Modeling Galaxy-Galaxy Collisions

Elizabeth Wesson and John Boccio
Swarthmore College

Introduction

Many instances of colliding galaxies have been observed, but the long timescales and often confusing perspectives involved make it difficult to study them directly. Therefore, it is useful to create computer models of galaxy-galaxy collisions to test theories of how these collisions actually occur.

Goals and Overview

This project aimed to create realistic galaxy models and analyze their behavior during and after collisions. We created galaxies based both on data from the Milky way and on mathematical models. We compared simulated collisions using dark matter with those using modified Newtonian dynamics (MOND). In some cases the galaxies fell apart, but in others we observed realistic behavior like the formation of spiral arms.

Methods

To accomplish the goals of this project it was necessary to learn to use the Xcode programming environment.

We created the initial data files in MATLAB. Small and medium-sized collisions were run in Particula, a program developed by Hiroshi Abe. Larger collisions used FORTRAN to generate the data sets and MATLAB to convert them into movies.

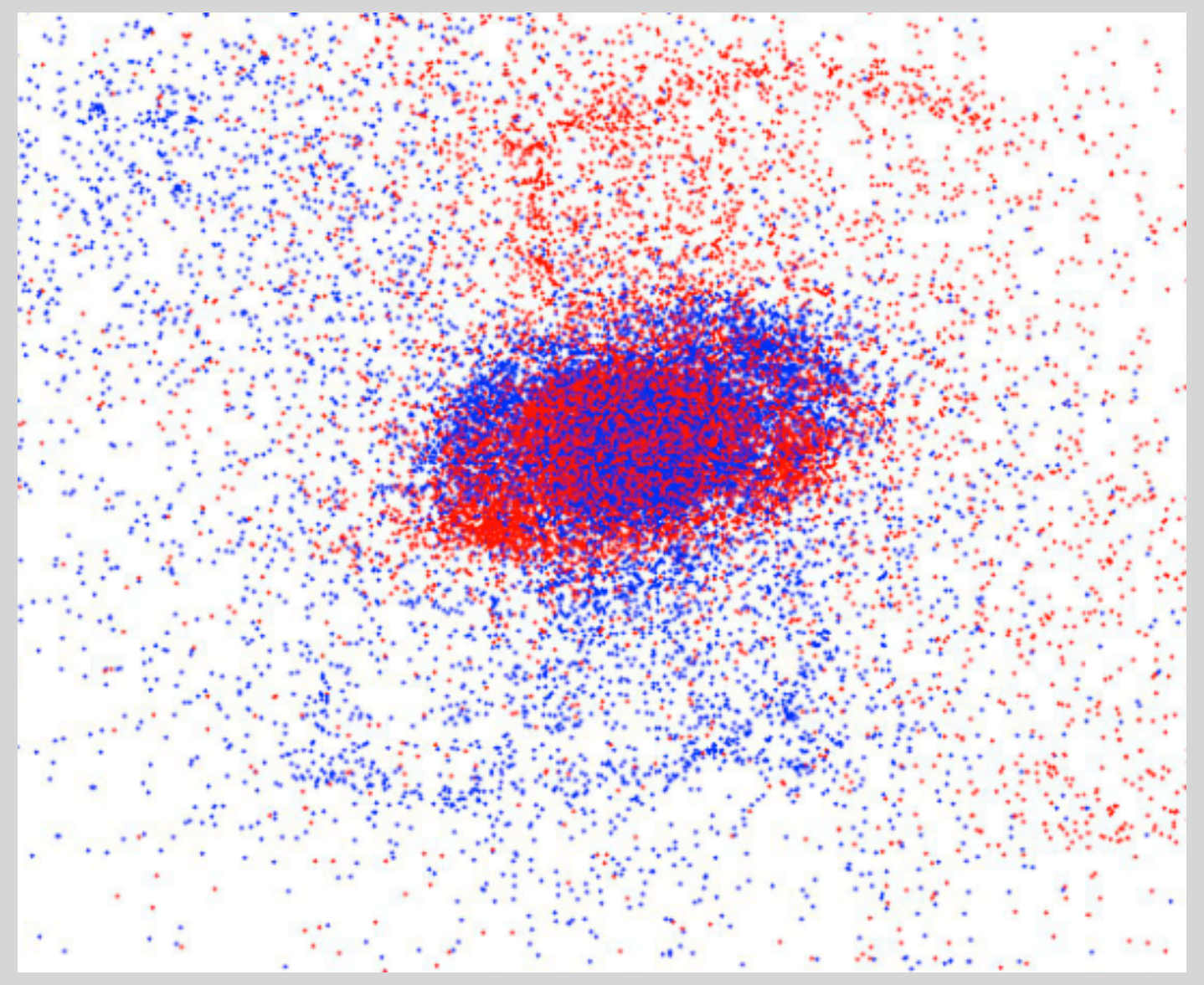

Figure 1: A newly formed galaxy develops a bar and spiral arms.

Learning Xcode

Particula was developed in Xcode, so we needed to learn this programming environment in order to modify Particula for our purposes. We used several books and online tutorials, but the most effective Xcode learning experience was making a pseudo-scientific (in that it has trigonometric functions and square root, but does not do order of operations) calculator.

The calculator was written in Xcode using Objective-C, and the GUI was created using Interface Builder.

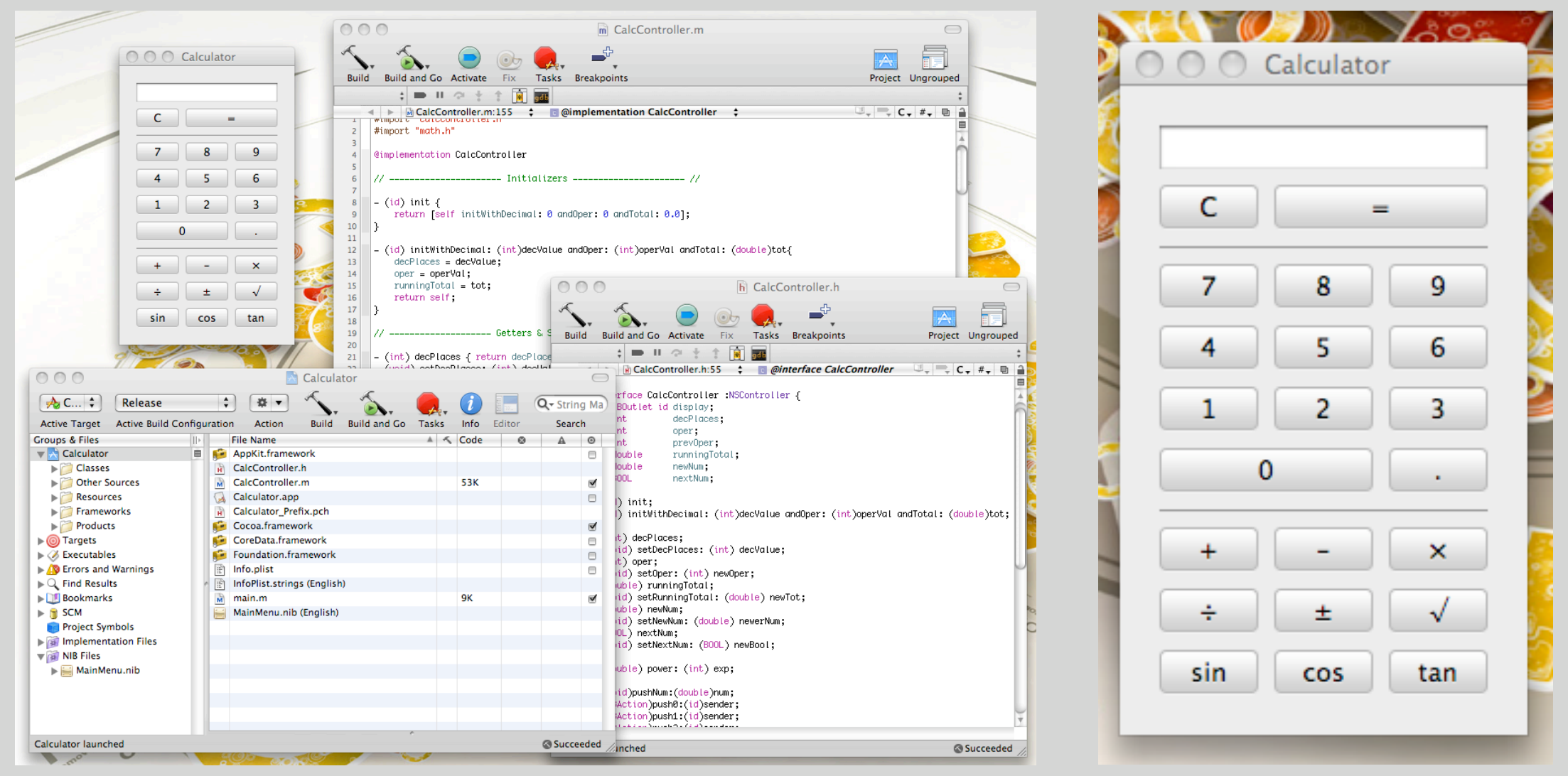

Figure 2: The calculator was developed using Xcode and Interface Builder.

The user can type in numbers or use the calculator's number buttons; however, the operation buttons must be used, even + and -. Also, the user must press C between calculations, rather than deleting the displayed number with the keyboard. These quirks exist because the calculator never converts the value in the display to a string, but always treats it as a number.

Dark matter and MOND

While not endorsed by a majority of physicists or astronomers, modified Newtonian dynamics (MOND) is a plausible and mathematically interesting theory. It can be approached as a theory of modified gravity or modified inertia; either way the effect is that objects whose Newtonian acceleration is very small are in fact accelerating much faster. In general,

$$a = a_N \left(\frac{1}{2} + \frac{1}{2}\sqrt{1 + 4\frac{a_0^2}{a_N^2}} \right)^{1/2}$$

where

$$a_0 \approx 1.2 \times 10^{-10}\, m \cdot s^{-2}$$

Note that

$$a_N \gg a_0 \Rightarrow a = a_N$$

$$a_N \ll a_0 \Rightarrow a = \sqrt{a_0 a_N}$$

This means that at short distances where the gravitational acceleration is not too small, MOND behaves like the Newton universe. At large distances, the dramatic increase in acceleration over Newtonian physics helps to keep the edges of galaxies together.

The comparison

We tested the robustness of model galaxies with dark matter halos against that of galaxies in the MOND universe.

For the MOND runs, we used a version of Particula with the force code edited accordingly. We were only able to edit the direct calculation method code, however, because the faster methods rely on converting all distributions past a certain distance to mass density functions.

Plummer galaxies hold together better under MOND than in the Newton universe; however, the center of the simulated Milky Way developed a hole under both regimes. In addition, we created a Kuzmin disk galaxy with a Plummer bulge and isothermal dark halo that held its shape well, but did not test it under MOND.

We could not conclusively declare one method better than the other, but we hope to create more sophisticated models in the future.

Dark Matter Models

Finding a good model for the dark halo of a galaxy involved a great deal of trial and error. A Plummer sphere worked reasonably well, as did an isothermal sphere. However, each simulation worked best with a dark halo of a different size. It was necessary to tinker with both the radius and the mass of each dark halo.

Generally, the ideal mass of a dark halo was between 15 and 35 times the mass of the galaxy, with the most common value about 20. Good values for the radius of the dark matter sphere range between 12 and 20 times that of the galactic disk.

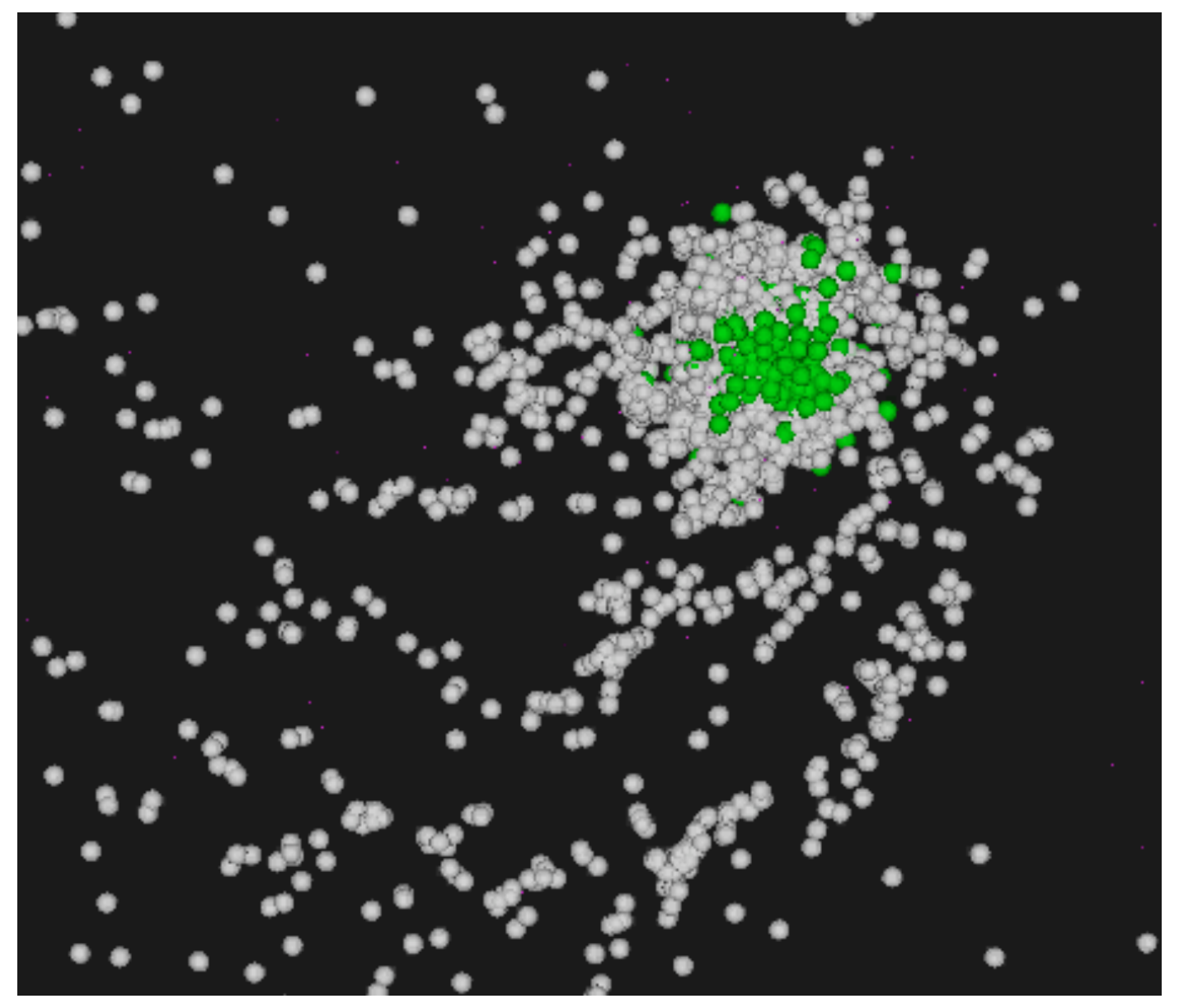

Figure 3 : Evolution of a Kuzmin galaxy. It has spontaneously developed asymmetric arms.

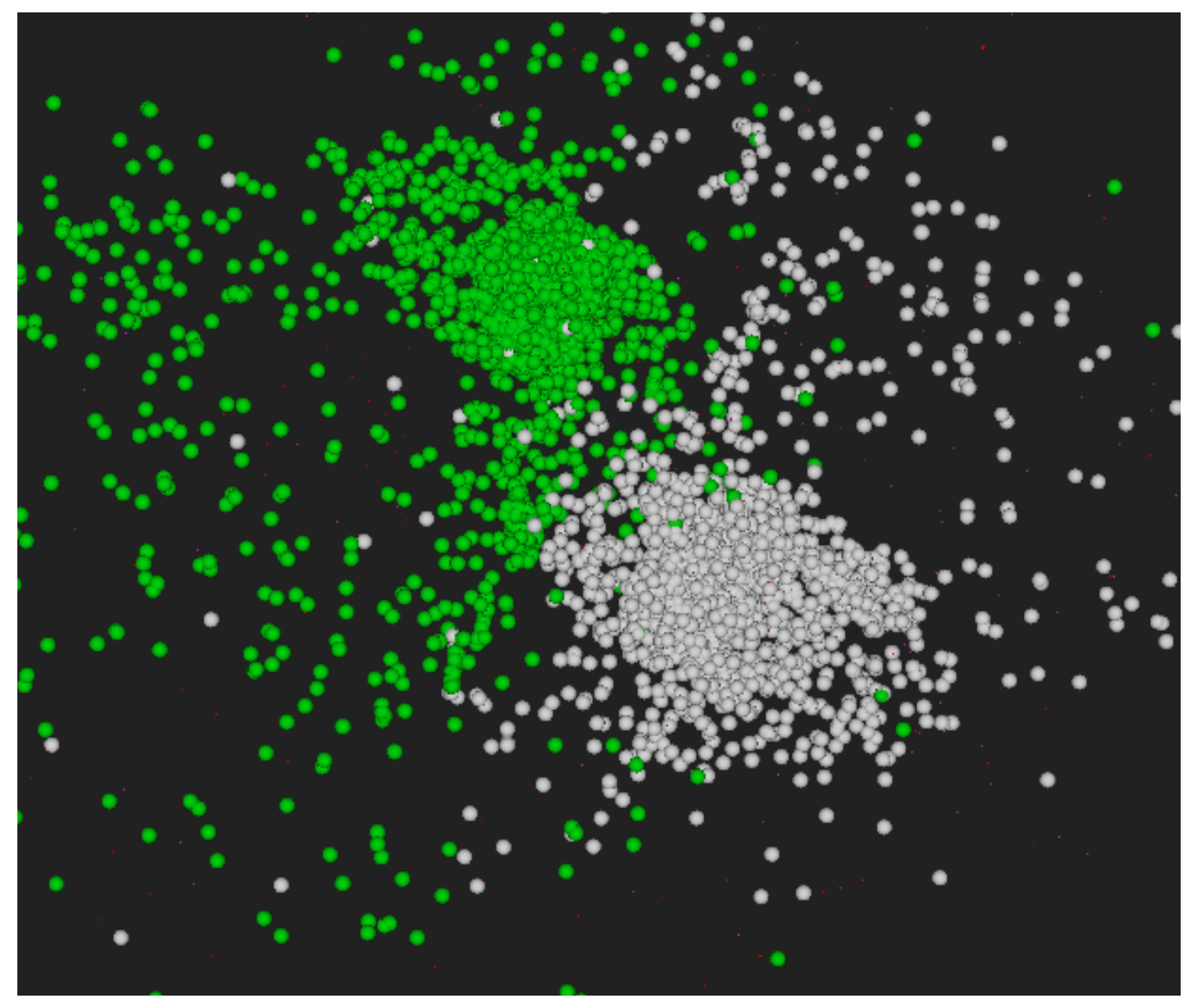

Figure 4: After colliding, these Kuzmin galaxies develop long, trailing tails.

Galaxy models

This project involved galaxies based both on real data from the Milky Way and on mathematical models. We used the Plummer model for spherical galaxies and dark matter halos, the Kuzmin model for disk galaxies, and isothermal spheres for dark matter halos.

One of our major goals was to determine whether MOND or a dark matter halo would do better at holding the model Milky Way together.

We found that with MOND the outer edges of the galaxies stayed together well, but the area near the center did not.

After trying various parameters, we found that a Plummer sphere much larger than the Milky Way made a reasonable dark halo. However, the center of each galaxy still developed a hole.

We speculate that the Milky Way data sets do not include a proportional number of stars near the center of the galaxy.

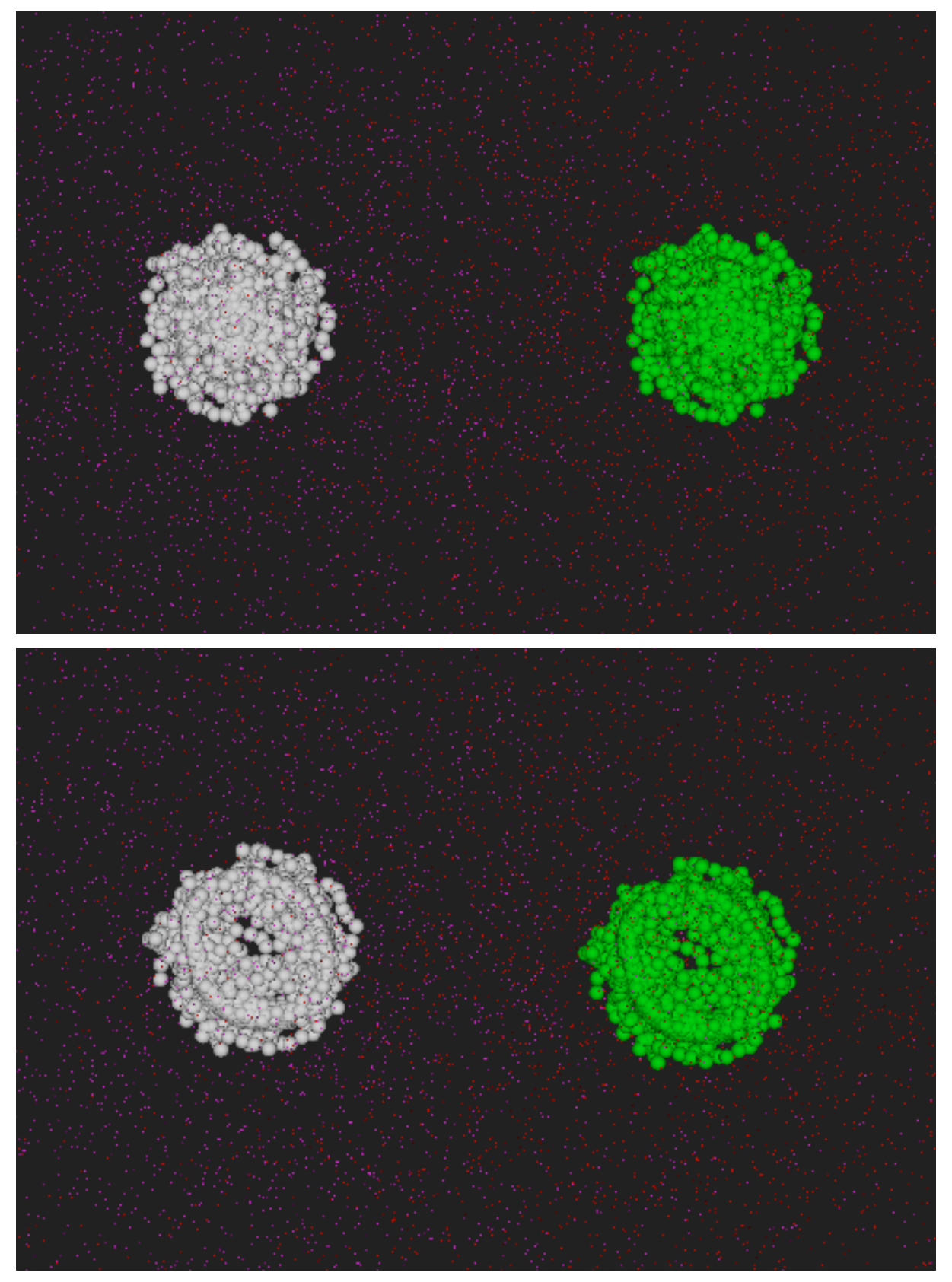

Figure 5: Galaxies generated from Milky Way data fail to hold together, though dark matter is present.

Plummer Sphere

The Plummer model is a spherically symmetric distribution of masses that supports itself by the random motion of each mass. Its mass density as a function of radius is

$$\rho(r) = \frac{3M}{4\pi} \cdot \frac{a^2}{\left(a^2 + r^2\right)^{5/2}}$$

where *a* is a constant. If we choose *a* = 1, we must choose the position of each star by

$$r = \frac{1}{\left(\frac{1}{X^{2/3}} - 1 \right)^{1/2}}$$

where X is a uniformly chosen random number between 0 and 1.

Kuzmin Disk

This model, which we used to simulate disk galaxies, is a flat, circularly symmetric distribution which has the Plummer potential in the plane of the galaxy. Its surface density in the plane is given by

$$\sigma(r) = \frac{M}{2\pi a^2} \cdot \left(1 + \frac{r^2}{a^2} \right)^{-5/2}$$

so that the radius of a given point mass is

$$r = a \cdot \sqrt{\frac{1}{(1 - X)^2} - 1}$$

where X is again a random number.

Isothermal Sphere

The isothermal sphere is useful for modeling the dark matter halo of a galaxy with a near-flat rotation curve, because its own rotation curve is constant. Our most successful disk galaxies were Kuzmin disks with Plummer bulges and isothermal dark halos.

The mass density of an isothermal distribution is

$$\rho(r) = \rho_0 \cdot \left(\frac{r}{a} \right)^{-2}$$

and we find that the radius of a particular point mass is directly proportional to the uniformly chosen random number X.

Particula

Improvements

Particula is written in C++, and it uses OpenGL to display the stars that comprise each galaxy. The original version of Particula uses non-relativistic Newtonian dynamics, and it displays all the stars the same size. It was convenient to create two new versions of Particula: Particula_tiny and Particula_MOND.

The first of these is useful for simulations with dark matter. Using a switch, we edited the display code so that the masses in the third and fourth files loaded are displayed as much smaller than the stars in the first and second files. This allows the viewer to display dark matter halos without obscuring the galaxies.

In the second new version, Particula_MOND, we edited the force code so that the acceleration of each point is calculated using MOND.

Limitations

Particula is only suited to small and medium-sized collision simulations, because it will only display up to 131,072 points at a time. This is due to settings in the OpenGL libraries, and not easily fixable.

Therefore, for collisions involving hundreds of thousands of points, we used FORTRAN to generate the data for each step and MATLAB to convert the resulting files into images. An advantage of this approach is that once the data files exist, it takes only a few minutes to create the movie. This allowed us to see collisions from several different perspectives.

Particula does not save the data it uses to create each image; watching the collision from a different viewpoint requires running the simulation again. We hope to fix this shortcoming in the future.

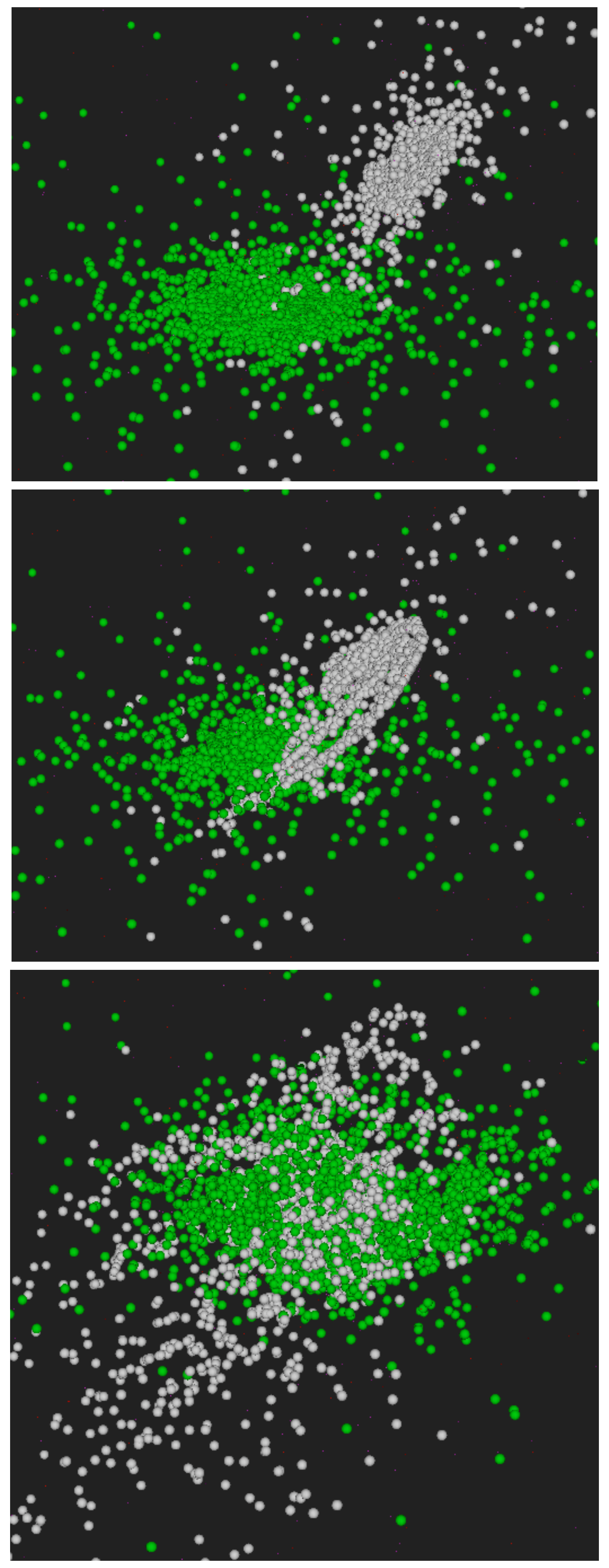

Figure 6: An oblique collision of Kuzmin disk galaxies.

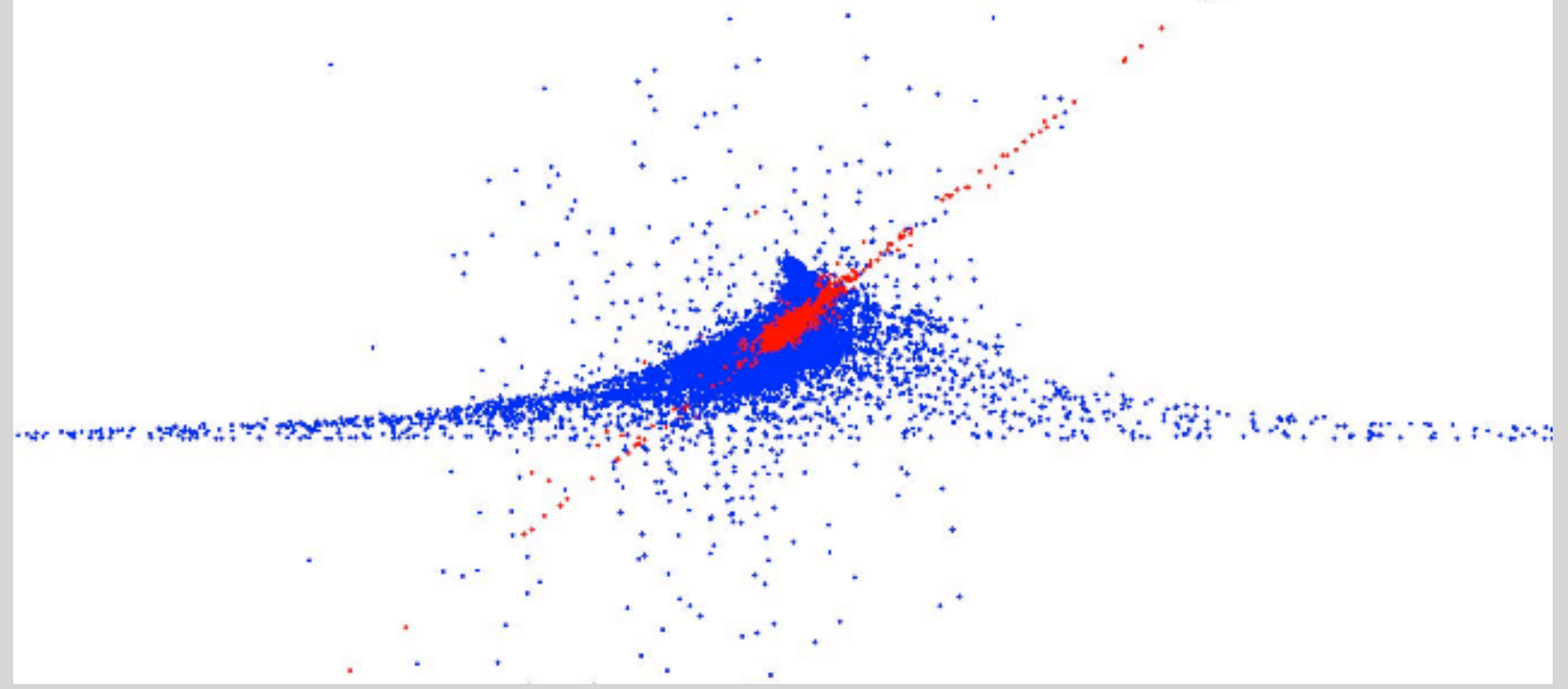

Figure 7: An oblique collision of two Kuzmin disk galaxies, seen edge-on.