A Deep Dive Into the Restricted Three-Body Problem

AM 205 Final Project

Shaan Desai and Graham Lustiber

December 14, 2016

Contents

1 Purpose

We explore various properties of the restricted three-body problem as introduced in assignment 3. The set-up is as follows: the Earth and the moon lie in a 2-dimensional plane at positions $(0, 0)$ and $(0, 1)$ respectively, with a mass ratio between the moon and Earth of $\mu = 0.01$. By assuming a co-rotating frame, the Earth and moon's orbits about each other can be ignored. We focus on the motion of a third object of negligible mass starting at various positions in the plane with some initial velocity. The state of this object is represented by (x, y, u, v), with x and y being its position and u and v being its velocity in the x and y directions.

2 Methodology

1. Surface of Section: The surface of section is a way of representing an n dimensional trajectory in an $n - 1$ dimensional space, thus reducing the number of free parameters. In order to precisely compute and plot the surface of section for the given problem we test three different methods:

    a) Newton's Method: fast but often converges to wrong points unless integration routine is given very small time steps

    b) Secant Method: slower but convergence isn't very precise – only depends on signs and not magnitudes

    c) Brent Method: converges to the points accurately but slightly slower than newtons method for the same time step.

2. Lagrange Points: These are points(initial conditions) that are at equilibrium between the Earth and moon exerting gravitational forces on them. An optimization approach is used to determine the lagrange points and the particular jacobi constants at which they occur. Furthermore, a study of their

stability is also undertaken by assessing perturbations of these points.

3. Find and Classify Periodic Critical Points

   Another set of points of interest are elliptic stable points. These are points that lead to elliptic oscillations in the x-y space and thus lead to singular points in the x-u surface of section space. In order to determine these points a root finding method was implemented and then an eigenvalue problem was used to assess stability of the given points.

3 Preliminary Equations

The surface of section is a way of representing a trajectory from (x,y) space in (x,u) space. In order to make this change we need to reduce the number of free parameters available to us and a classical way to do this is to look at trajectories that cross $y = 0$ for a particular jacobi constant (since a majority of trajectories cross this axis at some point). Where the hamiltonian is:

$$J(x, y, u, v) = (x - \mu)^2 + y^2 + \frac{2(1-\mu)}{\sqrt{x^2+y^2}} + \frac{2\mu}{\sqrt{(x-1)^2+y^2}} - u^2 - v^2 \tag{1}$$

We can rearrange this equation to solve for the surface of section by fixing the Jacobi constant since energy is conserved.

$$C = (x - \mu)^2 + y^2 + \frac{2(1-\mu)}{\sqrt{x^2+y^2}} + \frac{2\mu}{\sqrt{(x-1)^2+y^2}} - u^2 - v^2 \tag{2}$$

Which can be recast into:

$$v^2 = (x - \mu)^2 + y^2 + \frac{2(1-\mu)}{\sqrt{x^2+y^2}} + \frac{2\mu}{\sqrt{(x-1)^2+y^2}} - u^2 - C \tag{3}$$

We want to choose a plane through which a majority of the trajectories pass through; as a consequence, we can pick $y = 0$ in our coordinate system since most of the

trajectories go through this point. This also adds an extra constraint to equation (3) since we can make one of our initial conditions $y = 0$ since we know that this is a trajectory that would have passed through the x-axis anyway. This then simplifies equation (3) to:

$$v^2 = (x-\mu)^2 + \frac{2(1-\mu)}{|x|} + \frac{2\mu}{|(x-1)|} - u^2 - C \tag{4}$$

The above equation tells us that we need to determine x, u, and v that will ensure the above equation is satisfied. We do know that v has to be real which imposes an extra constraint on the values of x and u. Thus:

$$v = +\sqrt{(x-\mu)^2 + \frac{2(1-\mu)}{|x|} + \frac{2\mu}{|(x-1)|} - u^2 - C} \tag{5}$$

Since v is chosen to be positive and needs to be real, as only real velocities are allowed, the value within the square root has to be positive, which implies:

$$(x-\mu)^2 + \frac{2(1-\mu)}{|x|} + \frac{2\mu}{|(x-1)|} - u^2 - C \geq 0 \tag{6}$$

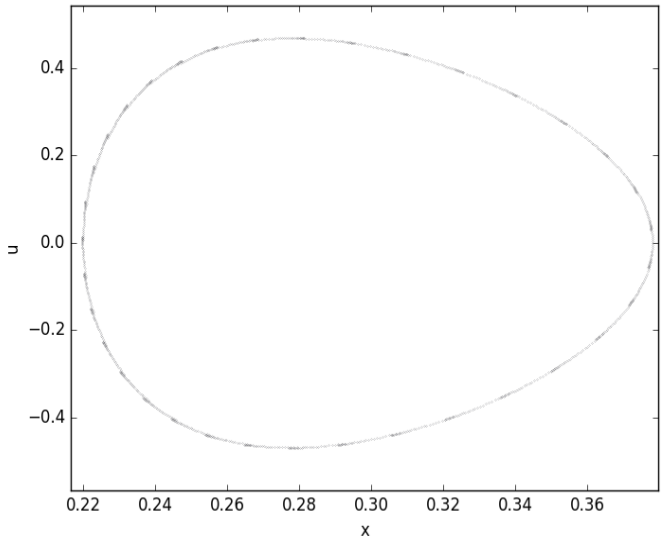
$$(x-\mu)^2 + \frac{2(1-\mu)}{|x|} + \frac{2\mu}{|(x-1)|} - C \geq u^2 \tag{7}$$

Which then gives us the bound of u given a range of different x's as:

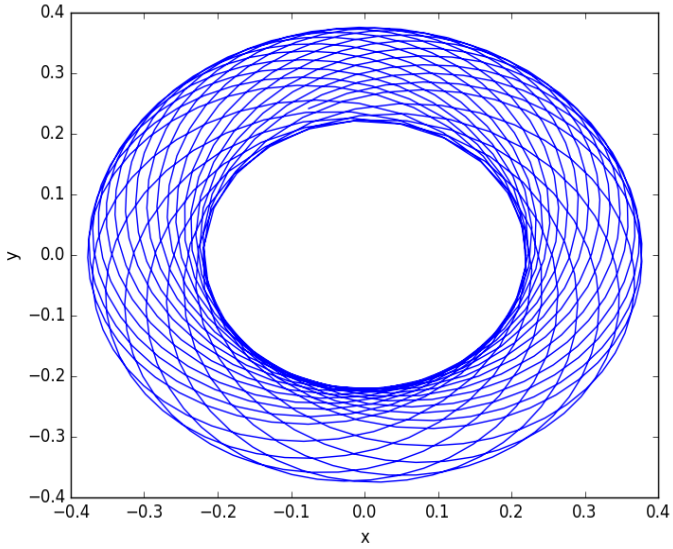
$$-\sqrt{(x-\mu)^2 + \frac{2(1-\mu)}{|x|} + \frac{2\mu}{|(x-1)|} - C} \leq u \leq \sqrt{(x-\mu)^2 + \frac{2(1-\mu)}{|x|} + \frac{2\mu}{|(x-1)|} - C}$$

Given these initial conditions, we are able to plot a range of different trajectories for a specified Jacobi constant in the x-y space. One of the interesting features about using this is that we can look for when we cross the $y = 0$ axis and note the x and u values for a particular trajectory. We can then make a plot of x against u for that specific

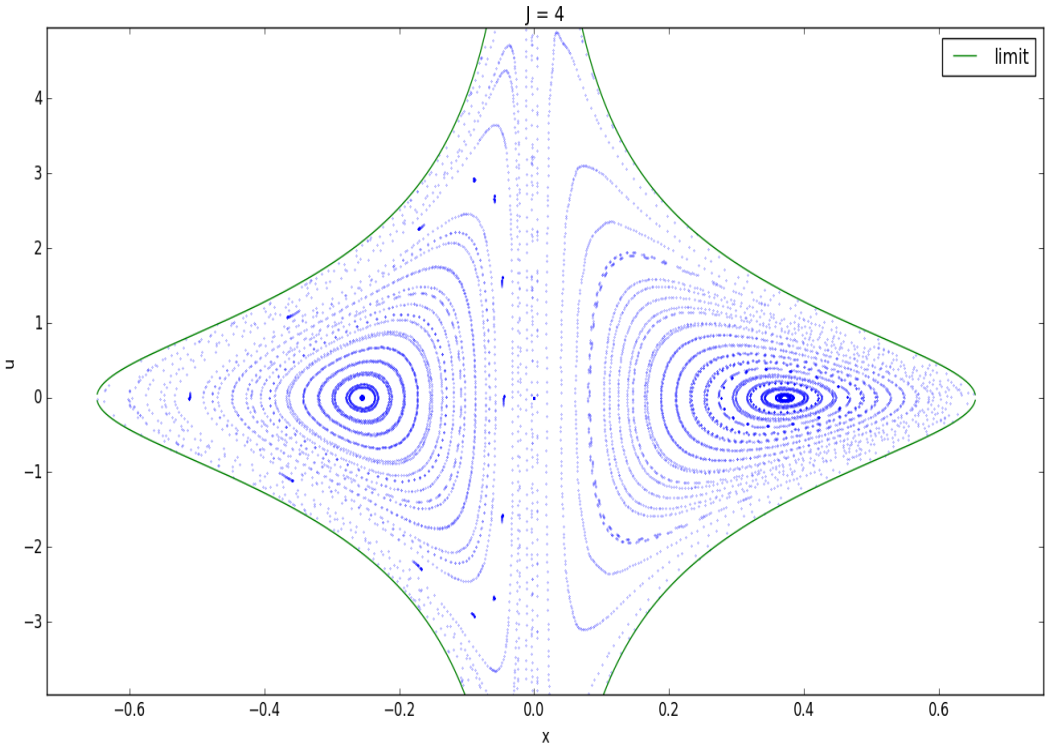
trajectory and then repeat the process for a range of different trajectories and overlay these plots, giving, for example:



The x,u space above illustrates the trajectory with a Jacobi constant of 4 and $(x, y, u, v) = (0.22, 0, 0, 2.25)$ for initial conditions which looks like a torus in x,y space:



In this case we can see that whenever we cross the x-axis and v has positive velocity, we can mark the x and u values. In the case above we took 1000 crossings. Using a range of different initial conditions that fall within our bounds, we can create such mappings to give us a phase space diagram:



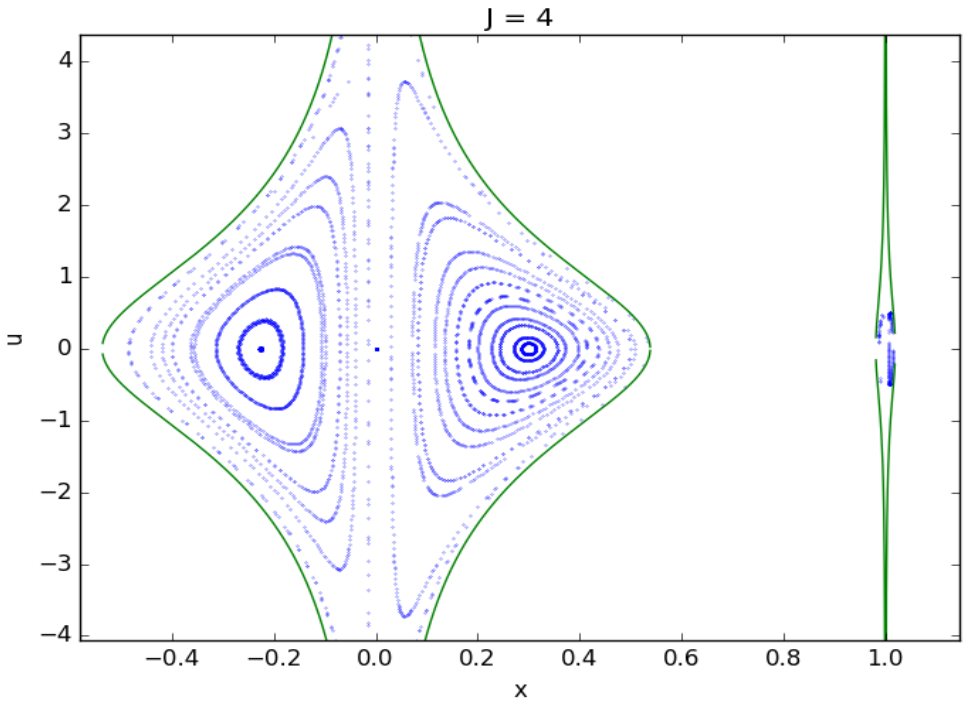
The above diagram illustrates a range of different initial conditions and their quasiperiodic curves. Essentially, if a trajectory starts off at one of the given x and u values, it eventually will return to that set after a number of iterations. The figures above were calculated for 200 crossings of the x-axis.

4 Root Finding Methods

In order to determine the precise x and u values at every crossing for a particular trajectory, we needed to find roots. We use 3 different approaches and evaluate them for accuracy and efficiency.

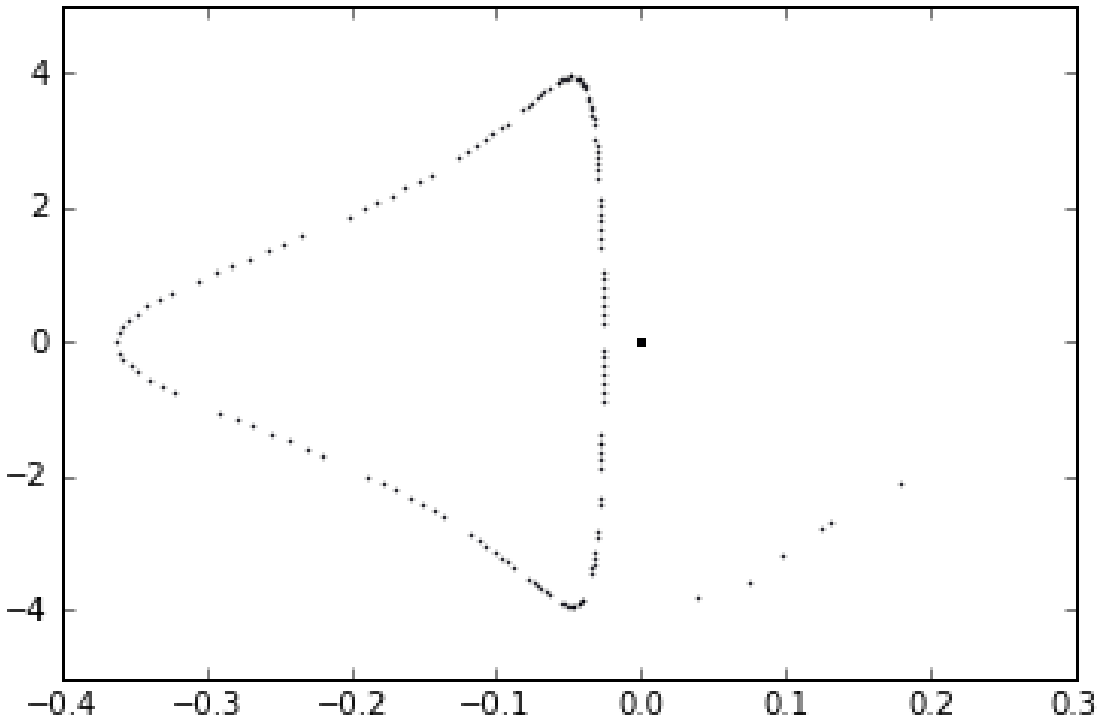
4.1 Newton's Method

We simply kept integrating in $dt = 0.01$ time steps up until we found a crossing. We then used Newton's method to determine the time step needed to ensure that $|y| < 10^{-10}$, saved the values of the crossing points, and plotted them. The graph looks like this:



Producing this took 155.873889 seconds. However, one of the major issues we faced

with using this method was that it led to some inaccuracies. For example, the root method was taking large time steps and converging to points off of the quasiperiodic contour. An example here shows this:

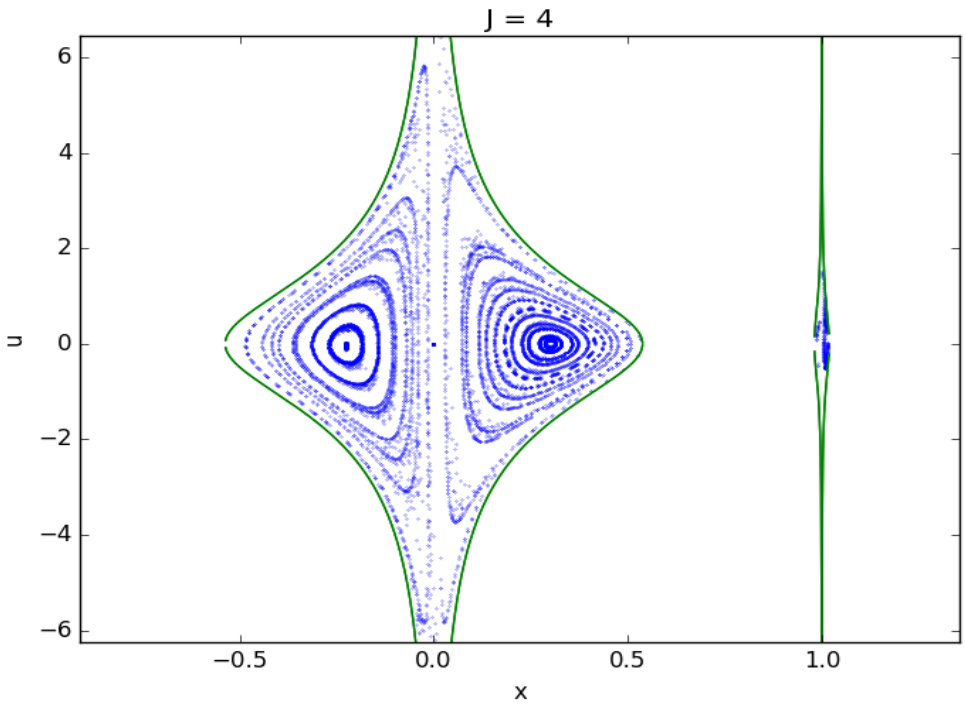


When smaller dt values were taken such that finer integration time steps took place, fewer errors of this nature occurred. However, in reducing the dt value, the algorithm took much longer to run and didn't guarantee that the obtained roots would be in the expected region.

4.2 Secant Method

Using the same initial conditions and root values, we also used the secant method by looking at the magnitude of y between the dt time step up until the difference in time

steps was less than 10^{-15}, as early figures showed many stray points off of the contours we were expecting. The graph generated looks as follows:

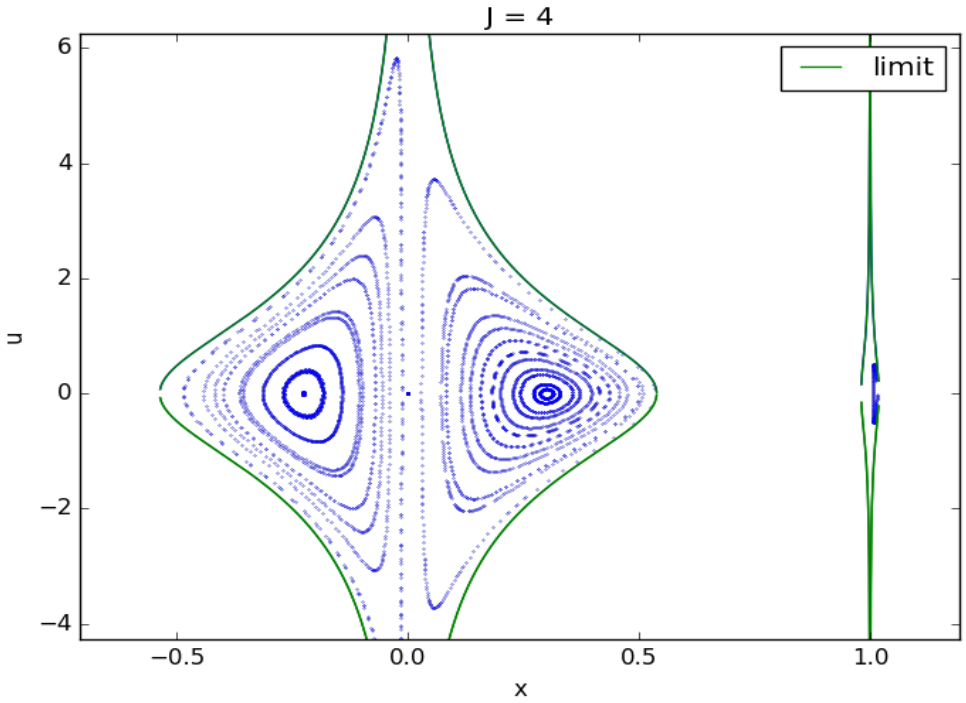


Producing this took 463.80928 seconds. Clearly this method is more problematic in that although we are guaranteed that the points don't converge to roots outside of the expected region, machine precision differences weren't able to allow for exact crossings, as only signs of the y values were used. We can see in the graph above how there are many stray points off of the contours.

4.3 Brent's Method

This method promises quadratic convergence as well as an ability to find a root between given bounds for y. The graph generated looks very similar to that of

Newton's method:



Producing this took 195.781139 seconds, which is faster than the secant method but slightly slower than Newton's method. However, this method produces accurate contours and to our knowledge prevents convergence to roots outside of the allowed region, since Brent's method stays within the inputted bounds.

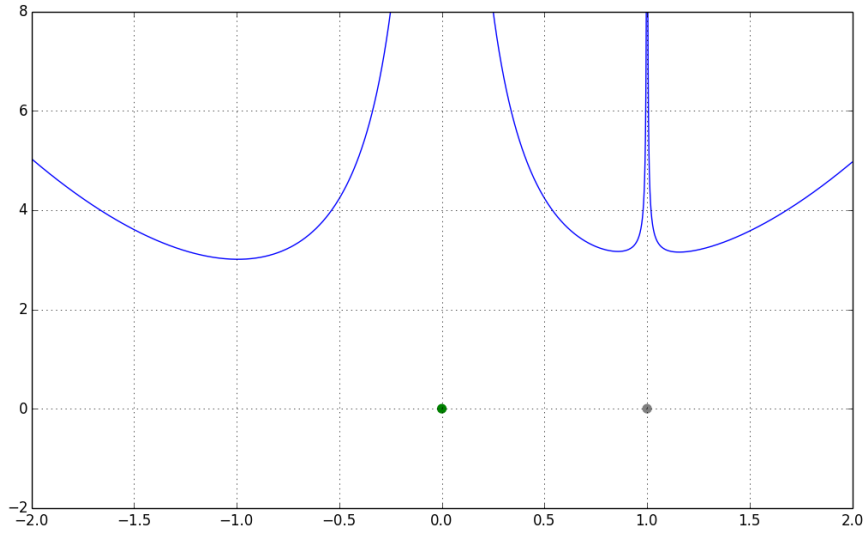
5 Lagrange Points

We are also interested in Lagrange points: points that are at equilibrium between the large bodies exerting gravitational forces on them. Because the Earth and moon are fixed in our co-rotating frame, the only points with gravitational pull in the y-direction will be along the x-axis. Thus, we can find these point analytically by setting

$y = u = v = 0$, giving

$$J(x) = (x - \mu)^2 + \frac{2(1-\mu)}{\sqrt{x^2}} + \frac{2\mu}{\sqrt{(x-1)^2}} \tag{8}$$

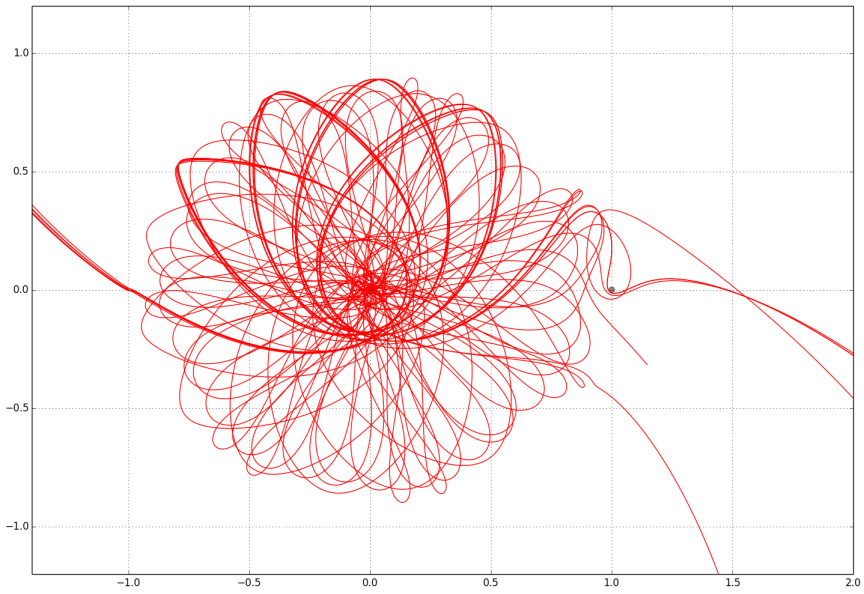
Plotting this equation yields



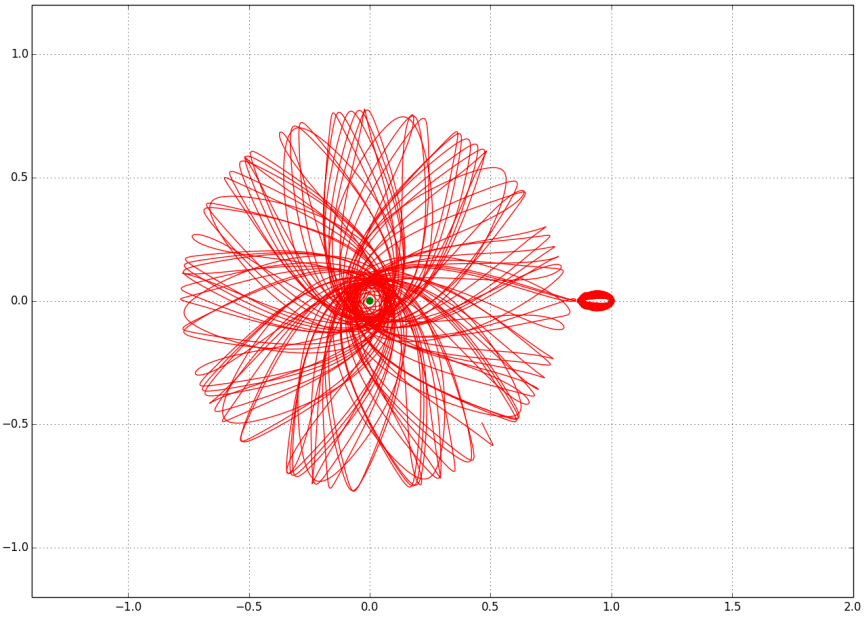
and the three minima can be found analytically by solving

$$\frac{\partial J}{\partial x} = 2x - 2\mu - \frac{2x(1-\mu)}{(x^2)^{3/2}} - \frac{2\mu(x-1)}{((x-1)^2)^{3/2}} \tag{9}$$

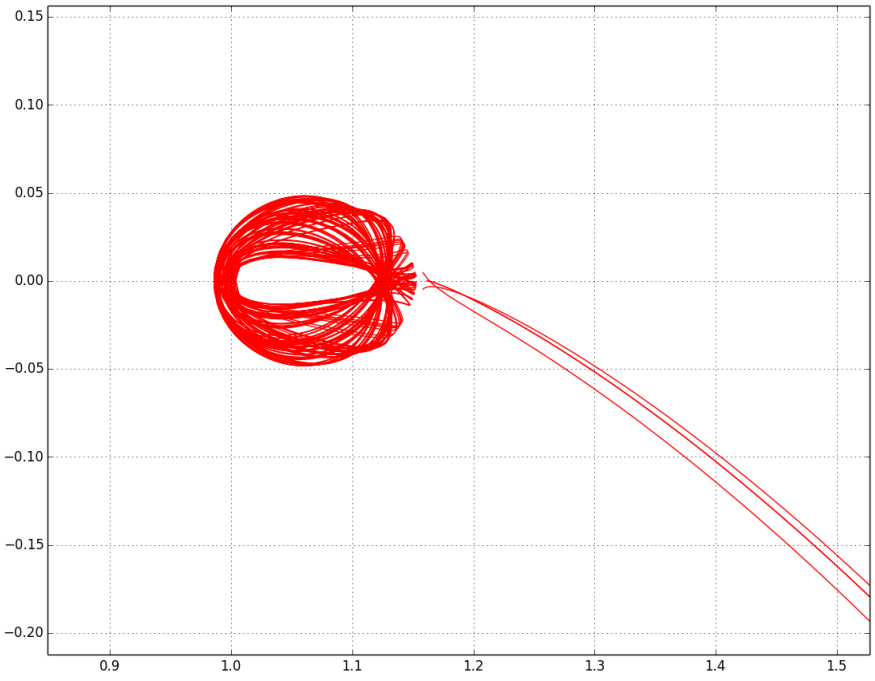
to produce points $L_1 = -0.994167$, $L_2 = 0.858079$, and $L_3 = 1.156765$. We do not consider possible points L_4 and L_5 that form adjacent equilateral triangles with the two large masses because we have a fixed system. Were we examining the orbit of the Earth, or any planet, around the sun, we would have these two other Lagrange points. Regardless, any small object placed at these Lagrange points will remain stationary. By considering a small ϵ-neighborhood about these points, we can examine the effect of perturbing the equilibrium. Letting $\epsilon = 0.005$, at L_1 we see:



at L_2:



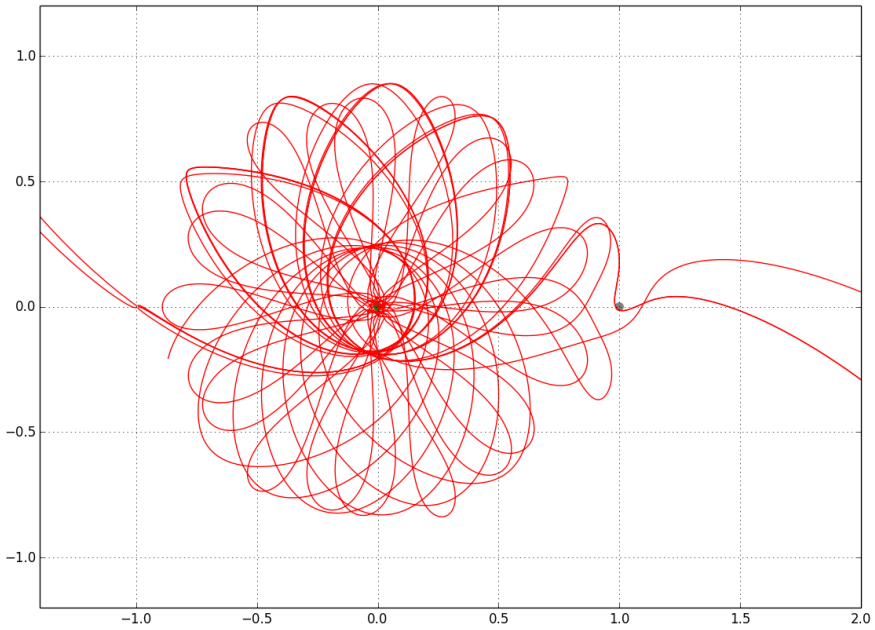
and at L_3:

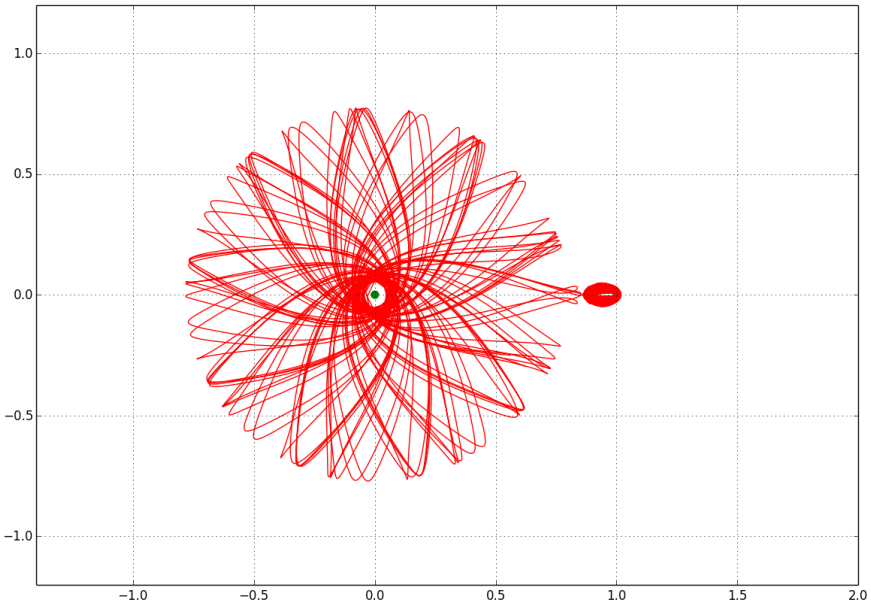


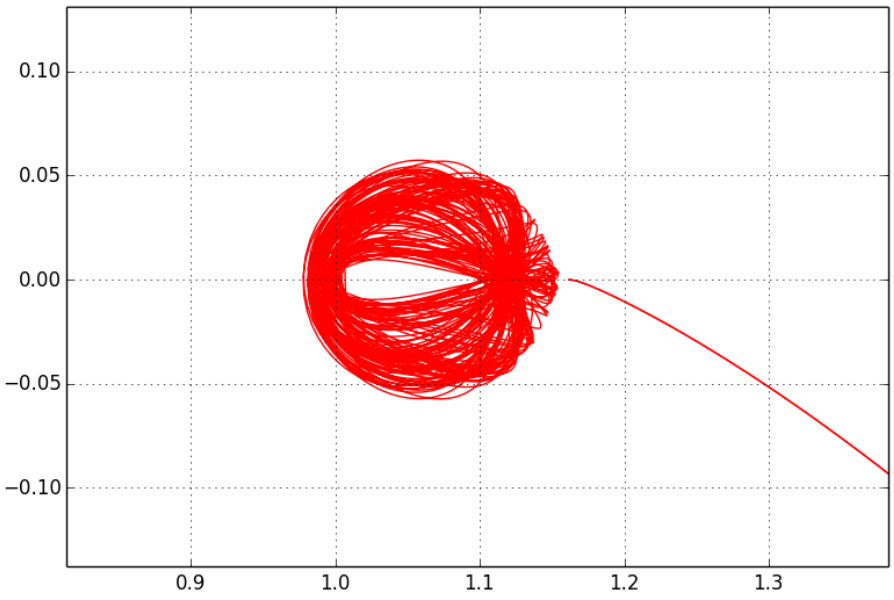
As expected, moving to the left of L_1 or right of L_3 causes the object to spin off. Placing the object Earth-side of L_1 or L_2 results in periodic, ellipsoid orbits about Earth with occasional spin-offs if an apogee is too close to the moon. Moon-side placement of L_3 similarly results in ellipsoid orbits between L_3 and the moon, but without spinoffs. It should be noted, however, that these slight shifts move onto new Jacobi constant curves. If we wish to remain at the same Jacobi constants, $3.009997, 3.167641, 3.154320$ for L_1, L_2, L_3, we cannot assume that u and v are both 0 off of the Lagrange point. We recalculate them to get new initial conditions for the trajectory integration, by solving

$$u^2 + v^2 = (x-\mu)^2 + y^2 + \frac{2(1-\mu)}{\sqrt{x^2+y^2}} + \frac{2\mu}{\sqrt{(x-1)^2+y^2}} - J \tag{10}$$

with u and v proportional to the 2 components of the perturbation vector $\langle \epsilon \cos(\theta), \epsilon \sin(\theta) \rangle$. These corrected trajectories, each now with some initial velocity, are basically identical:

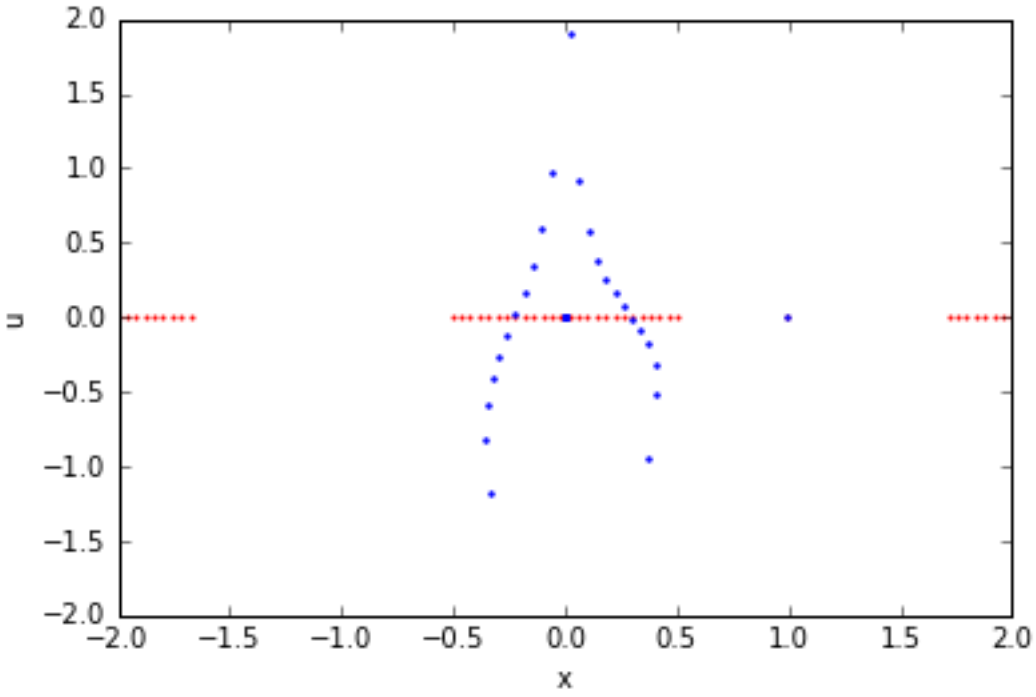



1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0


0.10
0.05
0.00
−0.05
−0.10
0.9
1.0
1.1
1.2
1.3

6 Elliptic Stable Points

Notice that one of the unique features of the plots shown earlier is that they appear to converge to some stable points e.g. given some initial x and u, the values of x and u after multiple crossings are the same. To determine their precise location, as well as their stability, we adopted Hénon et al's approach. Elliptic stable points are those classified as having a set of initial conditions that lead to intersections at that initial crossing. In essence they make a full loop and come back to the same point in x, y space. In order to determine these points, we sampled over a range of different initial conditions, found the "next" crossing, and marked both initial condition and the next crossing. Refer to the figure below:



The red dots represent initial conditions of different permissible x values and u set to 0 (since we wanted to focus on stable points that have $u = 0$). The blue dots represent

the x and u values at the next crossing. We can see that there are 4 places where the red dots coincide with the blue, and we can classify these points as fixed points. However, the central dot and the far right dot are dots within the centers of mass and hence must be neglected. Therefore, there are 2 fixed points for Jacobi constant $C = 4$. Given this we can root find and determine the exact crossings, and once we have these conditions we can use matrix operations to see how perturbations of these fixed points behave. Let's define a function g that maps a set of initial conditions to the next crossing such that:

$$\begin{pmatrix} X_n \\ U_n \end{pmatrix} = g \begin{pmatrix} X_c \\ U_c \end{pmatrix}$$

To understand how changes in the function parameters lead to changes in output we can compute the Jacobian matrix which is:

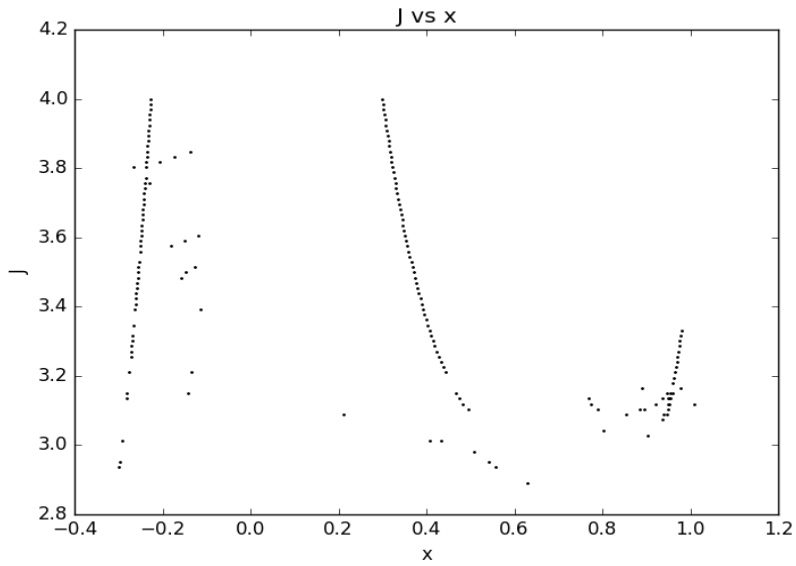
$$\begin{pmatrix} \frac{dX_n}{dX_c} & \frac{dX_n}{dU_c} \\ \frac{dU_n}{dX_c} & \frac{dU_n}{dU_c} \end{pmatrix}$$

We are able to numerically compute each of the terms in the Jacobian by applying the finite difference method:

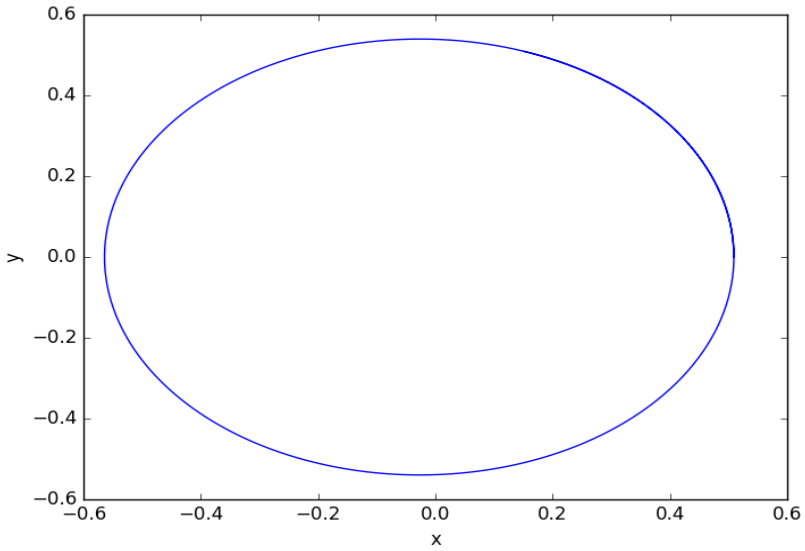
$$\frac{dX_n}{dX_c} = \frac{g_0 \begin{pmatrix} X_c + h \\ U_c \end{pmatrix} - g_0 \begin{pmatrix} X_c \\ U_c \end{pmatrix}}{h}$$

Where g_0 represents the first term in the two column vector. Once we compute this Jacobian we are able to determine its eigenvalues and thus make sense of its stability. A majority of the roots we found had eigenvalues of 1, thus meaning that they were elliptically stable and hence were points that corresponded to the foci of the contours shown earlier. We then iterated over a range of different Jacobi constants, found the

stable roots, and plotted the Jacobi constants against the roots to get the following figure:



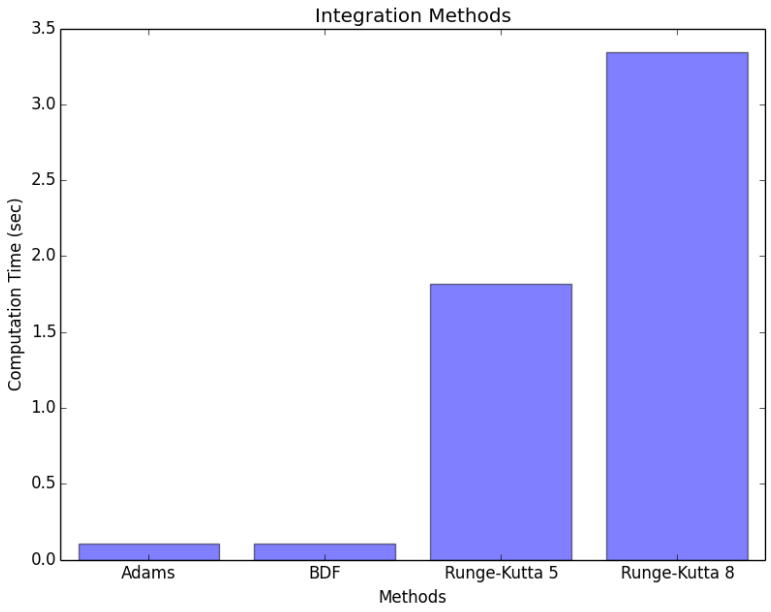
Using the figure above and picking a random point such that $x = 0.50908047464573825$ and $J = 2.9822784810126581$ and plotting these in x, y space gave us:



as expected. Therefore, having used both a root finding method and linear algebra, we were able to determine elliptic stable points of the problem.

7 Integration Methods

Without an explicit formulation for positions x and y in terms of time, the third object's position must be numerically integrated overtime. This is a nontrivial computational task when using a small enough timestep to achieve an accurate trajectory. We explored a number of different integration methods to find trajectory paths: Adams method (5th order), backward differentiation (5th order), Runge-Kutta (5th order), and Runge-Kutta (8th order). The times to compute an average trajectory from $t = 0$ to $t = 10$ with $dt = 0.001$ are as follows:



The Adams and BDF methods outperform Runge-Kutta, likely because the trajectory curves are close to linear for much of their orbits, especially if a trajectory breaks free from orbit. These conditions favor the variable step size and wider range of stability in Adams and BDF.

8 Conclusion

In this exploration we have discovered that brent's method is a good method for rapid and accurate convergence for root finding trajectory crossings. We have shown that analytically computed lagrange points are unstable by taking small ϵ perturbations about the points while preserving jacobi constants. Lastly, we have shown that elliptic stable points can be determined by root finding and then assessing the stability of these roots by computing the jacobi matrix. Certain points to work on in the future include:
1. a root finding method that uses interpolating points instead of passing an integration routine to brents method
2. a means to extend the current eigenvalueproblem.py algorithm to also find fixed points that are not elliptic stable points and determine the stability of those as well as plot homoclinic curves.
3. create an interactive interface that can allow a user to enter a specific u and v value such that both its trajectory in x,y space can be followed as well as its crossings marked in the x,u space.

9 Acknowledgments

Thanks to Matthew Holman for much assistance and guidance.

References

[1] Richard Fitzpatrick. *Newtonian Dynamics*. 2011.

[2] Michel Hénon. Numerical exploration of hamiltonian systems. *Les Houches Session XXXVI*, 1981.

[3] Richard Martin. Poincaré surface-of-section plots. University Lecture, 2013.

[4] Rick Moeckel. Chaos in the three-body problem. Presentation, 2000.