HPLA Final Project
Adam Heaton

Conservation of Energy for an N-Body Simulation

Purpose:

An N-Body simulation involves a number of physical bodies mechanically interacting with each other, whether it molecules bouncing off each other in a chamber, or numerous planets being pushed and pulled by the gravitational forces felt by every other planet within the system. For my simulation I decided to examine a physical system where bodies in space of varying masses would interact with each other through gravitational forces. There are several approaches or methods to calculating these interactions over time, and the purpose of this project is to compare the effectiveness of those approaches.

Background:

There are many physical properties exhibited by physical bodies in space, perhaps one of the most useful properties could be Mechanical Energy. Mechanical Energy is the sum of Potential Energy and Kinetic Energy, which are defined as:

$$KE = \frac{1}{2}m(\vec{v} \cdot \vec{v})$$

$$PE = -\frac{G \cdot m_1 \cdot m_2}{|\vec{r}_2 - \vec{r}_1|}$$

And the Total Kinetic Energy of the system is defined as:

$$KE_i = \frac{1}{2}\sum_i^N m(\vec{v}_i \cdot \vec{v}_i)$$

And the Total Potential Energy of the system is every unique potential between two masses.

The law of the Conservation of Energy states that the total energy of an isolated system will remain constant over time. By choosing to ignore collision and not introducing any outside forces into my system, the total energy of all bodies, which will only be mechanical, should remain constant. The methods of calculating the interactions between bodies are not perfectly accurate however, and because of this there will be some loss in Mechanical Energy over time. This is why I will be using Mechanical Energy as a benchmark to compare the effectiveness of different approaches to modeling the interactions of my simulation.

Approaches for Calculating Forces:
Given a number of physical bodies floating in space, I will assume the only force that can act on them is Gravity. The net force acting on any one body is the sum of all the individual forces acting on that body. So in order to calculate to calculate the net force acting on a body I will need to sum up all the gravitational forces felt between that body and every other body.

Gravitational Force between two bodies:

$$\vec{F} = -G \cdot \frac{m_1 \cdot m_2 \cdot (\vec{r}_2 - \vec{r}_1)}{|\vec{r}_2 - \vec{r}_1|^3}$$

Net Force on body (i):

$$\vec{F}_i = \sum_j^N \vec{F}_j$$

This can be called the **Direct Solve** method of calculating the net force acting on a body. This method has very little error but it is on the order of O(n^2) in time complexity. This is especially costly considering that if we have a large number of bodies and/or a large number of time step iterations, there will have to be O(n^2) calculation within each time step. Another method of calculating the net force acting on each body involves is the **Tree Method**.

This method involves placing the bodies into the leaves of either a Quad-Tree (for 2D) or a Octo-Tree (for 3D). The leaves of the tree contain the position and mass of individual bodies and the nodes contain the Center of Mass, and CoM's position for every child of that node. The children of each node represent one sub quadrant (in the case of a Quad-Tree) of the the nodes super quadrant within the entire spatial domain of the problem. With this organization, a body on one end of the spatial domain can calculate the net force acting on itself by looking at the positions and mass of other individual bodies (represented by leaves) close by, and every other cluster of bodies (represented by nodes) further away. This reduces the time complexity to O(n log n) which can make a huge difference in larger simulations.

Approaches for Calculating Positions and Velocities:
If we have a net force for each body we can use that force to calculate the acceleration of a body at each time step from Newton's laws of motion:

$$F = m \cdot a$$

$$a = \frac{F}{m}$$

With this acceleration we can integrate to find new positions and velocities through various methods.

Verlet:

The Verlet Integrator is an addition of two taylor series expansions of positions, one of them a time step ahead, and the other a time step back. the resulting addition gives the following equation used in calculating position:

$$\mathbf{r}(t+\Delta t) = 2\mathbf{r}(t) - \mathbf{r}(t-\Delta t) + \mathbf{a}(t)\Delta t^2$$

And velocity of the previous time step is given by:

$$\mathbf{v}(t) = \frac{\mathbf{r}(t+\Delta t) - \mathbf{r}(t-\Delta t)}{2\Delta t}.$$

Fourth order Runge-Kutta:

The Runge-Kutta method takes an initial value problem:

$$\begin{aligned} &y' = f(t,y) \\ &y(t_0) = y_0 \end{aligned}$$

And gives an approximation of an update to **y** with:

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + \tfrac{h}{2}, y_n + k_1\tfrac{h}{2}) \\ k_2 &= f(t_n + \tfrac{h}{2}, y_n + k_2\tfrac{h}{2}) \\ k_2 &= f(t_n + \tfrac{h}{2}, y_n + k_3 h) \end{aligned}$$

In this case we need to split the second order differential equation into two first order equations:

$$\begin{aligned} \dot{\vec{r}} &= \vec{v} \\ \dot{\vec{v}} &= \vec{a} \end{aligned}$$

Then we can determine the position and velocity of a body at the next time step using:

$$\vec{v}_{i+1} = \vec{v}_i + \frac{h}{6}\left(\vec{k}_{1_{v_{i+1}}} + 2\vec{k}_{2_{v_{i+1}}} + 2\vec{k}_{3_{v_{i+1}}} + \vec{k}_{4_{v_{i+1}}}\right)$$

$$\vec{r}_{i+1} = \vec{r}_i + \frac{h}{6}\left(\vec{k}_{1_{r_{i+1}}} + 2\vec{k}_{2_{r_{i+1}}} + 2\vec{k}_{3_{r_{i+1}}} + \vec{k}_{4_{r_{i+1}}}\right)$$

Which are calculated from accelerations and velocities at varying times steps:

$$\begin{array}{ll}
\vec{k}_{1_{r_{i+1}}} = \vec{v}_i & \vec{k}_{1_{v_{i+1}}} = \vec{a}\left(\vec{r}_i\right) \\
\vec{k}_{2_{r_{i+1}}} = \vec{v}_i \vec{k}_{1_{v_{i+1}}} \frac{h}{2} & \vec{k}_{2_{v_{i+1}}} = \vec{a}\left(\vec{r}_i + \vec{k}_{1_{r_{i+1}}} \frac{h}{2}\right) \\
\vec{k}_{3_{r_{i+1}}} = \vec{v}_i \vec{k}_{2_{v_{i+1}}} \frac{h}{2} & \vec{k}_{3_{v_{i+1}}} = \vec{a}\left(\vec{r}_i + \vec{k}_{2_{r_{i+1}}} \frac{h}{2}\right) \\
\vec{k}_{4_{r_{i+1}}} = \vec{v}_i \vec{k}_{3_{v_{i+1}}} h & \vec{k}_{4_{v_{i+1}}} = \vec{a}\left(\vec{r}_i + \vec{k}_{3_{r_{i+1}}} h\right)
\end{array}$$

Setup and Assumptions:

So in the case of this project, four distinct methods for evaluating Total Mechanical Energy of the system over time will be simulated.

Methods:

1. Direct Solve with Verlet
2. Direct Solve with Runge-Kutta
3. Tree method with Verlet
4. Tree method with Runge-Kutta

I will be examining KE, PE and Total Energy as they evolve over the course of the simulation with different initial conditions and different problem sizes.

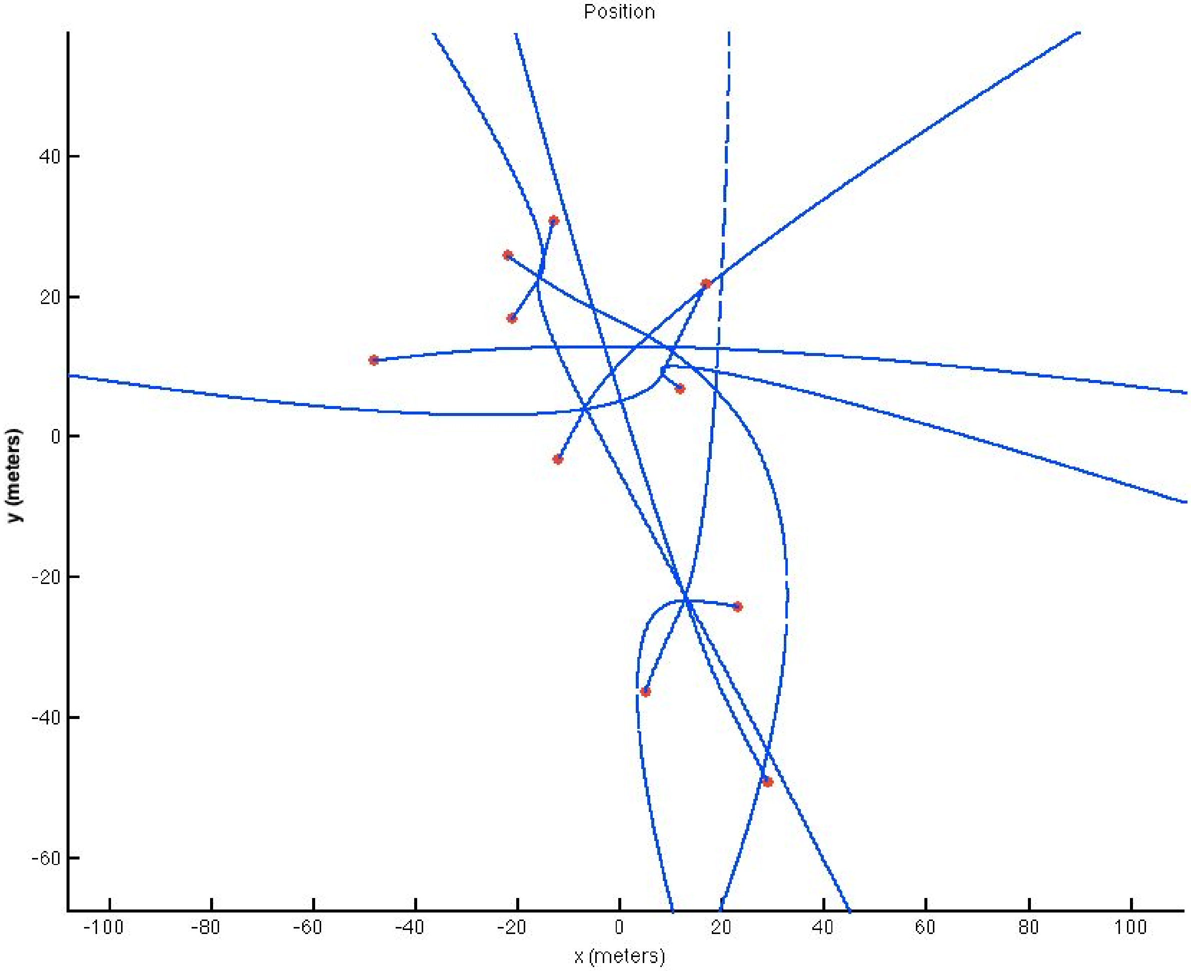
Assumptions:

1. Bodies are point masses - all of their mass is concentrated at their exact coordinates
2. Bodies cannot collide
3. There are no boundaries
4. Gravity is the only force acting on the bodies
5. There are no external forces acting on the system

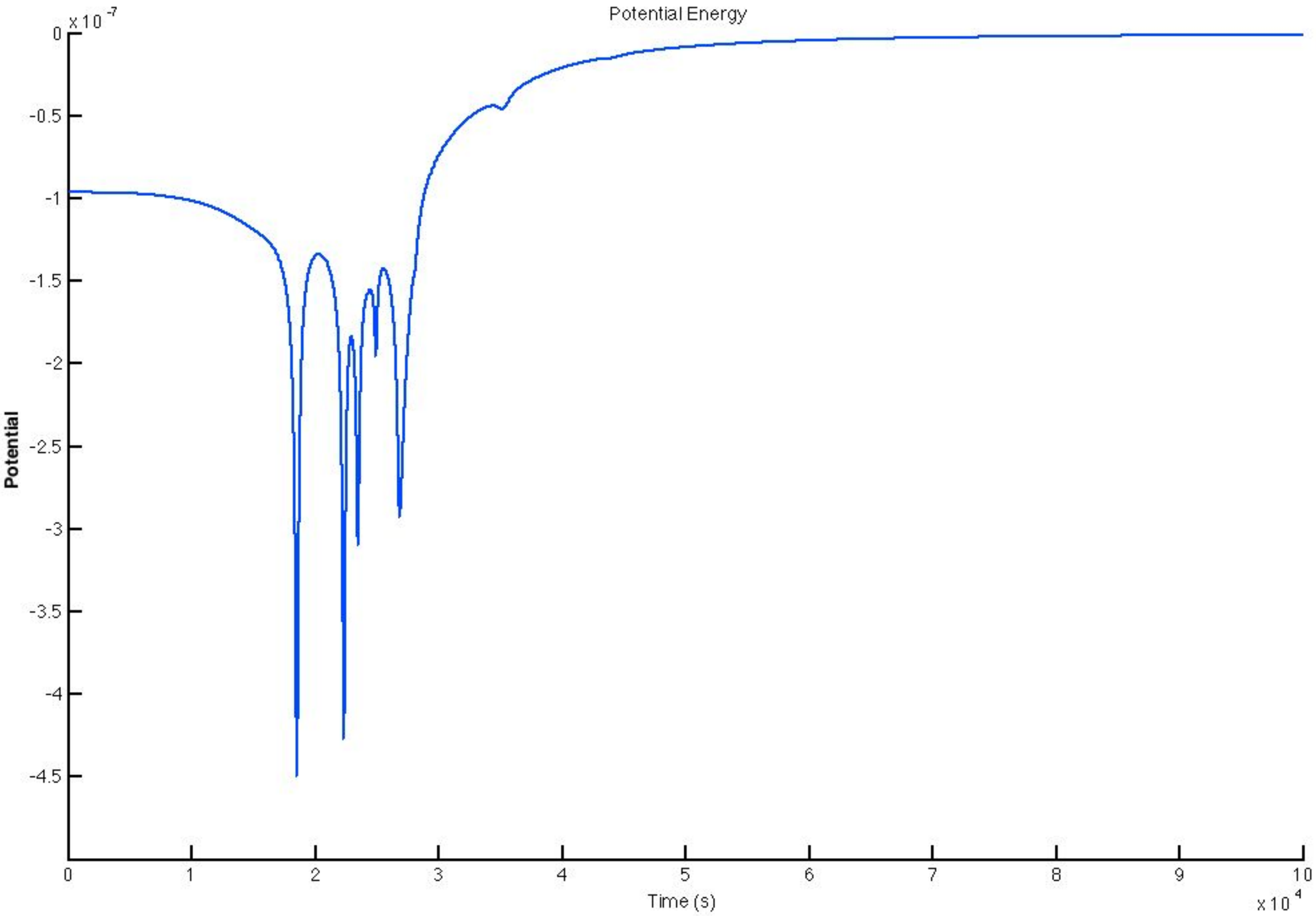
Results:

Experiment 1:

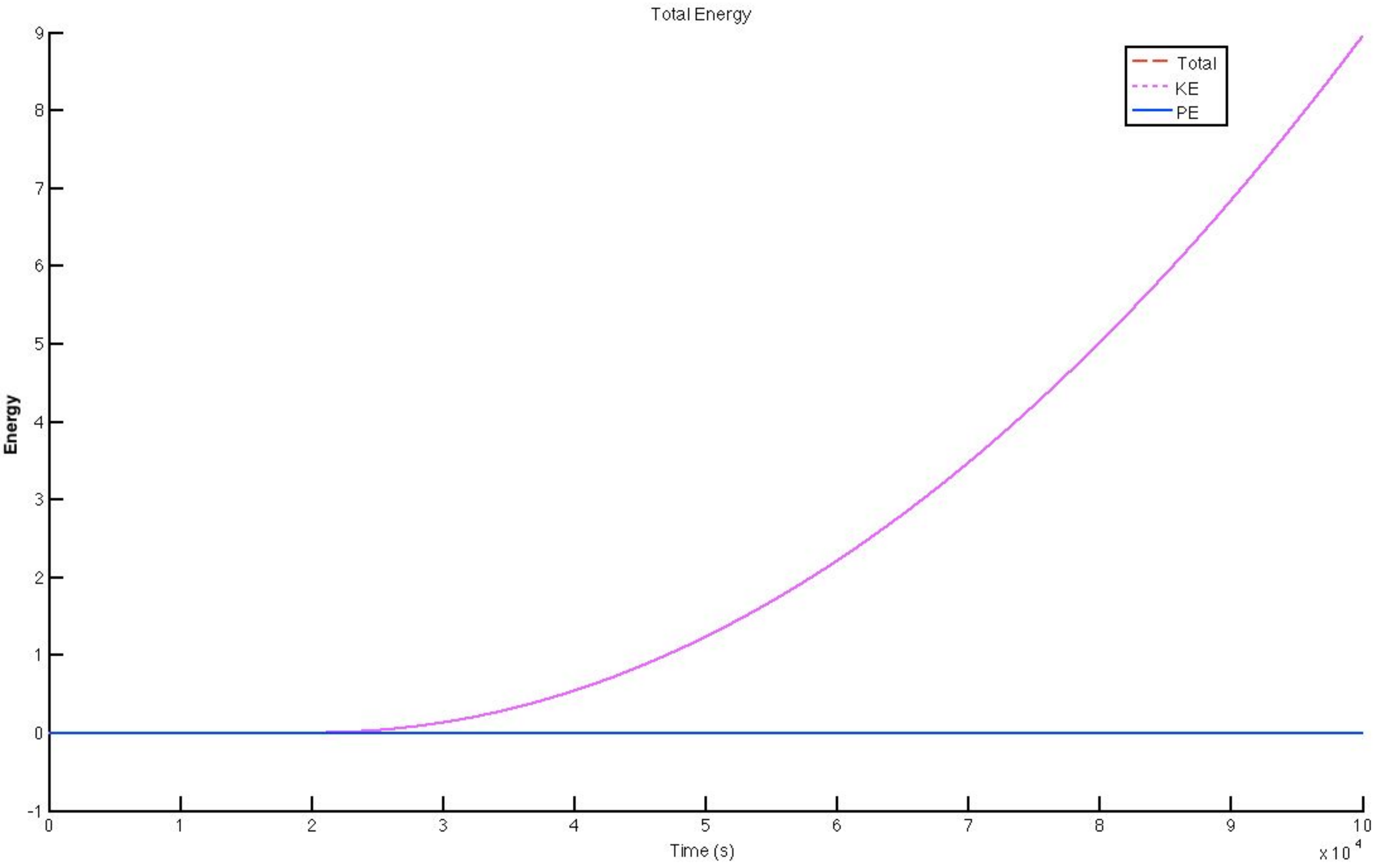
Number of Bodies	- 10
Initial Velocities	- 0 (m/s)
Initial Positions	- (0 to 100) meters before shifting CoM
Number of steps	- 100000
Delta T	- 0.5 seconds



This figure shows the the paths taken by each individual body as the simulation progressed. The red dots represent a starting point for each body. This was mostly used for a sanity check to see if the interactions between bodies more or less “made sense”. The bodies do seem to interact nicely when coming it comes to close encounters, however; what may be interesting is how every body eventually shoots off on its own path.



This figure shows the Total Potential Energy of the system as a function of time passed. As you can see all of the bodies start of with a small amount of potential, until many of them start moving towards a central location, or another individual body. When this happens the total potential of the system decreases into what can be called a well. This is when most of the bodies are very close to a source of gravitational attraction. Since the bodies have gained Kinetic Energy in the process of entering the well, they will shoot out the other side and this is exactly what all the bodies do. They gain kinetic energy as they encounter other objects, and that kinetic energy carries them out of the influence of those objects where potential reaches a plateau.

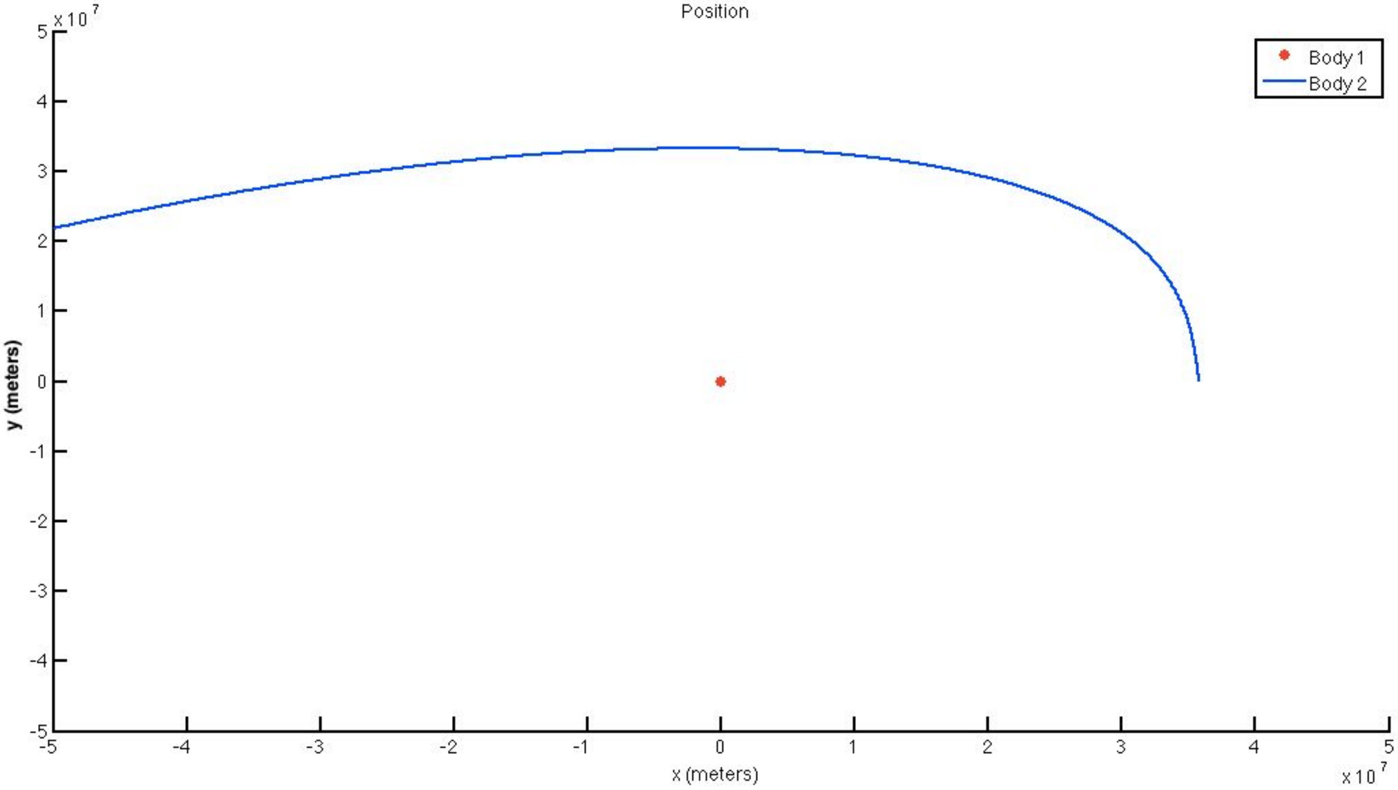


This figure shows Total Energy, KE, and PE. Right away it becomes noticeable that after the bodies escape their potential wells and shoot off into space their Kinetic Energy increase exponentially and with it Total Energy. This is not the expected behaviour as what should be occurring after the bodies start to shoot off into space is the following: their respective velocities should start to decrease as they are attracted gravitationally back towards the approximate center of mass the system. This decrease in velocity should translate to a decrease in KE.

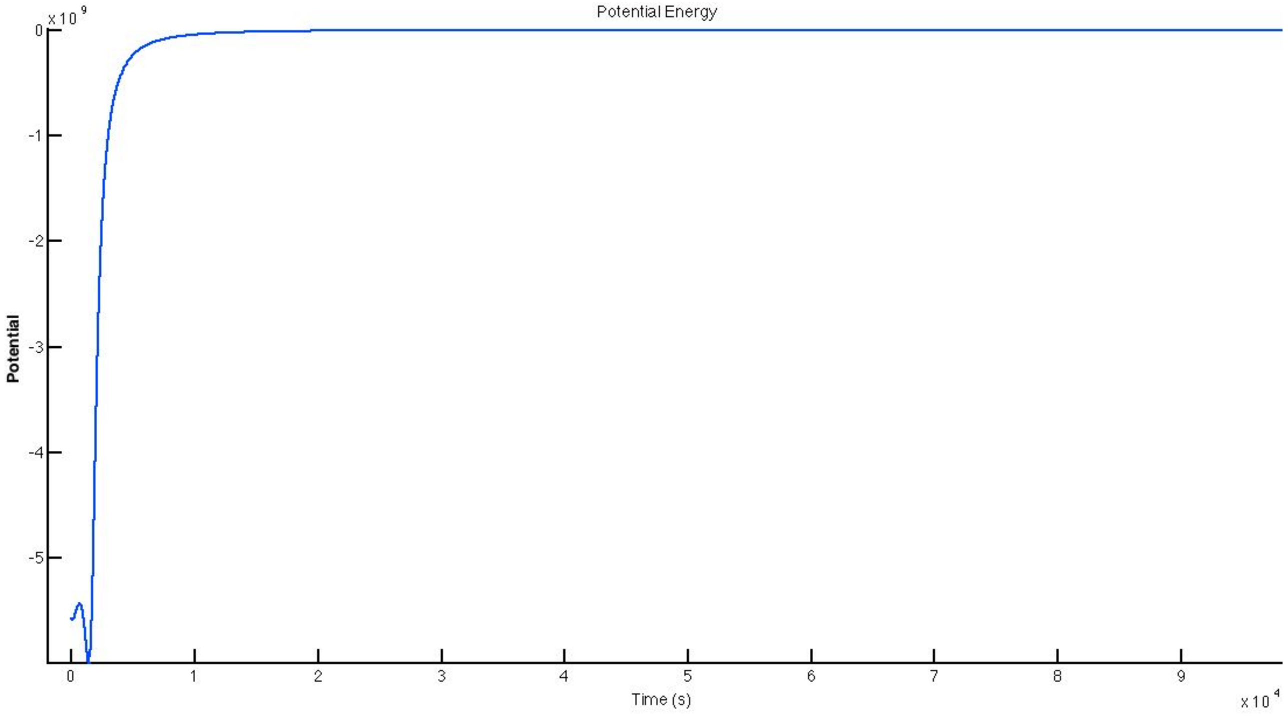
Experiment 2:

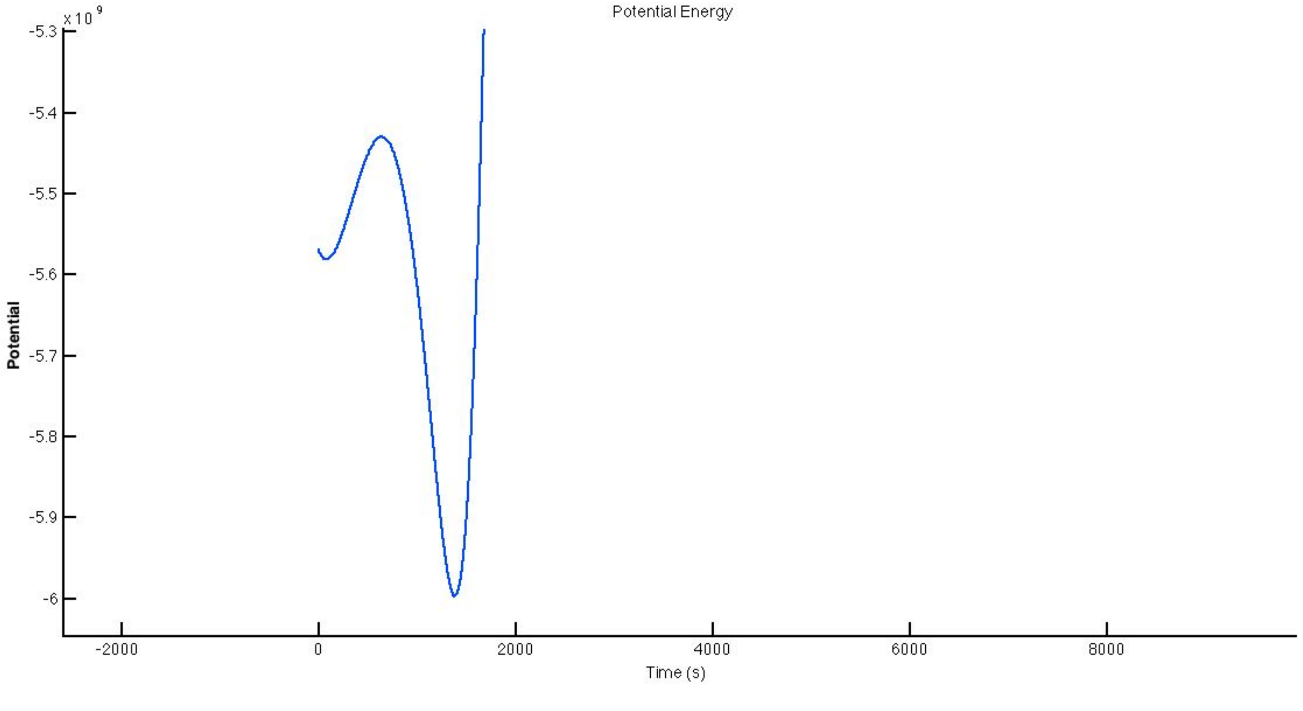
Number of Bodies	- 2
Initial Velocities	- ~15,500 (m/s)
Initial Positions	- (0,0) and (35784000,0) meters
Number of steps	- 100000
Delta T	- 0.5 seconds
Masses	- (0,0) = ~Earth's Mass kg // (35784000,0) = 1000 kg

This experiment was one of many designed to simplify the simulation in order eliminate uncertainties that could have arisen with a larger number of bodies and randomly generated positions and masses. By eliminating some uncertainties my hope was that I would be able to find the reason for the exponential increase in KE and Total Energy as seen in the above experiment. This did not turn out to be the case.

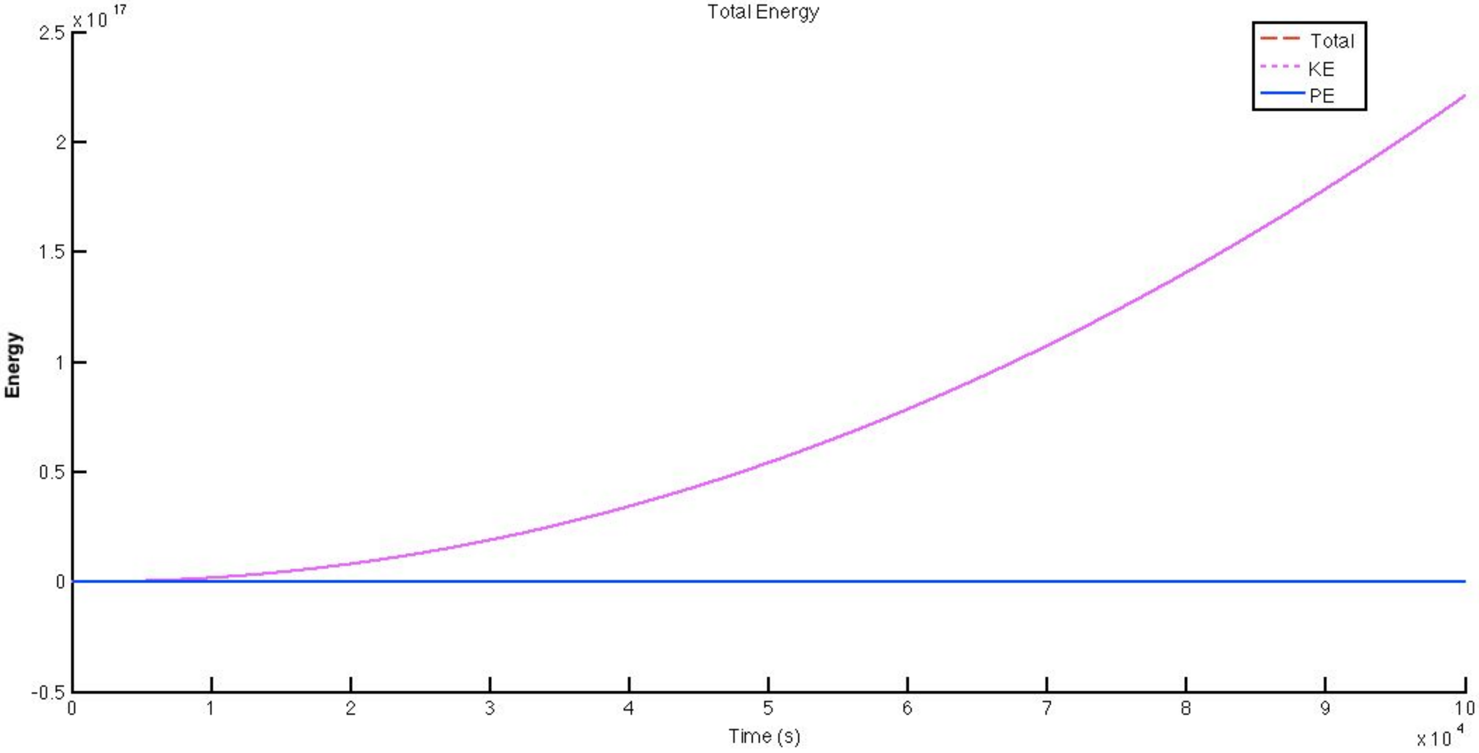


Here we see a very massive body placed at the origin which moves a negligible amount, as expected. The blue mass starts far away from the red mass with a significant initial velocity in the Y direction. As the blue mass starts to travel in the Y direction it is pulled gravitationally by the red mass as to alter it's course and shoot by the red mass.





These images represent the total PE of the 2 mass system but since the the red mass feels a negligible amount of pull from the blue mass these graphs can be interpreted as the blue masses potential over time. As shown, the mass starts out by trying to leave the potential well of the red mass, attributed by its initial velocity, but the blue mass is quickly pulled in the massive red body where by being in its influence, gains a huge amount velocity until it is spat out of the potential well and shot off into space.



From this figure we see that KE, and with it Total Energy, increase exponentially. Whereas what should be expected is decrease in velocity as the blue mass leaves the red's influence and starts to be attracted towards coming back. And with this decrease in velocity, a bigger decrease in KE. This is not the case and because of this error it becomes increasingly difficult to compare Total Energy of a system across different methods.

Conclusion:

After performing countless experiments, and varying initial conditions and simulation parameters, I was not able to observe a conservation of energy (constant Total Energy) in any of my simulations. Due to this unforeseen blockage I was not able to implement methods 2, 3, and 4 and compare the conservation of energy across methods. Had this problem not arisen I would have implemented and tested these methods, as outlined in earlier sections. Given that energy was mostly conserved in the testing of the different methods. I could have looked at a linear best fit of Total Energy for each method, and compared those best fits to see which one had the least slope, i.e. conserved energy the best.