Project number:

Part IB Computational Projects 2020

C7920B

Attach to front of report

Mathematical Tripos, Part 1B
Computational project

The Restricted Three-Body Problem

University of Cambridge

April 26, 2020

1 Introduction

1.1 Question 1

From project introduction, we have the following equations:

$$\ddot{x} - 2\dot{y} = -\frac{\partial \Omega}{\partial x} \tag{1.1a}$$

$$\ddot{y} - 2\dot{x} = -\frac{\partial \Omega}{\partial y} \tag{1.1b}$$

And

$$\Omega = -\frac{1}{2}(x^2 + y^2) - \frac{\mu}{\sqrt{(x+1-\mu)^2 + y^2}} - \frac{1-\mu}{\sqrt{(x-\mu)^2 + y^2}} \tag{1.2}$$

Which has time derivative:

$$\dot{\Omega} = \frac{\partial \Omega}{\partial x}\dot{x} + \frac{\partial \Omega}{\partial y}\dot{y} = (2\dot{y} - \ddot{x})\dot{x} + (2\dot{x} - \ddot{y})\dot{y} = -(\dot{x}\ddot{x} + \dot{y}\ddot{y})$$

The quantity

$$J = \frac{1}{2}(\dot{x}^2 + \dot{y}^2) + \Omega(x, y)$$

has time derivative:

$$\dot{J} = \dot{x}\ddot{x} + \dot{y}\ddot{y} + \dot{\Omega} = 0$$

Therefore it is a constant of the motion. Hence,

$$\frac{1}{2}(\dot{x}^2 + \dot{y}^2) + \Omega(x, y) = J(x, y) = J(x_0, y_0) = \Omega(x_0, y_0) + \frac{1}{2}(u_0^2 + v_0^2)$$

Therefore, we have the following

$$\Omega(x, y) = \Omega(x_0, y_0) + \frac{1}{2}(u_0^2 + v_0^2) - \frac{1}{2}(\dot{x}^2 + \dot{y}^2) \leqslant \Omega(x_0, y_0) + \frac{1}{2}(u_0^2 + v_0^2)$$

equality holds when $\dot{x} = \dot{y} = 0$

Programming Task

See appendix

2 Space travel

2.1 Question 2

Given that the spaceship is close enough to P_h, (2) becomes

$$\Omega = -\frac{0.5}{\sqrt{(x-\mu)^2 + y^2}}$$

Apply coordinate transformation $x(t) - \mu = r(t)\cos\theta(t),\ y(t) = r(t)\sin\theta(t)$, we obtain:

$$\begin{aligned}
\dot{x} &= \dot{r}\cos\theta - r\sin\theta\dot{\theta} \\
\dot{y} &= \dot{r}\sin\theta + r\cos\theta\dot{\theta} \\
\ddot{x} &= \ddot{r}\cos\theta - 2\dot{r}\sin\theta\dot{\theta} - r\cos\theta\dot{\theta}^2 - r\sin\theta\ddot{\theta} \\
\ddot{y} &= \ddot{r}\sin\theta + 2\dot{r}\cos\theta\dot{\theta} - r\sin\theta\dot{\theta}^2 + r\cos\theta\ddot{\theta}
\end{aligned}$$

And

$$\begin{aligned}
\frac{\partial\Omega}{\partial x} &= \frac{\cos\theta}{2r^2} \\
\frac{\partial\Omega}{\partial y} &= \frac{\sin\theta}{2r^2}
\end{aligned}$$

Hence, after evaluating $\cos\theta * (1a) + \sin\theta * (1b)$ and $\sin\theta * (1a) - \cos\theta * (1b)$,

$$\ddot{r} - r\dot{\theta}^2 - 2r\dot{\theta} = -\frac{1}{2r^2} \tag{2.1.1a}$$

$$2\dot{r} + r\ddot{\theta} + 2\dot{r}\dot{\theta} = 0 \tag{2.1.1b}$$

from (2.1.1b) we have the following:

$$\frac{d(r^2\dot{\theta})}{dt} = -2r\dot{r} \Longrightarrow r^2\dot{\theta} = k - r^2$$

i.e.

$$\dot{\theta} = -1 + kr^{-2}$$

where k is an arbitrary constant. Then from (2.1.1a) we have:

$$\ddot{r} - r(-1 + kr^{-2})^2 - 2r(-1 + kr^{-2}) = -\frac{1}{2r^2}$$

i.e.

$$\ddot{r} = \frac{k^2}{r^3} - \frac{1}{2r^2} - r = -V'(r)$$

So

$$V'(r) = -\frac{k^2}{r^3} + \frac{1}{2r^2} + r \Longrightarrow V(r) = \frac{k^2}{2r^2} - \frac{1}{2r} + \frac{r^2}{2} + C$$

where C is an arbitrary constant. If the spaceship moves in a circular orbit with radius a, then we have the following constraint $r = a$, and then $\dot{r} = \ddot{r} = 0$.

Therefore, from (2.1.1b) we have $\ddot{\theta} = 0,\ \dot{\theta} = const$ and from (2.1.1a) we have $\dot{\theta}^2 + 2\dot{\theta} - \frac{1}{2a^3} = 0$, take the positive root $\omega = -1 + \sqrt{1 + \frac{1}{2a^3}}$. Then the initial conditions for circular motion are as follow:

$$\dot{r(0)} = 0,\ \dot{\theta(0)} = \omega,\ r(0) = a,\ \theta(0) = 0$$

which corresponds to the initial conditions in Cartesian coordinates:

$$x(0) = \mu + a,\ y(0) = 0,\ \dot{x(0)} = 0,\ \dot{y(0)} = a\omega$$

From $\ddot{r} = 0$,

$$-\frac{k^2}{a^3} + \frac{1}{2a^2} + a = 0$$

and since $\omega > 0$,

$$k = \sqrt{\frac{a}{2} + a^4}$$

. The analytic periodic solution is: $x = \mu + a\cos(\omega t),\ y = a\sin(\omega t),\ \dot{x} = -a\omega\sin(\omega t),\ \dot{y} = a\omega\cos(\omega t)$.

Put in the initial values for a = 0.01 and run the solver, I have the following graph:

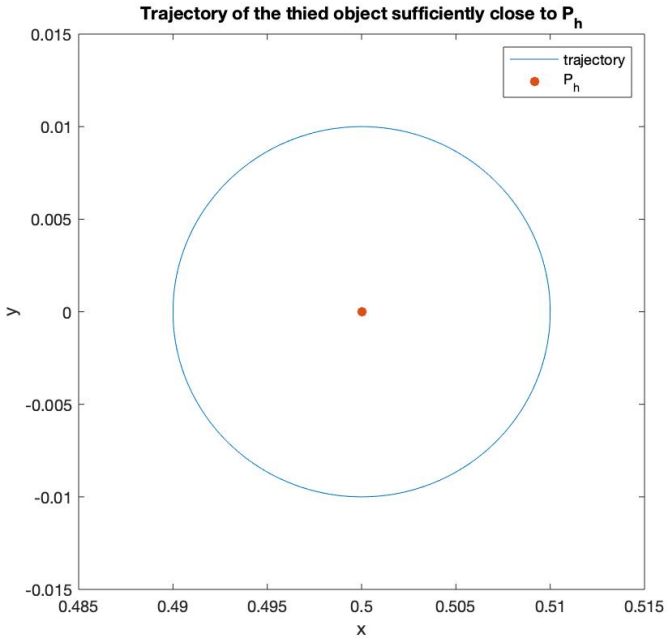
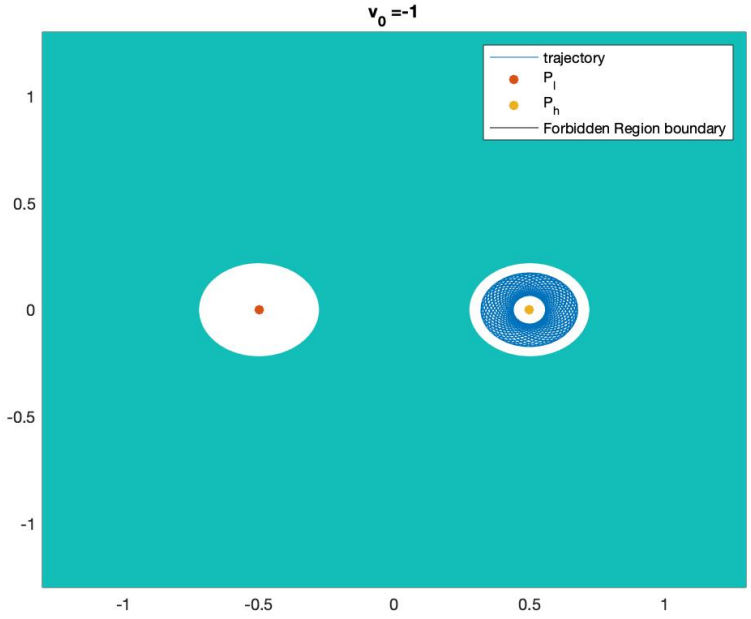


Figure 2.1: Numerical solution of the system for a = 0.01

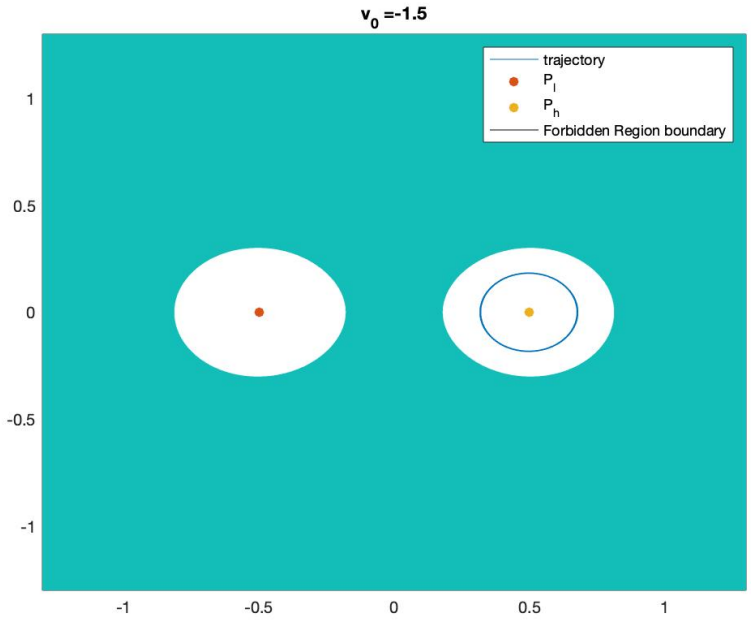
which is precisely a circular motion.

2.2 Question 3

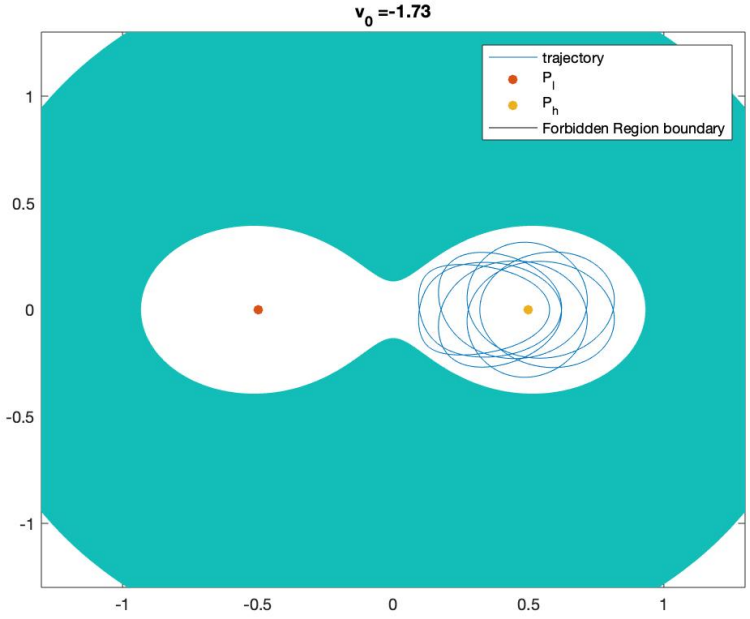
The graphs show the forbidden region (teal) of this motion.



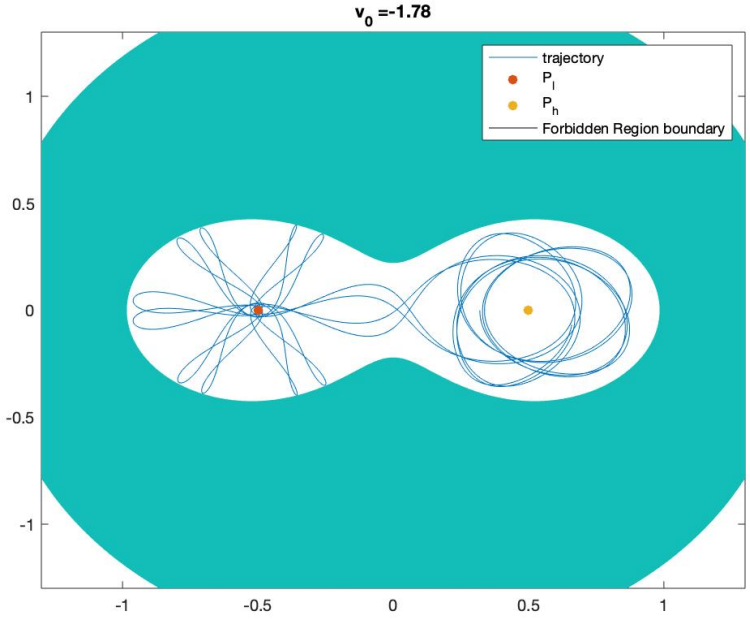
$x(30) = 0.4280$, $y(30) = -0.0666$



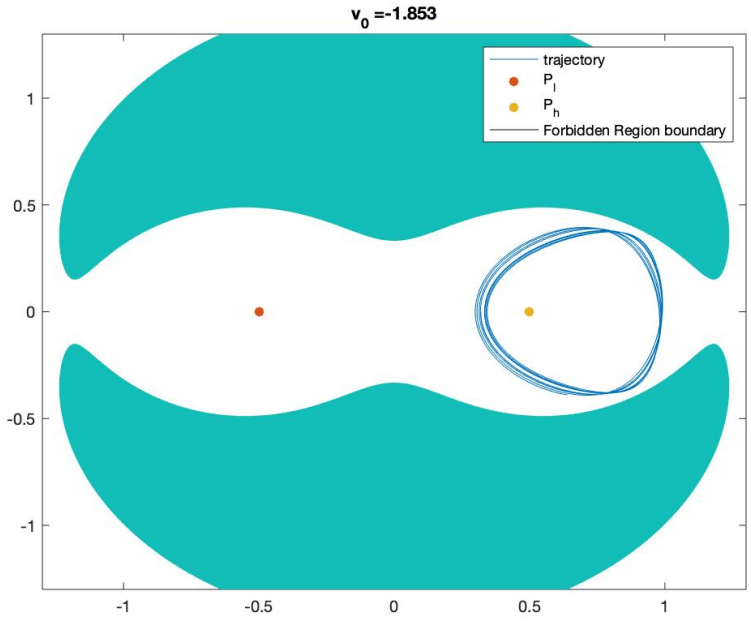
$x(30) = 0.3215$, $y(30) = 0.0415$



$x(30) = 0.1820$, $y(30) = 0.1696$



$x(30) = 0.6597$, $y(30) = -0.0675$

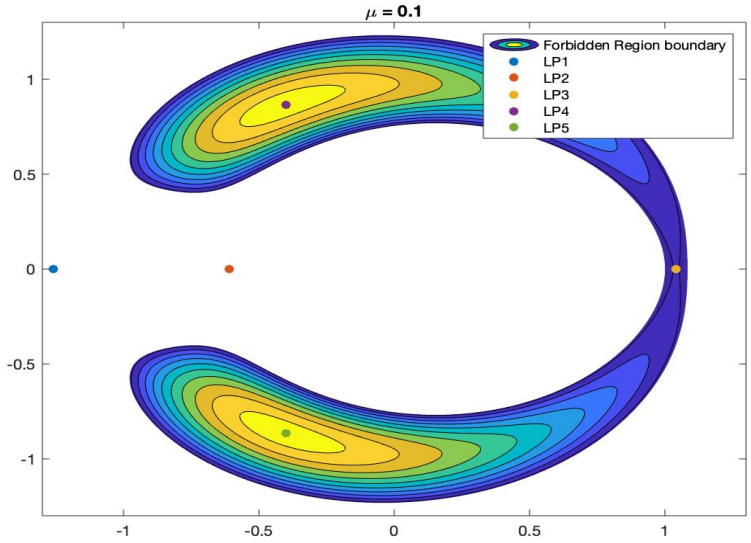


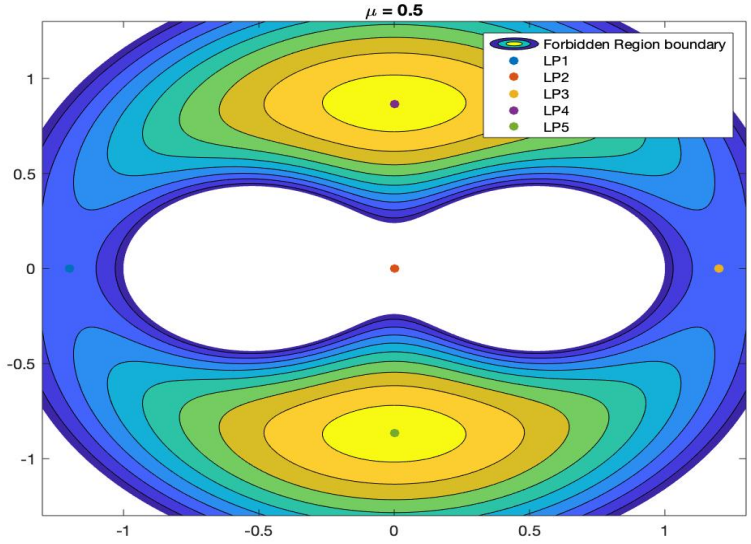
$$\text{x}(30) = 0.6400,\ \text{y}(30) = \text{-}0.3890$$

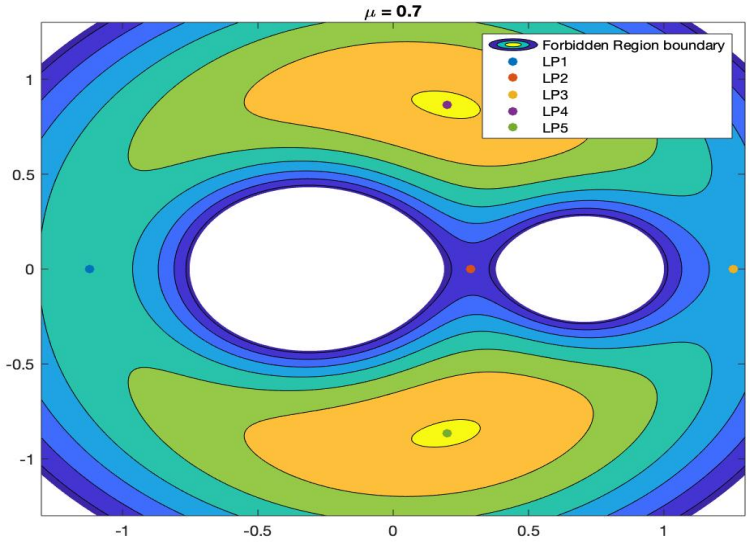
The relative tolerance accounts for the accuracy of each component of the solution, these graphs are accurate to the tenth significant figure as I have set the relative tolerances to 1e-10. The absolute tolerance is for the solution when it approaches 0, and as the solution for each v_0 is at a distance much greater than 1e-10 from 0, they are precise. The trajectories are elliptical, and as $-v_0$ increases, the eccentricity get larger. The third body gradually overcomes of the potential barrier of P_h and the forbidden region shrunk as the initial constant J increases with $|v_0|$.The allowed the region is a useful guide to the size of the trajectory Therefore when we move from P_h to P_l it would be most suitable to choose $v_0 = -1.78$.

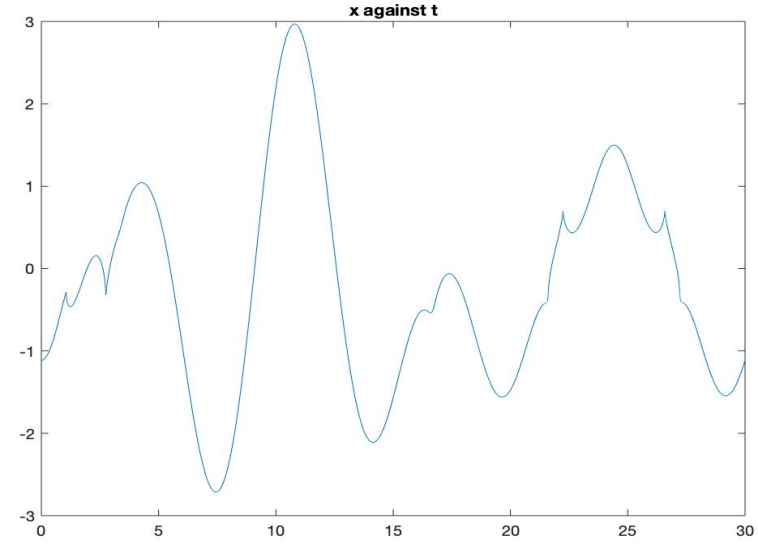
3 Lagrange points and asteroids

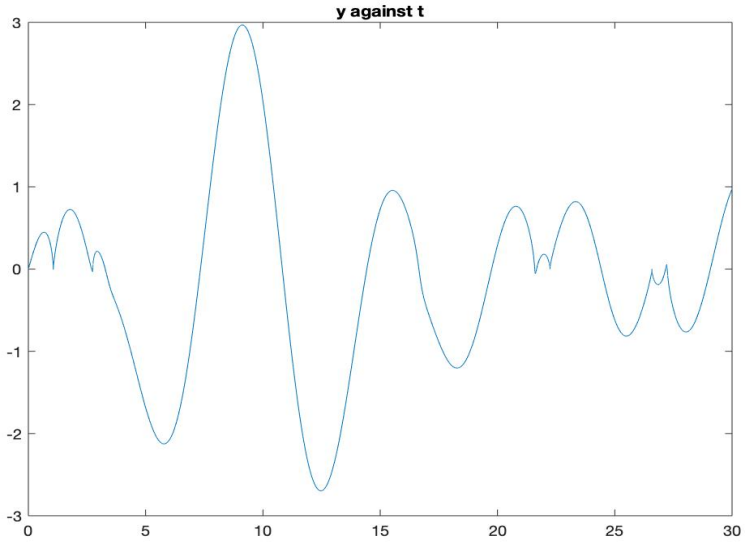
3.1 Question 4

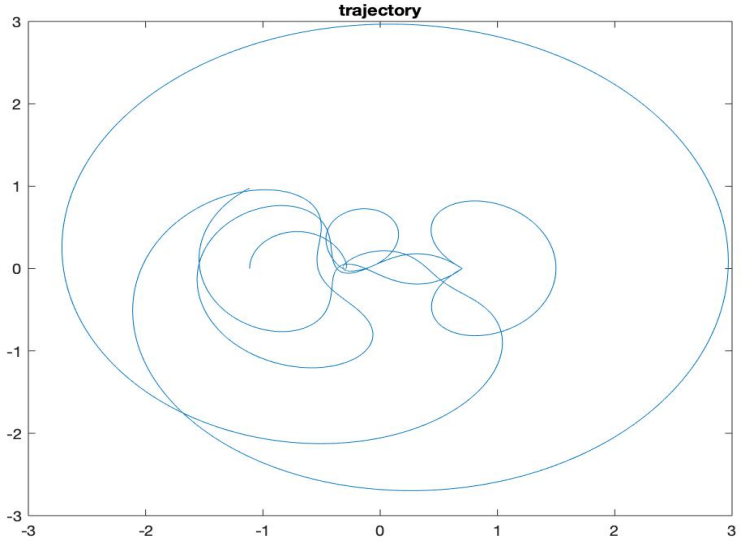

μ = 0.1
Forbidden Region boundary
LP1
LP2
LP3
LP4
LP5
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

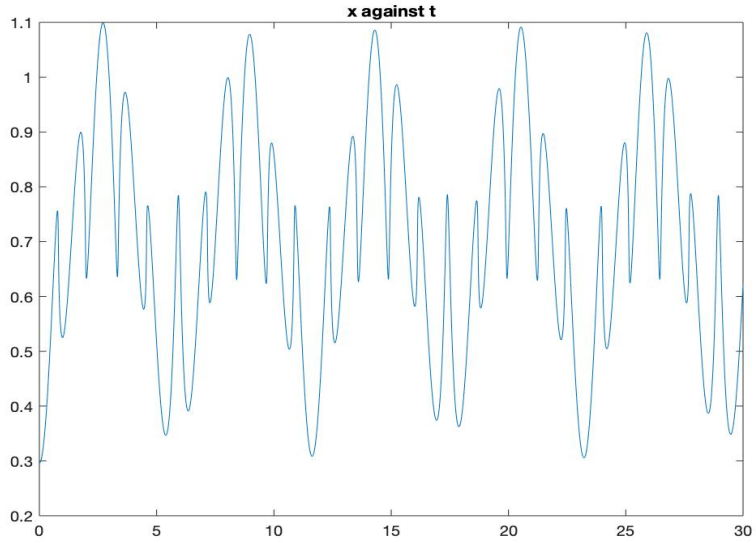

μ = 0.5
Forbidden Region boundary
LP1
LP2
LP3
LP4
LP5
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

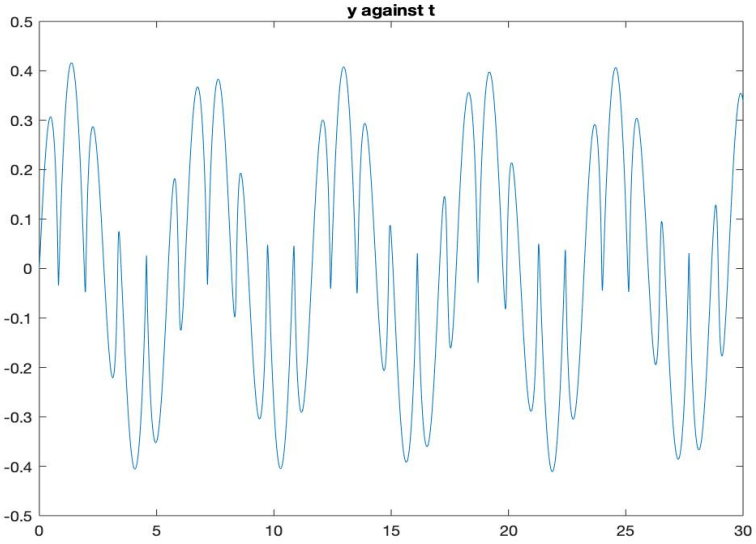

μ = 0.7
Forbidden Region boundary
LP1
LP2
LP3
LP4
LP5
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

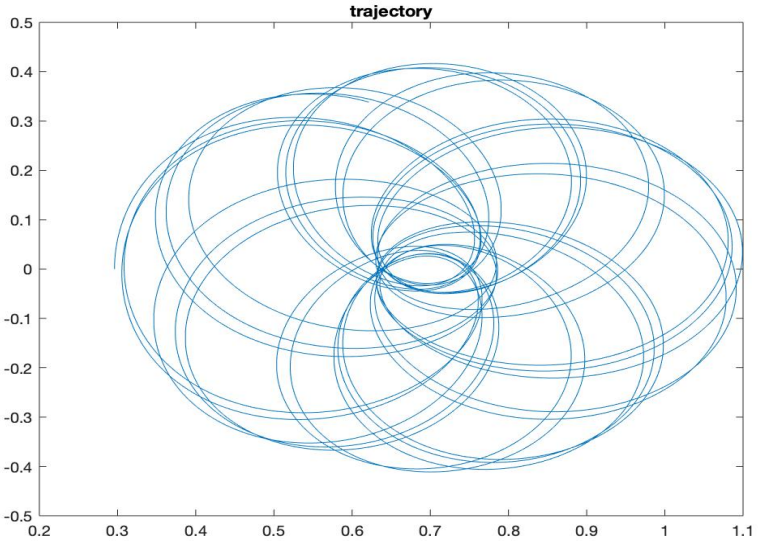

x against t
3
2
1
0
-1
-2
-3
0
5
10
15
20
25
30

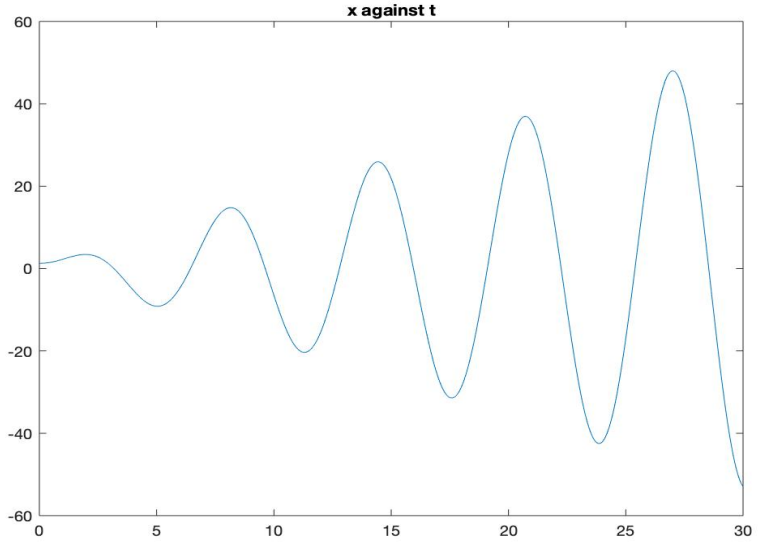

y against t
3
2
1
0
-1
-2
-3
0
5
10
15
20
25
30

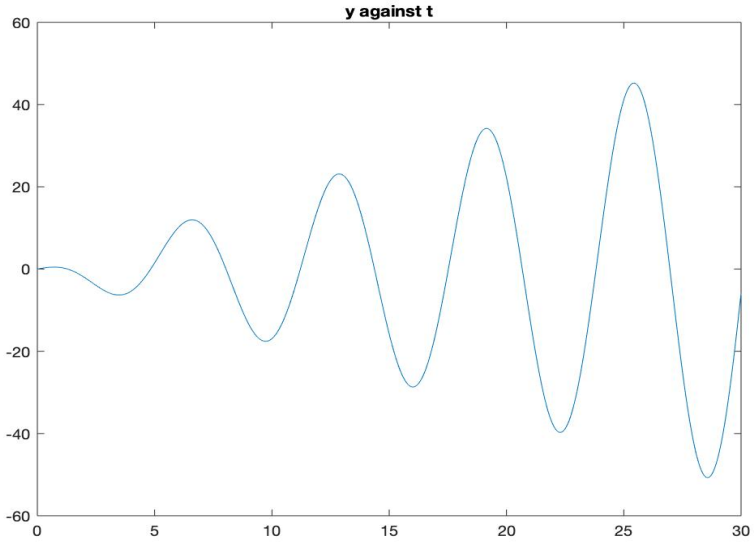

trajectory
3
2
1
0
-1
-2
-3
-3
-2
-1
0
1
2
3

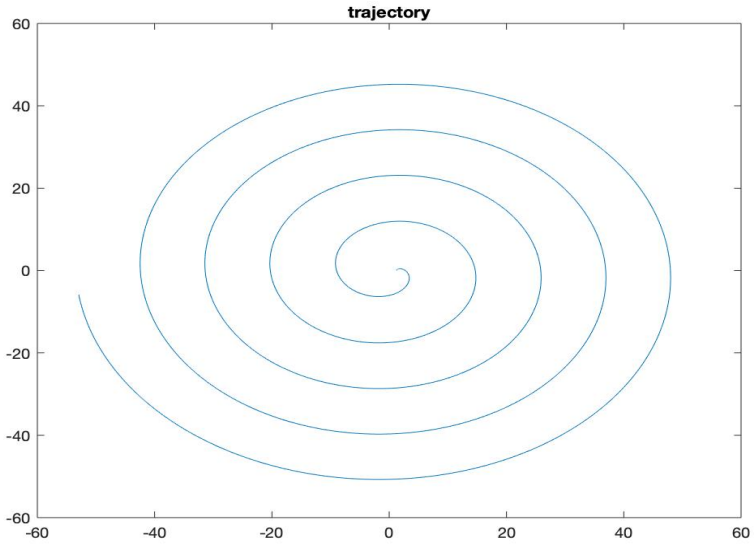

x against t
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
5
10
15
20
25
30


y against t
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
0
5
10
15
20
25
30


trajectory
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1


x against t
60
40
20
0
-20
-40
-60
0
5
10
15
20
25
30



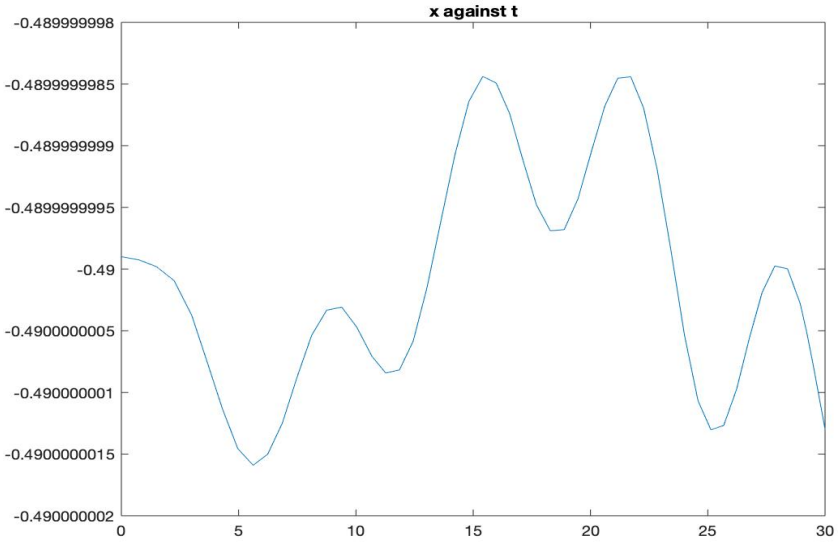


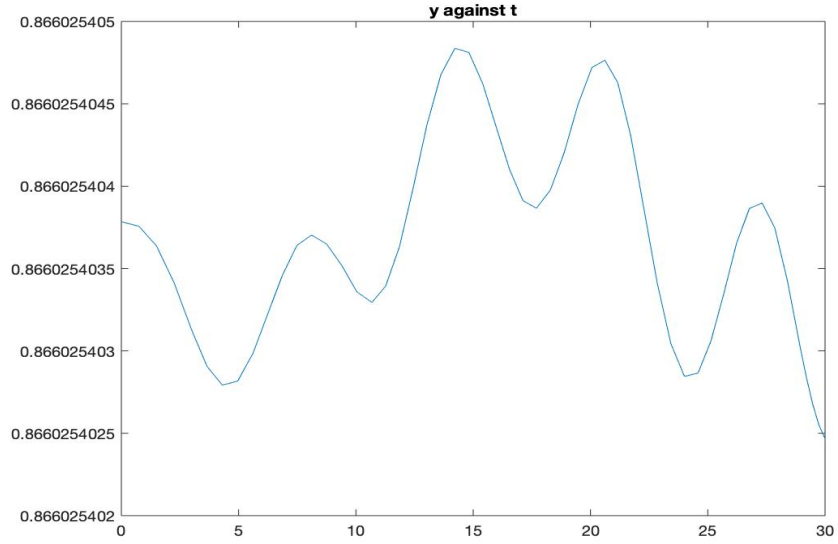
The first three graphs show the positions of five Lagrange points and contours of $\Omega > J$ for different values of μ(here i took $\Omega > J$ as it is not evident to draw a contour graph for all values of Ω, as they differ only by a little), and the 9 following show the evolution of x against t , y against t, and the trajectories of the third body starting near the collinear Lagrange

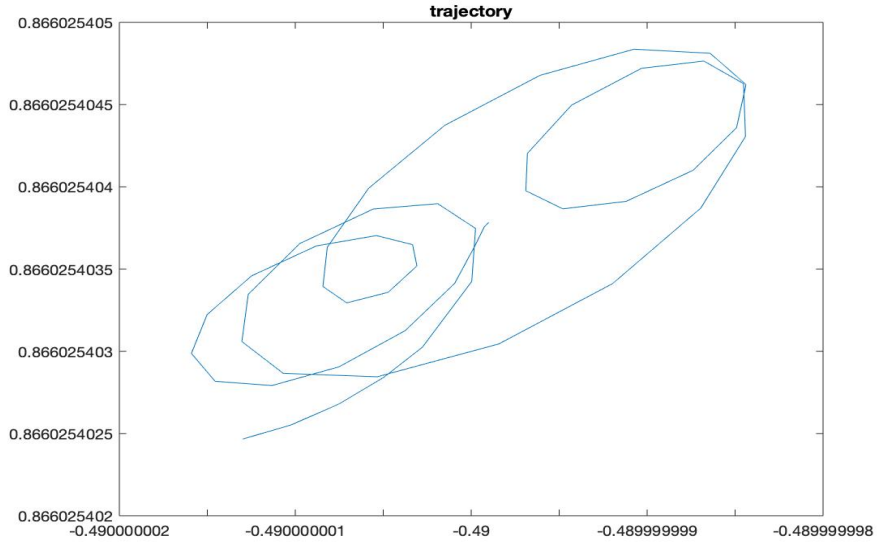
points, with a perturbation of 0.01. It is evident from the diagrams that the collinear lagrange points are unstable. From the contours of Ω we can observe that the collinear Lagrange points are saddle points, and hence they must be unstable, no matter how big μ is. To perform the linearised analysis, we need to check the Hessian of Ω and find out its eigenvalues. The contour plot has shown that it must have eigenvalues of distinct signs, hence unstable.

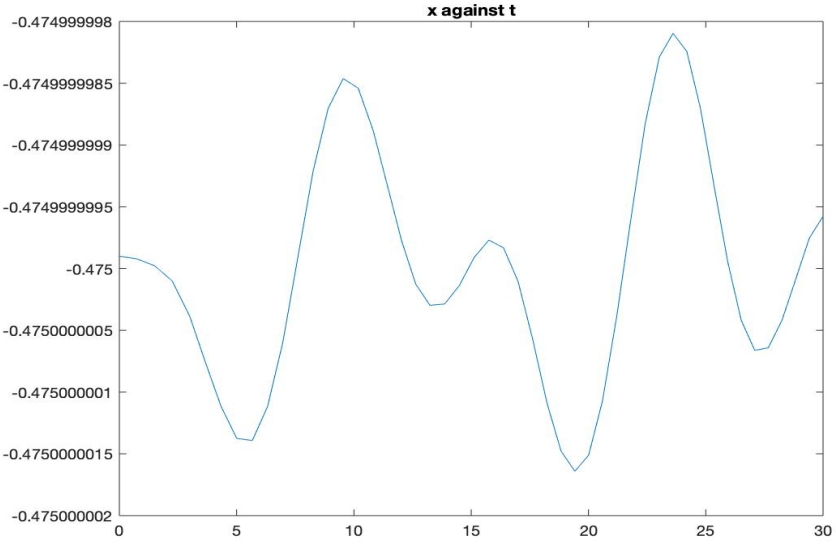
3.2 Question 5

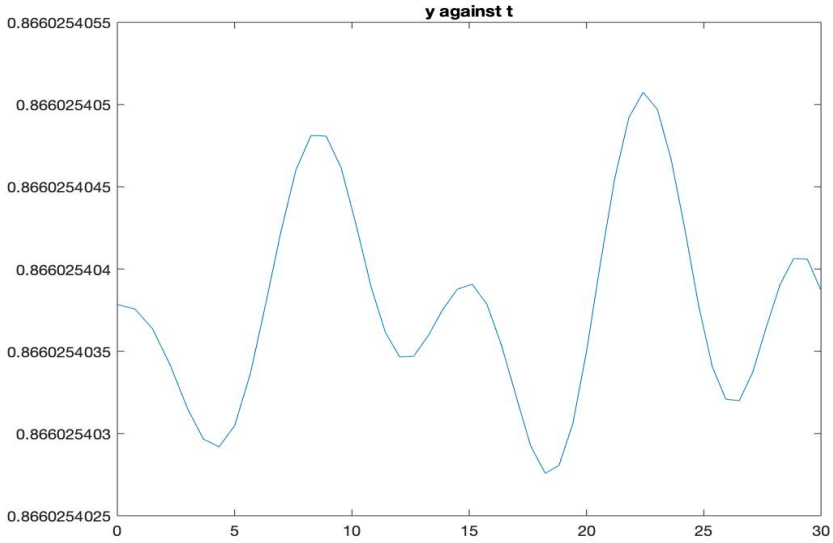
The graphs below are as requested by the question. The stability of Equilateral Lagrange points seems to be weakened as μ increases.

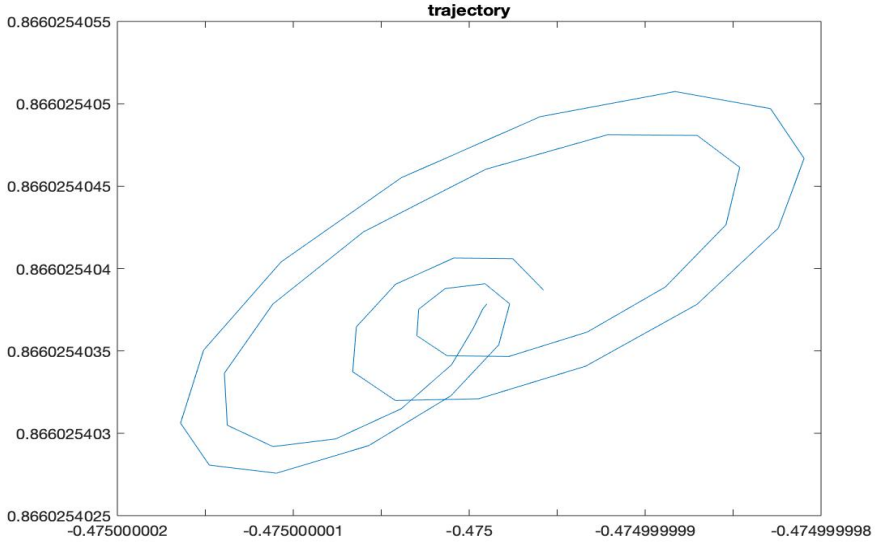

x against t
-0.489999998
-0.4899999985
-0.489999999
-0.4899999995
-0.49
-0.4900000005
-0.490000001
-0.4900000015
-0.490000002
0
5
10
15
20
25
30

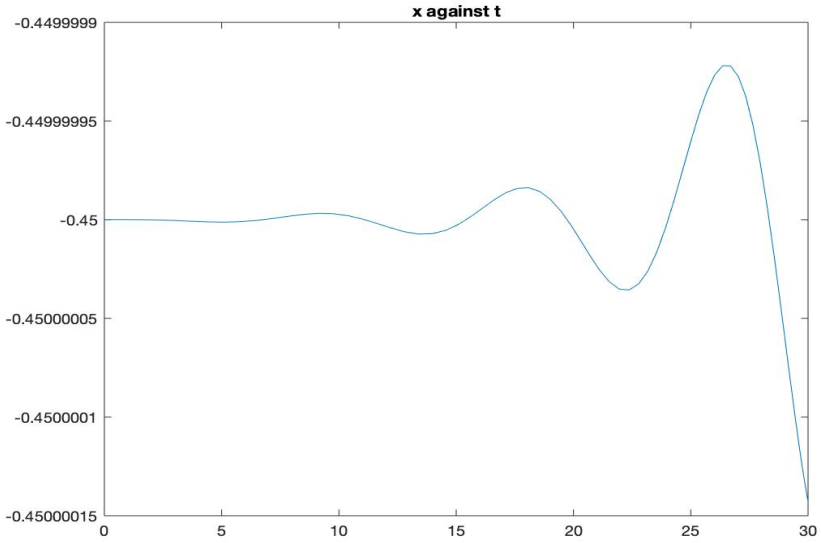

y against t
0.866025405
0.8660254045
0.866025404
0.8660254035
0.866025403
0.8660254025
0.866025402
0
5
10
15
20
25
30

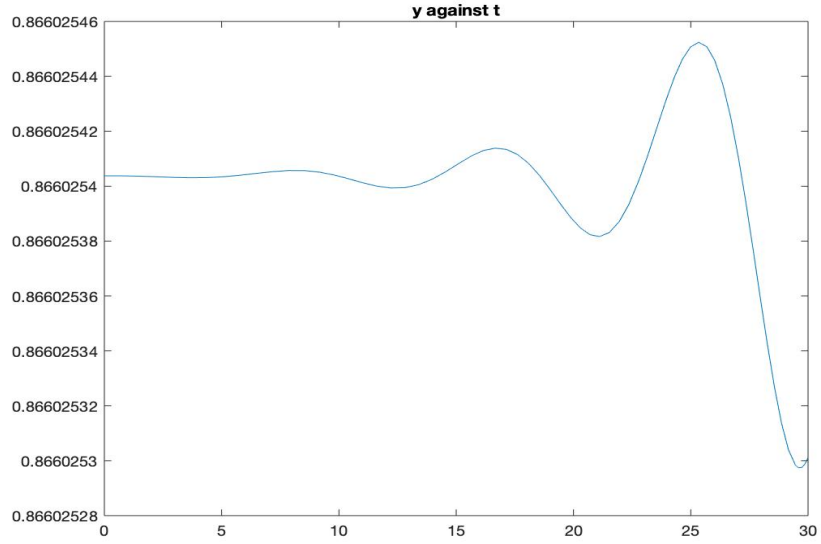

trajectory
0.866025405
0.8660254045
0.866025404
0.8660254035
0.866025403
0.8660254025
0.866025402
-0.490000002
-0.490000001
-0.49
-0.489999999
-0.489999998

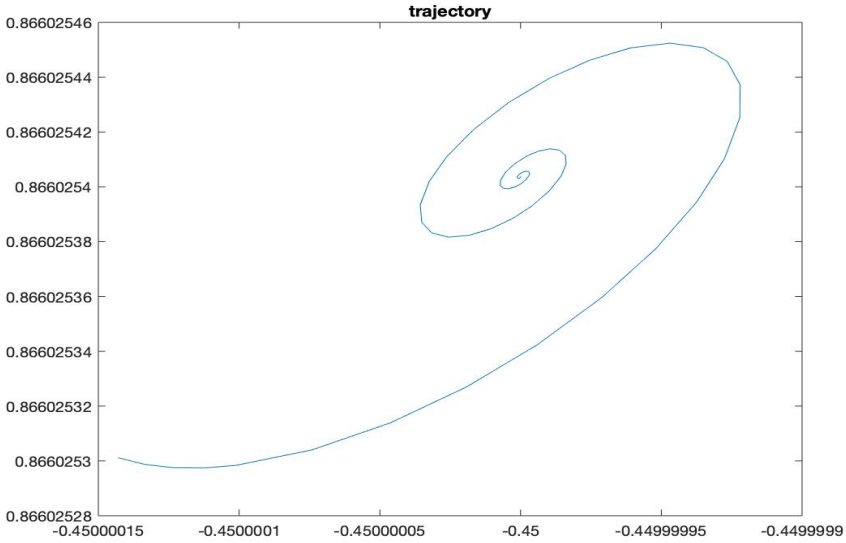

x against t
-0.474999998
-0.4749999985
-0.474999999
-0.4749999995
-0.475
-0.4750000005
-0.475000001
-0.4750000015
-0.475000002
0
5
10
15
20
25
30

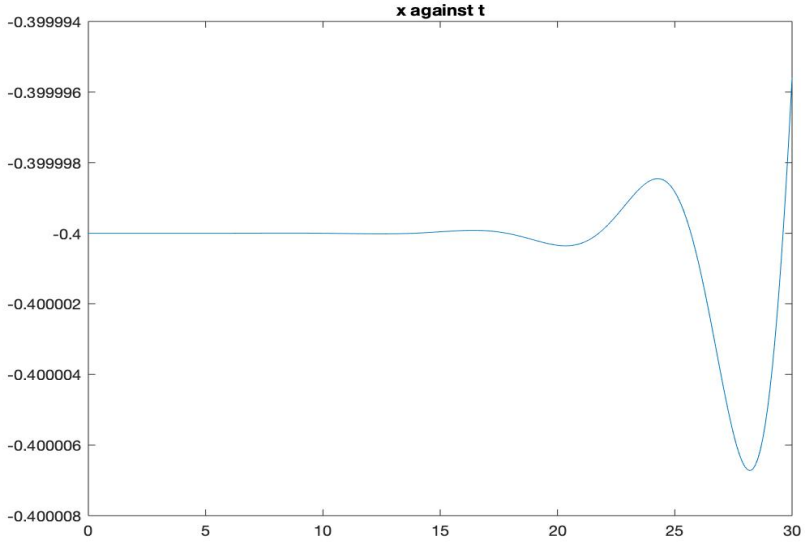

y against t
0.8660254055
0.866025405
0.8660254045
0.866025404
0.8660254035
0.866025403
0.8660254025
0
5
10
15
20
25
30

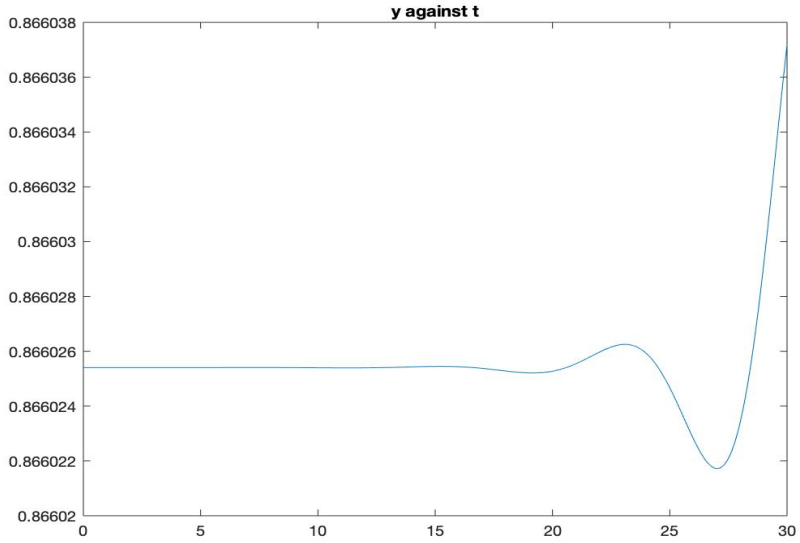

trajectory
0.8660254055
0.866025405
0.8660254045
0.866025404
0.8660254035
0.866025403
0.8660254025
-0.475000002
-0.475000001
-0.475
-0.474999999
-0.474999998

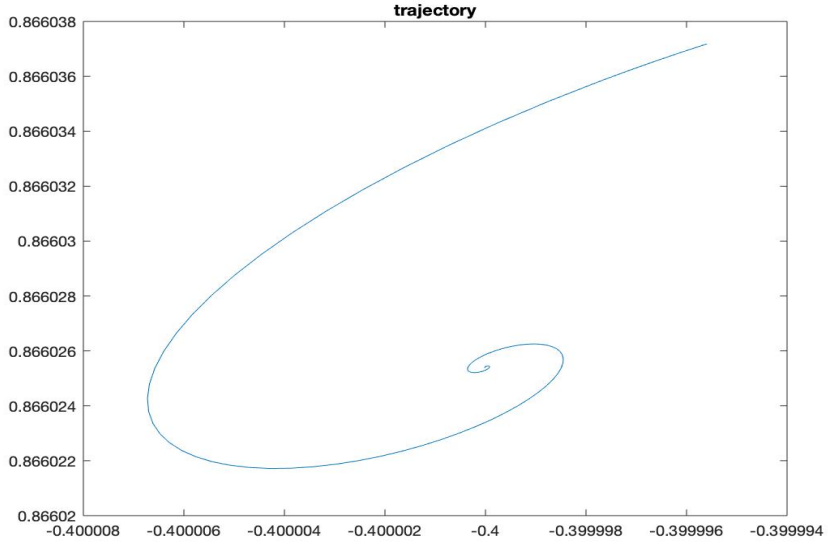

x against t
-0.4499999
-0.44999995
-0.45
-0.45000005
-0.4500001
-0.45000015
0
5
10
15
20
25
30

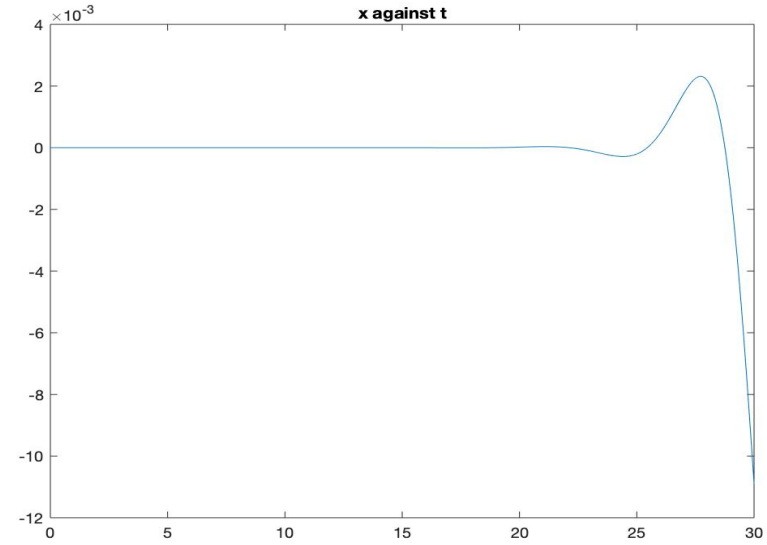

y against t
0.86602546
0.86602544
0.86602542
0.8660254
0.86602538
0.86602536
0.86602534
0.86602532
0.8660253
0.86602528
0
5
10
15
20
25
30

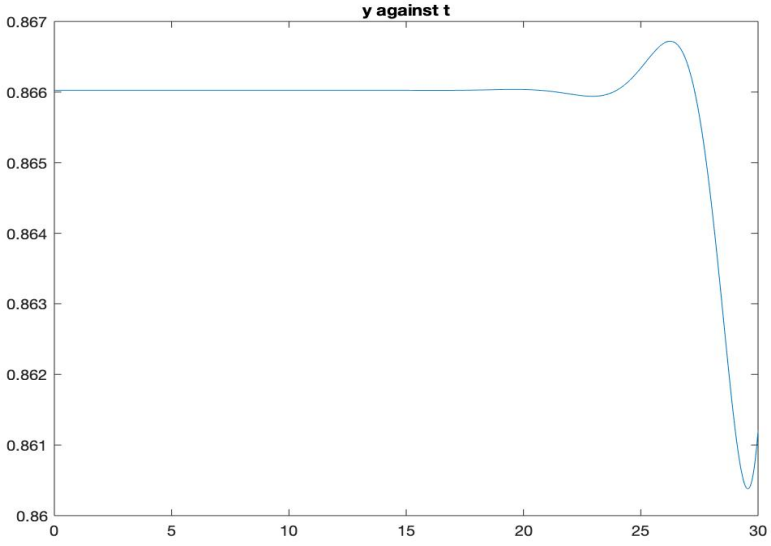

trajectory
0.86602546
0.86602544
0.86602542
0.8660254
0.86602538
0.86602536
0.86602534
0.86602532
0.8660253
0.86602528
-0.45000015
-0.4500001
-0.45000005
-0.45
-0.44999995
-0.4499999

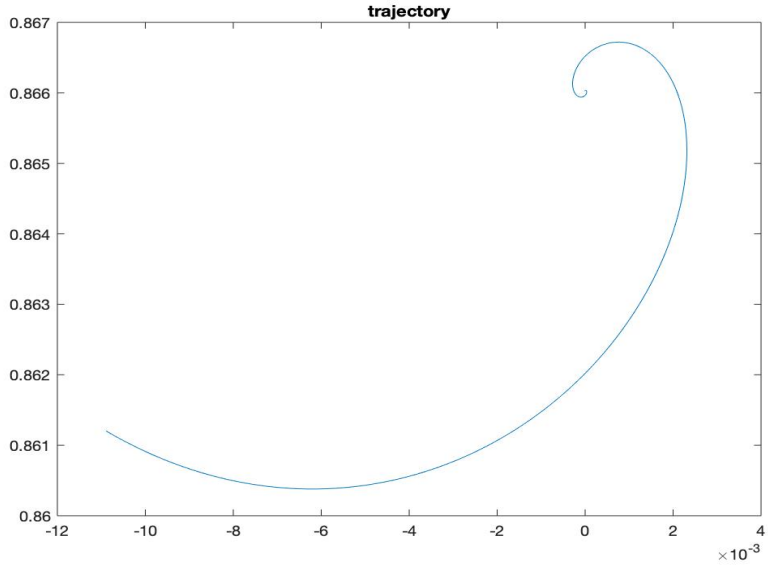

x against t
-0.399994
-0.399996
-0.399998
-0.4
-0.400002
-0.400004
-0.400006
-0.400008
0
5
10
15
20
25
30


y against t
0.866038
0.866036
0.866034
0.866032
0.86603
0.866028
0.866026
0.866024
0.866022
0.86602
0
5
10
15
20
25
30

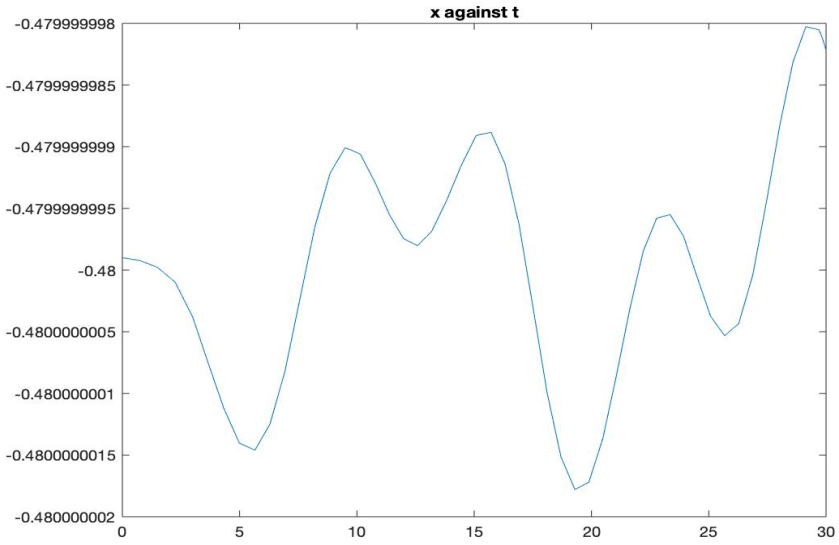

trajectory
0.866038
0.866036
0.866034
0.866032
0.86603
0.866028
0.866026
0.866024
0.866022
0.86602
-0.400008
-0.400006
-0.400004
-0.400002
-0.4
-0.399998
-0.399996
-0.399994

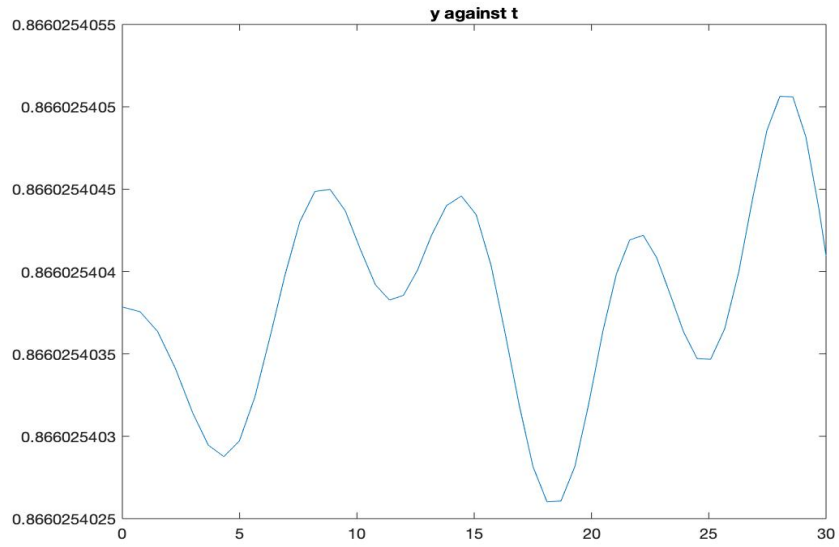

x against t
×10^-3
4
2
0
-2
-4
-6
-8
-10
-12
0
5
10
15
20
25
30

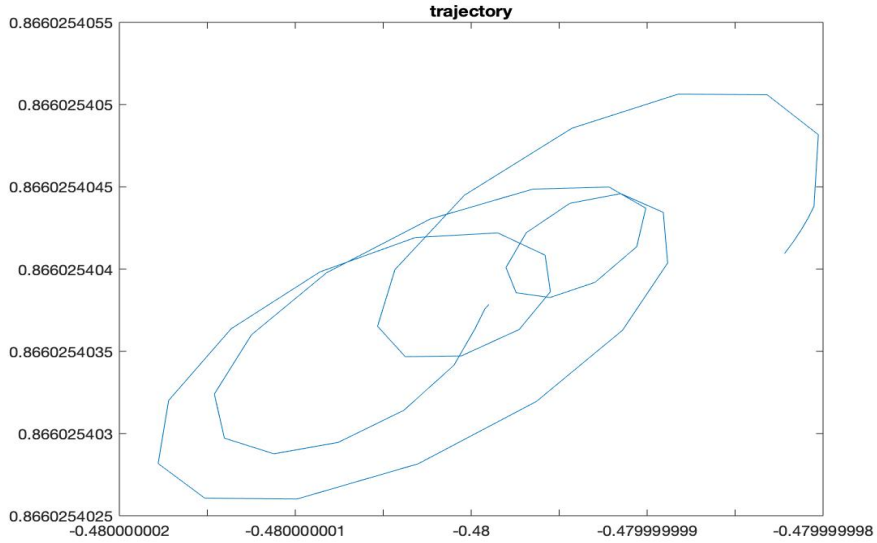

y against t
0.867
0.866
0.865
0.864
0.863
0.862
0.861
0.86
0
5
10
15
20
25
30

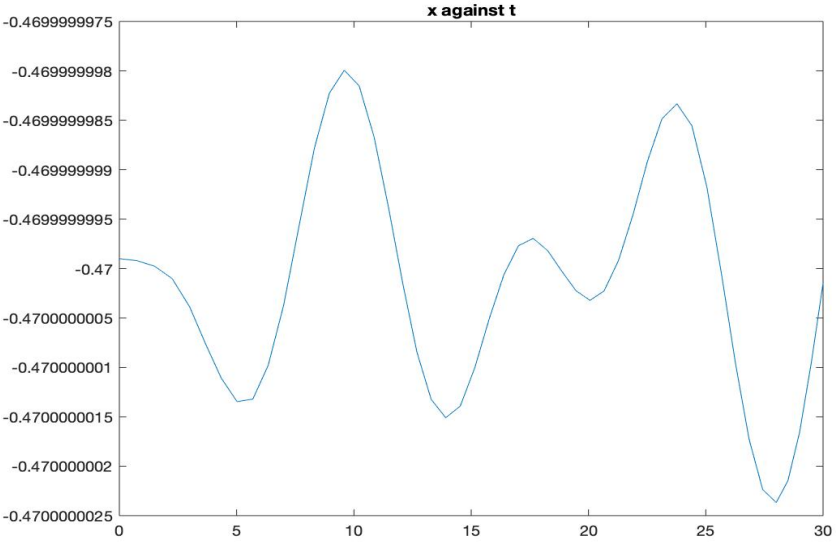

trajectory
0.867
0.866
0.865
0.864
0.863
0.862
0.861
0.86
-12
-10
-8
-6
-4
-2
0
2
4
×10^-3

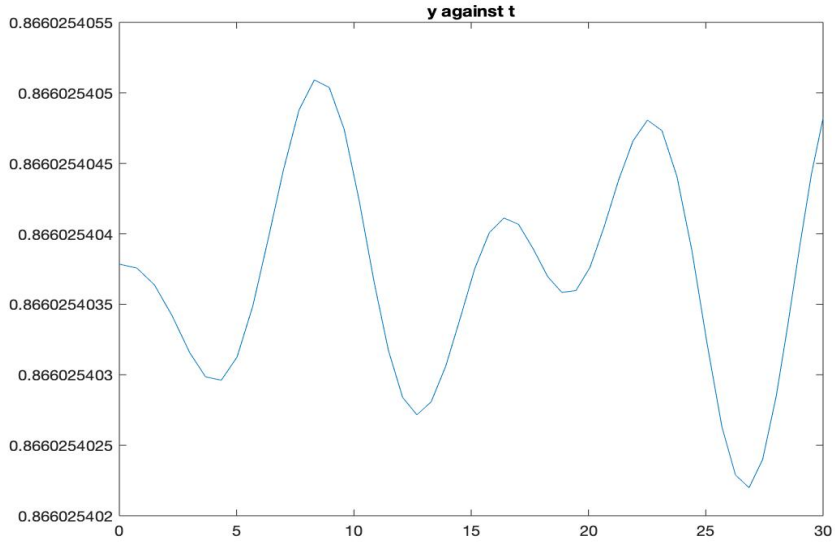
We then do further numerical investigation from 0.02 to 0.05, with step size of 0.01.From below you can see that $\mu_c \in (0.03, 0.04)$

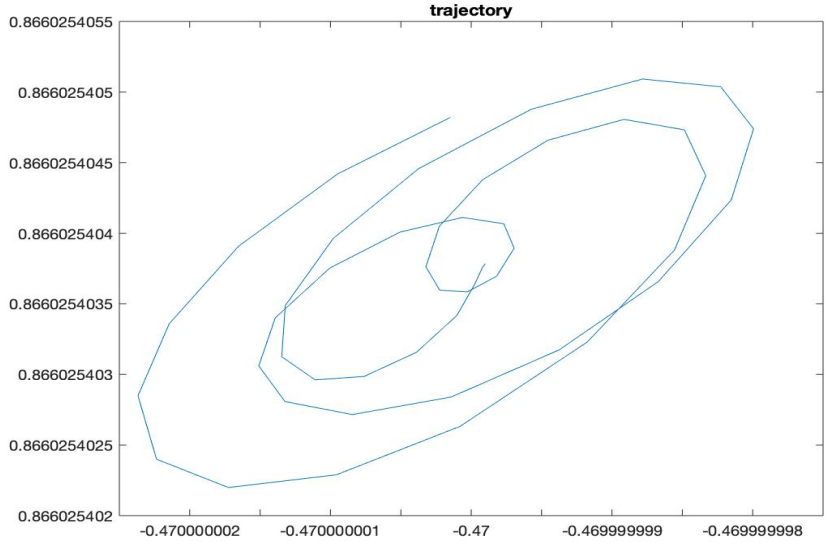


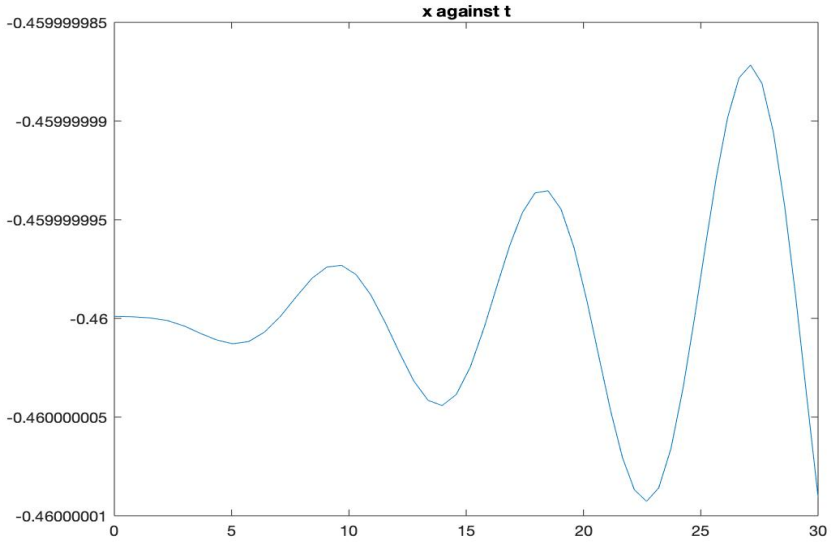


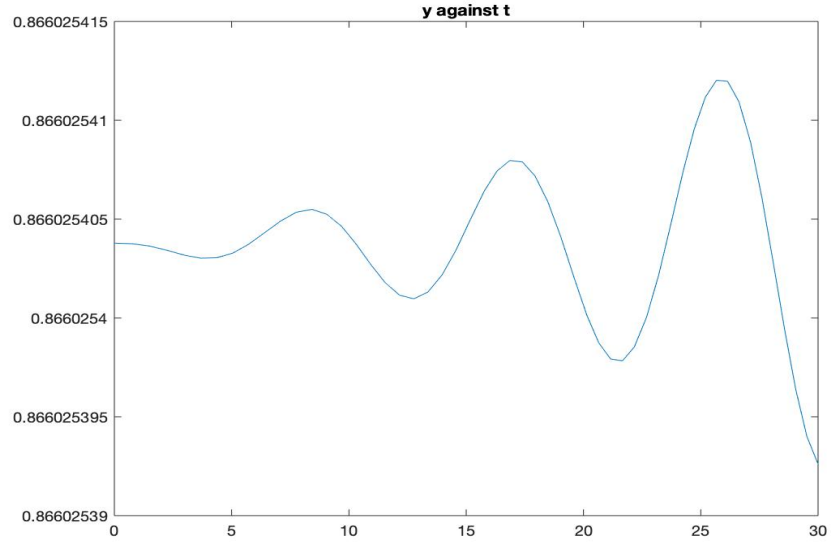

trajectory
0.8660254055
0.866025405
0.8660254045
0.866025404
0.8660254035
0.866025403
0.8660254025
-0.480000002
-0.480000001
-0.48
-0.479999999
-0.479999998

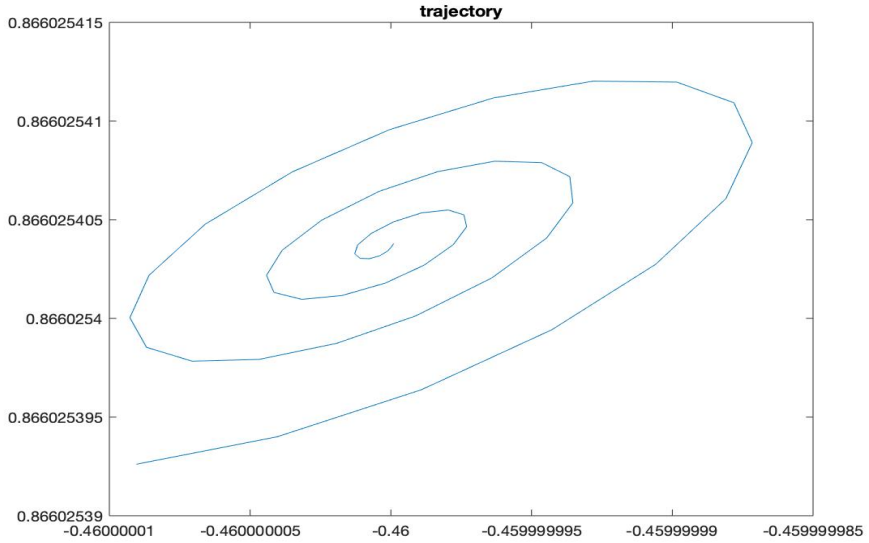

x against t
-0.4699999975
-0.469999998
-0.4699999985
-0.469999999
-0.4699999995
-0.47
-0.4700000005
-0.470000001
-0.4700000015
-0.470000002
-0.4700000025
0
5
10
15
20
25
30

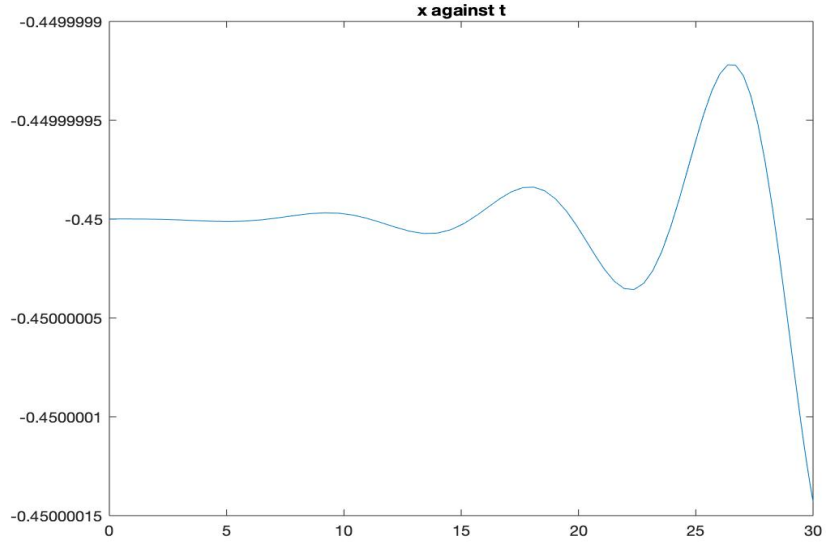

y against t
0.8660254055
0.866025405
0.8660254045
0.866025404
0.8660254035
0.866025403
0.8660254025
0.866025402
0
5
10
15
20
25
30

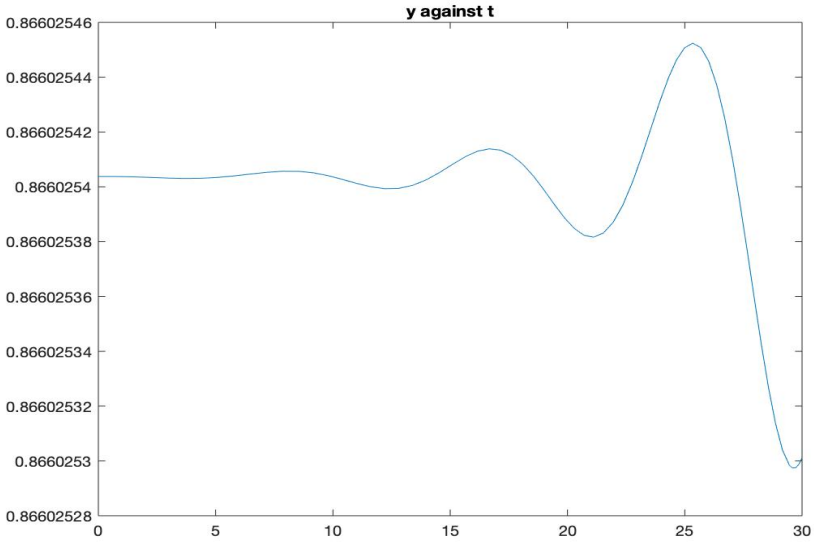

trajectory
0.8660254055
0.866025405
0.8660254045
0.866025404
0.8660254035
0.866025403
0.8660254025
0.866025402
-0.470000002
-0.470000001
-0.47
-0.469999999
-0.469999998

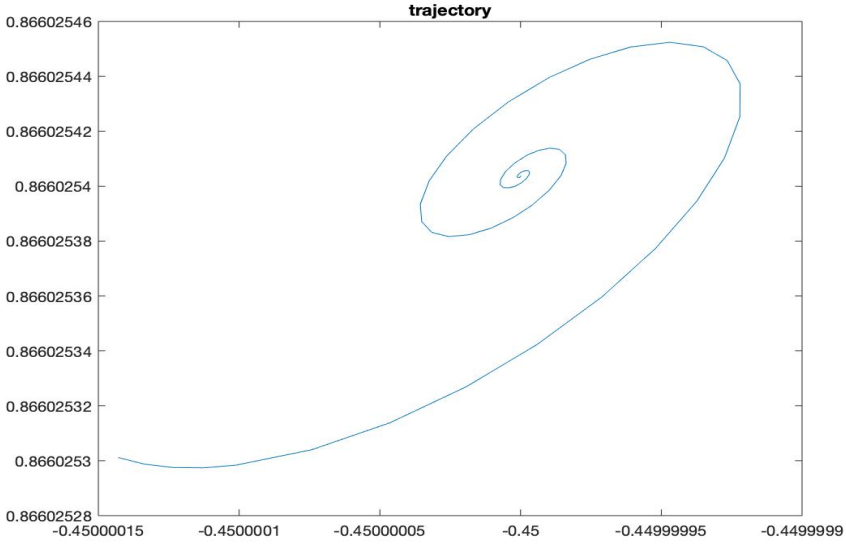

x against t
-0.459999985
-0.45999999
-0.459999995
-0.46
-0.460000005
-0.46000001
0
5
10
15
20
25
30


y against t
0.866025415
0.86602541
0.866025405
0.8660254
0.866025395
0.86602539
0
5
10
15
20
25
30


trajectory
0.866025415
0.86602541
0.866025405
0.8660254
0.866025395
0.86602539
-0.46000001
-0.460000005
-0.46
-0.459999995
-0.45999999
-0.459999985


x against t
-0.4499999
-0.44999995
-0.45
-0.45000005
-0.4500001
-0.45000015
0
5
10
15
20
25
30


y against t
0.86602546
0.86602544
0.86602542
0.8660254
0.86602538
0.86602536
0.86602534
0.86602532
0.8660253
0.86602528
0
5
10
15
20
25
30


trajectory
0.86602546
0.86602544
0.86602542
0.8660254
0.86602538
0.86602536
0.86602534
0.86602532
0.8660253
0.86602528
-0.45000015
-0.4500001
-0.45000005
-0.45
-0.44999995
-0.4499999

Calculate the second derivatives of Ω at equilateral Lagrange points, we have the following:

$$\Omega_{xx} = -\frac{3}{4}$$

$$\Omega_{xy} = \Omega_{yx} = \pm\frac{3\sqrt{3}}{2}(\frac{1}{2} - \mu)$$

$$\Omega_{yy} = -\frac{9}{4}$$

To investigate the stability of the Lagrange points, we perturb ξ and η in the x and y directions separately. Then we have the matrix:

$$\begin{pmatrix} \dot{\xi} \\ \dot{\eta} \\ \ddot{\xi} \\ \ddot{\eta} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \Omega_{xx} & \Omega_{xy} & 0 & 2 \\ \Omega_{xy} & \Omega_{yy} & -2 & 0 \end{pmatrix}$$

So we can see the matrix has 0 trace, and the sum of eigenvalues is 0. Calculate the characteristic equation : $x^4 + x^2 + \frac{27}{4}\mu(1-\mu)$. The eigenvalues come in conjugate pairs, and since their sum is 0, the real parts of eigenvalues sum to 0, and therefore one must be positive if the other is negative. In order to have stability, we do not want a positive real part which leads to exponential growth, so all roots must be purely imaginary. Then the quadratic equation $x^2 + x + \frac{27}{4}\mu(1-\mu)$ must have two real negative roots, hence the determinant of it must be non-negative. No further restriction is needed as the sum of roots is -1, the product is $\frac{27}{4}\mu(1-\mu) > 0$ as $mu \in (0, 0.5]$. Hence the critical value satisfies

$$1 - 27\mu_c(1-\mu_c) = 0,\ \mu_c = 0.038521...$$

After executing numerical experimentation we can justify the result. For the stable cases, as we have two real eigenvalues, we are expected to have periodic motion, which is seen from some graphs of x,and y against t previously.

4 Question 6

The distribution of the Trojans are consistent with my findings above, as μ is below the critical value, both equilateral Lagrange points are stable. No analogue is observed in Earth-Moon system because the gravitational forces from the sun and other planets of the system are significant, while in the Solar-Jupiter system the distance is large and both are massive.

A Codes

A.1 Question 1

```
function dydt = Dynsyst(y,k)
%Dynsyst defines the dynamical system.
%
a = (y(1)+1-k)^2+y(2)^2;
b = (y(1)-k)^2+y(2)^2;
z = -y(1)+k*(y(1)+1-k)*a^(-3/2)+(1-k)*(y(1)-k)*b^(-3/2);
w = -y(2)+k*y(2)*a^(-3/2)+(1-k)*y(2)*b^(-3/2);
dydt = [y(3); y(4); 2*y(4)-z; -2*y(3)-w];

function [t,y] = Solver1(mu,t_0,y_0)
%Solver1 solves the system numerically .
%          Here we used the function Dynsyst()
%          which defines the Dynamical System.
options = odeset('Reltol',1e-10,'AbsTol',1e-10);
[t,y] = ode45(@(t,y) Dynsyst(y,mu),t_0,y_0,options);
end

function [J] = Check(X,mu)
%Const calculates the constant of motion J

x=X(:,1);
y=X(:,2);
xdot=X(:,3);
ydot=X(:,4);
a=sqrt(((x+1-mu).^2)+(y.^2));
b=sqrt(((x-mu).^2)+(y.^2));
J = (-1/2)*((x.^2)+(y.^2)) -((1-mu)./b)-(mu./a)+(1/2)*((xdot.^2)+(ydot.^2)
end
```

A.2 Question 2

```
function dydt = ModDynsyst(y,k)
%Modified defines the dynamical system.
%   Here we set (dx/dt, dy/dt) = (y(3), y(4)), mu = k
a = (y(1)-k)^2+y(2)^2;
z = (1/2)*(y(1)-k)*a^(-3/2);
w = (1/2)*y(2)*a^(-3/2);
```

```
dydt = [y(3); y(4); 2*y(4)-z; -2*y(3)-w];

function [t,y] = Modsolver(k,t_0,y_0)
%Modsolver solves the system in Question 2 numerically.
%          Here we use function Modified defined previously.
options = odeset('Reltol',1e-10,'AbsTol',1e-10);
[t,y] = ode45(@(t,y) ModDynsyst(y,k),t_0,y_0,options);
end

%These codes show the position of P_h,
%   and draw the trajectory of the third object
%    corresponding to the numerical solution of the
%    dynamical system with specified initial conditions
a = 0.01;
[t,y] = Modsolver(0.5,[0,10],[0.5+a;0;0;a*(-1+(1+1/(2*a^3))^(1/2))]);
plot(y(:,1),y(:,2),'DisplayName','trajectory')
hold on
scatter(0.5,0,'filled','Displayname','P_h');
legend
xlabel('x');
ylabel('y');
title('Trajectory of the thied object sufficiently close to P_h');
```

A.3 Question 3

```
function [J] = Const(X,mu)
%Const calculates the constant of motion J

x=X(1);
y=X(2);
xdot=X(3);
ydot=X(4);
a=sqrt(((x+1-mu).^2)+(y.^2));
b=sqrt(((x-mu).^2)+(y.^2));
J =(-1/2)*((x.^2)+(y.^2)) -((1-mu)./b)-(mu./a)+(1/2)*((xdot.^2)+(ydot.^2))
end

function [Omega] = Potent(X,mu)
%Const calculates the potential of motion J

x=X(1);
```

```
y=X(2);
z=X(3);
w=X(4);
a=sqrt(((x+1-mu).^2)+(y.^2));
b=sqrt(((x-mu).^2)+(y.^2));
Omega =(-1/2)*((x.^2)+(y.^2)) -((1-mu)./b)-(mu./a);
end

function [] = Fob_Reg(v_0,mu)
J = Const([0.32,0,0,v_0],mu);
x = -1.3:.001:1.3;
y = -1.3:.001:1.3;
Z = NaN(length(x),length(y));
for a=1:length(x)
    for b=1:length(y)
        X=[x(a),y(b),0,0];
        Omega = Potent(X,mu);
        if (Omega>J)
            Z(a,b)=1;
        end
    end
end
contourf(x,y,Z',[1,1],'DisplayName','Forbidden Region boundary')
end


V_0 = [-1,-1.5,-1.73,-1.78,-1.853];
for i = 1: length(V_0)
    str = strcat('v_0 = ',num2str(V_0(i)));
    [t,y] = Solver1(0.5,[0,30],[0.32;0;0;V_0(i)]);
    figure(i)
    plot(y(:,1),y(:,2),'DisplayName','trajectory')
    title(str)
    hold on
    scatter(-0.5,0,'fill','DisplayName','P_l')
    scatter(0.5,0,'fill','DisplayName','P_h')
    Fob_Reg(V_0(i),0.5)
    legend
end
```

A.4 Question 4

```
function [] = Contour(X,mu)
J = Const(X,mu);
x = -1.3:.001:1.3;
y = -1.3:.001:1.3;
Z = NaN(length(x),length(y));
for a=1:length(x)
    for b=1:length(y)
        X=[x(a),y(b),0,0];
        Omega = Potent(X,mu);
        if (Omega>J)
            Z(a,b)=Omega;
        end
    end
end
contourf(x,y,Z','DisplayName','Forbidden Region boundary')
end

function [] = LaPt(mu)
%LaPt finds and draws the Lagrangian points of the system
func = @(x) -x+(1-mu)*(x-mu)/(abs(x-mu)^3)+mu*(x+1-mu)/(abs(x+1-mu)^3);
x = linspace(-2,2);
y = [];
for i = 1:length(x)
    z = fzero(func,x(i));
    if (abs(func(z)) < 1)
        y(length(y)+1) = round(z,4);
    end
end
y = unique(y);
scatter(y(1),0,'fill','DisplayName','LP1')
hold on
scatter(y(2),0,'fill','Displayname','LP2')
hold on
scatter(y(3),0,'fill','DisplayName','LP3')
hold on
scatter(mu-1/2,sqrt(3)/2,'fill','DisplayName','LP4')
hold on
scatter(mu-1/2,-sqrt(3)/2,'fill','DisplayName','LP5')
```

```
legend

opts = [0.1,0.5,0.7];
X = [1,0,0,0];
for i = 1:length(opts)
    figure(i)
    Contour(X,opts(i))
    hold on
    LaPt(opts(i))
end

x = [-1.1232,0.2861,1.2567];
delta = 0.01;
for i = 1:length(x)
    [t,y] = Solver1(0.7,[0,30],[x(i)+delta;0;0;1]);
    x_gainst_t = figure();
    plot(t,y(:,1))
    title('x against t')

    y_against_t = figure();
    plot(t,y(:,2))
    title('y against t')

    trajectory = figure();
    plot(y(:,1),y(:,2))
    title('trajectory')
end
```

A.5 Question 5

```
function [] = Equi(mu)
x = mu-1/2;
delta = 1e-10;
[t,y] = Solver1(mu,[0,30],[x+delta;sqrt(3)/2;0;0]);
x_gainst_t = figure();
    plot(t,y(:,1))
    title('x against t')
y_against_t = figure();
    plot(t,y(:,2))
    title('y against t')
trajectory = figure();
```

```
    plot(y(:,1),y(:,2))
    title('trajectory')
end

opts = [0.01,0.025,0.05,0.1,0.5];
for i = 1:length(opts)
    Equi(opts(i))
end

opts = [0.02,0.03,0.04,0.05];
for i = 1:length(opts)
    Equi(opts(i))
end
```