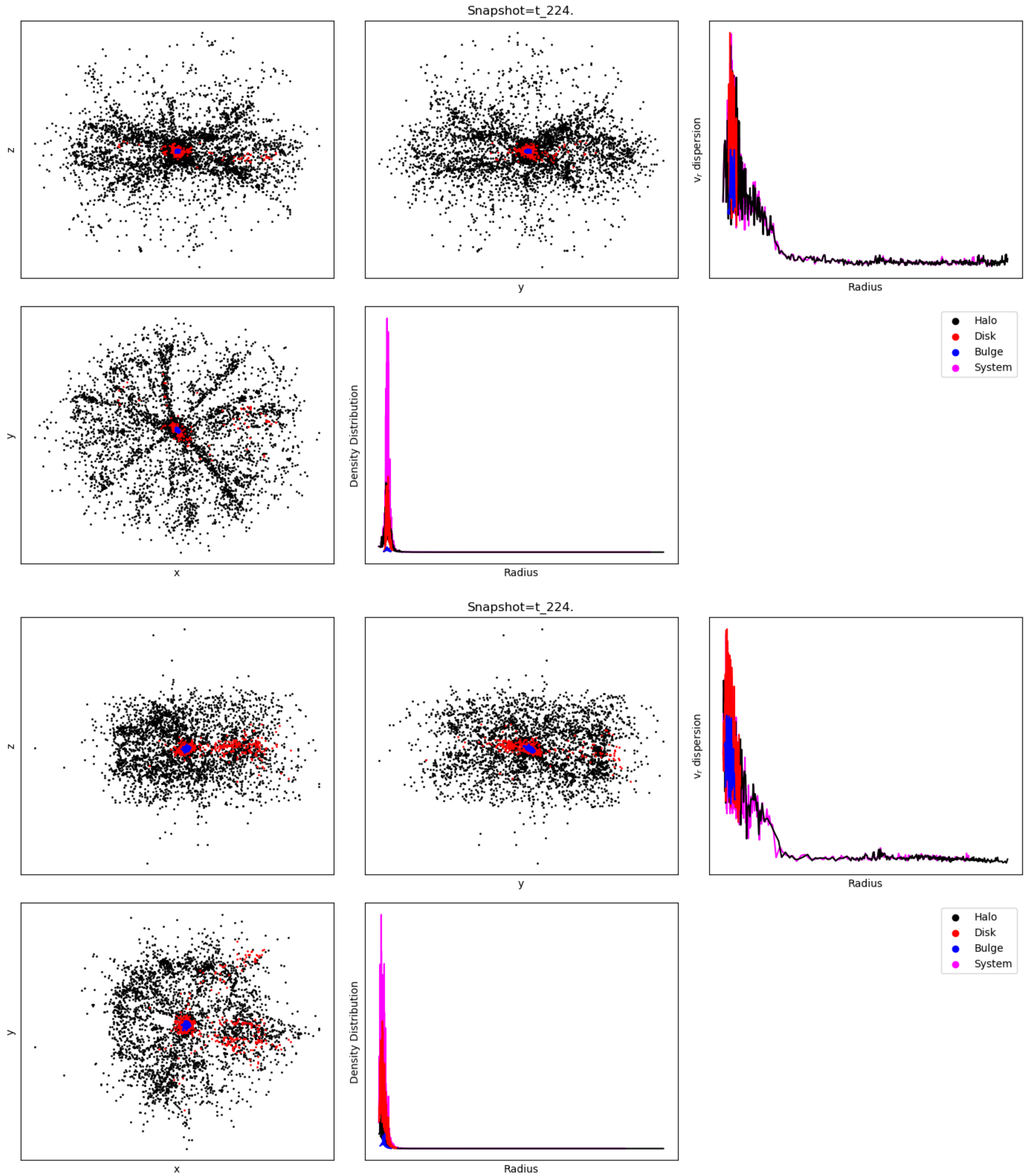

Figure 11. To compare the different homogeneous models, the homogeneous I model is presented on top, while the homogeneous II model is presented on the bottom. Each is of the same snapshot, after the particles have collapsed and expanded. The halo appears to form tendril structures more when the number of disk particles is much less than number of halo particles. The .gif is a bit more illustrative, but the simulations change little up until the last snapshot at t=10.01. The density distribution and velocity dispersion appears to be relatively the same.

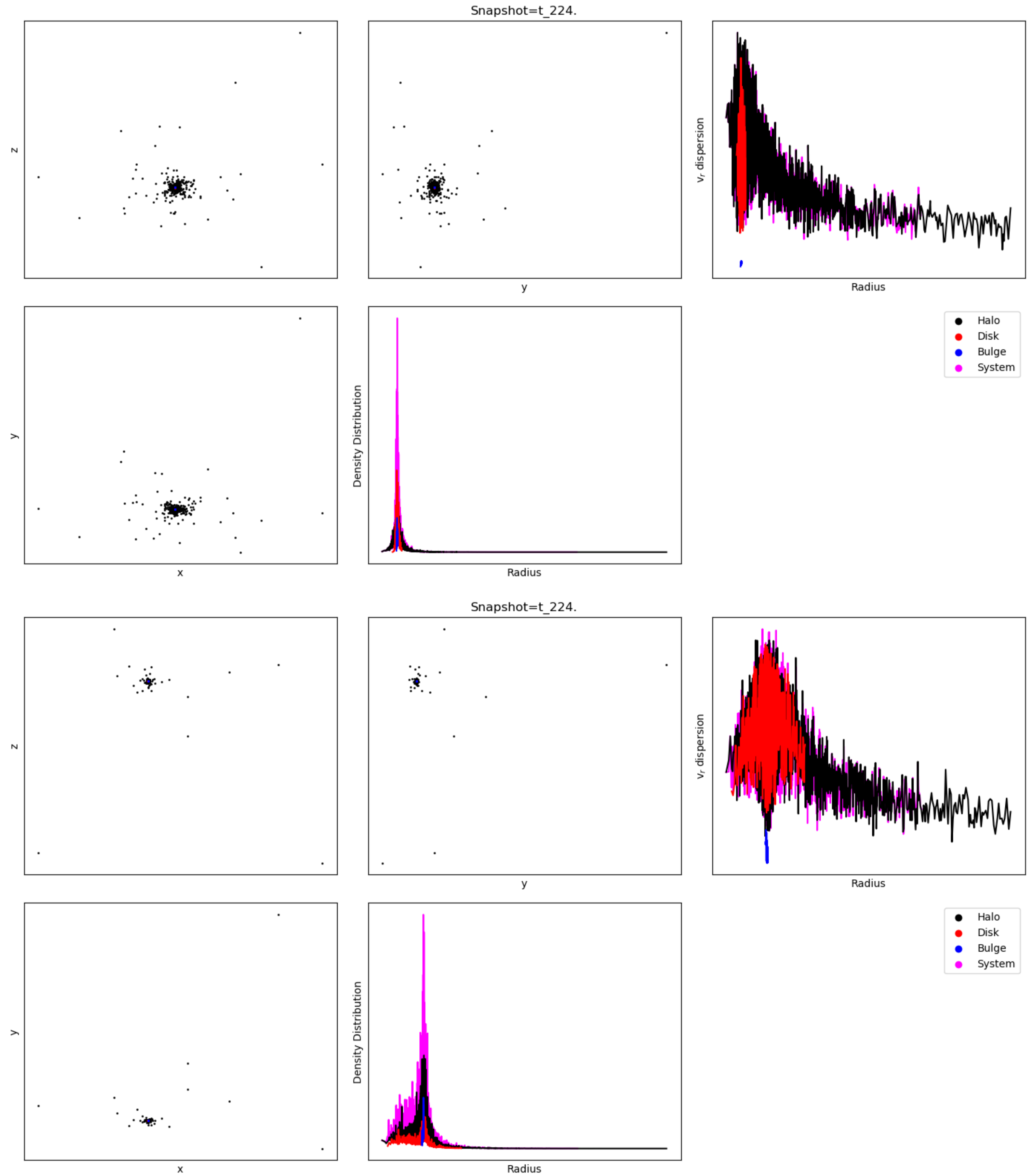

Figure 12. To compare the different Hernquist models, the Hernquist model with the interpolated radius is presented on top, while the Hernquist model with the approximated radius is presented on the bottom. Each is at the same time as in Figure 11. In both instances, it is difficult to understand how the profile has changed.

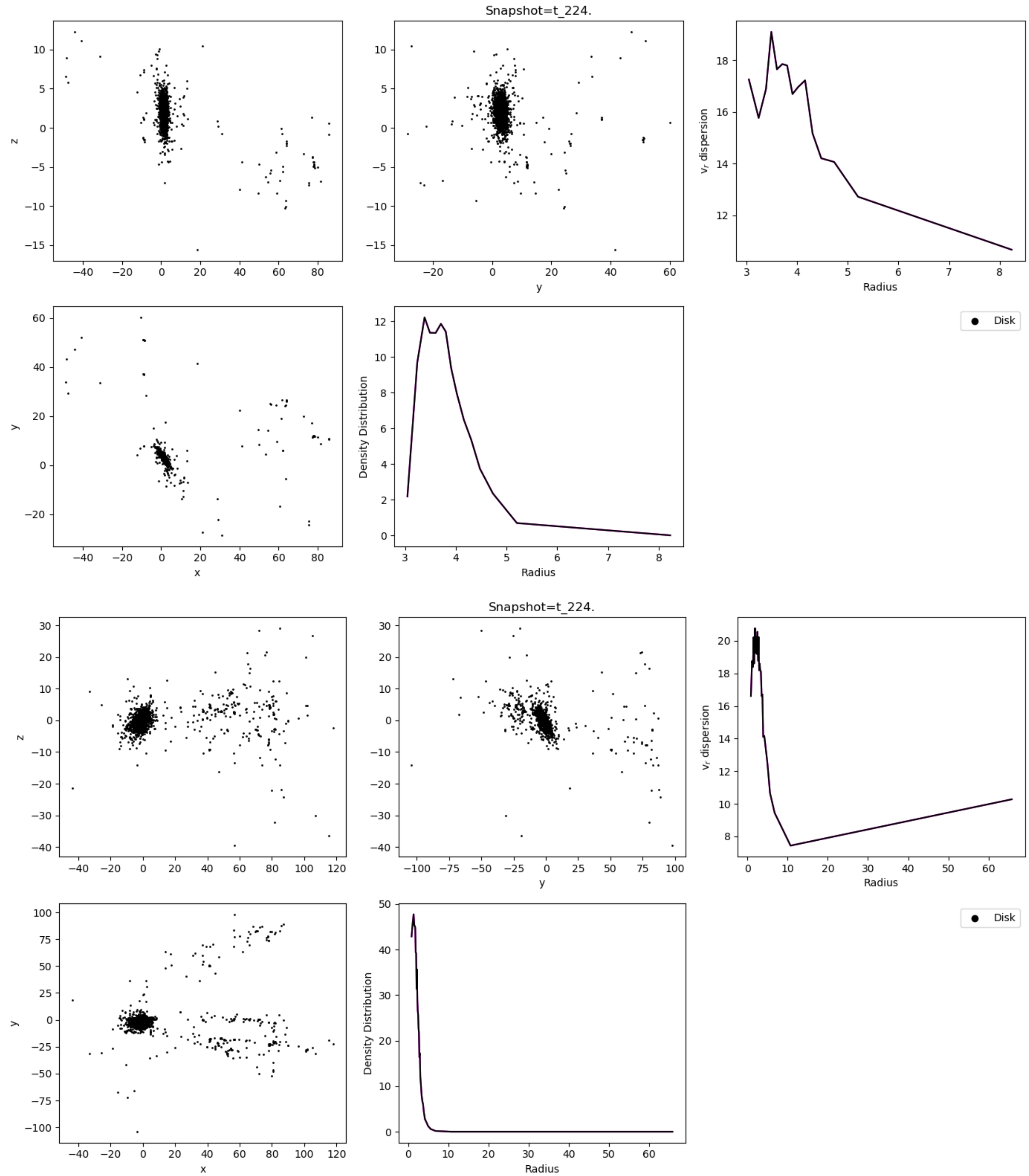

Figure 13. To compare the different homogeneous models, the homogeneous I model is presented on top, while the homogeneous II model is presented on the bottom. Each is of the same snapshot of the simulation with a halo+disk+bulge, but with different initial conditions as described in Section 3, after the particles have collapsed and expanded. The disk appears to form tendril structures more when the number of disk particles is about the same as the number of halo particles. The disk expands a lot more when the halo mass can't contain it.

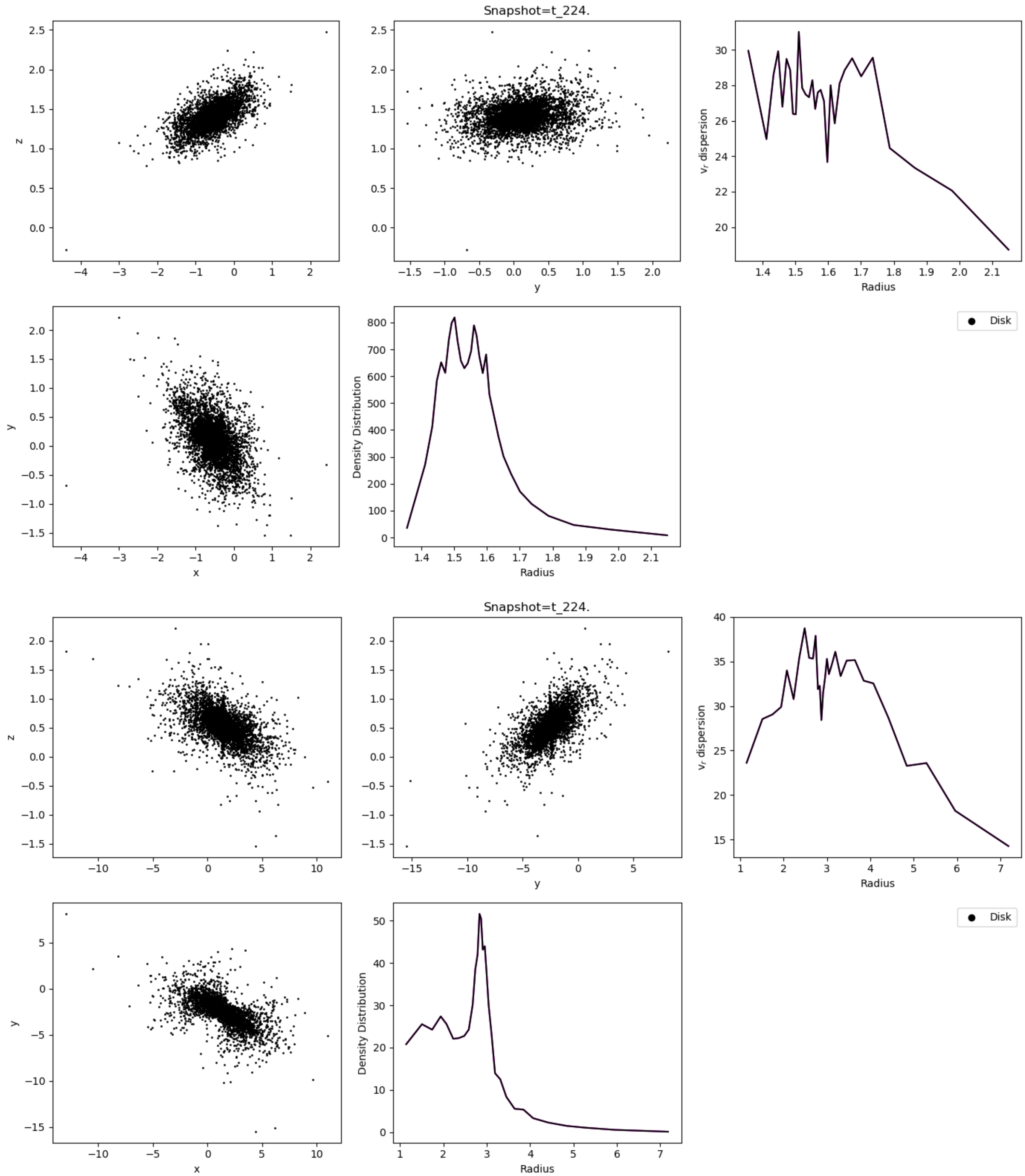

Figure 14. To compare the different Hernquist models, the Hernquist model with the interpolated radius is presented on top, while the Hernquist model with the approximated radius is presented on the bottom. Each is of the same snapshot of the simulation with a halo+disk+bulge, but with different initial conditions as described in Section 4. With the second set of initial conditions, a peanut shape forms. In neither case does the disk expand much, likely because the halo mass is three times more massive and can contain the disk. This changes the density distribution and velocity distribution.

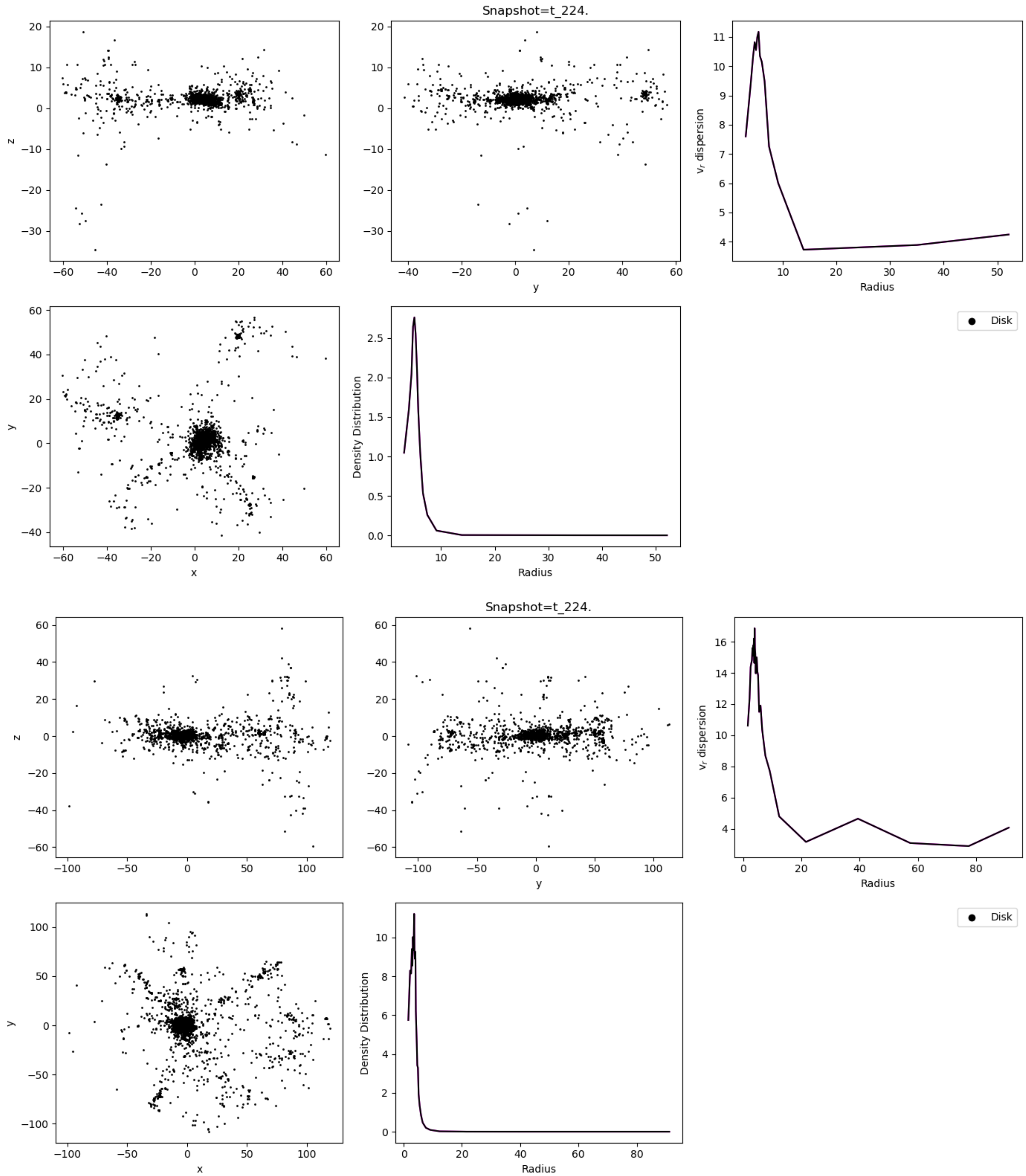

Figure 15. To compare the different homogeneous models, the homogeneous I model is presented on top, while the homogeneous II model is presented on the bottom. Each is of the same snapshot of the simulation with a a disk in isolation, but with different initial conditions as described in Section 3, after the particles have collapsed and expanded. It doesn't expand much more than the simulation in Figure 13, but the structure changes a bit. In this model, the particles are less dense with a lower velocity dispersion.

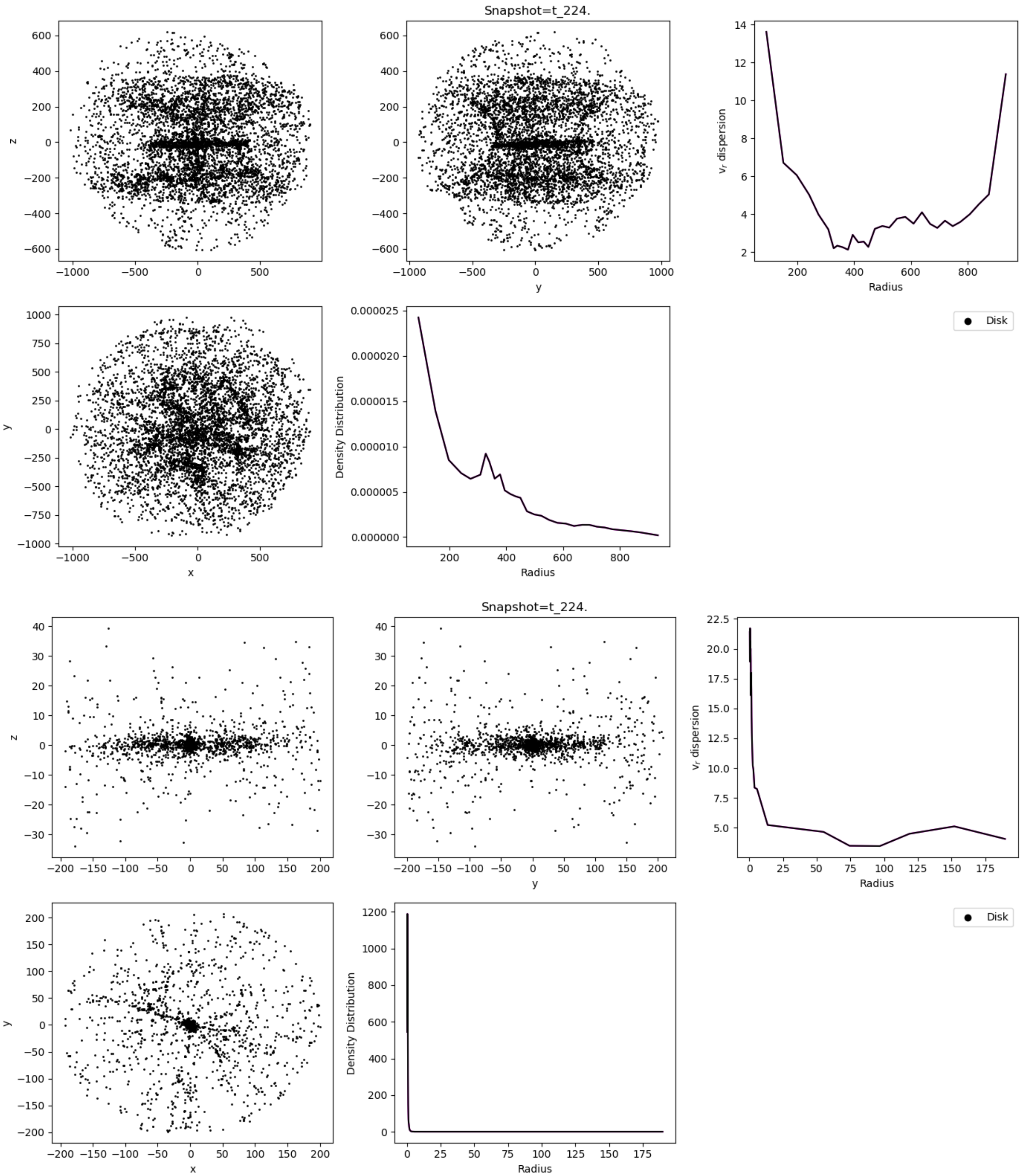

Figure 16. To compare the different Hernquist models, the Hernquist model with the interpolated radius is presented on top, while the Hernquist model with the approximated radius is presented on the bottom. Each is of the same snapshot of the simulation with a disk in isolation, but with different initial conditions as described in Section 4. While both expand much more than the simulation in Figure 14, the approximated exponential disk behaves much better.

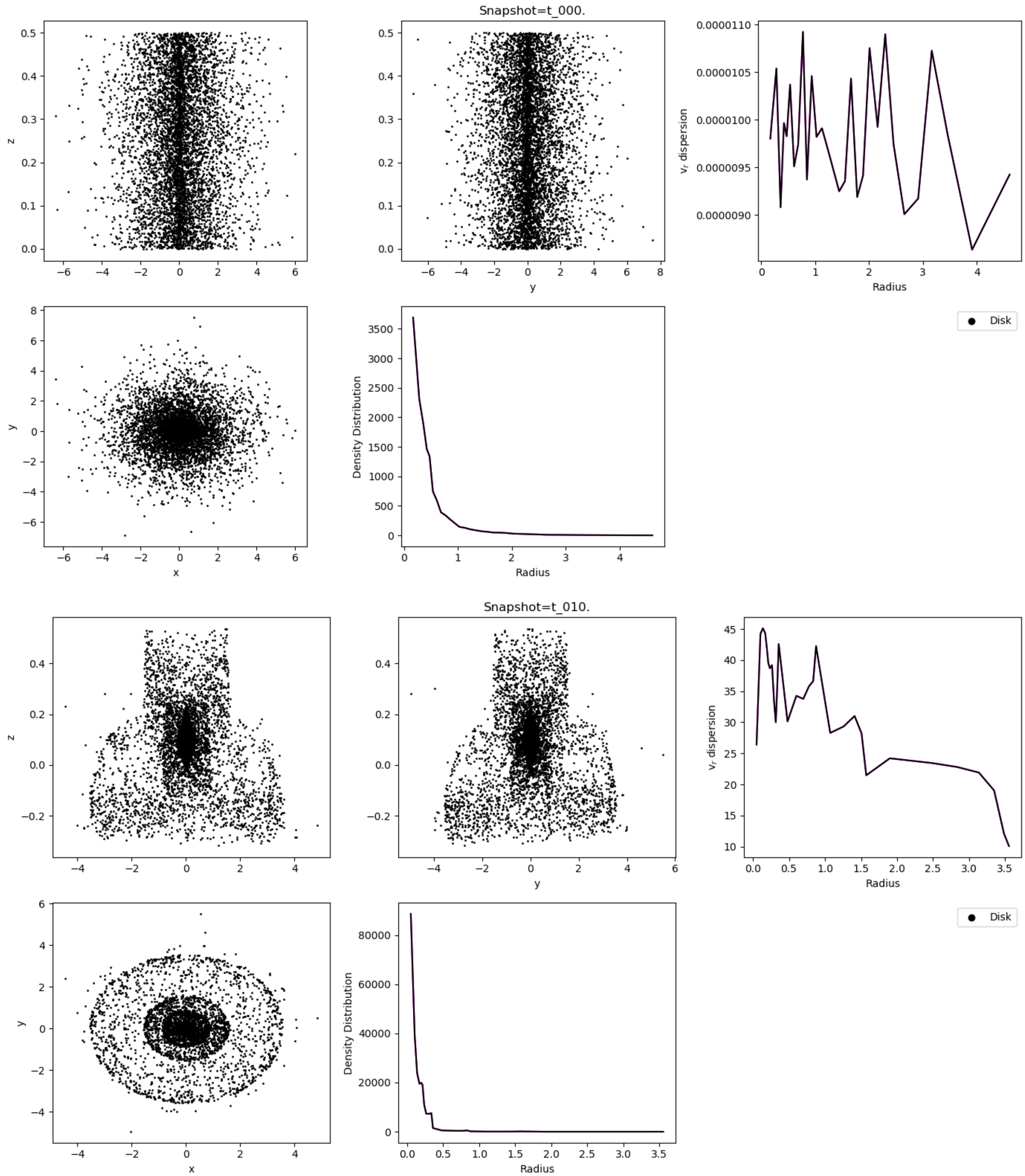

Figure 17. In the Hernquist model with an approximated exponential disk has an interesting feature where the particles form waves as they collapse and bounce off each other. The top image are the initial conditions, which appear to be relatively smooth in density distribution with a very small velocity distribution.

more. I had assumed the Gaussian would provide enough random directions to allow the galaxy to rotate. However, my .gif files don't show any rotation at all.

So for my 17*th* attempt, I would change how the velocities were chosen, as well as use the corrected `bulge_r_v2()` function.

7. ATTEMPTS

The results presented in Section 3.1 are from the thirteenth attempt at running Gadget2. Section 3.2 is from the (fourteenth) attempt. The previous attempts were mostly building the machinery necessary to run a simulation. I have yet to use a version control system, but I now see why it is important. Particularly, I uploaded all the files to GitHub, which is a version control system that I didn't utilize. It also took a long time because I decided to upload everything at once instead of committing small changes. Instead of keeping track in the way I'm doing now, the GitHub repository would do it for me. The terminal commands allow me to make commit change comments.

The first eight attempts were utilizing some derivations from Section **??**. It was on the ninth attempt I switched to homogeneous spheres and a homogeneous disk.

I switched for two reasons. Firstly, the simulation would either crash or blow up almost immediately. Nothing interesting happened; the simulation simply expanded. Secondly, I could not tell if it was my parameters or my initial conditions that were causing these issues. Using a simple homogeneous model would allow me to distinguish between the two.

I made one change at a time, hence the multiple attempts. While I kept track of the errors and what managed to resolve them, I did not keep track of which attempt had which error.

1. Periodic Boundaries error. The Makefile to compile Gadget2 has to have the `OPT += -DPERIODIC` (under the 'Basic operation mode of code' header) commented out if the parameter files have periodic boundaries switched off.

2. Before given the `binary.c` code, I originally attempted to make my own input files in .hdf5 format. It didn't work out, but I did learn a lot about the file format and structure. I also tried to make my own parameter files, but that also didn't work out. I ended up editing a parameter given to me.

3. Segmentation Faults: implies the simulation is too large. I therefore scaled down the number of particles found in the simulation. Upping the PartAllocFactor and TreeAllocFactor in the parameters also helped.

4. Timestep of size zero: implies the TreeAllocFactor needs to be increased. I also thought it was possible my particles were too far away since I was having difficulty getting reasonable radii with the Hernquist model.

5. Simulation only expands: implied to me the velocities were too high or the units were wrong. I therefore changed all the units in the parameter file to 1. I also initially changed time between snapshots to 0.01 with a max time of 0.1.

6. `binary.c` code only allows for halo and disk particles. I wanted halo,disk, and bulge. I therefore edited the `binary.c` code to allow for 3 groups and named it `binary3.c`.

7. Once including 3 components, I received errors because the softening values have to be nonzero for any component added. I therefore made it 1.0. I'm unsure of what value I actually should add. Hernquist93 has an actual method to derive these, but I couldn't solve them. Therefore, it was left as 1.

8. I was originally unsure what the mass was for the `binary.c` code; was it the total mass of the component or was it the mass of each individual particle? I originally believed it was the mass of the total component, but when inputting a total mass instead of an individual mass, I received segmentation faults. After seeing successful simulations, I'm positive it is the mass of each individual particle. Since the number of particles is meant to reflect the mass of the component, every particle is given a mass of 1.

9. The tenth attempt ran smoothly, but the disk was off center and the bulge was too small in radius. The eleventh attempt I made slight changes to the function for initial conditions, the radius of the bulge, and increased the number of particles.

10. The tenth attempt is when I first started to read in the snapshots. I started with just plotting the positions. I also created the .gif file of the images. The original .gifs are grainy. This was due to the scaling of the image. When the figsize parameter was reduced, the .gif became much cleaner.

11. Over the eleventh and twelfth attempt, I slowly made adjustments to the plots and .gif to get them to look nice. I also added in the density profiles (calculated, not theoretical). In the thirteenth attempt, I added in the radial velocity dispersion. I also added in calculating the total system's velocity dispersion and density profile.

12. The thirteenth attempt included looking at the different combinations of components of a disk-like galaxy. I edited the `read_snapshots.py` and the `create_gif.py` to take command line arguments. I edited the parameter files to extend the time to a max time of 10, but still have snapshots every 0.01 unit of time. It stopped doing anything interesting around snapshot 150 ($t \approx 1.5$). It simply started to move out of view. I had to run the snapshots several times to make the plots pretty enough for presenting.

13. The fourteenth attempt I kept the time stamp the same (just in case, although I did doubt anything exciting would happen after 1.5 time units). I changed the ratio of particles for each component.

14. The fifteenth attempt and sixteenth attempt are described in Section 4. I also started working on the newer versions of the plots.

REFERENCES

Athanassoula, E. 2013, Bars and secular evolution in disk galaxies: Theoretical input, ed. J. Falcón-Barroso & J. H. Knapen, 305

Binney, J., & Tremaine, S. 2008, Galactic Dynamics: Second Edition

Dubinski, J., Berentzen, I., & Shlosman, I. 2009, ApJ, 697, 293, doi: 10.1088/0004-637X/697/1/293

Hernquist, L. 1993, ApJS, 86, 389, doi: 10.1086/191784

Ostriker, J. P., & Peebles, P. J. E. 1973, ApJ, 186, 467, doi: 10.1086/152513

Springel, V. 2005, MNRAS, 364, 1105, doi: 10.1111/j.1365-2966.2005.09655.x

Springel, V., Yoshida, N., & White, S. D. M. 2001, NewA, 6, 79, doi: 10.1016/S1384-1076(01)00042-2