



Figure 1: The Antennae Galaxies (NGC 4038-39)

0. Important!

This handout is to be read in conjunction with the notes Finite Difference Solution of the Gravitational N -Body Problem, available via the course homework page.

1. Introduction

Galaxies come in a broad range of sizes and shapes. Many are regularly structured and can be classified as spirals or ellipticals. In the former, nearly all of the constituent stars orbit in the same direction about some center; in the latter, the stars are distributed in complex three-dimensional orbits.

Other galaxies, however, have very irregular features, and are often known as “peculiar” galaxies. The Antennae Galaxies shown in the figure above provide a good example. Over the past few decades it has become clear that the structure—or *morphology*—of many peculiar galaxies is due in part to interactions, or collisions, with other galaxies.

In this project, you will implement a *very* simplified model of galaxy collisions, motivated by work due to Alar and Jüri Toomre in the early 1970’s. Although this model is grossly inadequate for the detailed simulation of galaxy interactions, it is nevertheless capable of reproducing some basic morphological features that are observed in actual galactic collisions.

2. Overview of the Toomre Model

In the Toomre model, an isolated galaxy is modeled as a central particle, which we will call a *core*, that has some gravitating mass, m_c . Orbiting this core, in *circular* orbits of varying radii, is some number, n_s , of stars. Importantly, the stars have *vanishing* gravitating mass, by which we mean that they experience an acceleration due to the core's gravitational influence, but do not contribute to the gravitational field themselves. Thus, the core acts on each star gravitationally, but the stars do not exert gravitational forces on either the core or the other stars.

To model a collision of two galaxies we simply consider two cores, with masses m_{c1} and m_{c2} , surrounded by n_{s1} and n_{s2} particles, respectively. We start the galaxies with an initial separation that ensures that the interaction of core 1 with the stars from galaxy 2 and vice versa is negligible. Provided that the initial conditions are set properly, the galaxies will approach one another and then interact in some fashion. Here, *all* of the stars feel the gravitational influence of both cores, and each core feels the gravitational influence of the other core. Thus, any given particle will experience a gravitational force (acceleration) from either one or two particles. Note that it is the fact that the stars do not enter the force/acceleration calculation which will make it feasible for you to simulate using, for example, 1000's of stars per galaxy. In particular, if there are $O(N)$ stars, then the cost of the simulation is $O(N)$ vs. $O(N^2)$ which would be the case if all of the particles were gravitating.

3. Specifications: please read and follow carefully

Your implementation of the Toomre model should have the following features:

1. You can assume that there are exactly 2 cores, but you can also generalize to more cores if you like.
2. The number of stars per core should be easily varied in your code (i.e. don't hardcode that number throughout the code).
3. Work with the units defined in the N -body handout; i.e. with $G = 1$. If you then make your core masses $O(1)$ numbers, good characteristic spatial scales for your galaxies will also be $O(1)$, as will characteristic time scales for simulations.
4. Use the finite difference equations of motion described in the N -body handout, modified as necessary to take into account that there are two types of particles in the simulation (core and star). *Important! Do not use an ODE integrator to solve the equations of motion.*
5. You should implement the equations of motion in all three spatial dimensions (while noting that you can use MATLAB's vector/matrix facility to treat those dimensions as a single entity \mathbf{r} as described in the N -body handout); however, it will likely be the case that the dynamics you study is restricted to some 2-d plane (most likely the xy plane).
6. It should be easy to change initial conditions for the galaxies in your code. Do *not* assume that the initial velocities of your galaxies vanish.
7. For each galaxy, restrict the stellar radii about the core to some minimum and maximum values, and, for any given radii, distribute the star's angular position randomly. This will yield a more natural appearance for your galaxies at early times.

4. Computations and Analysis

1. Before you begin studying galactic collisions, you must convergence test your basic finite difference solution using two particles (cores) with distinct masses (say 1.0 and 0.5) in mutual circular orbit about each other. Choose one coordinate, say x , from one of the particles and demonstrate using at least a three-level convergence test that your implementation appears to have $O(\Delta t^2)$ error in that quantity. The convergence test should parallel—in construction and execution—the one for the nonlinear pendulum covered in Tutorial 3.
2. You can also *optionally* convergence test your two-particle code by testing for convergence to conservation of the total energy, using the formulae described in the N -body handout.

3. Once you have convergence tested your code for two particles, move on to galaxy dynamics. Start with a single galaxy at rest and verify that, when evolved, the stars remain on circular orbits about the core. Your test here can be qualitative: ensure, for example, that the stars don't fly away from the core. Choose initial conditions so that all motion is in a single plane (nominally the xy plane).

Now simulate a galaxy which is moving with some velocity. Note that you can translate a galaxy, or give it some overall velocity, simply by adding the translation or velocity vector to the position or velocity, respectively, of the core and each of the stars. Again, use a velocity vector such that the motion of all particles is restricted to the initial plane of the galaxy. Ensure that the stars remain in their circular orbits as the galaxy propagates.

4. It's now time to consider galaxy collisions. Once again, you should begin with interactions where all particles are initially in some plane and remain in that plane (again, probably the xy plane), and it is fine if *all* of your work has the dynamics restricted to a plane. (If you have time and the inclination, though, you can also try non-planar collisions.) Your goal here is to perform some simulations which generate "interesting" morphologies, such as the example shown in the movie on the course homework page. Note that the collisions should tend to be "glancing": the model will cease to be very meaningful if the cores get too close to one another. You will need to experiment with initial conditions to get things to your satisfaction. Also note that the direction of rotation of the stars about a core relative to the overall angular momentum of the collision will have an impact on the collision morphology. You should thus be able to change that direction in your code.
5. You can use basic MATLAB graphics to provide a rudimentary animation of the dynamics of your simulations. Consider introducing a frequency parameter ν_{freq} that limits output to every ν_{freq} time steps if the animation excessively slows the calculation. Also note that the MATLAB `drawnow` command will cause graphical elements in the "pipeline" to be displayed immediately.
6. In your write-up, include plots of the results from the convergence test(s) you perform, as well as a few representative snapshots of the particle configurations from one or more simulations. You should also make at least one AVI (MPEG) movie of a simulation that develops interesting structure and include that in your submission. Note that the script `bounce.m`, available from the course homework page, illustrates the creation of an AVI movie for the case of a bouncing ball simulation.
7. Your write-up should also contain a description of the theory behind the simulation, the finite difference equations used, and a discussion of any difficulties that you encountered in implementing the discrete equations of motion and then finding initial conditions which produced interesting results. It should also describe the overall structure of the code that you wrote to perform the simulations.
8. You can use separate scripts and/or functions for the convergence testing and galactic collisions phases of the project. However, the finite difference equations and their solution for the two cases should be identical.
9. On my office PC, a simulation with 10,000 stars total (5000 per galaxy) for 2500 time steps takes about 15 seconds. If you find that your simulations are taking an inordinate amount of time, chances are good that you are doing one or more things that need correcting.

If you get stuck with this or any other aspect of the project, don't be afraid to ask for help.