

Computer Simulation of Dark Matter Effects on
Galaxy Collisions

New Mexico
Supercomputing Challenge
Final Report
April 4, 2012

Team Number 72
Los Alamos Middle School

Team Members:

Cole Kendrick

Teachers:

Brian Kendrick

Project Mentor:

Brian Kendrick

Summary

The main goal of this project is to develop a computer program to model two galaxies colliding including dark matter mass. The questions this project will answer are: How will dark matter effect colliding galaxies, how accurately can this be modeled, how big of an impact does dark matter have on colliding galaxies. Last year, a computer program was developed to model the effects dark matter had on a galaxy's rotational curve. That program was written in C, and this year that program has been modified to fit this year's problem. This year, the code has been modified to do two galaxies and include galaxy-galaxy interactions. Currently, for this model dark matter is being treated by a large mass point at the center of the galaxy. This model is also using the same computational methods as last year, the velocity Verlet method (2nd order) is being used for Newton's laws of motion. Last year, a nearest neighbor method was implemented. This method is a much quicker way of solving a n-body calculation without doing full n-body computation. This method will be described in more detail later on. Andromeda and the Milky Way (MW) were the main focus of this project, 3 different cases were ran to determine how much dark matter really affected a collision. All simulations that were ran had 4000 stars (2000 per galaxy) and a time step (dt) of 200 years. The three cases that were ran are: (1) Normal mass – the realistic mass values we predict the galaxies to have, (2) MW has $\frac{1}{4}$ of Andromeda's dark matter mass, and (3) Normal mass with same rotational direction. My main results show that dark matter is needed to keep the galaxies stable throughout the collision. With case 2, the Milky Way galaxy was torn apart by Andromeda and eventually orbited it, with the normal mass cases 1 and 3, the galaxies eventually merge and become an elliptical galaxy over time. Without enough dark matter mass in each galaxy, the collision will not be stable. My results were compared to professional simulation results of the Andromeda and Milky Way collision as well as realistic mass distributions (what we observe).

Table of Contents

Introduction.....	4
Problem.....	5
Galaxy Model	6
Results	9
MPI	18
Conclusion.....	19
Future Work.....	19
References.....	20
Appendix 1.....	21
Appendix 2.....	22
Appendix 3.....	25

Introduction

What is dark matter and why do we need it?

No one knows what dark matter really is, although it has been one of the many theories that exist to help explain the flat rotational curve of all galaxies. Dark matter was first introduced in the 1930's but many people did not really support it until later in the

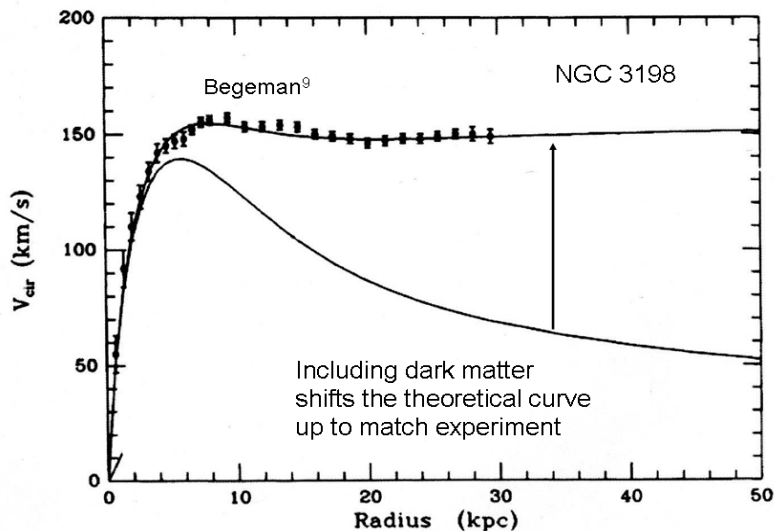


Figure 1: Rotational curves and experimental data for NGC 3198

1970's. In the 1970's people had begun collecting experimental data from galaxies by taking the brightness of the galaxy. They also made computer programs to test the dark matter theory. According to Newton's laws of motion, a galaxy rotation curve should dip down towards the edge of the galaxy. However, the experimental curve is relatively flat (see Fig. 1). Dark matter is one of the theories to resolve the difference between the two curves. Another theory that exists is called MOND (Modified Newtonian Dynamics) which changes Newton's laws of gravity to fit the flat rotational curve. By using the dark matter and MOND theory we can better understand how galaxies function and learn more about our tiny section of the universe.

How are galaxy rotational curves measured?

Galaxy rotational curves are measured from the Doppler shifts of the Hydrogen 21 cm line. The shift is broken down into red and blue, the red shift is going away and blue shift is coming towards you.

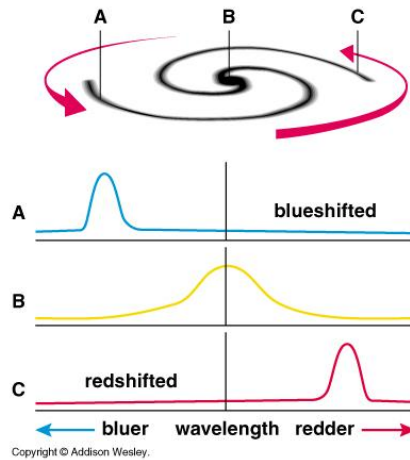


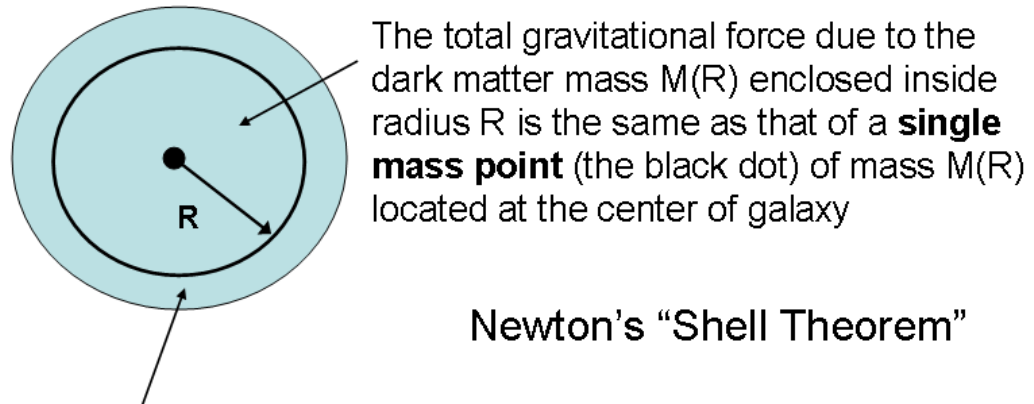
Figure 2: Schematic showing the blue and red Doppler shifts which are used to measure a galaxy's rotational velocity.

Problem

The two main questions my project will answer are: Can I successfully create a computer model to simulate the effects dark matter has on galaxy collisions? How accurate will my model be? Dark matter in my model is being treated by using the Navarro-Frenk-White (NFW) dark matter mass distribution. This distribution is a spherically symmetric static mass that encompasses the galaxy (Fig. 3). The same basic simulation methods that were used last year are also being implemented. Newton's law of motion $F=ma$ (total force = mass times acceleration) will be used to move the visible matter ("stars") in all of my simulations. The force is due to gravity acting between all of the visible matter plus a large central force at the center of the galaxy due to dark matter. The force between any two mass points m_1 and m_2 is given by Newton's law of gravity $F=G \frac{m_1 m_2}{r^2}$ where G is the universal gravitational constant and r is the distance between the two points.

Dark Matter Model

Dark matter mass distribution (the blue sphere) is static, spherically symmetric, and surrounds entire galaxy



The gravitational force due to the dark matter mass outside radius R adds to zero

Galaxy Model

The same galaxy model is being implemented into this years code, although this year there is two galaxies instead of one. My galaxy model is very similar to our solar system, instead of planets orbiting the Sun, stars orbit the dark matter and black hole of the galaxy. Also, instead of 8 planets, there are trillions of stars. My galaxy model takes those stars and represents many stars as one because doing a trillion stars on single computer will take forever. My galaxy model calculates the mass of the stars by taking the total mass of the core or disk (M_{core} or M_{disk}) and dividing that by the number of stars in the core ($N-N_0$) or disk (N_0). The total number of stars is N . The formula in my

model is $M = M_{disc}/N_0$ or $M_{core}/(N - N_0)$ to get the mass of a star. The figure 4 below shows how my galaxy model is similar to a solar system.

Galaxy Model

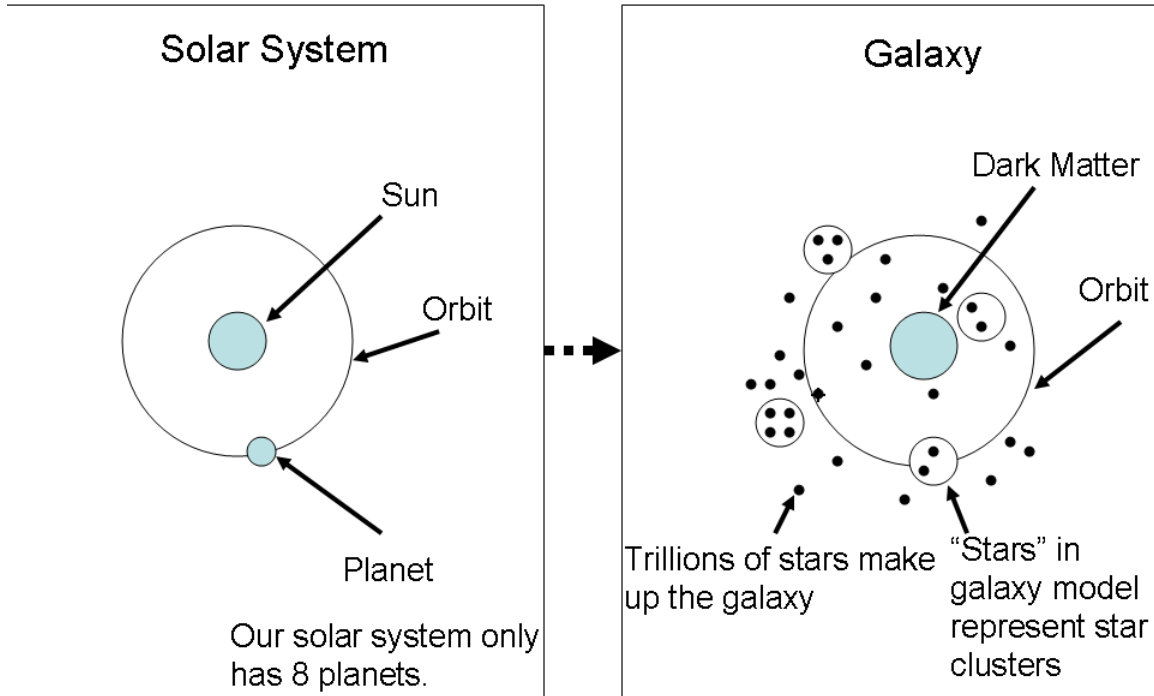


Figure 4: Schematic showing how my galaxy model resembles the solar system.

Nearest Neighbor

The nearest neighbor method is used in all of my calculations, and is a much quicker way of doing a n-body problem. Nearest neighbor creates a radius around each star and determines whether or not another star is in that radius. If the star is inside of the radius then the program stores that stars ID in an index and computes the gravitational force between it. The nearest neighbor radius size changes over the distance from the center of the galaxy. Therefore, the radius is small near the center of the galaxy because of the higher density of stars. On the edge of the galaxy the radius is larger because of the less dense areas. The optimal range is 1-11 kpc which was determined in my last year's project. (See fig. 5 below)

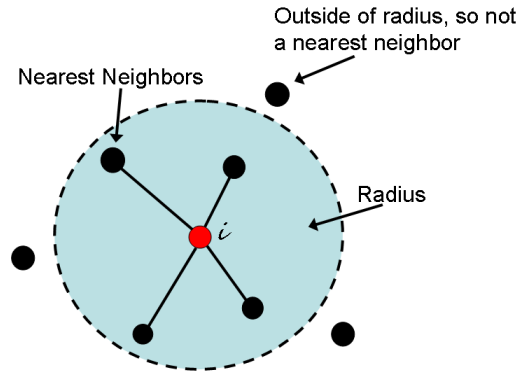


Figure 5: Schematic showing how the nearest neighbor method works.

Collision Model

This year, galaxy to galaxy interactions were added, which is the force behind the collisions. This year the program was also parallelized using MPI (see pg. 18) whereas last year the program was parallelized using OpenCL which utilizes the GPU instead of the CPU. The collision model shows how the interactions between galaxies are implemented in my model. Figure 6 (below) shows two galaxies colliding. The green arrow represents core to core interactions which are the dark matter halos. The blue arrow represents the core to star interactions, and the orange arrow represents the star to star interactions which is the nearest neighbor method. Each galaxy has internal interactions and external interactions. The nearest neighbor method that was above is used in both internal and external interactions.

Gravitational interactions between the two galaxies

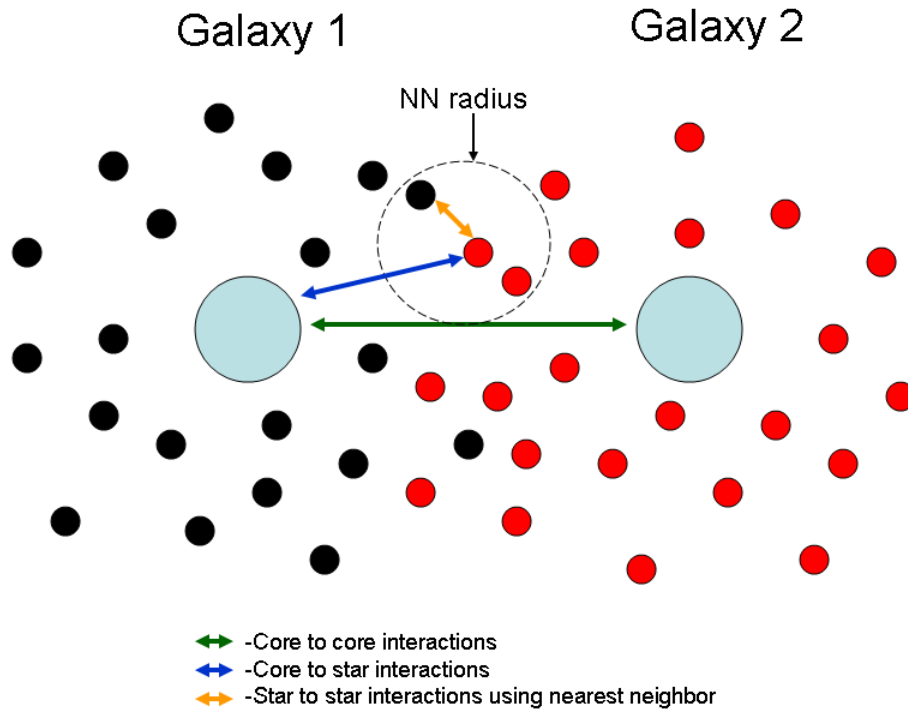


Figure 6: Interacting forces used in my model between colliding galaxies

Results

This project has mainly focused on the predicted future collision between the Andromeda and Milky Way galaxy. Three main simulations were ran: (1) Normal mass simulation (realistic mass values experimentally measured), (2) Milky Way has $\frac{1}{4}$ dark matter mass of Andromeda, (3) Normal mass both galaxies rotate in the same direction. Since the Milky Way is really far apart from Andromeda, another program placed a point (which represented the whole galaxy's mass) and ran a two point simulation until the two galaxies got within a certain distance (100 kpc). When the two galaxies reached that distance, the program will use those conditions and use them in the main program as initial conditions, place the galaxies, and start running the simulation. This method allows an easy assumption of the galaxies while they are far apart, which saves a lot of calculation time. Figure 7 (below) shows how the initial conditions were determined.

Initial Conditions for Andromeda (purple) and Milky Way (green)

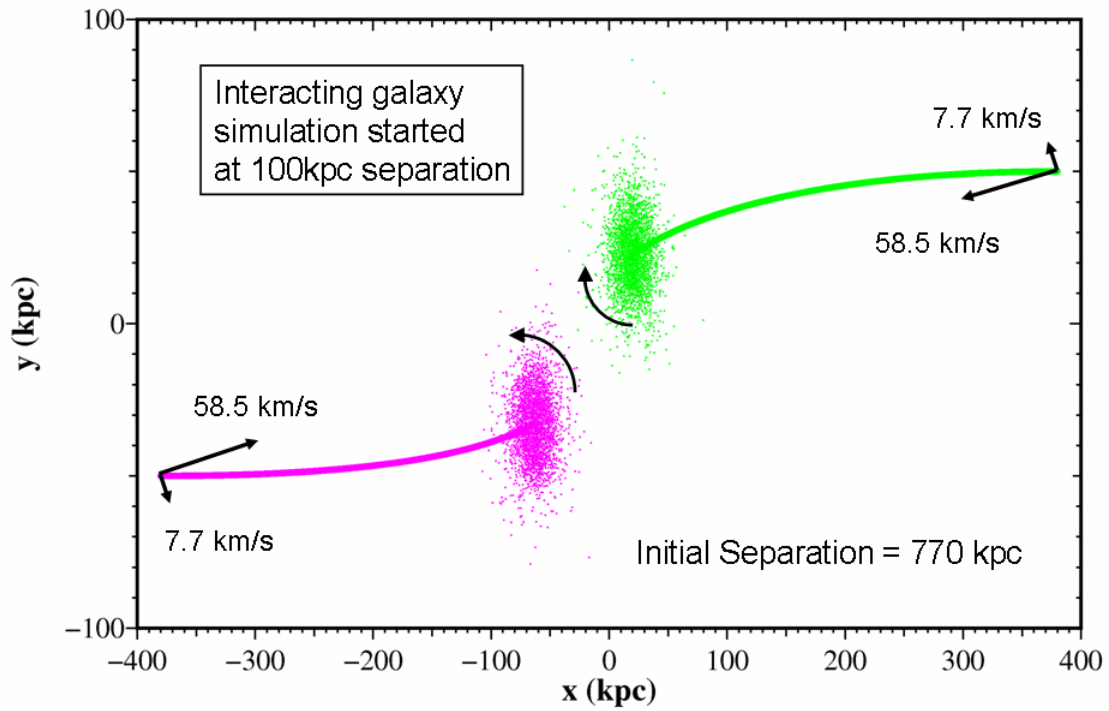


Figure 7: Shows initial separation between two galaxies and how starting conditions were obtained (MW – green, Andromeda – Purple)

All simulations that were run used a total of 4000 particles (2000 in each galaxy), a 200 year time step (dt), velocity Verlet method, and ran to about 12 billion years. Each simulation took about 2 days on a 3.0Ghz Intel Core 2 Duo. Figure 8-12 show results from the normal mass simulation. In the following figures, green represents the Milky Way and purple represents Andromeda.

Simulation 1

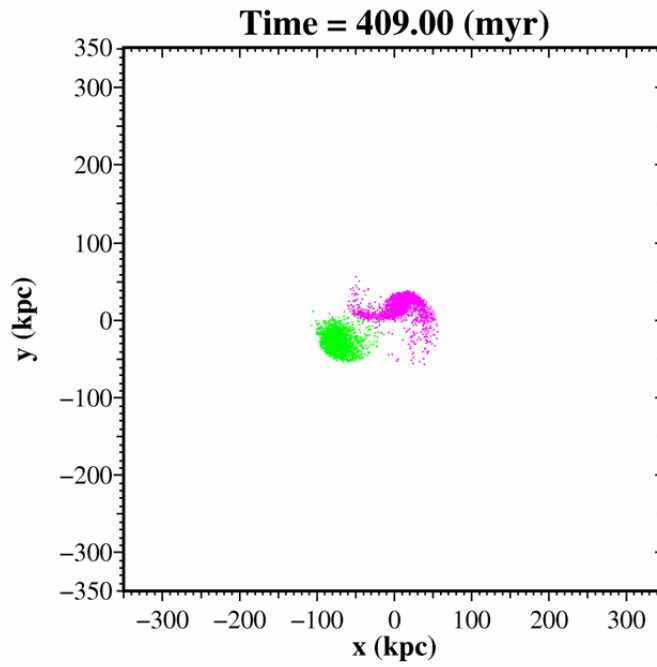


Figure 8: After initial collision, Andromeda has tails on the edges

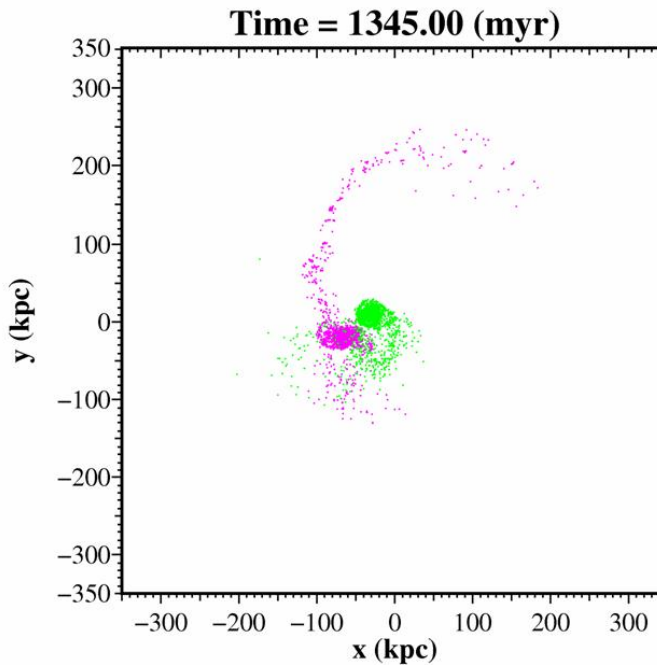


Figure 9: After several collisions, Andromeda has a large tail that extends out into space.

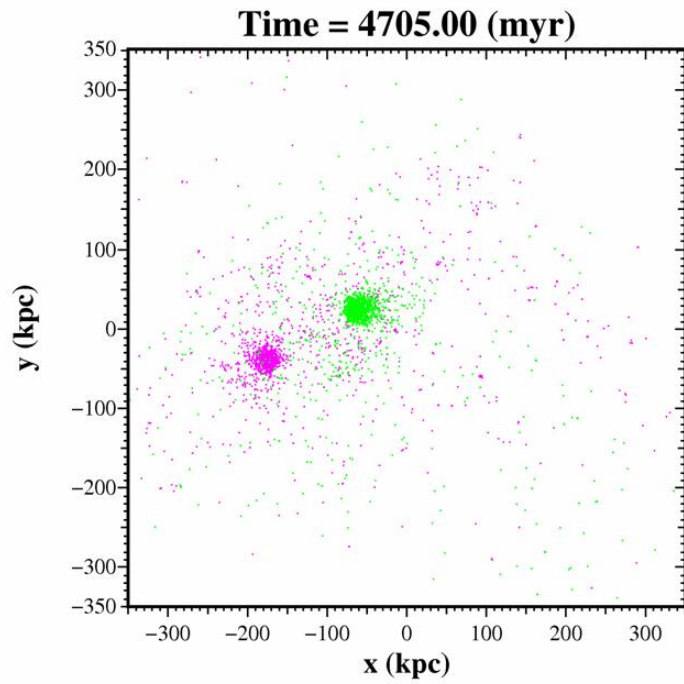


Figure 10: After several collisions, both galaxies slowly joining and slowing down.

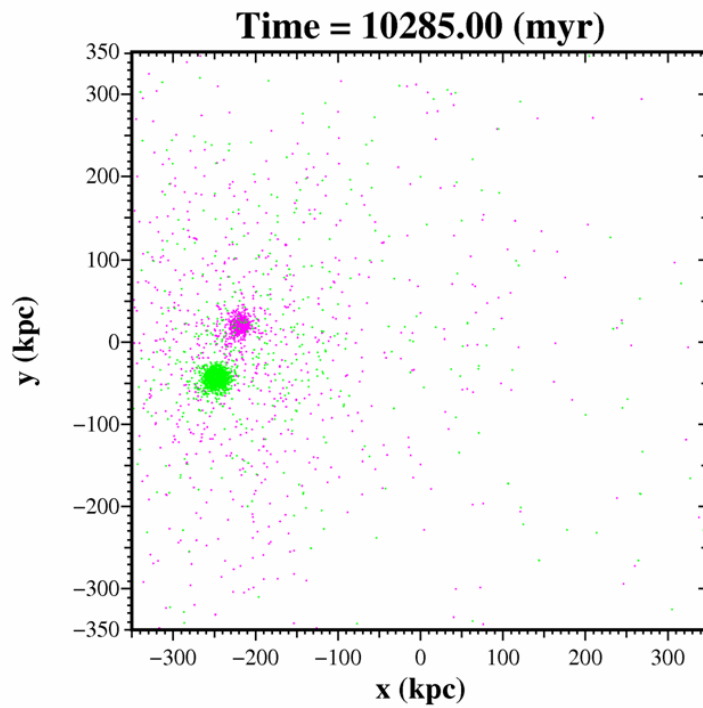


Figure 11: Near end of simulation, both galaxies have basically formed an elliptical galaxy.

Figure 8 shows the galaxies right after the initial collision, Andromeda starts to form two tails on either end of the galaxy. In figure 9, after several collisions, the tail extends out farther into space. This feature matches colliding galaxies viewed from observatories, this image below shows that this is a realistic feature.



Figures 10 and 11 shows the galaxies slowing down and accumulating stars around the two central forces which will lead to eventually causing the galaxies will merge and form an elliptical galaxy because they have about the same visible mass and dark matter mass. If one galaxy had less dark matter than the results will change. This was proven with simulation 2 where the Milky Way has $\frac{1}{4}$ of Andromeda's dark matter mass.

Simulation 2

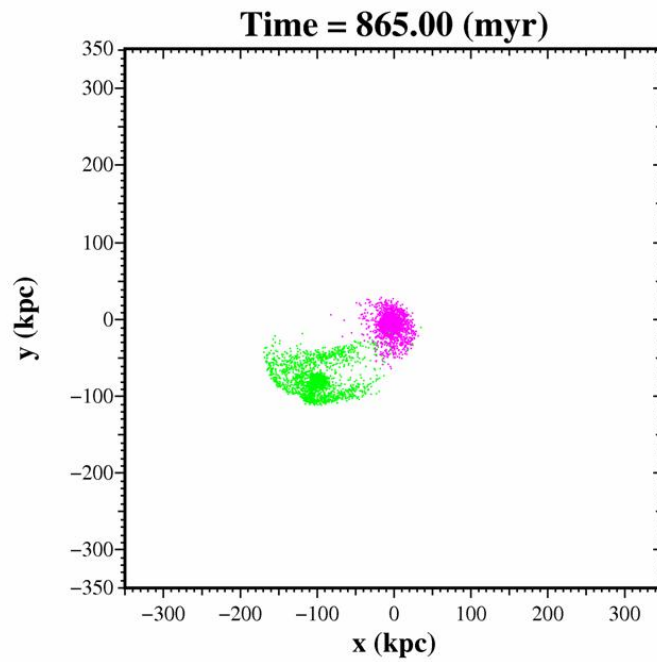


Figure 12: After initial collision, Milky Way (green) is getting thrown apart by Andromeda

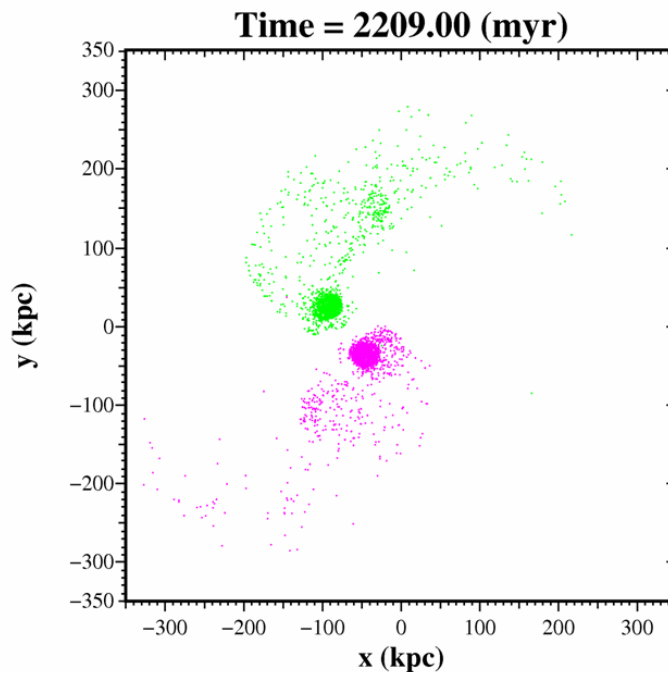


Figure 13: Several collisions have taken place, Milky Way has more matter slung out of the galaxy.

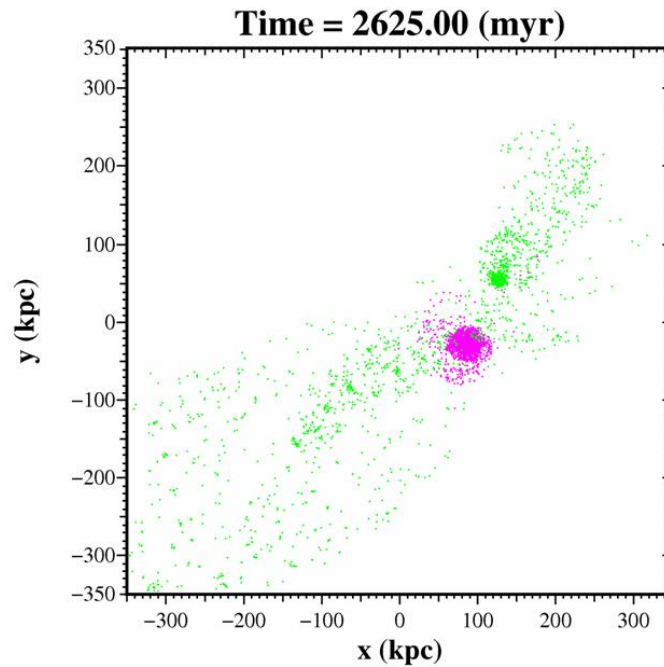


Figure 14: Milky Way has a smaller central mass, and has ‘clumps’ of stars further away. Andromeda stays compact and is not really affected.

In this simulation (figures 12-14), the Milky Way has $\frac{1}{4}$ the dark matter mass of Andromeda. Since the Milky Way has less dark matter mass, its stars are not as compact and do not remain intact throughout the collision. Early on in the simulation (figure 12) the Milky Way is already pretty much separated, and scattered. Andromeda remains stable and compact throughout the collision and is not impacted by the Milky Way at all. This simulation proves that in order to keep a galaxy stable throughout the collision, it has to have enough mass. Often if a collision has a large and small galaxy, the small galaxy will be absorbed or become a satellite galaxy. If two galaxies collide and they have about the same mass, they will both remain stable and eventually form a large elliptical galaxy. Both of the galaxies will orbit each other like a binary star. Below are the dark matter core trajectories for both galaxies. For simulation 1 (figure 15), the two galaxies orbit each other and become an elliptical like mentioned above, for simulation 2 (figure 16), Andromeda does not move a lot, it stays pretty much in a line whereas the Milky Way gets thrown around Andromeda and becomes a satellite galaxy to Andromeda. These two figures show how the galaxies move throughout the simulation.

Trajectories of central dark matter mass for Andromeda (purple) and Milky Way (green)

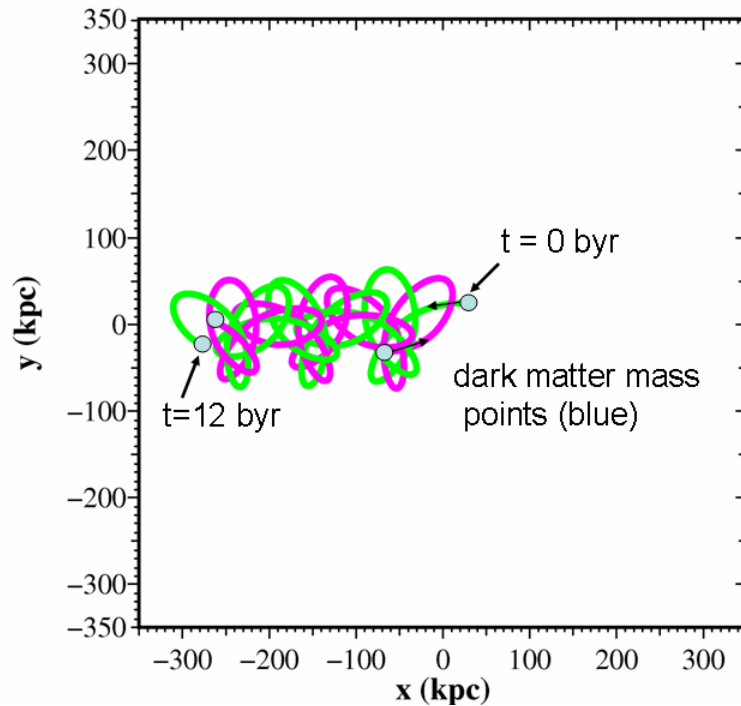


Figure 15: Shows simulation 1 trajectories, Andromeda and the Milky Way stay in a compact orbit and form an elliptical.

Trajectories of central dark matter mass for Andromeda (purple) and Milky Way (green)

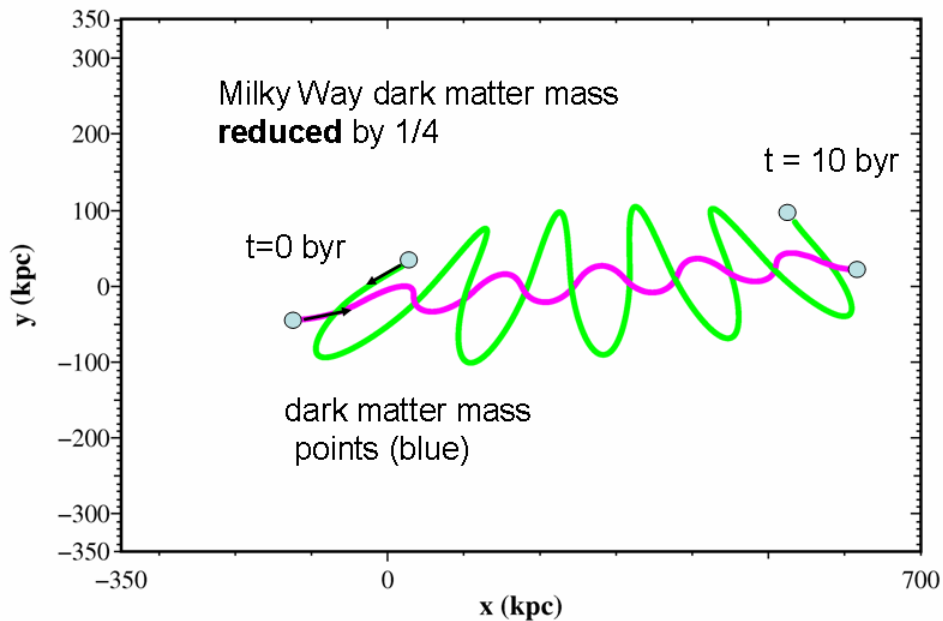


Figure 16: Shows trajectories for simulation 2, Andromeda is not really affected but the Milky Way gets slung around Andromeda.

Simulation 3

In simulation 3, the galaxies have the realistic mass values (same as simulation 1) and are rotating in the same direction. Having different or same rotation does not change the end result, however it does change how the galaxy collides. Meaning that whether or not they are rotating in the same direction they will eventually form an elliptical galaxy, they will just collide different. Figure 17 below shows the two galaxies rotating in the same direction (both are going counter-clockwise).

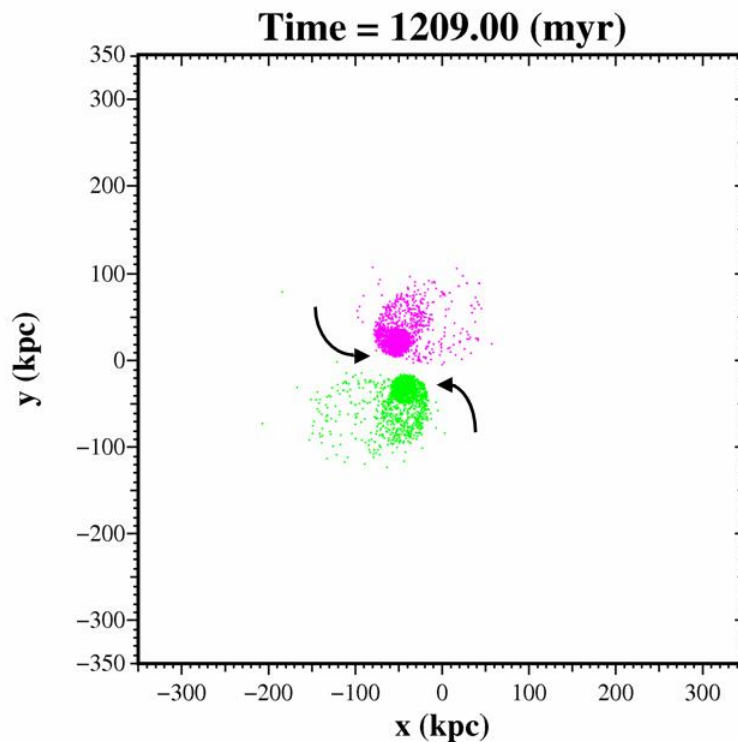


Figure 17: Both galaxies are rotating counter-clockwise and are even in mass.

MPI

MPI (Message-Passing-Interface) is a parallel programming language that is mostly used on supercomputers. It uses the machines multiple cores to work on one program, making it run much faster rather than a program that only utilizes one core. An MPI version of my galaxy model was written to parallelize my code and see how fast multiple cores would compare to a single core. After running the MPI version and comparing it to the normal C code, there was about a 50% speed up once the calculation reached about 16-32 thousand stars. Since this ran much faster, this would allow my model to have much more stars, which would also make it more accurate. Figure 18 (below) shows the amount of time it takes to do a single time step (with a 200 year dt) with different amounts of stars. The green line shows a single cpu, the red line shows two cpus using the MPI version of the code.

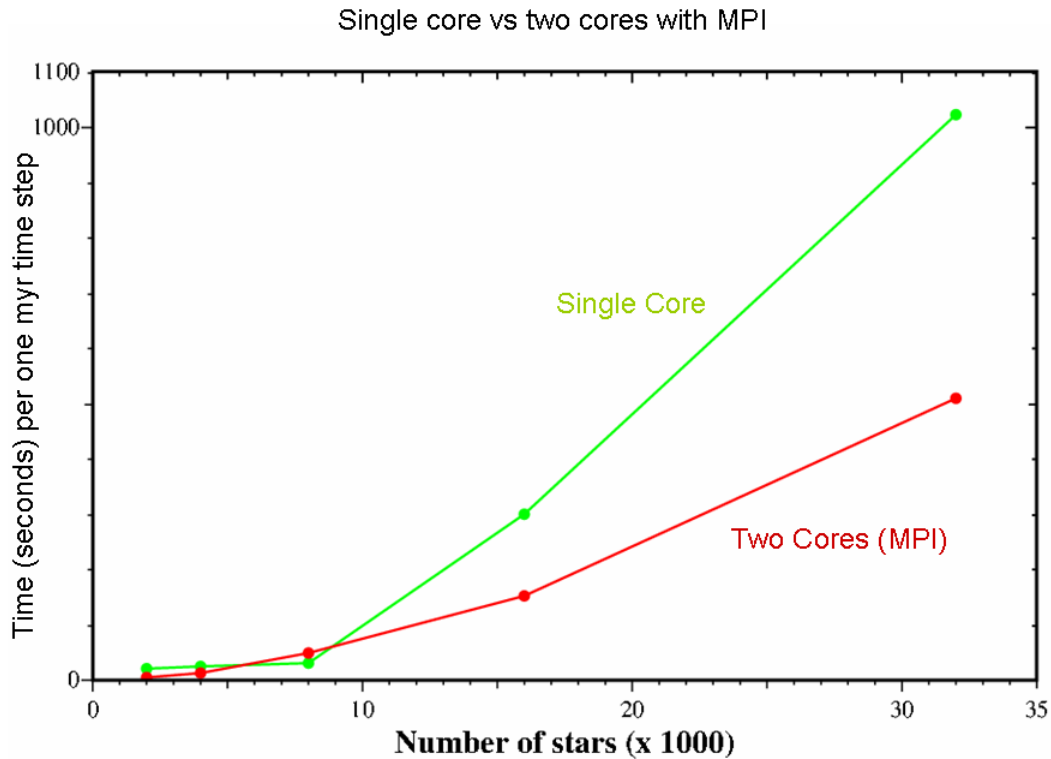


Figure 18: Shows the speed up between a MPI version and a normal code. MPI (red) takes about half the time it takes a single core from 16 – 32 thousand stars

Conclusion

I was able to successfully create a computer simulation of galaxy collisions using C. My 2D simulation results are consistent with many observed galaxy collisions and are also similar to 3D professional models. My simulation shows that dark matter has large effects on spiral galaxy collisions: (1) If the two galaxies dark matter halos are similar in mass, then both galaxies are distorted and eventually merge to form a larger elliptical galaxy, and (2) If one of the two galaxies dark matter halo is large, then the smaller galaxy is largely distorted, it orbits the larger one, and it is eventually absorbed. The larger galaxy remains mostly intact. The third simulation proves that if galaxies rotate in the same direction, in the end it will give the same result. In conclusion, this project can be used to help understand how dark matter effects galaxy collisions and what it really is.

Future Work

Future work includes comparing my nearest neighbor model to a full N-body calculation, replacing my static halo model with interacting/moving dark matter particles, going to three dimensions, and including gas in my galaxy models.

References

- 1) Navarro J. F., Frenk C.S, and White S.D. “A Universal Density Profile from Hierarchical Clustering.” *The Astrophysical Journal* **490** (1997): 493-508.
- 2) Begeman, K.G., “HI Rotation Curves of Spiral Galaxies. I. NGC 3198”, *Astronomy and Astrophysics* **223** (1989), 47-60.Print.
- 3) Zwicky, F., “Die Rotverschiebung von extragalaktischen Nebeln”, *Helvetica Physica Acta* **6** (1933): 110–127.
- 4) Milgrom, M., "A modification of the Newtonian dynamics as a possible alternative to the hidden mass hypothesis". *Astrophysical Journal* **270** (1983): 365–370.
- 5) Bennett, J.O., Donahue, M., Schneider, N., and Voit, M., “The Essential Cosmic Perspective”, Pearson; Addison Wesley, (3rd edition) 2005.Print.
- 6) Widrow, L.M. and Dubinski, J., “Equilibrium Disk-Bulge-Halo Models for the Milky Way and Andromeda Galaxies”, *The Astrophysical Journal* **631** (2005): 838-855.
- 7) Barns, J.E. and Hernquist, L, “Dynamics of Interacting Galaxies”, *Annual Reviews of Astronomy and Astrophysics*, **30** (1992): 705-742.
- 8) Withagen, J.C.J.G., “On the Collision between the Milky Way and the Andromeda Galaxy”, Masters Thesis, Universiteit van Amsterdam, 2008.
- 9) Dubinski, J., “The Great Milky Way - Andromeda Collision”, in *Sky and Telescope*, October 2006, pg. 31.

Appendix 1

Four different methods were tried for solving
Newton's equation of motion (1D examples):

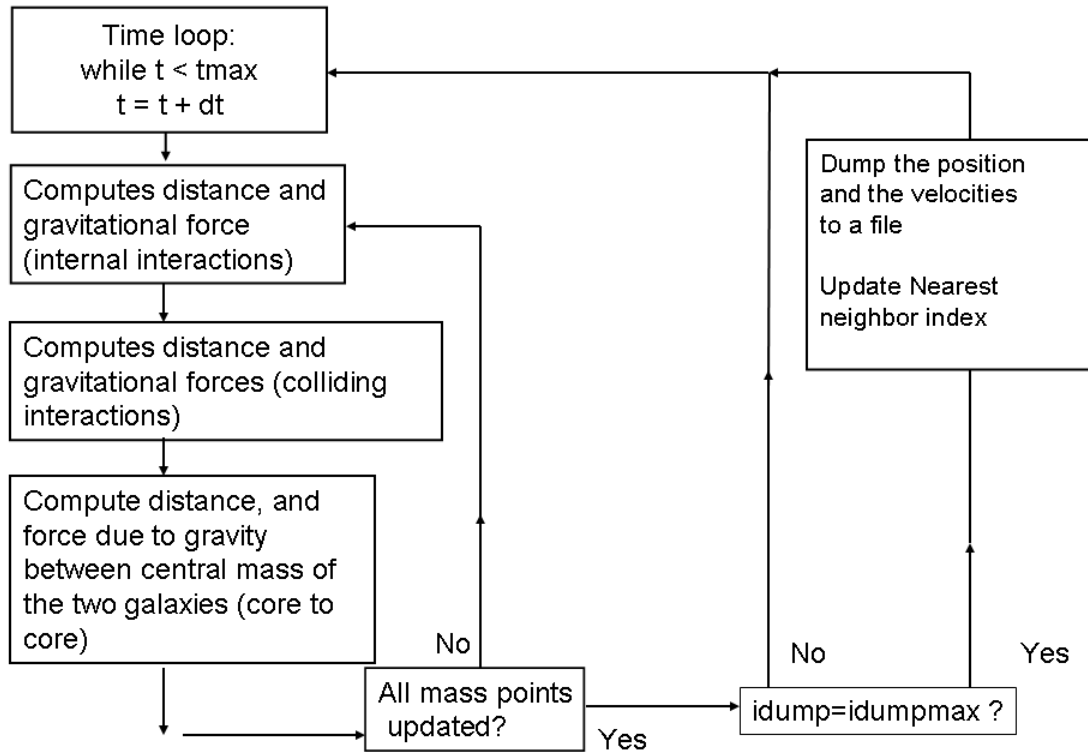
- A) Euler (1st order)
- 1 compute a at x_0
 - 2 $x = x_0 + v_0 dt$
 - 3 $v = v_0 + a dt$
- B) Symplectic Euler (1st order)
- 1 compute a at x_0
 - 2 $v = v_0 + a dt$
 - 3 $x = x_0 + v dt$
- C) Velocity-Verlet (2nd order)
- 1 compute a at x_0
 - 2 $x = x_0 + v_0 dt + 0.5 a dt dt$
 - 3 $v' = v_0 + 0.5 a dt$
 - 4 compute a at new x
 - 5 $v = v' + 0.5 a dt$
- D) Predictor-Corrector (2nd order)
- 1 compute a at x_0
 - 2 $x_p = x_0 + v_0 dt$
 - 3 $v_p = v_0 + a dt$
 - 4 $x_c = x_0 + v_p dt$
 - 5 compute a at x_p
 - 6 $v_c = v_0 + a dt$
 - 7 $x = (x_p + x_c) 0.5$
 - 8 $v = (v_p + v_c) 0.5$
- Method C: Fast, accurate, and stable for galaxy simulations

Method D: Requires 2 force evaluations per time step (2x slower)

Method A is Euler's method, B is Symplectic Euler, C is Velocity Verlet, and D is Predictor Corrector. The velocity Verlet and the Predictor Corrector are the most accurate methods. For galaxy simulations, the velocity Verlet method C was essentially identical to the Predictor Corrector but is two times faster. So the velocity Verlet (method C) was chosen for all of my simulations.

Appendix 2

Flow Chart of Main Computational Loop



```

while (t<tmax)
{
    t=t+dt;
/* Loop over all of the mass points */
    for(j=0;j<n;j++) {
/* update the position of mass j in galaxy #1 */
        x[j] = x[j] + vx[j]*dt + 0.5*Ax[j]*dt*dt;
        y[j] = y[j] + vy[j]*dt + 0.5*Ay[j]*dt*dt;
/* form 1st half of velocity update for mass points in galaxy #1*/
        vx[j]=vx[j] + 0.5*Ax[j]*dt;
        vy[j]=vy[j] + 0.5*Ay[j]*dt;
/* update the position of mass j in galaxy #2 */
        x2[j] = x2[j] + vx2[j]*dt + 0.5*Ax2[j]*dt*dt;
        y2[j] = y2[j] + vy2[j]*dt + 0.5*Ay2[j]*dt*dt;
/* form 1st half of velocity update for mass points in galaxy #2 */
        vx2[j]=vx2[j] + 0.5*Ax2[j]*dt;
        vy2[j]=vy2[j] + 0.5*Ay2[j]*dt;    }
/* Update positions and velocities of each galaxy's central mass */
        dx = dx + vxg*dt + 0.5*Axg[0]*dt*dt;
        dy = dy + vyg*dt + 0.5*Ayg[0]*dt*dt;
        vxg=vxg + 0.5*Axg[0]*dt;
        vyg=vyg + 0.5*Ayg[0]*dt;
        dx2 = dx2 + vxg2*dt + 0.5*Axg[1]*dt*dt;
        dy2 = dy2 + vyg2*dt + 0.5*Ayg[1]*dt*dt;
        vxg2=vxg2 + 0.5*Axg[1]*dt;
        vyg2=vyg2 + 0.5*Ayg[1]*dt;
/* ----- */
/* Now compute the Ax and Ay at the new x and y
/* So that the Velocity can be updated (velocity Verlet algorithm)
/* ----- */
/*compute accelerations for stars in each galaxy */
        getaccel(n, inn, cdm, mcdm0, m0, A, r, x, y, Ax, Ay, mcdm, mt, m,
jnA, nnA, drmin,dx,dy,Axg,Ayg,id);
        getaccel(n, inn, cdm, mcdm02, m02, A, r2, x2, y2, Ax2, Ay2, mcdm2,
mt2, m2, jnB, nnB, drmin,dx2,dy2,Axg,Ayg,id2);
/*compute accelerations for the central mass in each galaxy */
        getaccelCore(n,ict,cdm, mcdm0,m0,A, r, x, y, Ax, Ay, mcdm, mt, m,
jnA, nnA, drmin,dx, dy,mcdm02, m02, r2,x2, y2, Ax2, Ay2, mcdm2, mt2,
m2, jnB, nnB, dx2, dy2, Axg, Ayg);
/*compute accelerations for the stars in each galaxy due to the other
galaxy's central mass */
        getaccelCollide(n,ict,cdm, mcdm0,m0,A, r, x, y, Ax, Ay, mcdm, mt,
m, jnA, nnA, drmin,dx, dy,mcdm02, m02, r2,x2, y2, Ax2, Ay2, mcdm2, mt2,
m2, jnB, nnB, dx2, dy2, Axg,Ayg,id2);
        getaccelCollide(n,ict,cdm, mcdm02,m02,A, r2, x2, y2, Ax2, Ay2,
mcdm2, mt2, m2, jnB, nnB, drmin,dx2, dy2,mcdm0, m0, r,x, y, Ax, Ay,
mcdm, mt, m, jnA, nnA, dx, dy,Axg,Ayg,id);
/*compute accelerations for the stars in each galaxy due to the other
galaxy's nearby stars */
        getaccelNeighborCollide(n,ict,cdm, mcdm0,m0,A, r, x, y, Ax, Ay,
mcdm, mtb, m, jn1, nn1, drmin,dx, dy,mcdm02, m02, r2,x2, y2, Ax2, Ay2,
mcdm2, mt2b, m2, jn2, nn2, dx2, dy2);
        getaccelNeighborCollide(n,ict,cdm, mcdm02,m02,A, r2, x2, y2, Ax2,
Ay2, mcdm2, mt2b, m2, jn2, nn2, drmin,dx2, dy2,mcdm0, m0, r,x, y, Ax,
Ay, mcdm, mtb, m, jn1, nn1, dx, dy);
/* update the velocities of all mass points in each galaxy */
        for(j=0;j<n;j++) {

```

```

    vx[j]=vx[j] + 0.5*Ax[j]*dt;
    vy[j]=vy[j] + 0.5*Ay[j]*dt;
    vx2[j]=vx2[j] + 0.5*Ax2[j]*dt;
    vy2[j]=vy2[j] + 0.5*Ay2[j]*dt;    }
/* Update velocities of each galaxy's central mass point */
    vxg=vxg + 0.5*Axg[0]*dt;
    vyg=vyg + 0.5*Ayg[0]*dt;
    vxg2=vxg2 + 0.5*Axg[1]*dt;
    vyg2=vyg2 + 0.5*Ayg[1]*dt; }

```

“getaccel” Subroutine

```

for(j=0;j<n;j++)    {
    rsq = ((x[j]-dx)*(x[j]-dx)) + ((y[j]-dy)*(y[j]-dy));
    r[j]=sqrt(rsq);
    rkpc=r[j]/kpc;
/* Dark Matter Halo is NFW: static, spherically symmetric mass
distribution */
    mcdm[j]=cdm*mcdm0*(-rkpc/(A + rkpc) - log(A) + log(A+rkpc));
/*Total mass inside radius rkpc = Black hole + (core + disk) + CDM*/
    mass=m0 + mt[j] + mcdm[j];
/* force due to gravity between mass point m[j] and central total mass
*/
    f = (g*m[j]*mass)/(rsq+drmin);
    ex = (x[j]-dx) / r[j];    /* cosine */
    ey = (y[j]-dy) / r[j];    /* sine */
    Fintx=0.0; /* Compute NEAREST NEIGHBOR (NN) force INTERACTIONS */
    Finty=0.0;
    if (nn[j] != 0) {
        for(i=0;i<nn[j];i++)    {
            ind=j*NNMAX + i; /* nearest neighbor index */
            iv=jn[ind];
            dr=(x[j] - x[iv])*(x[j] - x[iv]) + (y[j]-
y[iv])*(y[j]-y[iv]);
/* Force between mass point m[j] and nearest neighbor iv */
            fint=g*m[j]*m[iv]/(dr+drmin);
            dr=sqrt(dr);
            exi = (x[j]-x[iv]) / dr;
            eyi = (y[j]-y[iv]) / dr;
            Fintx=-fint*exi + Fintx;
            Finty=-fint*eyi + Finty;} }
/* Total force due to Gravity of central mass + NN interactions on
m[j]*/
    Fx = -f * ex    + Fintx;
    Fy = -f * ey    + Finty;
/* Total acceleration of mass m[j] */
    Ax[j] = Fx / m[j];
    Ay[j] = Fy / m[j]; }

```


Appendix 3

Equations Used

1) $\mathbf{F} = \mathbf{m a}$

Newton's law of motion:

Force = mass * acceleration

2) $\mathbf{F} = \mathbf{G m_1 m_2 / r^2}$

Newton's law of gravity used to compute gravitational force between two objects:

Force = Gravitational Constant * mass1 * mass2 / distance squared

3) $\mathbf{m_1 V^2 / r} = \mathbf{G m_1 M / r^2}$

The centripetal force is due to gravity, used to set initial velocities (see diagram below)