PHY480 REPORT

**1. Abstract**

**2. Introduction**

**3. Body**

**3.1 Literature review**

**3.1.1 NBODY problem**

Newton’s law of gravitation describes how a group of stars interact in a star cluster. These gravitational interactions cause the dynamical properties (velocity, position, acceleration) of the stars to change. This causes a change in the dynamical properties of the whole region. The evolution of this stellar region can be observed in a NBODY simulation. The NBODY problem is incorporates the prediction of future dynamical properties of the stars in the system. The initial properties of the particles in the system are known. In a system of N particles, the acceleration of a particle can be defined as,

-1

Where, mass of the test particle

the unit vector along the direction of the distance vector

the modulus square of the distance between the bodies considered

Integrating equation (1) provides the position and velocity of a particle at any time t. For *N*=2 the above equation is analytically solvable. Since we are considering stellar clusters as our system, where N2, numerical methods are considered.

Numerical integration of equation 1 provides the below solutions.

-2

-3

Where, and the new position and velocity of the particles.

and the initial position and velocity of the particles

, , , the initial acceleration of the particles with the latter three being the 1st, 2nd, 3rd time derivatives.

timestep for the simulation

The timestep *dt* determines the accuracy of the values of the future position and velocity of the particles. It goes inversely with the computational time. There is a rise in the number of calculations done in a single simulation as *dt* drops.

Accuracy in predicting the motion of stars in the cluster is the main task. The error in the solution is proportional to the timestep. As we go to higher orders, a small drop in *dt* will imply a large reduction in the error.

For our project we consider the 2nd order method, with the 4th order predictor-corrector (Hermite scheme) method used in semester 2. Below are the equations used for the second order method.

-4

-5

The 4th order predictor-corrector method improves on the accuracy of the previous methods.

**3.1.2 History of NBODY simulation**

**3.1.3 Astrophysics applications of NBODY method**

**3.2 Progress**

The initial work on the project was carried by constructing a simple second order code. Equation 5 can be corrected by considering an assumption.

-6

Initially a three-body system of Sun-Earth-Jupiter was considered for the problem, with the rest of the planets added later.

For the first block of the code, the declaration of all the variables used was done. For certain known variables like the gravitational constant *G*, the mass, velocity and initial positions of the planets, the initialisation is done. From the planetary fact sheet on the NASA website[1], the initial positions and velocities were taken. Certain initial conditions for the position and velocity are mentioned below. From these conditions, the orbits of the planets were forced to be on the XY plane.

For this system, we added the code which did centre-of-mass and velocity corrections. Without including this correction, the system would drift away from the origin during the simulations. The centre-of-mass and centre-of-velocity in the x direction is calculated as below,

-7

-8

Where, , is the total mass of the system

An infinite loop was created which would run the simulations to a time , with a time-step *dt.* Future positions and velocities determined using equation (4) and (6). Modified form of equation (1) was used to calculate the future accelerations.

-9

Where,

The orbits of the planets were produced where the runtime for the simulations was 1000 yr. The timestep was chosen to be 1000 sec.

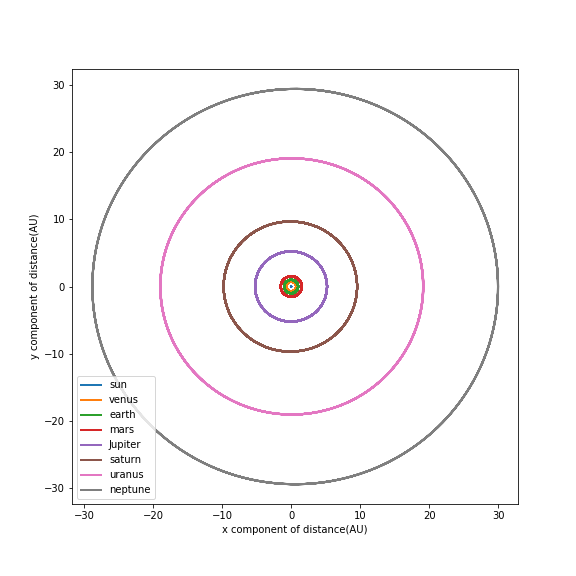


Figure 1: Orbits of the planets in the solar system.

To check the stability of this system, an energy conservation check was done. This was carried out by determining the fractional energy of this system using the equations below. A counter was set so that after every 6 months this calculation was possible.

-10

-11

-12

Where,

total energy of the system at time *t*

and the fractional energy and the initial energy of the system respectively

A plot of the fractional energy varying with time was produced for a time step, seconds.

A screenshot of a cell phone

Description automatically generated

Figure 1: fractional energy of the system against time (in yrs). The simulation was run for 1000 yrs with a time-step of 1000 seconds.

The fractional energy of the system depends on the timestep, dt. For good accuracy, the time-step should be very low. This significantly increases the computation time, making the second order code too slow. For getting to Million yr timescales, the fourth order method presents a good method.

Another test for the code was done by looking at the distances from the sun varying with time for the planets. For every 1000th iteration, the distance was calculated for the bodies. This helped reduce the computation time.

A screenshot of a social media post

Description automatically generated Figure 2: The separation from the sun (in AU) against time (in yrs). The periodic variations present in the curves is due to the interactions between the planets.

We declare all the variables (, etc.) to be included in the code. The initialisation of the known variables is done (i.e. setting , number of bodies, *n*=3, etc.)

For this system, we set certain initial conditions for the velocity and position of the bodies.

This forces the bodies to orbit in the XY plane, starting from the X axis with the initial velocity in the Y direction.

The most important part of this code comes in creating an acceleration loop. This loop is used in determination of the initial acceleration, and the future acceleration, .

We produce a time loop which runs up-to 30 yrs. With the initial and future accelerations in hand, the future position and velocity of the bodies is calculated using equation 4 and 6. A figure is produced of the orbits of the three bodies using the x and y positions. The orbits of the Sun-Jupiter-Earth will drift away from the origin. However, it is not seen here since the running time was significantly short. This is due to the presence of a common centre of mass for the bodies which was not included at the beginning.

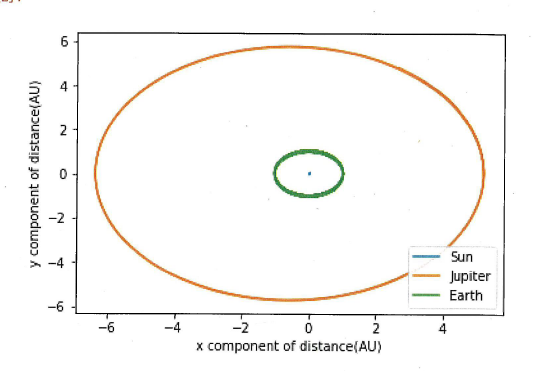


Figure 1: Orbits of the Sun-Jupiter-Earth with the simulations run for 30 yrs. The orbits are not corrected for the centre-of-mass and velocity.

To correct that, we introduce the centre-of-mass and velocity corrections (given in the code in the appendix). Then with the runtime changed to 100 yrs, the orbits did not show the drifts.

The stability of this system is checked using the fractional energy calculation using the equation given below.

Where and – the initial and the current energy of the system (at the current time, )

This energy conservation check was done with the timestep at 100 sec.

3.2.2 Whole solar system

To make the code more useful, the whole solar system is included. The known values of the position, velocity and mass are given. These values are taken from the Planetary fact sheet in the NASA website[1]. The simulations are made to run to 1000 yrs.

For this system, the energy conservation check is carried out, with a timestep of 1000 seconds. For every 6 months, the fractional energy values are noted. A fractional energy plot is produced for this system.

A screenshot of a cell phone

Description automatically generated

Figure 2: Fractional energy varying with time (in yrs) for the planets in the solar system.

Due to the very small values of the fractional energies, the system is stable.

Another stability test comes from investigating the how the distance from the sun () varies for the planets. The simulations are run for 1000 yrs with a similar timestep used. To reduce the computation time, the values for are taken at every 1000th iteration.

A screenshot of a social media post

Description automatically generatedFigure 3: Distance from the Sun (AU) against time (yrs).

The periodically varying curves give good idea of the stability of this system. The features in the curve for Saturn arrive from its interactions with Jupiter.

3.3 Project plan

At the start of week 1, the core part of the work in semester 2 begins with task1, i.e., building a basic fourth order predictor-corrector code. A time of 2-3 weeks is given for this task based on its difficulty. After forming the base, we require an adaptive timestep for the code (task 2). Depending on the errors obtained from the energy checks, the code either doubles or halves the timestep, *dt*. A similar amount of time of 3 weeks is assigned to this task. The two main components of the code are ready. This leads to task 3 of testing the code. The results obtained from these tests determine whether the code is working. For example, we should observe milankovic cycles when we produce the plot of the orbits of the planets in our Solar System.

Task 4 provides the motivation of writing this piece of code. We apply this code to any astrophysics problem (like late oligarchic phase of planet formation). Moreover, certain tweaks are added at this point to increase the speed of the code. With a time of 2 weeks, it overlaps with the easter break. Then we lead to the most important task 5, the write-up of the report. All the figures and results produced in tasks 3 and 4 are included in the report. Tasks 4 and 5 overlap at the beginning of easter break. A time of 3-4 weeks is assigned so that a draft of the final report can be submitted to the supervisor approximately two weeks before the deadline (17/05/2019). The final task involves in refining the report. Within a week the report is checked for any mistakes before the final submission.

4. Conclusion

5. References

[1]- <https://nssdc.gsfc.nasa.gov/planetary/factsheet/>