MATH2305 - Applied Mathematics I (2021), **ASE**, *pp.* 1–9
Printed in Great Britain 31 May 2021

Trisolaran

BY DANIEL TAN - U7132133
MATH2305 ASE - Regular and Chaotic Dynamics

LECTURER - DR. ZHISONG QU

SUMMARY

The goal of this project is to find livable and stable orbits for a small planet that lies within a system of three stars orbiting about a point. This comes from the science fiction novel 'The Three Body Problem' by Liu Cixin. Initially we consider Lagrange points as possible candidates for livable and stable orbits but move onto search for other points using brute-force methods and Poincare Sections.

In order to make the system easier to study we use coordinates in the rotational frame

$$x' = x\cos(\omega t) + y\sin(\omega t)$$
$$y' = y\cos(\omega t) - x\sin(\omega t)$$

We were then given the Hamiltonian in terms of the new coordinates

$$H(x', y', p_{x'}, p_{y'}) = \frac{1}{2}({p_{x'}}^2 + {p_{y'}}^2) + \omega y' p_{x'} - \omega x' p_{y'} + \sum_{i=1}^{3} \frac{1}{r_i}$$

1. VERIFYING THE HAMILTONIAN

The Hamiltonian is given in the original project description as

$$H(x', y', p_{x'}, p_{y'}) = \frac{1}{2}({p_{x'}}^2 + {p_{y'}}^2) + \omega y' p_{x'} - \omega x' p_{y'} + \sum_{i=1}^{3} \frac{1}{r_i}$$

Hamilton's equations give us

$$\frac{dx'}{dt} = \frac{\partial H}{\partial p_{x'}} = p_{x'} + \omega y' \tag{1}$$

$$\frac{dy'}{dt} = \frac{\partial H}{\partial p_{y'}} = p_{y'} - \omega x' \tag{2}$$

$$\frac{dp_{x'}}{dt} = -\frac{\partial H}{\partial x'} = \omega p_{y'} - \sum_{i=1}^{3} \frac{x' - x'_i}{r_i^3} \tag{3}$$

$$\frac{dp_{y'}}{dt} = -\frac{\partial H}{\partial y'} = -\omega p_{x'} - \sum_{i=1}^{3} \frac{y' - y'_i}{r_i^3} \tag{4}$$

But we also know

$$p_{x'} = \dot{x}' - \omega y' \tag{5}$$
$$p_{y'} = \dot{y}' + \omega x' \tag{6}$$

Then taking the derivatives

$$\frac{dp_{x'}}{dt} = \ddot{x}' - \omega\dot{y}' \tag{7}$$
$$\frac{dp_{x'}}{dt} = \ddot{y}' + \omega\dot{x}' \tag{8}$$

But then we can equate the derivatives of the momentum giving and subsitute equation (3) and (4).

$$\ddot{x}' - \omega\dot{y}' = \omega p_{y'} - \sum_{i=1}^{3}\frac{x' - x_i'}{r_i^3}$$
$$\therefore x' - \omega\dot{y}' = \omega(\dot{y}' + \omega x') - \sum_{i=1}^{3}\frac{x' - x_i'}{r_i^3}$$
$$\therefore x' - 2\omega\dot{y} - \omega^2 x' = -\sum_{i=1}^{3}\frac{x' - x_i'}{r_i^3}$$

Similarly, to find the second equation of motion

$$\ddot{y}' + \omega\dot{x}' = -\omega p_{x'} - \sum_{i=1}^{3}\frac{y' - y_i'}{r_i^3}$$
$$\ddot{y}' + \omega\dot{x}' = -\omega(\dot{x}' - \omega y') - \sum_{i=1}^{3}\frac{y' - y_i'}{r_i^3}$$
$$\ddot{y}' + 2\omega\dot{x}' + \omega^2 y' = -\sum_{i=1}^{3}\frac{y' - y_i'}{r_i^3}$$

So we have obtained equations (17) and (18) from the project description from Hamilton's equations.

2. Finding Lagrange Points

The lagrange points occur when $\dot{x}' = \dot{y}' = 0$. Then from the equations of motion from above we have that

$$\sum_{i=1}^{3}\frac{x' - x_i'}{r_i^3} = 0$$
$$\sum_{i=1}^{3}\frac{y' - y_i'}{r_i^3} = 0$$

But letting a solve function run searching for all solutions is computationally expensive and very time consuming. So instead note that both of these equations is the derivative of the original sum

$$V = \sum_{i=1}^{3} \frac{1}{r_i}$$

So instead we plot this function and from inspection we find very approximate guesses to where local minimum, local maximum and saddles points are to compute the lagrange points faster using fsolve.

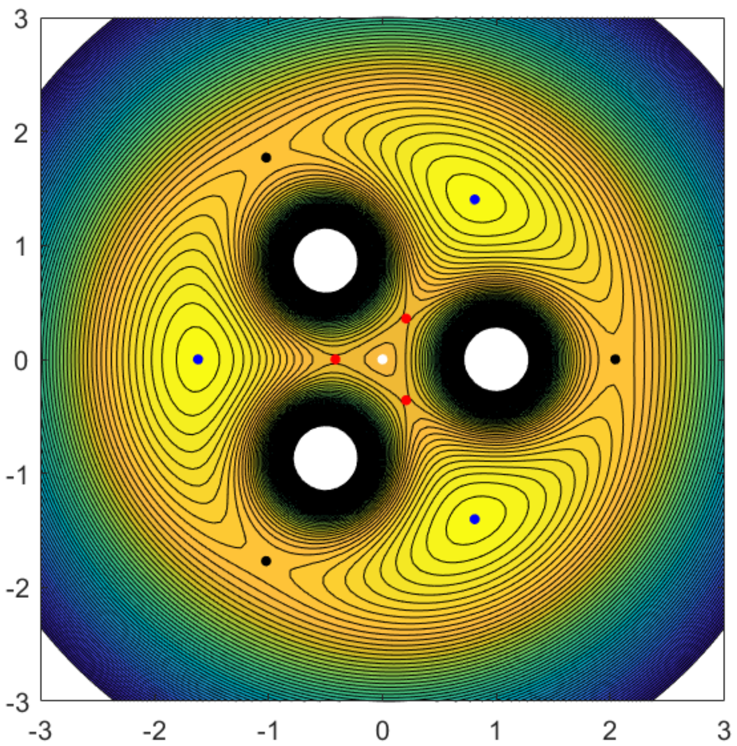


Fig. 1: The lagrange points found over the contour map of V where different colours represent the classes of points

From figure 1 we can see that there are four classes of Lagrange points due to the rotational symmetry of the system. The points have coordinates (one from each class)

$$\begin{aligned} \text{L1} &= (0,0) & \qquad \text{L2} &= (-0.4139, 0) \\ \text{L3} &= (2.0438, 0) & \qquad \text{L4} &= (-1.6198, 0) \end{aligned}$$

3. Stability of Lagrange Points

Before discussing the stability of the Lagrange points I used the ODE solver ODE15s as it preserved the conservative nature of the Hamiltonian better then non-stiff ODE solvers. From figure 2 below it is clear stiff ODE solvers preserve the constant Hamiltonian much more accurately.

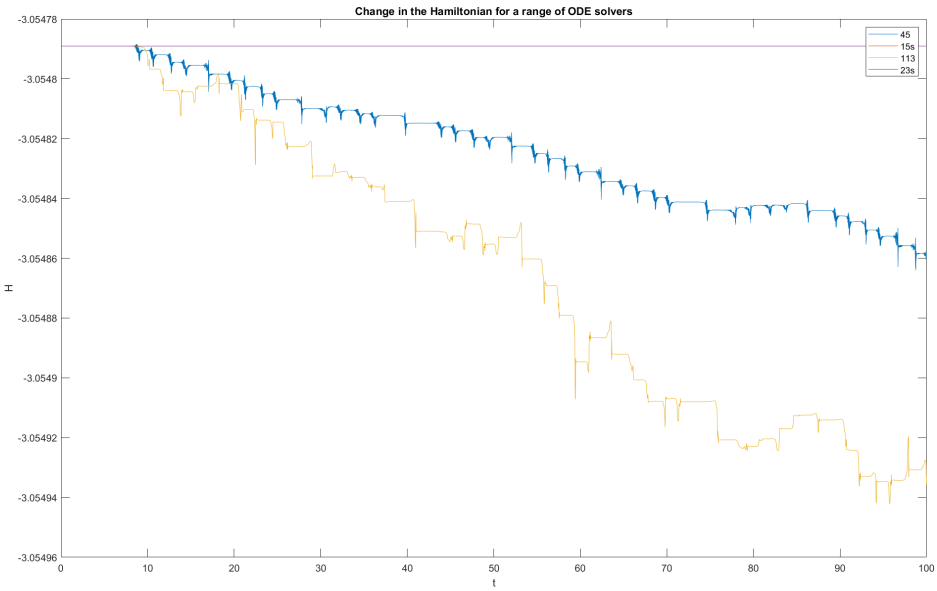


Fig. 2: The change in Hamiltonian over time with different differential equation solvers

The Lyapunov exponent was computed in order to test the behaviour of the Lagrange points when the initial conditions are slightly perturbed. To do this, the Jacobian is calculated from Hamilton's Equations (equations 1-4)

$$\mathcal{M} = \begin{bmatrix} 0 & \omega & 1 & 0 \\ -\omega & 0 & 0 & 1 \\ -\sum_{i=1}^{3} \frac{1}{r_i^3} + \sum_{i=1}^{3} \frac{3(x-x_i)^2}{r_i^5} & \sum_{i=1}^{3} \frac{3(x-x_i)(y-y_i)}{r_i^5} & 0 & \omega \\ \sum_{i=1}^{3} \frac{3(x-x_i)(y-y_i)}{r_i^5} & -\sum_{i=1}^{3} \frac{1}{r_i^3} + \sum_{i=1}^{3} \frac{3(y-y_i)^2}{r_i^5} & \omega & 0 \end{bmatrix}$$

$$\frac{d\delta\mathbf{y}}{dt} = \mathcal{M} \cdot \delta\mathbf{y}$$

The Lyapunov Exponent is given as $\lambda(\mathbf{y_0}) = \lim_{t\to\infty} \lim_{\|\delta\mathbf{y_0}\|\to 0} \frac{1}{t} \ln \frac{\|d\delta\mathbf{y}(t)\|}{\|\delta\mathbf{y_0}\|}$

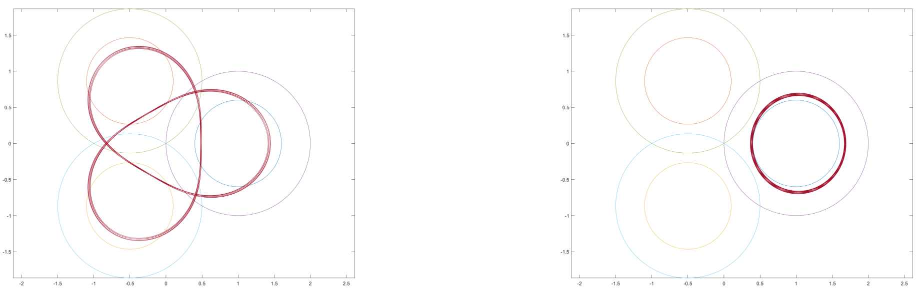
From the figure below, it is clear for classes of lagrange points 1, 3, and 4 that the limit is non-zero positive number. Similarly, this is noticeable when comparing the values of p_y of time when the initial conditions are slightly perturbed from the Lagrange points.

But for class 2 of Lagrange points it is unclear from figure (b) below if the exponent is approaching zero or small positive number. But now consider figure (f) below which compares

p_y over time to a small in change in the initial condition. Immediately we see that the change causes large differences immediately. But also note that over time the value of p_y deviates from its constant line. This is due to the build of precision error from the ODE solver. But for stable orbits we wouldn't see small precision error cause complete deviation to a chaotic orbit. From the Lyapunov Exponents it is clear that none of the Lagrange points are stable orbits.

4. Livable Orbits

I have found only one class of stable livable orbits but more stable orbits that are not livable.



(a) Non-livable stable orbit

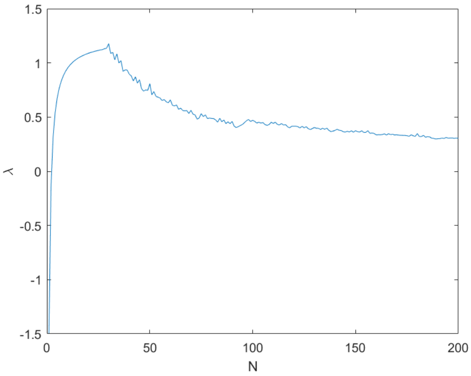
(b) Livable stable orbit

4·1. *Brute Force Method*

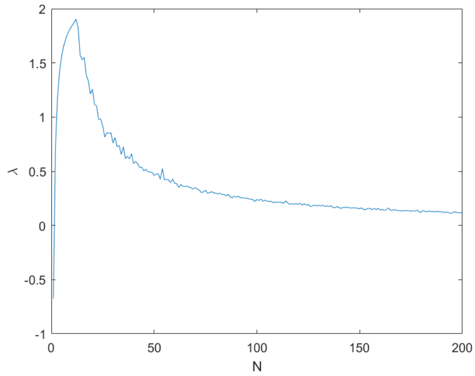
There are three different brute force methods used in order to find stable orbits. The first two are similar and introduce a 'tolerance' in which the ODE is solved numerically and when the system crosses the x-axis the difference between the initial conditions and the current conditions is calculated using the euclidean norm to see if it is within a certain 'tolerance'. The idea is if the system returns to the initial conditions with similar momentum and position values then it is in a somewhat regular orbit as it we nearly repeat it self. But there are problems with this search method. A brute force method will search hundred or thousands of different initial conditions and solving that many differential equations over a large interval is computationally expensive and time consuming. Also it become clear that the tolerance must be set quite high around 0.01 in order to find stable orbits in a reasonable amount of time as low tolerance would require increasing the interval to be solved on significantly. This leads to many false positives. The first method simply searched a range of xp_x, p_y values of the x-axis. The second method fixes a constant Hamiltonian and then searches x, p_x values.

The third method is significantly faster and finding livable orbits and works by simply stopping the ODE valuation whenever it enters a non-livable area. It searches through different Hamiltonian constants and x, p_x values. This method was mentioned at office hours in the last week by Zhisong Qu.

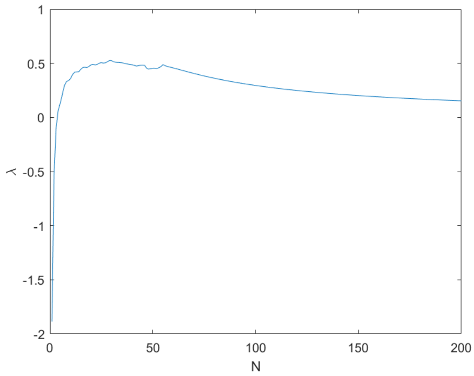
Method 1: bruteforce.m
Method 2: brute_ham.m
Method 3: brute_nontol.m



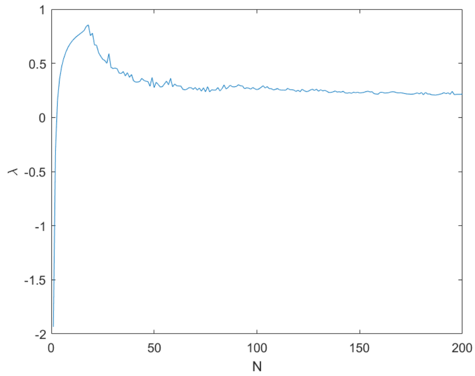
(a) L1



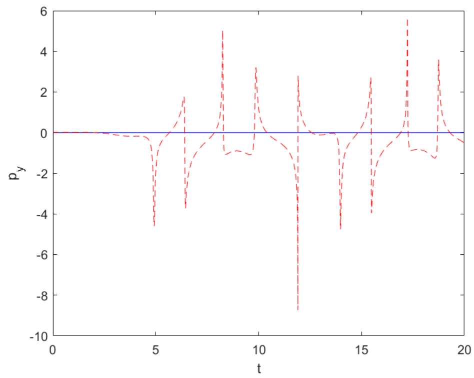
(b) L2



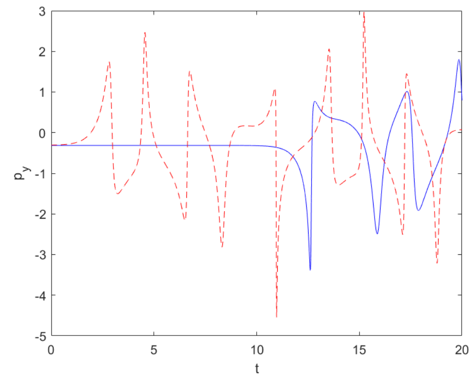
(c) L3



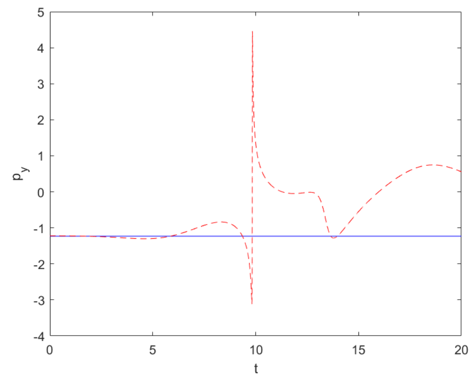
(d) L4



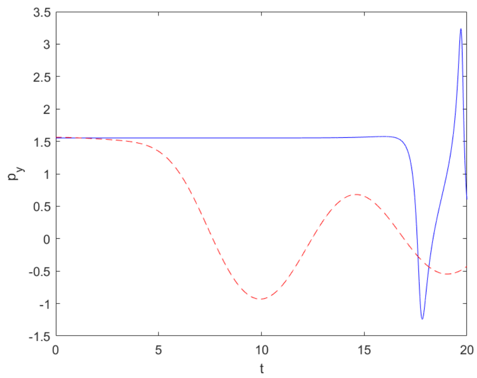
(e) L1



(f) L2



(g) L3



(h) L4

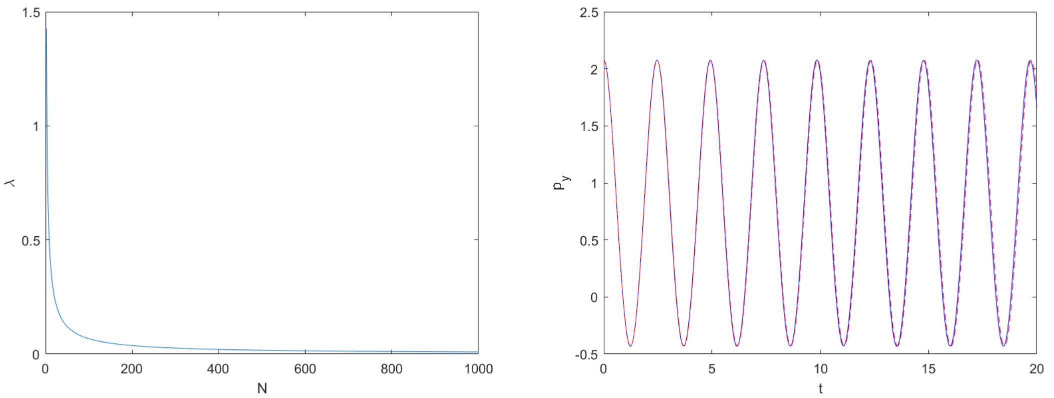
Lyapunov Exponents for different Lagrange classes and comparing the change in p_y over time between Lagrange point and a small deviation from the Lagrange point

4·2. *Stability of Orbits*

Firstly the approximate initial values of the stable orbit are approximately for the livable orbit.

$$x = 0.38 \;\; y = 0 \;\; p_x = 0 \;\; p_y = 2.0733$$
$$H(x, y, p_x, p_y) = -1.6821$$

To test the stability of the orbit we calculate the Lyapunov Exponent, the same as testing the stability of the Lagrange points. It is clear from figure (a) below that the Lyapunov Exponent converges to 0 which indicates a non-chaotic orbit. Similarly when adding a small perturbation we see that the p_y hardly deviates from the non-perturbed orbit indicating a stable orbit.



(a) Liable orbit Lyapunov Exponent

(b) Livable orbit difference with small perturbation

In order to investigate whether or not this is a KAM surface or a fixed point we plot the Poincare section around the initial conditions

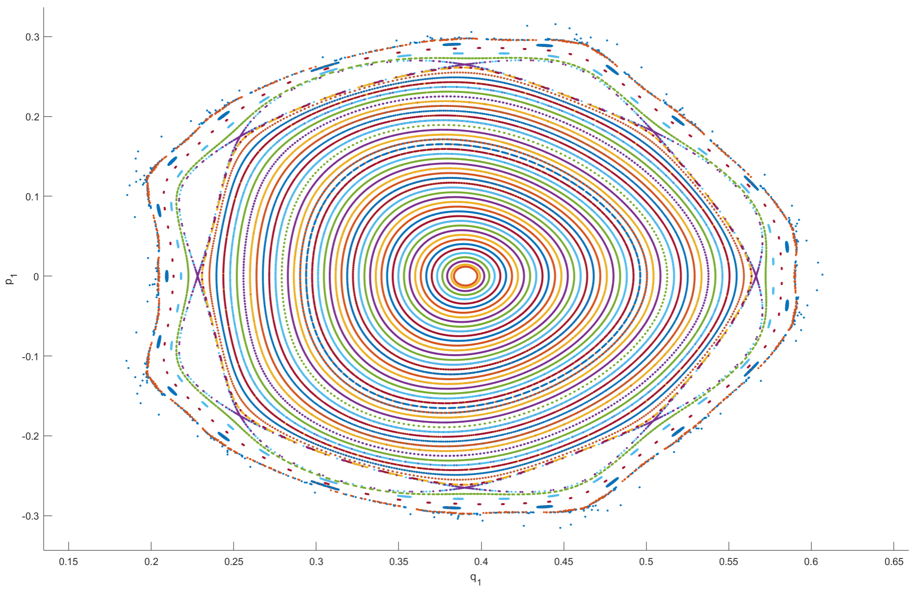


Fig. 6: Poincare Section around the stable orbit

We can see the class of solutions are the conditions that lie around the fixed point in the Poincare section which have more regular orbits.

From inspection the Poincare we can see that this is in the middle of an island and intuitively from the orbit we can see that is has a rotational number of 1. To show that this is a fixed point more rigorously we would use Newton's Method to find the fixed point but I was unable to figure out how to implement Newton's method for this system.

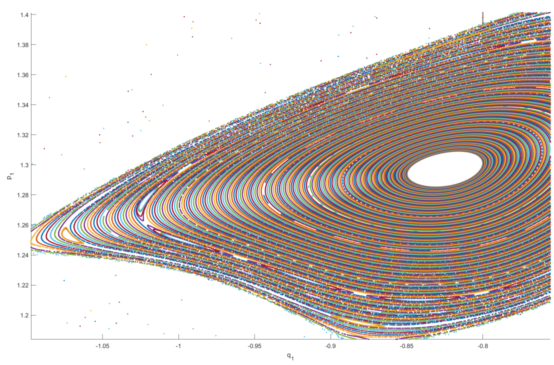


Fig. 7: Poincare Section around the non-livable but stable orbit

In order to potentially find more livable orbits I would recommend using Newton's method to find many more fixed points which are good candidates for stable orbits and therefore livable orbits. From the plot below there are many islands that surround the 'larger' fixed point. Using Newton's method to find these more accurately then inspection and potentially finding more will likely give rise to more classes of livable orbits.

References

[1] Zhisong Qu. *Regular and Chaotic Dynamics Lecture Notes*.

[2] Zhisong Qu [*Regular and Chaotic Dynamics Research Project - Trisolaran*].

[3] Zhisong Qu: Matlab scripts
`https://github.com/zhisong/ANU_MATH6405ASE_codes`