Numerical Simulations on the Nature of the Universe

Michio Tanaka

May 8, 2017

1 Abstract

We used gravipy a symbolic manipulation library for python in order to obtain the geodesic equations for the Schwarzchild metric of space-time. From there we numerically solved the resulting geodesic equations to obtain the proper time evolution of the system for various test masses representing the planets Mercury, Venus, Earth, and Mars. We input relevant initial conditions for each planet and plotted the subsequent motion of the planets around a point mass representing a sun.

2 Assumptions

2.1 The only relevant mass is the Sun

To simplify our description of the inner Solar System, we assumed that the planets would not interact with eachother gravitationally. The only determinate of their trajectory would be their interaction with the curvature generated by the point mass at the center of the system. This enabled us to use the Schwarzchild metric to determine the geodesics of the planet's motion through space as opposed to having to solve Einstein Equations for the stress energy tensor resulting from all of the other planets.

2.2 The proper time of the planets is the same as the coordinate time

We assumed that coordinate time was the same as proper time in order to simplify our animation of the motion of the planets. Without this assumption we would have had to interpolate the trajectories of our planets in order to account for an asynchronous time step. This is due to the fact that the coordinate time on one planet may not line up properly with the coordinate time of another planet and thus we would not have data points for that specific time.

2.3 Orbits are in Plane

We assume that the orbits of the plants are in plane with $\theta = \pi/2$. Thus we do not need to set initial values for the derivative of θ

3 Methodology

In addition to the numpy and matplotlib libraries that make up the backbone of most computing in python. We used gravipy, a symbolic computing library that is tailored to computations relevant to General Relativity. It was this library that enabled us to compute the Geodesic equations from the metric that we input.

Another library that we used was the symbolic computing library upon which the gravipy package is built, sympy. We used sympy aside from its implementation in gravipy in order to solve the geodescic for the second derivatives of our coordinates with respect to the proper time. We also used methods within sympy in order to put the geodesic equations into a form that could be input into the ode solver.

Finally we used the odeint solver within scipy.integrate in order to solve the geodesic equations for relevant initial conditions. In order to solve these second oder differential equations we used reduction of order to turn the four second order differential equations into eight first order differential equations. The first four differential equations correspond to the derivatives of the coordinates with respect to proper time and the second four differential equations correspond to the derivatives of the first derivatives of the coordinates. This demanded that we provide eight initial conditions corresponding to the first and second derivatives of the coordinates. We set the initial time arbitrarily to zero, the initial r to the perihelion of the orbit of the planet whose orbit we were calculating,the initial θ to $\pi/2$ under the assumption that all of the planets orbit in plane, and the ϕ to zero arbitrarily. We set the initial time derivative with respect to proper time to 1 under our assumption that the coordinate time and proper time is the same, the initial r derivative to zero as the zero as is the property of the orbit at the perihelion, the ϕ derivative to zero under the assumption of orbit in plane, and finally the ϕ derivative to the velocity of the orbit at perihelion divided by the perihelion radius.

The odeint solver returns arrays with each of the coordinates indexed by the time delta of the proper time, from this we plot the position of each of the planets at different proper times. Under our assumption that the proper time and the coordinate time are the same, we assume that the coordinates of each planet at a given proper time are also their concurrent location in coordinate time.

4 Conclusion

Using just the curvature of space defined by the Schwarzchild metric, we were able to derive approximations of the orbits of the inner planets in the solar system. The orbits that we ended up with closed in on each other and were elliptical as expected by classical calculation. The method that we used to solve the metric can be applied to other metrics including the FLRW metric. So in the future we have the capability to extend our code in order to observe the behavior of particles in other curved space times.