Project: 2.1 The Restricted Three-Body Problem

May 1, 2018

1. **Show from (1a) and (1b) that the quantity**
$$J = \frac{1}{2}\dot{x}^2 + \frac{1}{2}\dot{y}^2 + \Omega(x, y)$$
is constant following the motion. Deduce that trajectories must be confined to the region
$$\Omega(x, y) \le \Omega(x_0, y_0) + \frac{1}{2}u_0^2 + \frac{1}{2}v_0^2$$
where x_0, y_0, u_0 and v_0 are the initial values of x, y, $\dot{x}$ and $\dot{y}$, respectively.

J is constant following the motion if $\frac{dJ}{dt} = \dot{J} = 0$.
$$\begin{aligned}\dot{J} &= \dot{x}\ddot{x} + \dot{y}\ddot{y} + \frac{\partial\Omega}{\partial x}\dot{x} + \frac{\partial\Omega}{\partial y}\dot{y}\\ \dot{J} &= \cancel{\dot{x}\ddot{x}} + \cancel{\dot{y}\ddot{y}} + (2\dot{y} - \cancel{\ddot{x}})\dot{x} + (-2\dot{x} - \cancel{\ddot{y}})\dot{y}\\ \dot{J} &= 2\dot{y}\dot{x} - 2\dot{x}\dot{y} = 0.\end{aligned}$$

For the second part, we can form the following inequality, since $\frac{1}{2}\dot{x}^2 + \frac{1}{2}\dot{y}^2 \ge 0$ for any value of $\dot{x}$ and $\dot{y}$,
$$\begin{aligned}&\Omega(x, y) \le \Omega(x, y) + \frac{1}{2}\dot{x}^2 + \frac{1}{2}\dot{y}^2\\ &\Omega(x, y) \le J\\ &\Omega(x, y) \le \Omega(x_0, y_0) + \frac{1}{2}u_0^2 + \frac{1}{2}v_0^2.\end{aligned}$$
This is true because J is constant and hence must be the same at the beginning of the motion as it is at any point following the motion.

2. **Consider motion in the neighbourhood of P_2 say, so that the effect of P_1 may be ignored and (2)[1] may be approximated by**
$$\Omega = -\frac{1}{2r_2}. \tag{\textbf{4}}$$
Show that the system (1a), (1b) and (4) has analytic solutions with the spacecraft in a circular orbit of radius a about P_2, where a has any sufficiently small value for which the approximation (4) is valid.

Modify your program to solve (1a) and (1b) with Ω specified by (4) instead of (2). Demonstrate, for one value of a, that the modified program can accurately reproduce the analytic solutions.

Using approximation **(4)**, the differential equations **(1a)** and **(1b)** become:
$$\ddot{x} - 2\dot{y} = -\frac{\partial\Omega}{\partial x} = \frac{\partial}{\partial x}\left(\frac{1}{2r_2}\right) \tag{\textbf{1a}}$$
$$\ddot{y} + 2\dot{x} = -\frac{\partial\Omega}{\partial y} = \frac{\partial}{\partial y}\left(\frac{1}{2r_2}\right), \tag{\textbf{1b}}$$
where $r_2^2 = (x - 0.5)^2 + y^2$. The (most general) solution where the spacecraft is in a circular orbit of radius a about P_2 would be when $(x(t), y(t))$ is:
$$(x(t), y(t)) = (a\cos(\omega t + \phi) + 0.5, a\sin(\omega t + \phi)). \tag{1}$$
We can check this is a solution by substituting (1) into **(1a)** and **(1b)** respectively:
$$\ddot{x} - 2\dot{y} = \frac{\partial}{\partial x}\left(\frac{1}{2r_2}\right) = \frac{\partial}{\partial x}\left(\frac{1}{2}\Big((x-0.5)^2 + y^2\Big)^{-\frac{1}{2}}\right) = \frac{0.5 - x}{2\Big((x-0.5)^2 + y^2\Big)^{\frac{3}{2}}}$$
$$\begin{aligned}\Rightarrow \quad & -a\omega^2\cos(\omega t + \phi) - 2a\omega\cos(\omega t + \phi) &&= \frac{-a\cos(\omega t + \phi)}{2\Big(a^2\cos^2(\omega t + \phi) + \sin^2(\omega t + \phi)\Big)^{\frac{3}{2}}}\\ \Rightarrow \quad & \cos(\omega t + \phi)\big(-a\omega^2 - 2a\omega\big) &&= \frac{-a\cos(\omega t + \phi)}{2a^3}\\ \Rightarrow \quad & \cos(\omega t + \phi)\left(a\omega^2 + 2a\omega - \frac{1}{2a^2}\right) &&= 0.\end{aligned} \tag{2}$$

[1] All labels that are in bold reference the labeling within the project questions given to us. I have used non-bold text to reference any of my equations in my project and tried not to overlap with the project question labels.

$$\ddot{y} + 2\dot{x} = \frac{\partial}{\partial y}\left(\frac{1}{2r_2}\right) = \frac{\partial}{\partial y}\left(\frac{1}{2}\Big((x-0.5)^2+y^2\Big)^{-\frac{1}{2}}\right) = \frac{-y}{2\Big((x-0.5)^2+y^2\Big)^{\frac{3}{2}}}$$

$$\Rightarrow \quad -a\omega^2\sin(\omega t+\phi) - 2a\omega\sin(\omega t+\phi) = \frac{-a\sin(\omega t+\phi)}{2\Big(a^2\cos^2(\omega t+\phi)+\sin^2(\omega t+\phi)\Big)^{\frac{3}{2}}}$$

$$\Rightarrow \quad \sin(\omega t+\phi)\big(-a\omega^2-2a\omega\big) = \frac{-a\sin(\omega t+\phi)}{2a^3}$$

$$\Rightarrow \quad \sin(\omega t+\phi)\left(a\omega^2+2a\omega-\frac{1}{2a^2}\right) = 0. \tag{3}$$

Equations (2) and (3) indicate a common condition on the angular frequency, ω, of the spaceship as it orbits around the planet P_2. The other condition that $\sin(\omega t+\phi) = \cos(\omega t+\phi) = 0$ does not hold since $|\omega| > 0$ if the spaceship is orbiting around the planet. Moreover, the functions cos and sin can never be zero simultaneously when they have the same argument. The value ϕ is just an arbitrary constant which indicates where the orbit begins on the circle at a specific time, i.e. $t = 0$. We can solve the common condition above that:

$$\left(a\omega^2+2a\omega-\frac{1}{2a^2}\right) = 0$$

Solving the quadratic equation in ω gives us solutions:

$$\omega^2+2\omega-\frac{1}{2a^3} = 0 \Rightarrow \omega_\pm = \frac{-2\pm\sqrt{4+\frac{4}{2a^3}}}{2} = \frac{-\cancel{2}\pm\cancel{2}\sqrt{1+\frac{1}{2a^3}}}{\cancel{2}}$$

$$\Rightarrow \omega_\pm = -1\pm\sqrt{1+\frac{1}{2a^3}} \tag{5}$$

Hence there exists solutions $(x(t), y(t))$ for this simplified problem if and only if ω is equal to one of the two solutions above in (5). The general solutions for this problem that satisfy the conditions above are:

$$x(t) = a\cos\left(\left(-1\pm\sqrt{1+\tfrac{1}{2a^3}}\right)t+\phi\right)+0.5$$

$$y(t) = a\sin\left(\left(-1\pm\sqrt{1+\tfrac{1}{2a^3}}\right)t+\phi\right), \tag{6}$$

where ϕ is an arbitrary constant. These functions represents an orbit of radius a about P_2 (for two specific angular velocities, $\omega_\pm$) and hence we have shown that these are valid analytic solutions for the given conditions.

We can use my function `threebody2` along with the MATLAB ODE solver, `ode45`, to check this analytic solution for a given value of a. I will choose $a = 0.1$ which will force $\omega_\pm = -1\pm\sqrt{1+\frac{1}{2\times 0.1^3}}$. So we have $\omega_+ = -1+\sqrt{501}$ and $\omega_- = -1-\sqrt{501}$.

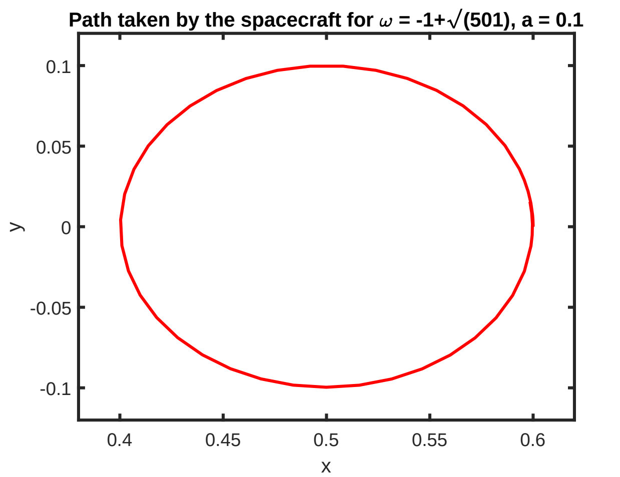

Figure 1: Diagram showing the path taken by the spaceship when $\omega = \omega_+ = -1+\sqrt{501}$. This is for $t \in [0, 0.3]$. The orbit is in an anti-clockwise direction.

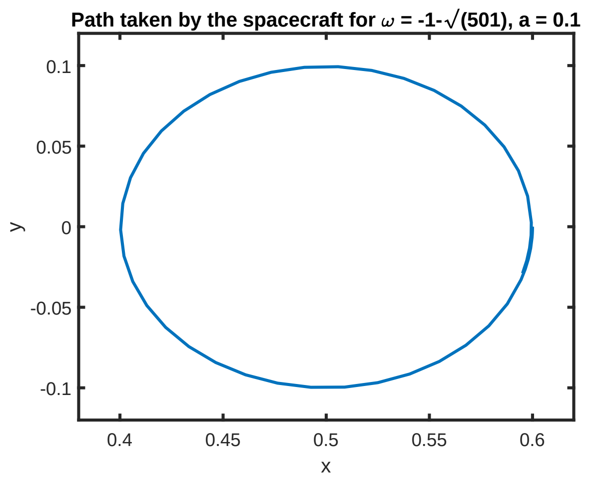

Figure 2: Diagram showing the path taken by the spaceship when $\omega = \omega_- = -1-\sqrt{501}$. This is for $t \in [0, 0.28]$. The orbit is in a clockwise direction.

Since ϕ is arbitrary, I chose $\phi = 0$ which corresponds to the spacecraft being at position $(a + 0.5, 0)$ at time $t = 0$. The graphs above indicate the solution is accurate since visually they reproduce an almost concentric circle, in both cases. I have let each program run for slightly more than the time period of the orbit ($T_{\pm} = \left|\frac{2\pi}{\omega_{\pm}}\right|$) to highlight this discrepancy more clearly. To further show the accuracy of the results, the graphs below indicate the magnitude of the difference between the actual analytic values (6) compared with the computed values from my program, for time $t \in [0, 0.3]$.

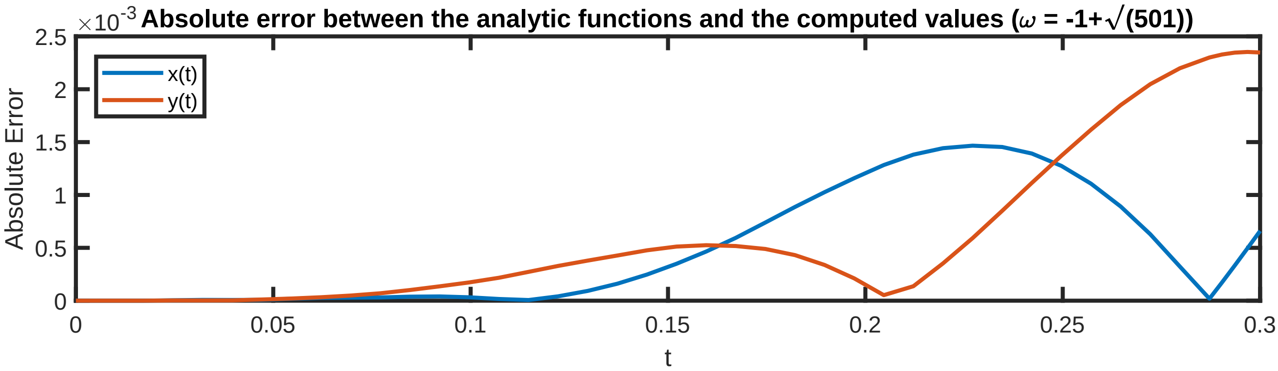

Figure 3: Graph showing how the absolute error in the computed values of $x(t)$ and $y(t)$ change for $\omega = -1 + \sqrt{501}$. The relative tolerance is 1×10^{-3}

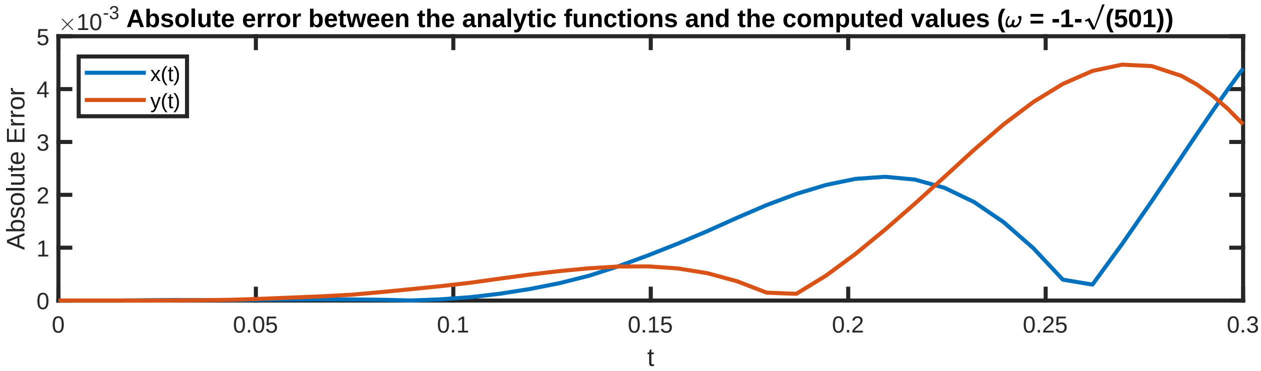

Figure 4: Graph showing how the absolute error in the computed values of $x(t)$ and $y(t)$ change for $\omega = -1 - \sqrt{501}$. The relative tolerance is 1×10^{-3}

Both of these graphs indicate that the absolute error in each case is of the order of 10^{-3} with $t \in [0, 0.3]$. This shows that the numerical solutions obtained are accurate to within a maximum of 5×10^{-3}, which is relatively accurate and hence is a good approximation for the trajectory of the orbit.

This question provides a check that my program works since it reproduces results very close to the analytic solutions solved. Furthermore, to verify the program, I increased the relative tolerance of the program (necessary for the accuracy of values that are not close to zero) to 1×10^{-6}. The results below indicate how increasing the relative tolerance reduces the error and hence the program becomes closer to the true value. This highlights how the program is accurate according to the differential equations it is solving.

The tolerance used in the initial graphs plotted above, Figure 3 and 4, use the default tolerance values [2] of the `ode45` function - which are: relative tolerance $= 1 \times 10^{-3}$, absolute tolerance $= 1 \times 10^{-6}$. The latter is of less significance for our purposes.

We can clearly see how the smaller tolerance used to produce Figures 5 and 6 made the results up to 1000 ($= 10^3$) times more accurate than the larger tolerance, equal to the factor between the two tolerances.

[2] https://uk.mathworks.com/help/matlab/ref/odeset.html

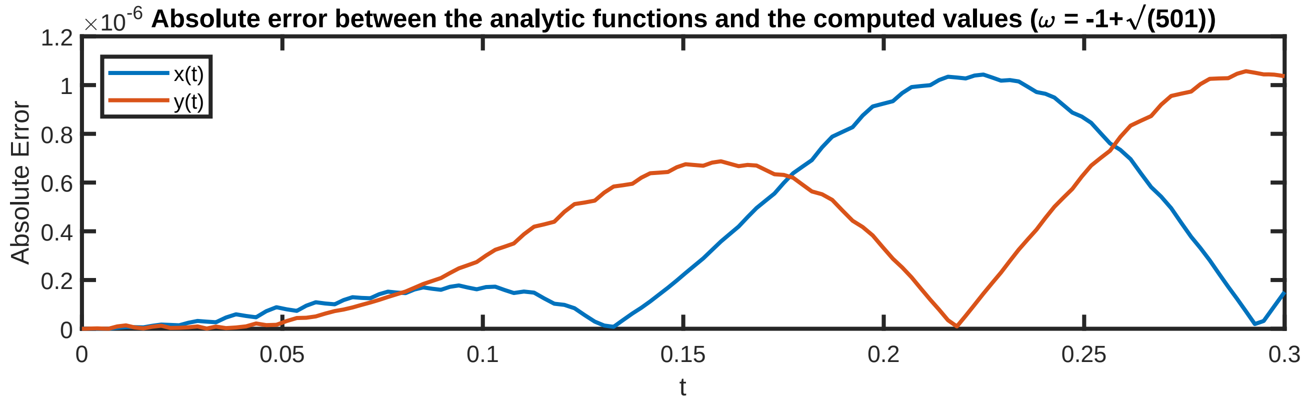

Figure 5: Graph showing how the absolute error in the computed values of $x(t)$ and $y(t)$ change for $\omega = -1 + \sqrt{501}$. The relative tolerance is 1×10^{-6}

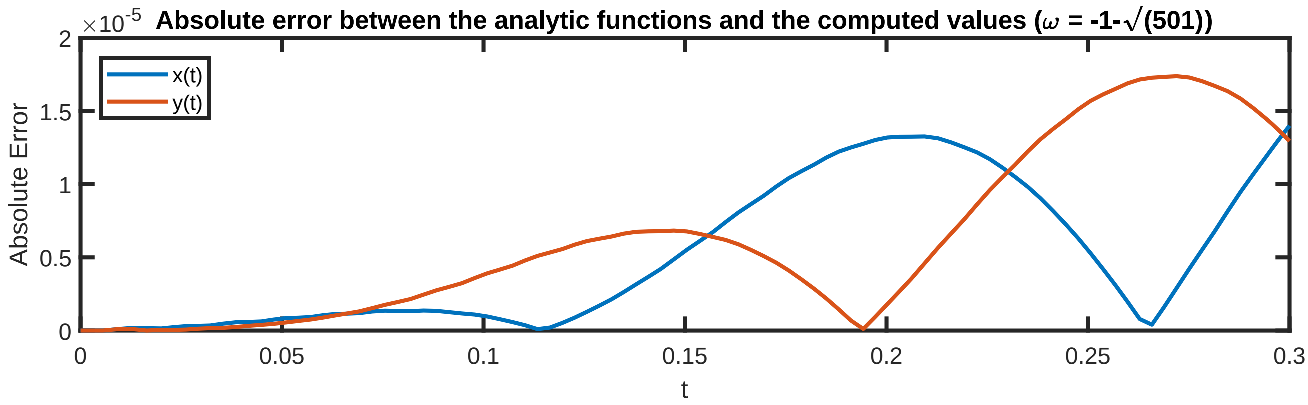

Figure 6: Graph showing how the absolute error in the computed values of $x(t)$ and $y(t)$ change for $\omega = -1 - \sqrt{501}$. The relative tolerance is 1×10^{-6}

For the final check to ensure the accuracy of my program, I plotted a numerically calculated J for each different analytic solution and also compared it at different relative tolerance values.

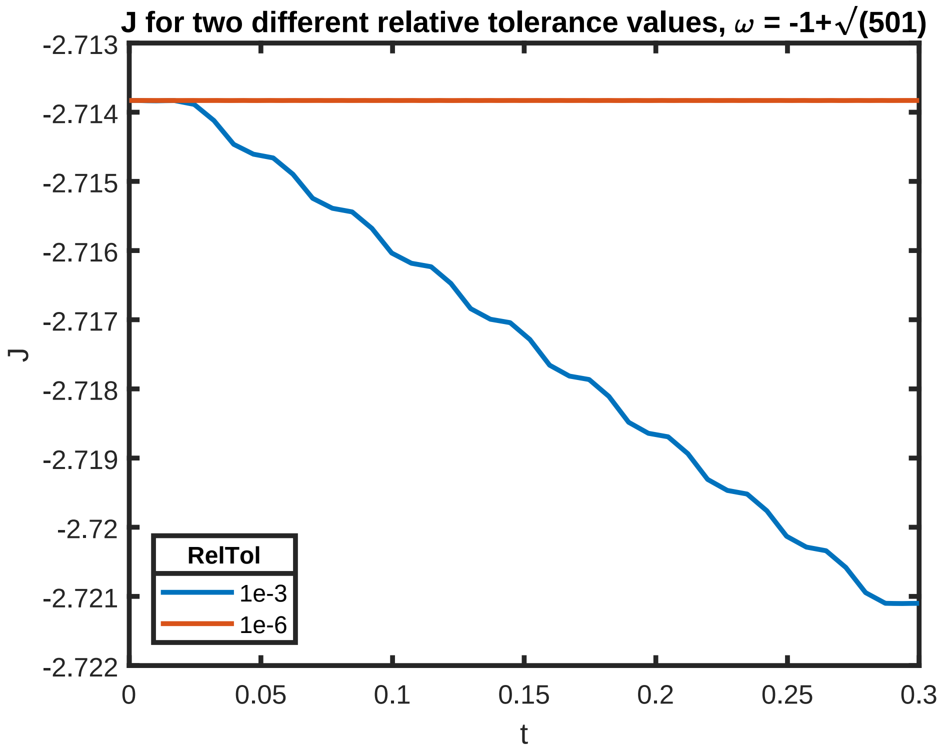

Figure 7: Graph showing how J varies following the motion for the case when $\omega = -1 - \sqrt{501}$, for two different relative tolerance values.

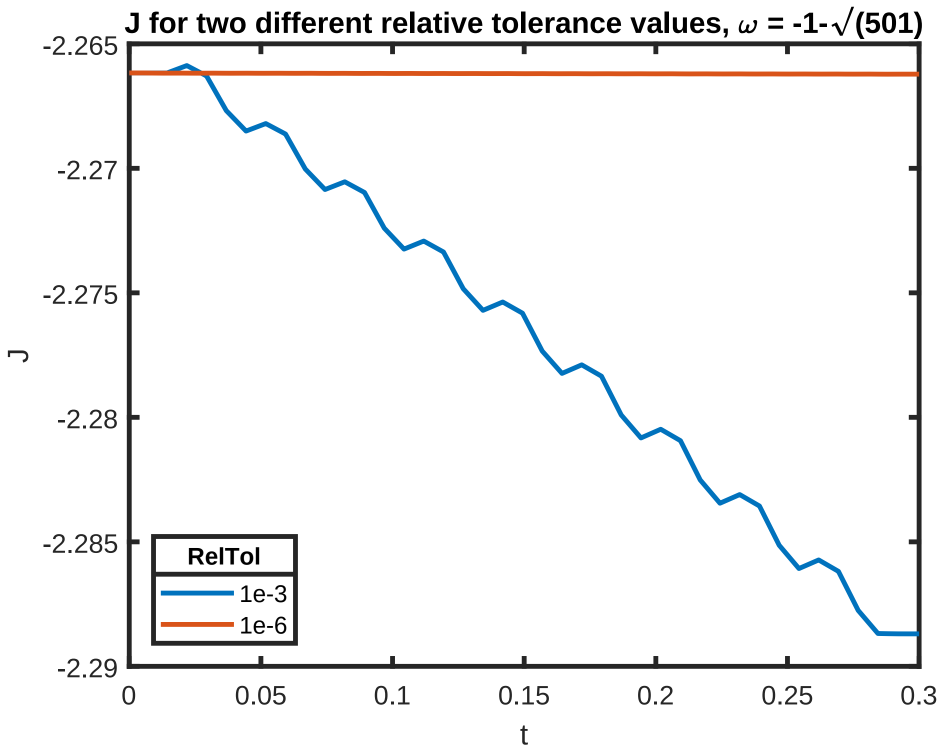

Figure 8: Graph showing how J varies following the motion for the case when $\omega = -1 - \sqrt{501}$, for two different relative tolerance values.

We can see that the value of J overall is constant to a particular number of significant figures. In both cases, the value of J has a much more smaller variation for when the relative tolerance is smaller. This highlights how reducing the relative tolerance leads to greater accuracy of results.

3. **Now return to the original system (1a), (1b) and (2), with $\mu = 0.5$, and take initial conditions**

$x = 0.5, y = 0.2, \dot{x} = u_0, \dot{y} = 0$ **with** $u_0 = -1.0, -1.39, -1.53, -1.54, -1.69, -1.7$ **and** -1.9 **in turn. For each case, use your programs to integrate from** $t = 0$ **to** $t = 30$**.**

Note, the tolerance values used in this question are as follows: Relative tolerance = 1×10^{-5}, Absolute tolerance = 1×10^{-5}. This enabled the program to prvide accurate results without compromising too much on the time taken to obtain the results.

(I) For $u_0 = -1.0$:

(i) Plot x against t:

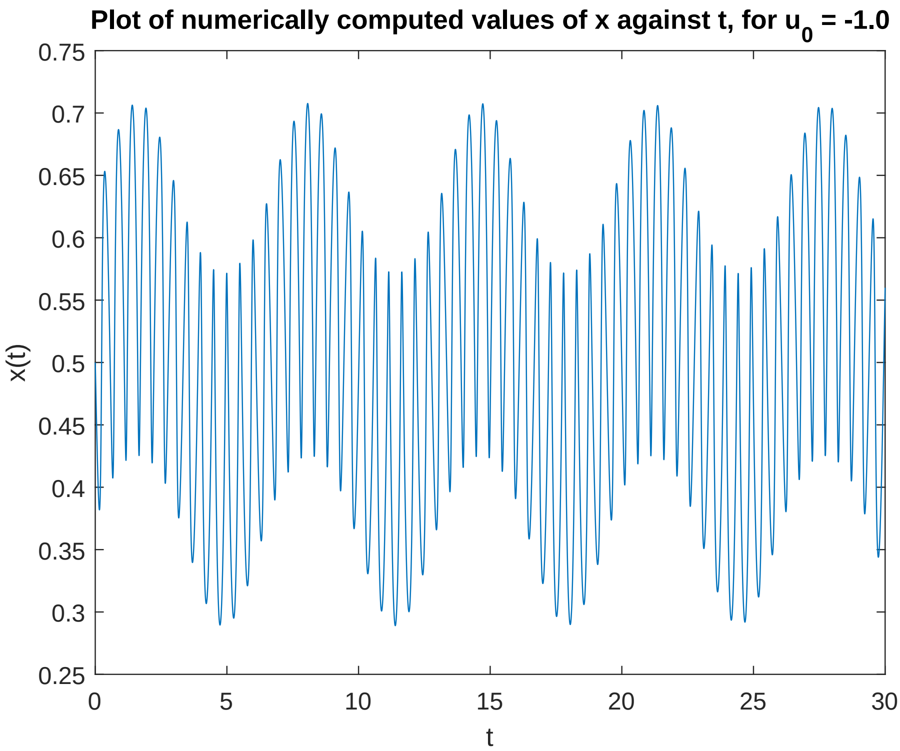

Figure 9: Plot of $x(t)$, computed by `ode45`, against $t \in [0, 30]$ for $u_0 = -1.0$.

(ii) Display the trajectory in the (x, y) plane, and the forbidden region $\Omega(x, y) > J = -2.7603$:

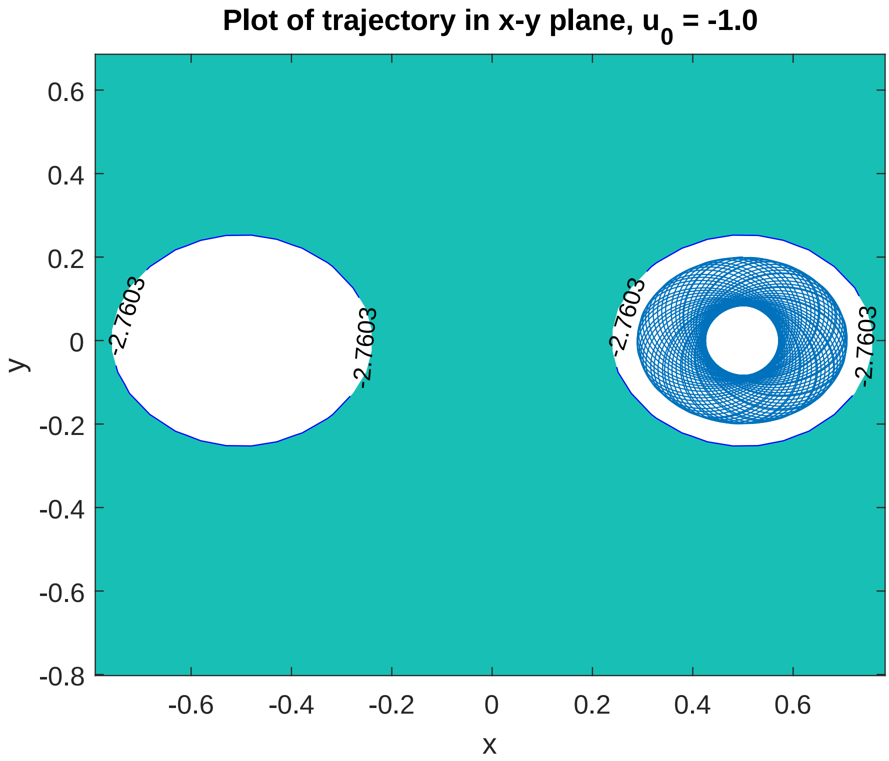

Figure 10: Plot of trajectory in the x-y plane, computed by `ode45`. The shaded region represents the forbidden region for which $\Omega(x, y) > J = -2.7603$. This motion is for time $t \in [0, 30]$ and initial speed $u_0 = -1.0$.

(iii) State the values of x and y at $t = 30$, giving a reasoned assessment on their accuracy*. For initial speed $u_0 = -1.0$, at $t = 30$, the values of (x, y) are:

$$x(30) = 0.5599 \text{ and } y(30) = -0.1222$$

(II) For $u_0 = -1.39$:

(i) Plot x against t:

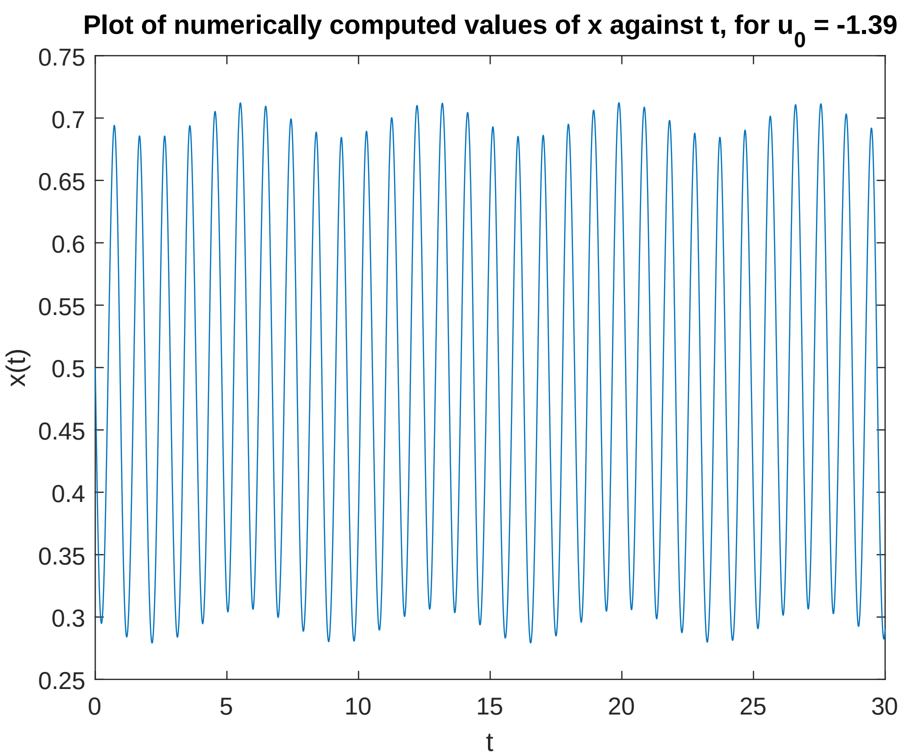

Figure 11: Plot of $x(t)$, computed by `ode45`, against $t \in [0, 30]$ for $u_0 = -1.39$.

(ii) Display the trajectory in the (x, y) plane, and the forbidden region $\Omega(x, y) > J = -2.2942$:

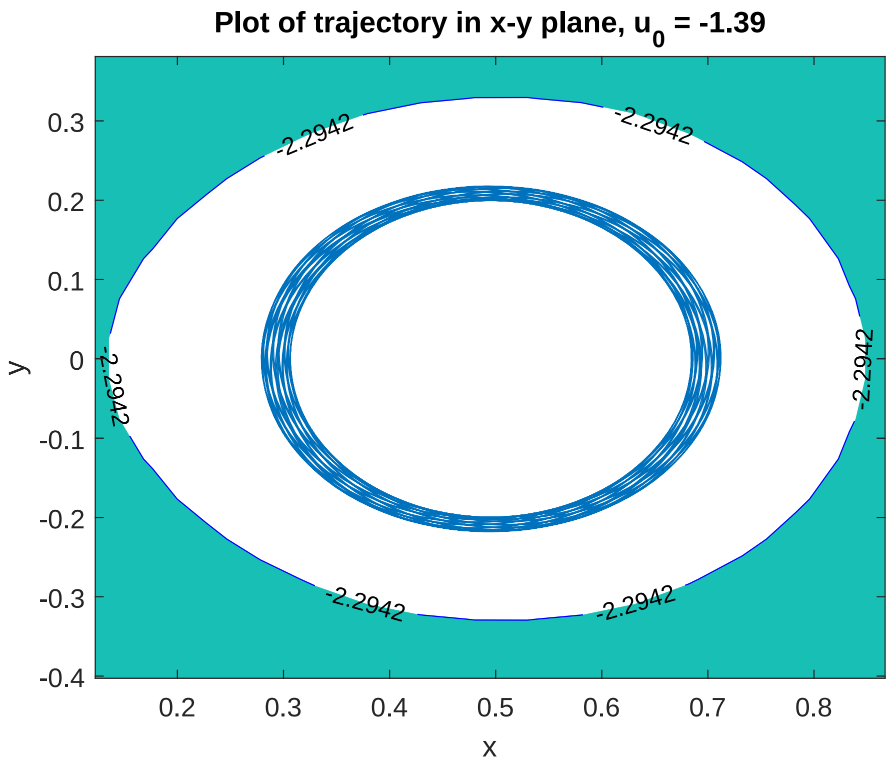

Figure 12: Plot of trajectory in the x-y plane, computed by `ode45`. The shaded region represents the forbidden region for which $\Omega(x, y) > J = -2.2942$. This motion is for time $t \in [0, 30]$ and initial speed $u_0 = -1.39$.

(iii) State the values of x and y at $t = 30$, giving a reasoned assessment on their accuracy*. For initial speed $u_0 = -1.39$, at $t = 30$, the values of (x, y) are:

$$x(30) = 0.2909 \text{ and } y(30) = -0.0674$$

(III) For $u_0 = -1.53$:

(i) Plot x against t:

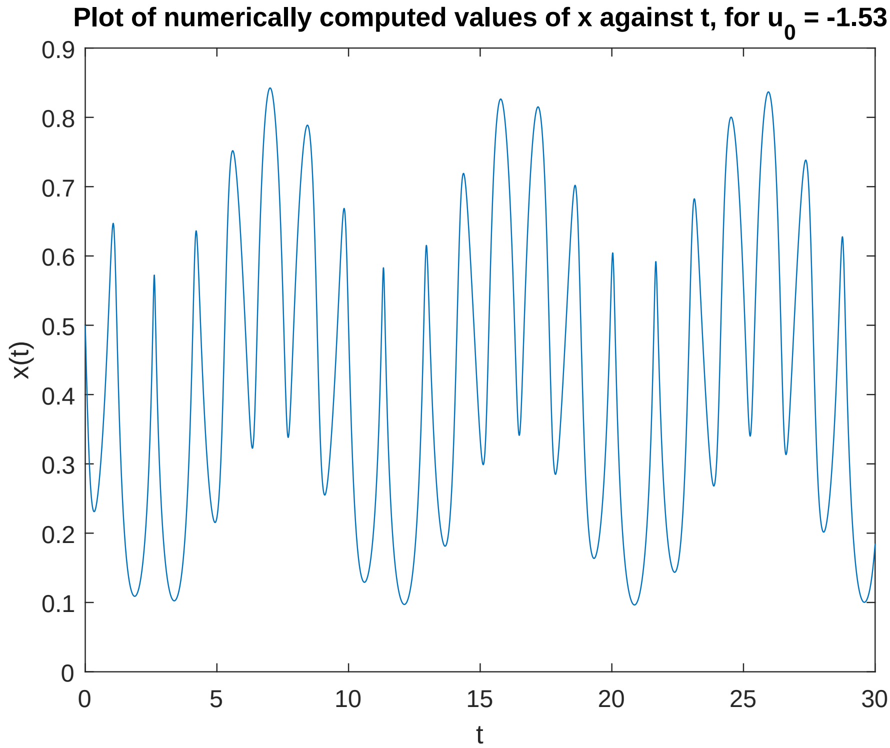

Figure 13: Plot of $x(t)$, computed by `ode45`, against $t \in [0, 30]$ for $u_0 = -1.53$.

(ii) Display the trajectory in the (x, y) plane, and the forbidden region $\Omega(x, y) > J = -2.0898$:

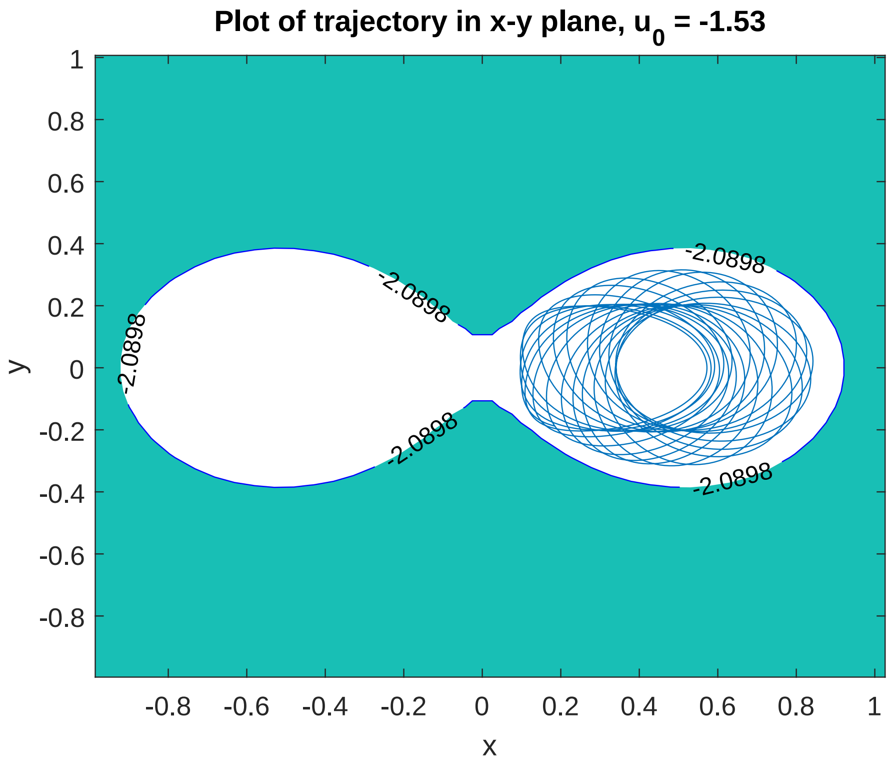

Figure 14: Plot of trajectory in the x-y plane, computed by `ode45`. The shaded region represents the forbidden region for which $\Omega(x, y) > J = -2.0898$. This motion is for time $t \in [0, 30]$ and initial speed $u_0 = -1.53$.

(iii) State the values of x and y at $t = 30$, giving a reasoned assessment on their accuracy*. For initial speed $u_0 = -1.53$, at $t = 30$, the values of (x, y) are:

$$x(30) = 0.1850 \text{ and } y(30) = -0.1880$$

(IV) For $u_0 = -1.54$:

(i) Plot x against t:

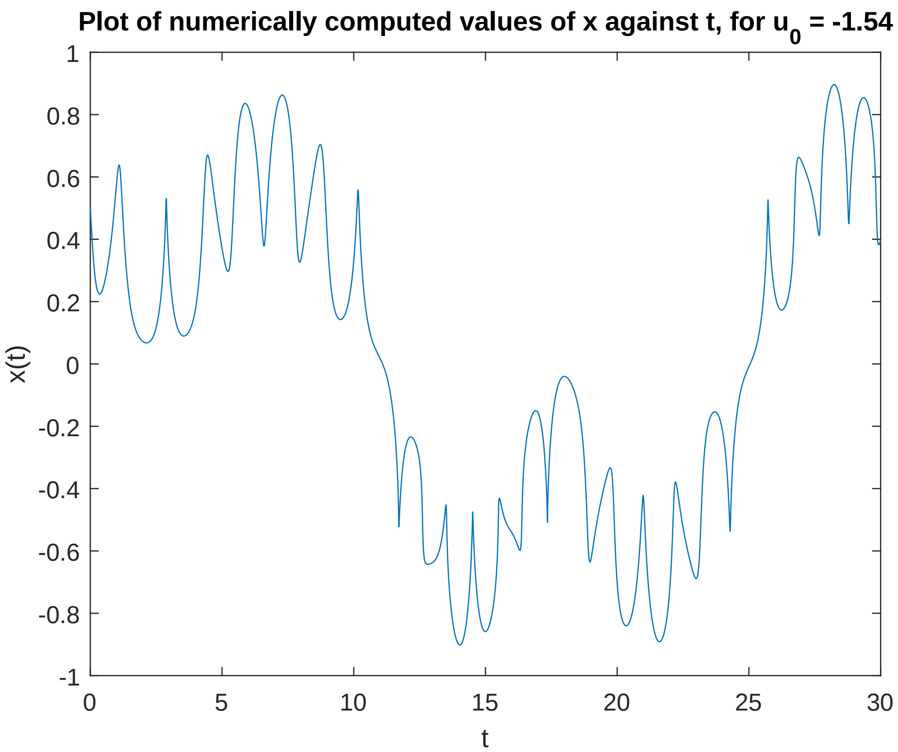

Figure 15: Plot of $x(t)$, computed by `ode45`, against $t \in [0, 30]$ for $u_0 = -1.54$.

(ii) Display the trajectory in the (x, y) plane, and the forbidden region $\Omega(x, y) > J = -2.0745$:

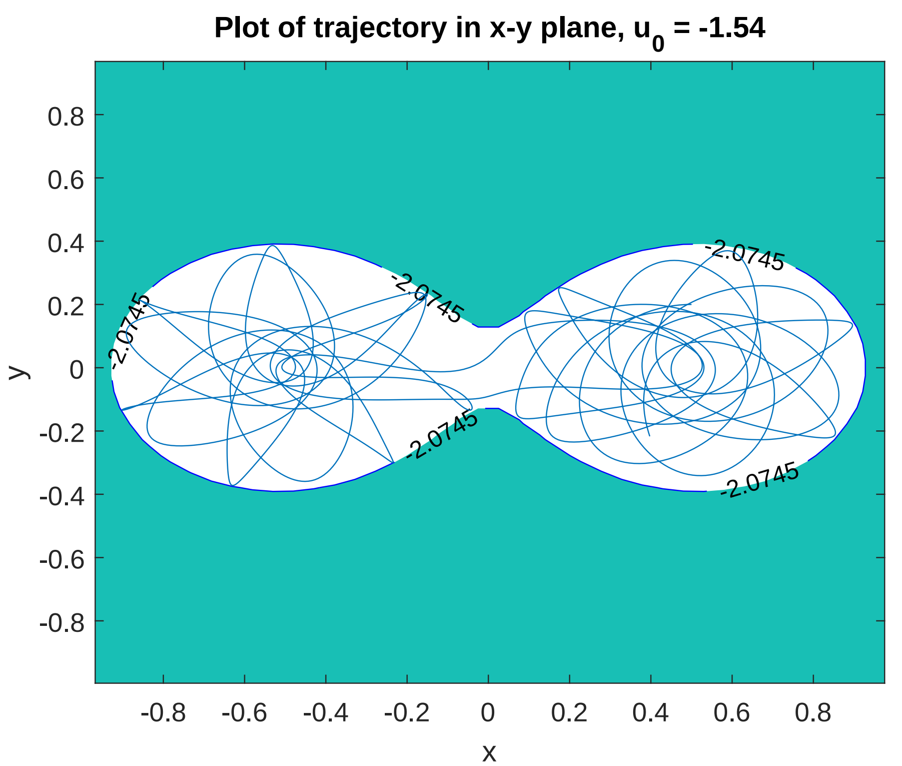

Figure 16: Plot of trajectory in the x-y plane, computed by `ode45`. The shaded region represents the forbidden region for which $\Omega(x, y) > J = -2.0745$. This motion is for time $t \in [0, 30]$ and initial speed $u_0 = -1.54$.

(iii) State the values of x and y at $t = 30$, giving a reasoned assessment on their accuracy*. For initial speed $u_0 = -1.54$, at $t = 30$, the values of (x, y) are:

$$x(30) = 0.3973 \text{ and } y(30) = -0.2168$$

(V) For $u_0 = -1.69$:

(i) Plot x against t:

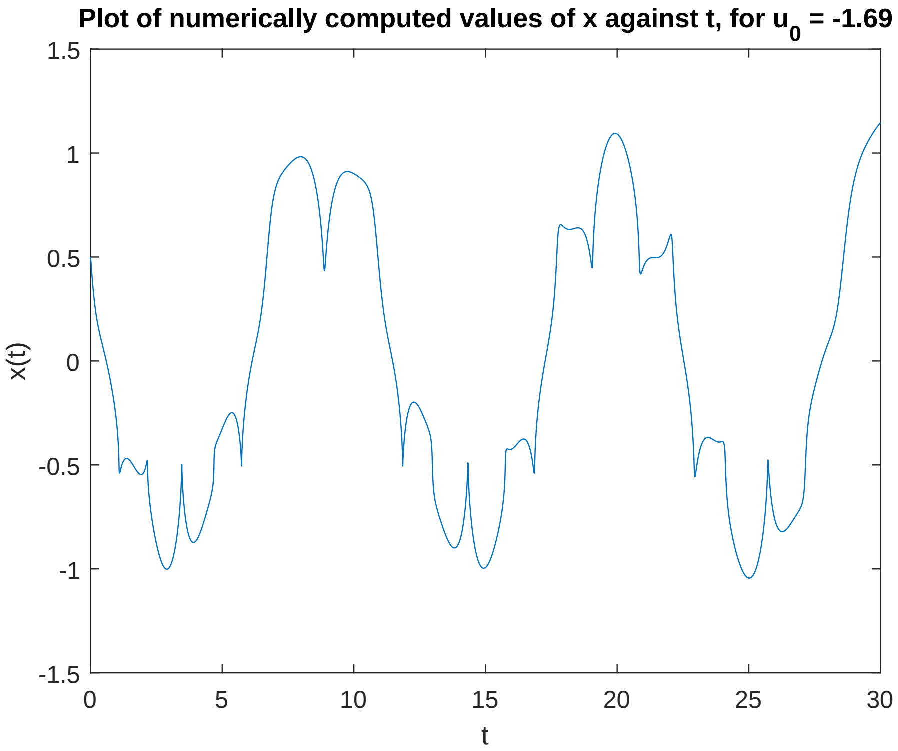

Figure 17: Plot of $x(t)$, computed by `ode45`, against $t \in [0, 30]$ for $u_0 = -1.69$.

(ii) Display the trajectory in the (x, y) plane, and the forbidden region $\Omega(x, y) > J = -1.8322$:

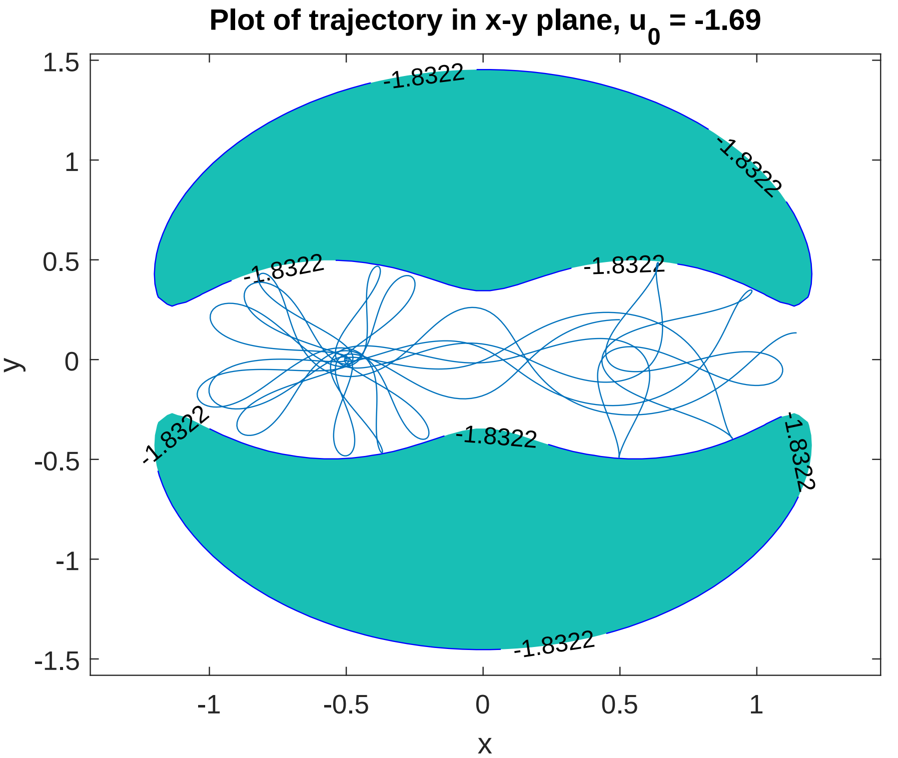

Figure 18: Plot of trajectory in the x-y plane, computed by `ode45`. The shaded region represents the forbidden region for which $\Omega(x, y) > J = -1.8322$. This motion is for time $t \in [0, 30]$ and initial speed $u_0 = -1.69$.

(iii) State the values of x and y at $t = 30$, giving a reasoned assessment on their accuracy*. For initial speed $u_0 = -1.69$, at $t = 30$, the values of (x, y) are:

$$x(30) = 1.1455 \text{ and } y(30) = 0.1340$$

(VI) For $u_0 = -1.7$:

(i) Plot x against t:

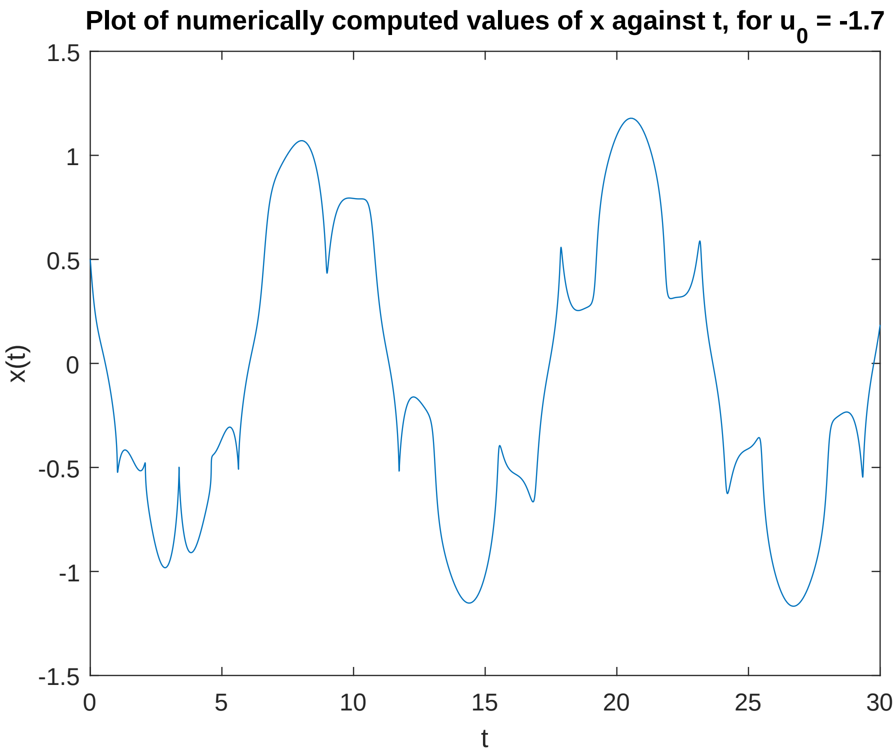

Figure 19: Plot of $x(t)$, computed by `ode45`, against $t \in [0, 30]$ for $u_0 = -1.7$.

(ii) Display the trajectory in the (x, y) plane, and the forbidden region $\Omega(x, y) > J = -1.8153$:

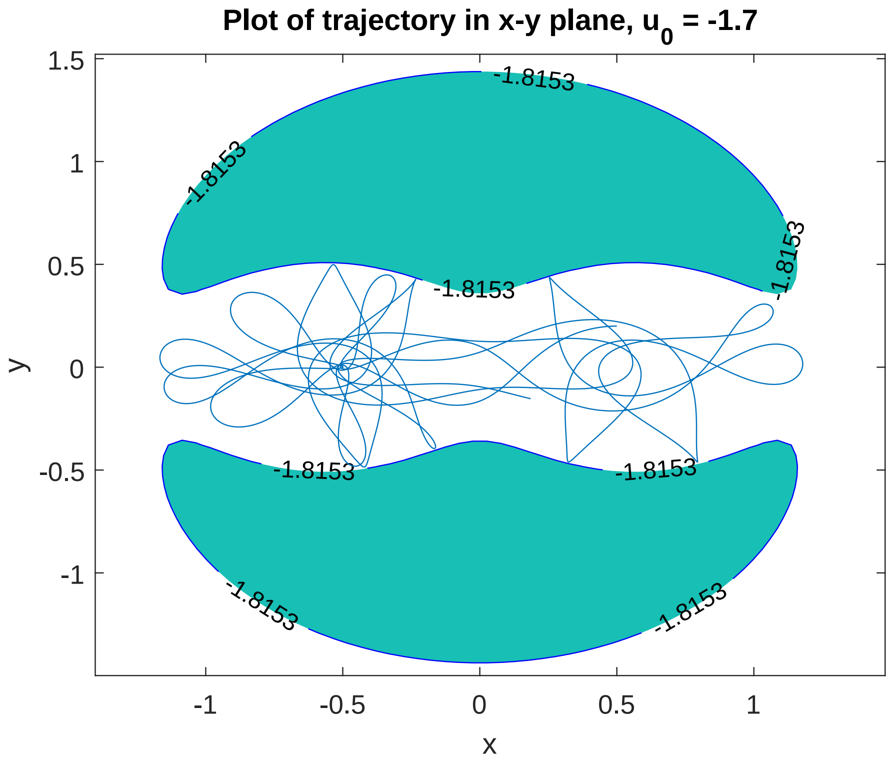

Figure 20: Plot of trajectory in the x-y plane, computed by `ode45`. The shaded region represents the forbidden region for which $\Omega(x, y) > J = -1.8153$. This motion is for time $t \in [0, 30]$ and initial speed $u_0 = -1.7$.

(iii) State the values of x and y at $t = 30$, giving a reasoned assessment on their accuracy*. For initial speed $u_0 = -1.7$, at $t = 30$, the values of (x, y) are:

$$x(30) = 0.1853 \text{ and } y(30) = -0.1531$$

(VII) For $u_0 = -1.9$:

(i) Plot x against t:

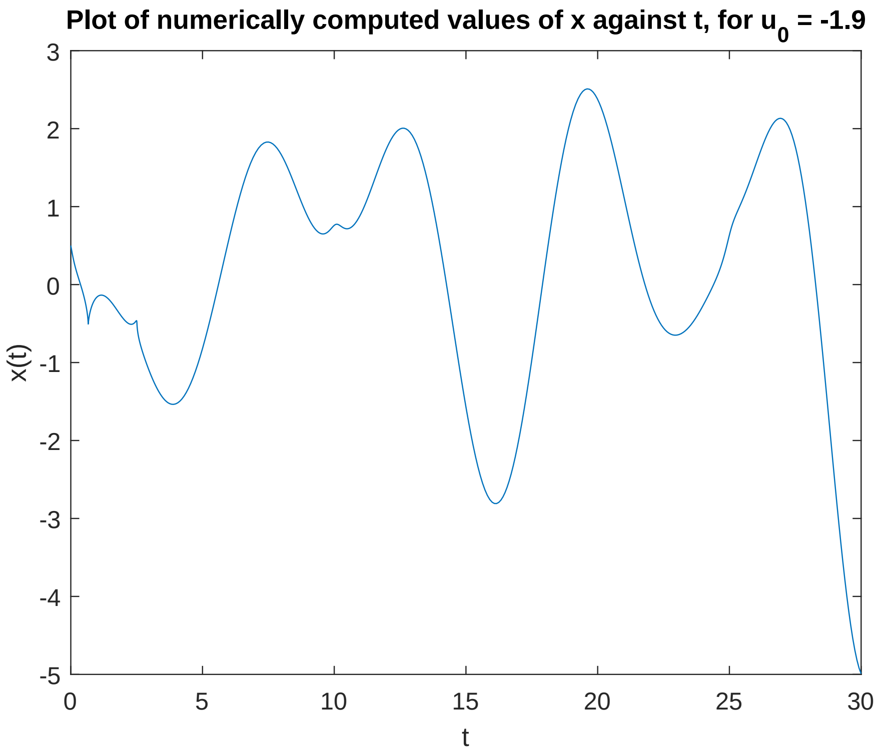

Figure 21: Plot of $x(t)$, computed by `ode45`, against $t \in [0, 30]$ for $u_0 = -1.9$.

(ii) Display the trajectory in the (x, y) plane, and the forbidden region $\Omega(x, y) > J = -1.4553$:

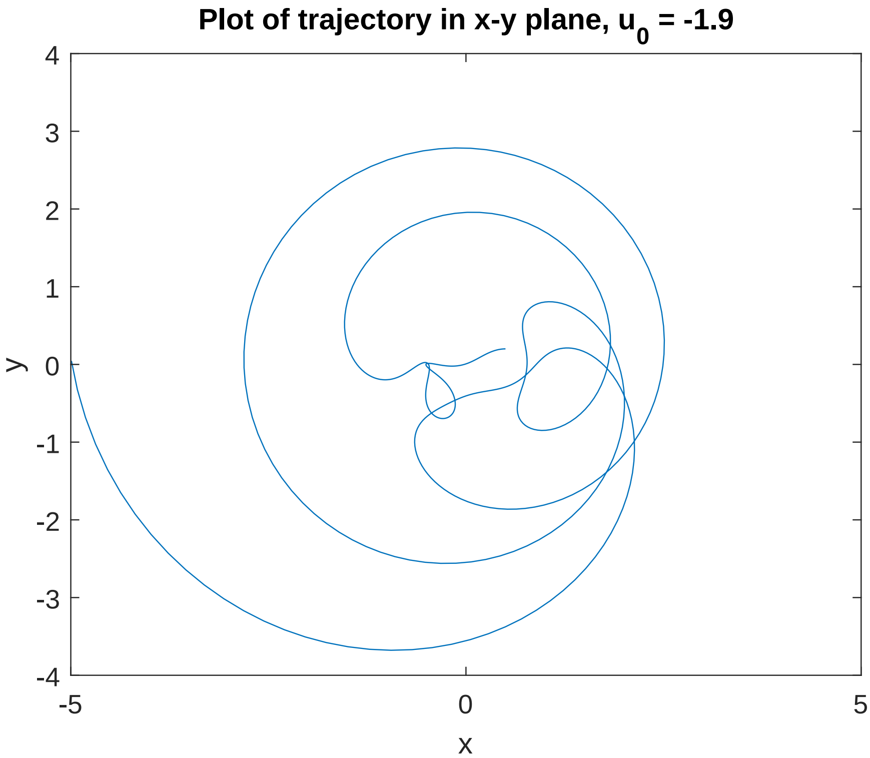

Figure 22: Plot of trajectory in the x-y plane, computed by `ode45`. There is no shaded region since there is no forbidden region. $\Omega(x, y) \leq J = -1.4553, \forall x, y \in \mathbb{R}$. This motion is for time $t \in [0, 30]$ and initial speed $u_0 = -1.9$.

It is true that there exists no forbidden region for all $x, y \in \mathbb{R}$ since we know:

$$\Omega(x, y) = -\left(\frac{(x - \frac{1}{2})^2}{4} + \frac{(x + \frac{1}{2})^2}{4} + \frac{1}{2((x - \frac{1}{2})^2 + y^2)^{\frac{1}{2}}} + \frac{1}{2((x + \frac{1}{2})^2 + y^2)^{\frac{1}{2}}} + \frac{y^2}{2}\right), \quad (7)$$

is strictly negative for all x, y. The function tends to $-\infty$ as $|x| \to 0.5, |y| \to 0$ and also as $|x|, |y| \to \infty$. In the plane $x \in [-2.5, 2.5]$, $y \in [-2.5, 2.5]$, the maximum value of $\Omega(x, y) \approx -1.50059$ (calculated by the function `max()` on the matrix that represents the height of the function $\Omega(x, y)$ in the x-y plane). This range is large enough to pass the maximum point of $\Omega(x, y)$ as Figure 23 below shows that the function starts to decrease again as $|x|, |y| > 1.5$. So beyond $|x| = |y| = 2.5$, the graph is strictly decreasing since it is continuous and must also satisfy the property that $\Omega(x, y) \to -\infty$ as $|x|, |y| \to \infty$. (checking $\nabla\Omega(x, y) = 0$ can verify where the maximum value of the function is).
Therefore the maximum value in the smaller range in Figure 23 below is indeed the maximum value of the whole function. Therefore $\Omega(x, y) \leq -1.50059 \leq J = -1.4553$, and so Ω cannot be greater than J for any $x, y \in \mathbb{R}$ for initial speed $u_0 = -1.9$.

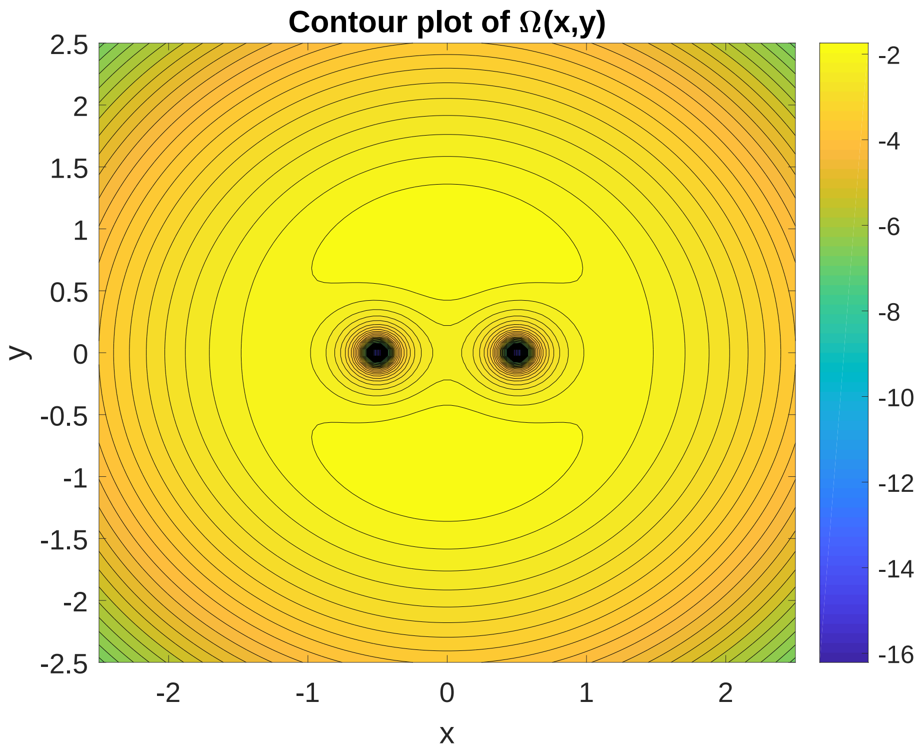

Figure 23: Contour plot of $\Omega(x, y)$ for $x \in [-2.5, 2.5]$, $y \in [-2.5, 2.5]$.

(iii) State the values of x and y at $t = 30$, giving a reasoned assessment on their accuracy*. For initial speed $u_0 = -1.9$, at $t = 30$, the values of (x, y) are:

$$x(30) = -4.9919 \text{ and } y(30) = 0.0445$$

*For each part (iii) that gives the value of x and y at $t = 30$, there is a similar justification as to why they are all accurate to approximately 4 decimal places. The relative tolerance that I manually set for my program is 1×10^{-5}. Hence the accuracy would be to this order of magnitude. Therefore it is a reasonable assumption to say that all of the values for x and y are accurate to 1×10^{-4} since the last accurate decimal place would appropriately round the penultimate decimal point.

Each trajectory is dependent on the initial condition of the speed of the particle in the x direction, u_0. Small changes in the speed, result in large differences in the path taken by the spacecraft. For example, Figure 14 is a closed and fairly regular path which is contained within the permitted area closer to P_2 (orbit is mainly about P_2). However Figure 16 shows a chaotic path which goes between the two planets P_1 and P_2, despite u_0 differing by only 0.01 units of speed.

As $|u_0|$ increases, the area of the forbidden region, $\Omega(x, y) > J$, decreases. This is because J is defined by the addition of the squares of the initial speeds (and Ω), hence the value of J increases as u_0 increases. Since the values of $\Omega(x, y)$ at given points in the x-y plane are fixed, for a fixed value of μ, the criteria that a forbidden region exists when $\Omega(x, y) > J$ becomes more improbable. Once $J > \sup_{x,y \in \mathbb{R}} \Omega(x, y)$, no forbidden region will exist for the path of the spacecraft.

For smaller values of $|u_0|$, where the allowed region is closed, we obtain a good idea of the span of the trajectory. The trajectory is likely to be either relatively periodic, and so the size of the trajectory would be consistent following the motion, such as in Figure 10. However if the motion is chaotic, but the allowed region is still bounded, then eventually the trajectory will go everywhere in the closed space. Hence the region is a good indicator of the size of the space, indicated by Figure 16. There are however cases where the allowed region is infinite (Figure 18) but the motion still remains relatively close to the two planets governing the motion of the spacecraft. Here the allowed region does not define the size of the trajectory.

The most suitable value of u_0 required to travel from the neighbourhood of P_2 to the neighbourhood of P_1 would be $u_0 = 1.55$ (to 2dp). This is because it is the second smallest value (to 2dp) after which the velocity enables you to get into the neighbourhood of P_1, but also does not require that much time. Figure 24 is for $t \in [0, 2.5]$, whereas Figure 25 is $t \in [0, 11.5]$. $u_0 = 1.55$ is more than 4 times faster at getting into the neighbourhood of P_1 than $u_0 = 1.54$ - hence the former is more efficient, despite the extra energy needed for the higher initial speed. This efficiency would depend on the units of time and speed however.

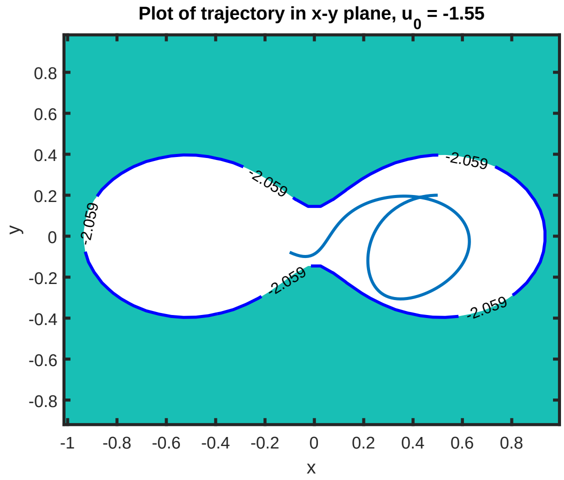

Figure 24: Plot for $u_0 = 1.55$ for $t \in [0, 2.5]$.

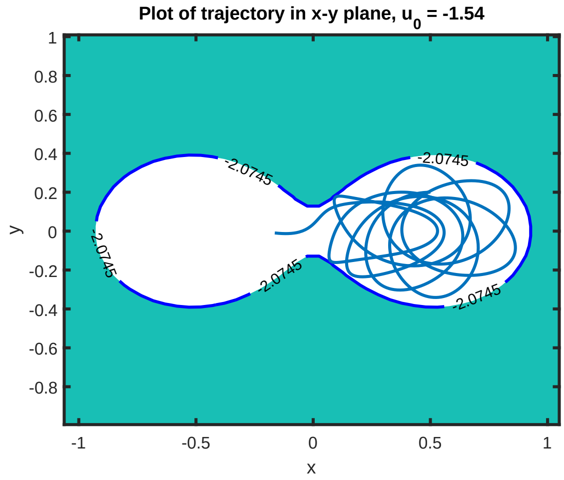

Figure 25: Plot for $u_0 = 1.54$ for $t \in [0, 11.5]$.

4. **By examining contour plots of show that the system (1a), (1b) and (2) generally has five equilibrium points; three 'collinear Lagrange points' on the x-axis and two 'equilateral Lagrange points' at the third vertex of an equilateral triangle whose other two vertices are at P_1 and P_2. Display contour plots for three values of μ with the equilibrium points marked. Note that without loss of generality you may assume that $\mu \in (0, 0.5]$; why?**

   We can assume, without loss of generality, that $\mu \in (0, 0.5]$ since the coordinates of the planets P_1 and P_2 are at $(\mu - 1, 0)$ and $(\mu, 0)$ respectively. Therefore for $0.5 < \mu < 1$, the system would be analagous to a system where $\mu \in (0, 0.5]$, up to a minus sign (reflection of the system in the y-axis), as a result of the symmetry of the problem. More explicitly: when $\mu = \mu_0 \in [0.5, 1)$, the system would be precisely analagous to the relection of the system $\mu = \mu_1$, in the y-axis, where $\mu_1 = 1 - \mu_0$. Therefore we only need to investigate the cases for $\mu \in (0, 0.5]$.

```
function dxdt = threebodyQ2(t,x)
%syms x y
%Omegasimple(x,y)=-(1/(2*(((x-0.5)^2+(y^2))^(1/2))))

dxdt = zeros(4,1);
dxdt(1) = x(2);
dxdt(2) = 2*x(4) + (1 - 2*x(1))./(4*((x(1) - (1/2)).^2 + (x(3)).^2).^(3/2));
dxdt(3) = x(4);
dxdt(4) = -2*x(2) + (-x(3))./(2*((x(1) - (1/2)).^2 + (x(3)).^2).^(3/2));
```

```
function dxdt = threebodyQ3(t,x)
mew=0.5;
%syms x_1 x_3
%r_1=@(x_1,x_3)((x_1+1-mew)^2+(x_3)^2)^(1/2);
%r_2=@(x_1,x_3)((x_1-mew)^2+(x_3^2))^(1/2);
%Omega(x_1,x_3)=(-0.5*mew*(r_1(x_1,x_3))^2-0.5*(1-mew)*(r_2(x_1,x_3))^2-(mew/(r_1(x_1,x_3)))-((1-mew)/(r_2(x_1,x_3))));
%Px = eval(['@(x_1,x_3)' char(diff(Omega(x_1,x_3),x_1))]);
%Py = eval(['@(x_1,x_3)' char(diff(Omega(x_1,x_3),x_3))]);

dxdt = zeros(4,1);
dxdt(1) = x(2);
dxdt(2) = 2*x(4) - (mew*(x(1)+1-mew).*(-1+ ( ((x(1)+1-mew).^2+(x(3)).^2).^(-3/2) )) + (1-mew)*(x(1)-mew).*(-1 + (((x(1)-mew).^2+(x(3).^2)).^(-3/2))) );
dxdt(3) = x(4);
dxdt(4) = -2*x(2) - (mew*x(3).*(-1+ ( ((x(1)+1-mew).^2+(x(3)).^2).^(-3/2) )) + (1-mew)*x(3).*(-1 + (((x(1)-mew).^2+(x(3).^2)).^(-3/2))) );
```

```
syms x y
Omega(x,y)= - (x - 1/2)^2/4 - (x + 1/2)^2/4 - 1/(2*((x - 1/2)^2 + y^2)^(1/2)) - 1/(2*((x +
    1/2)^2 + y^2)^(1/2)) - y^2/2;
Initial1=[0.5,-1.0,0.2,0];
Initial2=[0.5,-1.39,0.2,0];
Initial3=[0.5,-1.53,0.2,0];
Initial4=[0.5,-1.54,0.2,0];
Initial5=[0.5,-1.69,0.2,0];
Initial6=[0.5,-1.7,0.2,0];
Initial7=[0.5,-1.9,0.2,0];
Initial8=[0.5,-1.54,0.2,0];

J_1=0.5*Initial1(2)^2+double(Omega(0.5,0.2));
J_2=0.5*Initial2(2)^2+double(Omega(0.5,0.2));
J_3=0.5*Initial3(2)^2+double(Omega(0.5,0.2));
J_4=0.5*Initial4(2)^2+double(Omega(0.5,0.2));
J_5=0.5*Initial5(2)^2+double(Omega(0.5,0.2));
J_6=0.5*Initial6(2)^2+double(Omega(0.5,0.2));
J_7=0.5*Initial7(2)^2+double(Omega(0.5,0.2));
J_8=0.5*Initial8(2)^2+double(Omega(0.5,0.2));

autoJ=[J_1,J_2,J_3,J_4,J_5,J_6,J_7,J_8];
initialspeed=[-1.0,-1.39,-1.53,-1.54,-1.69,-1.7];

options=odeset('RelTol',1e-5,'AbsTol',1e-5);
[tsol,xsol]=ode45(@threebody3,[0,11.5],[0.5,Initial8(2),0.2,0],options);
yy=linspace(-2.5,2.5); xx=linspace(-2.5,2.5);
[X,Y]=meshgrid(xx,yy);
Z= - (X - 1/2).^2/4 - (X + 1/2).^2/4 - 1./(2.*((X - 1/2).^2 + Y.^2).^(1/2)) - 1./(2.*((X +
    1/2).^2 + Y.^2).^(1/2)) - Y.^2/2;
contourf(X,Y,real(Z),[autoJ(8),autoJ(8)],'b-','ShowText','on')
hold
plot(xsol(:,1),xsol(:,3))
xlabel('x')
ylabel('y')
title('Plot of trajectory in x-y plane, u_0 = -1.54')
```