e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:05/May-2023 Impact Factor- 7.868 www.irjmets.com

SIMULATION OF ORBITAL MECHANICS THROUGH ITERATION

Sanil Arora*1, Dr. Anant Prakash Pandey*2

*1Student, Heritage Xperiential Learning School, Gurugram, India

*2Guide, Phd From Banaras Hindu University, Postdoctoral Researcher At Indian Institute Of Technology Jodhpur, India.

DOI : https://www.doi.org/10.56726/IRJMETS40107

ABSTRACT

Different aspects of orbits of celestial bodies can be calculated by using different formulae. In this paper it is proposed that iterating over some initial conditions, using only one formula, many other phenomena of orbits, such as the variation in speed during the orbit, can be simulated. By calculating only the force between two bodies for some number of discrete time steps, it is possible to compute velocities, kinetic energies, and potential energies for the particles in question which match with the expected behaviour as predicted by other formulae. This study uses such a simulation to simulate the orbits of bodies with different initial conditions and then compares then analyses their behaviour with respect to quantities like kinetic and potential energies, values which the simulation is not programmed to take into account but does so anyway as an emergent effect. The study thereby validates this method of simulation of forces through iteration with discrete time steps.

Keywords: Gravitation, Simulation, Iteration, Orbital Mechanics.

I. INTRODUCTION

When simulating a natural phenomenon, such as gravity, it can often be useful to break it down into its simplest interactions. While orbits may be observed to follow many complex patterns, they all stem from the simple interaction of gravity between two masses. In this simulation, we will break the problem down into a single computation, the force between two particles due to their masses, calculated using Newton's Law of Gravitation

The simulation will be run in python with its output being displayed using either Pygame, to show the motion of bodies in real time, or using libraries like Matplotlib to show graphs of physical quantities over a period of time.

The goal of the simulation is not to be computationally effective. Other methods such as moving a particle along a predefined ellipse and changing its velocity to conserve angular momentum can be used to achieve a similar goal. The purpose of this simulation is to achieve that result through iteration of a simple interaction. Also unlike the previously mentioned method, due to the iterative nature of this method, it can be extended to an arbitrary number of bodies; however, in the scope of this paper, only the outputs produced by the interactions between two bodies have been verified.

II. METHODOLOGY

Computational aspects of the simulation

The simulation works by keeping track of an n number of objects of class "pointmass" which have certain numerical quantities associated with them. These quantities are mass, position, and velocity. Position is stored as two values, x and y coordinates on a cartesian plane. Velocity is also stored as two values: magnitude and direction measured in radians from the positive x axis. Arithmetic operations are then performed on these quantities for every discrete timestep. These timesteps can be decided arbitrarily; in this paper the results shown will use a value of 300 timesteps per second. On every timestep, every point mass calculates the force between itself and another point mass then updates its velocity accordingly. It does so for every other point mass in the simulation other than itself, giving the simulation a Big O Notation of n^2. After the calculation of forces and updating of velocities, every particle's position is moved based on the velocity associated with it.

Physical aspects of the simulation

This section will elaborate on finding the force between two point masses and then updating their velocities. First a vector between the two points is calculated using inverse trigonometric functions. It is stored in terms of its magnitude and direction relative to the positive x axis.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:05/May-2023 Impact Factor- 7.868 www.irjmets.com

The force is calculated using Newton's law of universal gravitation. Let 'm_1' be the mass of the body we are calculating the force on.

$$F = Gm_1m_2/r^2$$

Dividing by 'm_1' on both sides and now writing 'm_2' as 'm'. 'a' being the acceleration.

$$a = Gm/r^2$$

'G' can be arbitrarily defined and tweaked to control how strong the effect of gravity should be in the simulation. It is important to note that whatever value of 'G' has to be the same value of 'G' used later when verifying the results of the simulation through energy conservation.

The value of 'r' is the magnitude of the vector between the points stored earlier.

'a' itself is stored as a vector with a magnitude equaling Gm/r^2 and direction the same as 'r'.

To update the velocity, the resultant of the current velocity vector and the acceleration vector is taken. This can be thought of as adding the acceleration, or the change in velocity for that discrete time step to the preexisting velocity. On every time step, all particles then change position based on their current velocity. The entire process is then repeated again and can be done so for an arbitrary number of steps.

III. MODELING

'G' is assigned the value 6 in the following simulations. The vertical axis line (drawn in light blue): y = 500. The horizontal axis line (drawn in light red): x = 350. Instead of showing the motion of the particles over time, all of their positions in a given time interval are shown at the same time to describe the shape of their orbit.

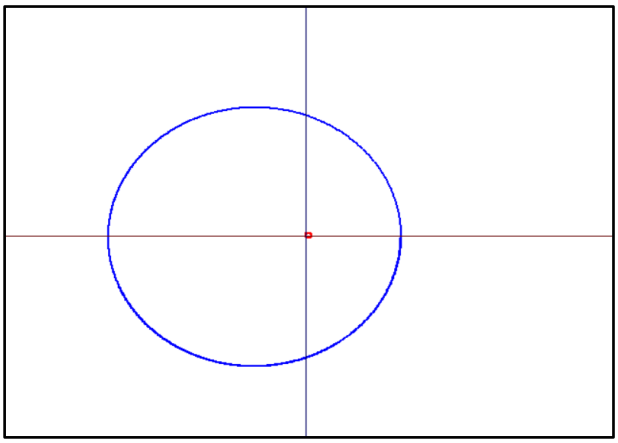

Figure 1: All positions of two bodies over 1000 timesteps. Mass ratio Blue:Red 1:50. Starting positions (500,350) (600,350). Starting velocities (magnitude, direction) (2/50, -pi/2) and (1, pi/2).

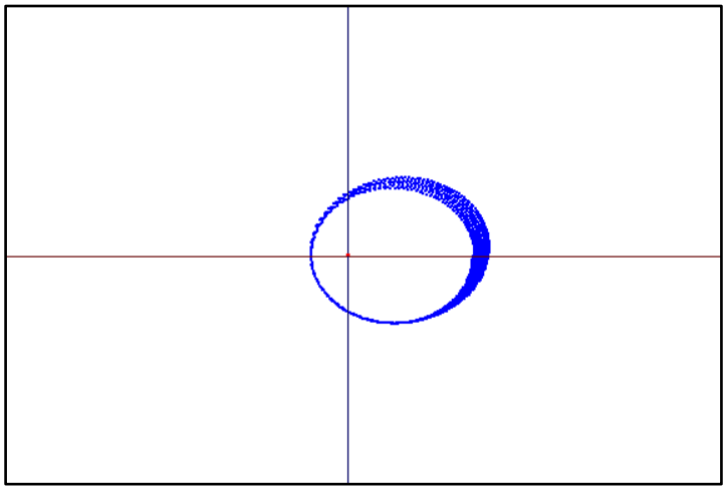

Figure 2: All positions of two bodies over 1000 timesteps. Mass ratio Red:Blue 1:150. Starting positions (500,350) (600,350). Starting velocities (magnitude, direction) (2/150, -pi/2) and (1, pi/2).

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:05/May-2023 Impact Factor- 7.868 www.irjmets.com

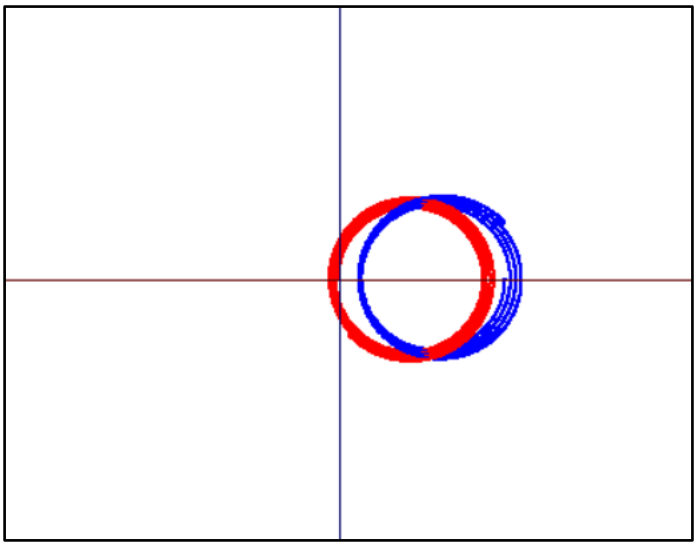

Figure 3: All positions of two bodies over 2000 timesteps. Mass ratio 1:1. Starting positions (500,350) (600,350). Starting velocities (magnitude, direction) (0.5, -pi/2) and (0.5, pi/2).

IV. RESULTS AND ANALYSIS

Visually, looking at the motion of their obit, we expect them to find the particles moving in close to elliptical orbits. Since, in a pair, both particles are orbiting each other, with neither of them the reference point, we expect to see spiral shapes. This can be thought of as the focus of the ellipse changing over time. This is particularly visible in Figure 2 and Figure 3.

Empirically, the validity of the simulation can be tested by computing the potential and kinetic energies of one of the particles throughout its orbit.

The potential energy can be computed from the expression-

$$-Gm_1m_2/r$$

The constants G, m_1 , m_2 are already known. 'r' is the magnitude of the distance between the two particles at that instance.

Kinetic energy is being computed through the expression-

$$m_1v^2/2$$

Here, m_1 is always known and 'v' is the magnitude of the velocity in that instance. Variables marked with subscript 1 denote the mass for which the energies are being calculated. Variables marked with subscript 2 are for the mass the first mass is orbiting. While the simulation can simulate the interactions between an arbitrary number of masses, only the results for a simulation of two masses are being verified in this paper.

It is expected that the kinetic and potential energy should always add up to a constant since after the initialization of the system, no external work is done by it or on it. Due to the elliptical shape of the orbit, it is also expected that the potential energy would drop near when the particle gets closer to the one it is orbiting. To compensate, the kinetic energy must increase. A cyclic tradeoff is expected to be found when the two energies are graphed for a specific orbit.

The expected result is found both in the shape of the obits and the graph of the energies. One thing to note is the slight fluctuation in the sum of kinetic and potential energy. This fluctuation could be an artefact caused by loss of precision in the floating point arithmetic driving this simulation. However the fluctuation is negligible compared to the shift in magnitude of both the kinetic and potential energy.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:05/May-2023 Impact Factor- 7.868 www.irjmets.com

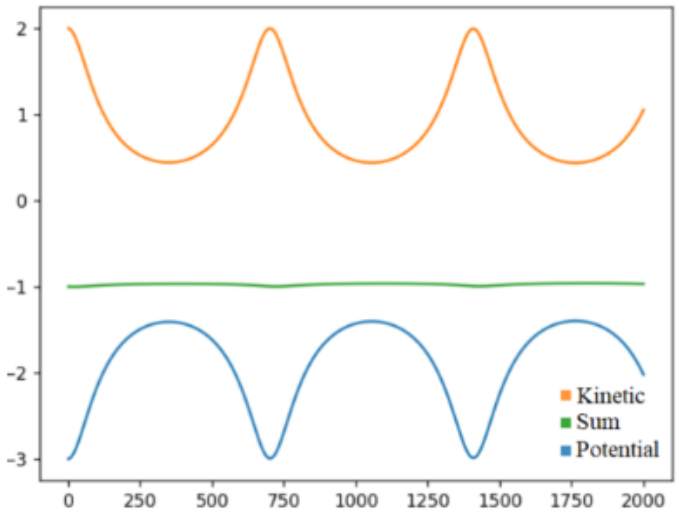

Figure 4: Graph of Potential energy (blue), Kinetic energy (Yellow), and sum of both (Green). For a simulation with 2000 timesteps. Mass ratio 5:1. Starting positions (500,350) (600,350). Starting velocities (magnitude, direction) (2/50, -pi/2) (2,pi/2)

V. CONCLUSION

From this research it can be concluded that this specific method of simulating orbits, characterised by computing forces on particles over discrete time steps, provides valid results which coincide with expected orbital behaviour. This statement is affirmed by analysing the potential and kinetic energy of the system of two particles throughout their orbit.

ACKNOWLEDGEMENTS

I want to express my sincere thanks to Dr. Anant Prakash Pandey for inspiring me to write this research paper. He not only taught me some of the ideas that are utilised in this thesis as my high school physics instructor, but he also went above and beyond to help me write and style this paper using his valuable time.

VI. REFERENCES

[1] Arora, S.L. (2022). New Simplified Physics (10th Edition).
[2] Verma, H.C. (2022). Concepts of Physics.
[3] Irodov, I.E. (1988). Problems in General Physics.
[4] Arora, P. (2022). Computer Science with Python.