AE 502 Project 1

Linyi Hou | linyih2 | github.com/TigerHou2

March 1, 2021

Problem 1

Q: Using the equations of motion for the Circular Restricted Three-Body Problem (CRTBP), investigate different trajectories in the dynamical system by propagating ten sets of initial conditions (one set = $[x, y, z, \dot{x}, \dot{y}, \dot{z}]$). Find orbits around each of the primary bodies as well as around both bodies.

A: The equations of motion (EOMs) for the CRTBP, given in the co-rotating frame, are as follows:

$$\ddot{x} - 2\omega\dot{y} = \frac{\partial U}{\partial x} \tag{1}$$

$$\ddot{y} + 2\omega\dot{x} = \frac{\partial U}{\partial y} \tag{2}$$

$$\ddot{z} = \frac{\partial U}{\partial z} \tag{3}$$

where U is the pseudo-potential and ω is the angular velocity, defined as:

$$\omega = \sqrt{\frac{G(M_1 + M_2)}{R^2}} \tag{4}$$

$$U = \frac{1}{2}\omega^2(x^2 + y^2) + \frac{1-\mu}{r_1} + \frac{\mu}{r_2} \tag{5}$$

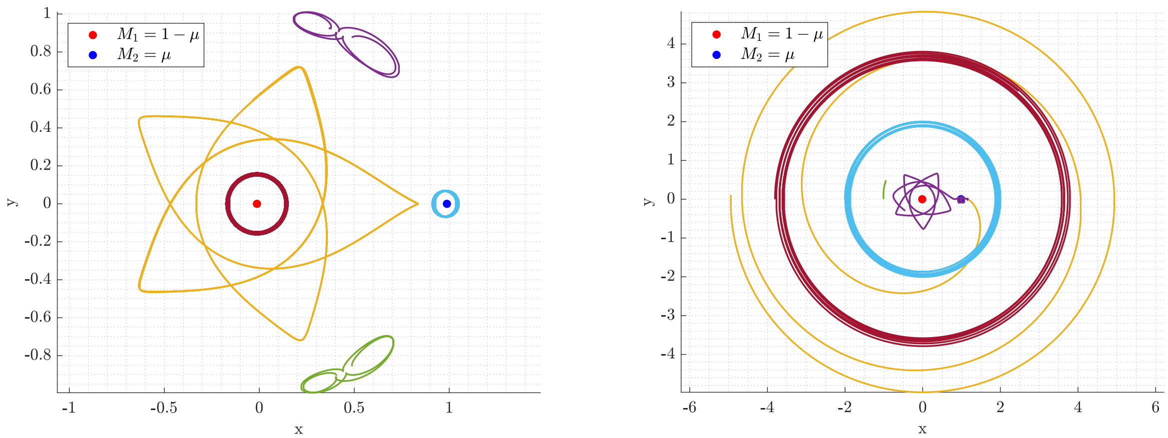
$$r_1 = \sqrt{(x - x_1)^2 + y^2 + z^2} \tag{6}$$

$$r_2 = \sqrt{(x - x_2^2) + y^2 + z^2} \tag{7}$$

$$R = x2 - x1 = 1 \tag{8}$$

Using the above relations, the partials on the RHS of Eqs. (1)-(3) can be easily computed either by hand or via tools such as MATLAB. Subsequently, the EOMs can be propagated.

Ten sets of initial conditions were propagated using the EOMs for the CRTBP, and the results are shown in Fig. 1. The plot uses canonical units with $G = 1$, $R = 1$, $m_1 + m_2 = 1$, where $m_2 = \mu = 0.012150585609262$, which corresponds to the Earth-Moon system.



(a) Orbits near L1, L4, L5, m1, and m2.

(b) Orbits around L2, L3, and m1+m2.

Figure 1: Ten orbits in the co-rotating frame of the circular restricted three-body problem.

The locations of the Lagrange points L1-L5 were computed analytically using the following equations [1]:

$$L_1 = R\left(1 - \left(\frac{\alpha}{3}\right)^{1/3}\right) \tag{9}$$

$$L_2 = R\left(1 + \left(\frac{\alpha}{3}\right)^{1/3}\right) \tag{10}$$

$$L_1 = -R\left(1 + \left(\frac{5\alpha}{12}\right)\right) \tag{11}$$

$$L4 = \left(\frac{R}{2}\frac{M_1 - M_2}{M_1 + M_2}, \ \frac{\sqrt{3}}{2}R\right) \tag{12}$$

$$L5 = \left(\frac{R}{2}\frac{M_1 - M_2}{M_1 + M_2}, \ -\frac{\sqrt{3}}{2}R\right) \tag{13}$$

where the x-coordinates of L1-L3 are shown, and both x- and x-coordinates of L4-L5 are shown. This is because in the CRTBP, the two masses M_1 and M_2 are places on the x-axis of the co-rotating frame, meaning L1-L3 simply have y-coordinates of zero.

In Fig. 1a, position perturbations are applied to L4 and L5, because the orbit is simply a point at the exact locations of L4 and L5. For L1, the initial position is slightly offset toward M_1, which resulted in a periodic orbit.

In Fig. 1b, the high sensitivity of L2 is shown by propagating two sets of initial conditions that differ in their y-velocities by $1 \times 10^{-14} DU/TU$. In one case, the third body falls towards M_1, whereas in the other case the third body approaches escape from the system.

Although L3 is also an unstable Lagrange point, the sensitivity of L3 was low enough that setting the initial condition to exactly L3 caused the orbit to remain at a single point. A small perturbation in position was applied to show that a body in orbit near L3 would experience only a slow drift away, as opposed to L2 where the drift was much faster.

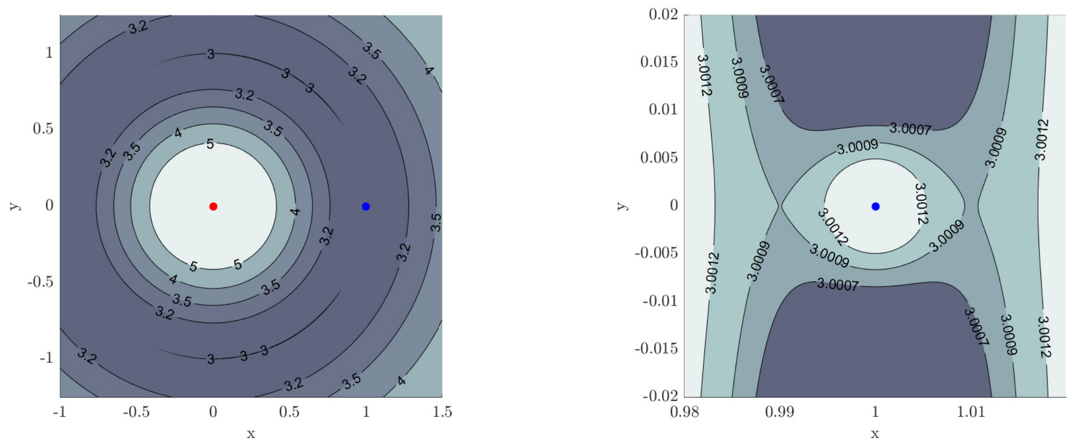
Problem 2

Q: Plot the zero velocity contours for various Jacobi constants in the Earth-Moon system and the Sun-Earth system. Discuss results.

A: The Jacobi constant C defines zero velocity surfaces according to the equation below:

$$(x^2 + y^2) + \frac{2(1-\mu)}{r_1} + \frac{2\mu}{r_2} = C \tag{14}$$

Solving the above equation in the Sun-Earth system yielded the following results:

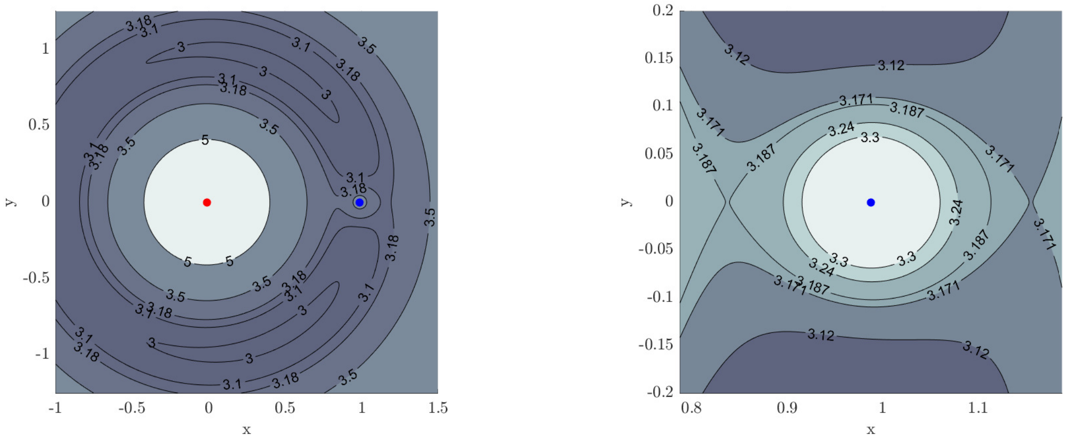


(a) (b)

Figure 2: Zero velocity surfaces of the Sun-Earth system.

In Fig. 2a, the Lagrange points L4 and L5 can be somewhat seen on the contour $C = 3$, but the region is very narrow, and L3 is not clearly visible. When zoom in, as shown by Fig. 2b, a more refined contour plot reveals L1 and L2 along the contour $C = 3.0009$. These results can be compared with those of the Earth-Moon system, shown below:

It is clear that the Lagrange points for the Earth-Moon system are less sensitive to changes in the Jacobi constant. For example, the difference between the Jacobi constants that distinguish L1 and L2 is on the order of 10^{-2}, whereas for the Sun-Earth system the



(a) (b)

Figure 3: Zero velocity surfaces of the Sun-Earth system.

difference is on the order of 10^{-4}. This means that the Earth-Moon system has Lagrange points that are far less sensitive to perturbations. More generally, in the CRTBP, if M_1 and M_2 are more similar in mass, their Lagrange points would be less sensitive to disturbances.

However, the sensitivity mentioned in the above paragraph is relative to the canonical units of the CRTB system. Since the Sun-Earth system has canonical units that correspond to much larger values in SI as compared to the Earth-Moon system, it is possible that the Sun-Earth system, in practice, is less sensitive to perturbations from a station-keeping standpoint.

Problem 3

Q: Given the following initial conditions and period for a Halo orbit at L2 in the Earth-Moon system, compute the stable and unstable manifolds and plot them.

$$\mu = 0.012150585609262$$
$$X0 = [1.118824382902157, 0.0, 0.014654873101278, 0.0, 0.180568501159703, 0.0]$$
$$P = 1.706067405636607$$

A: Given the time derivative of state

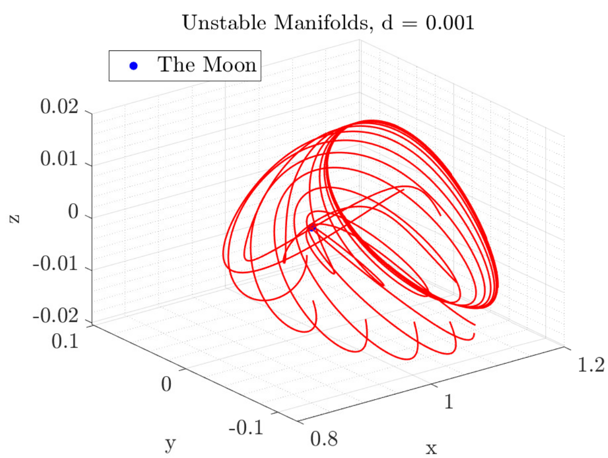
$$\dot{x} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_5 \\ f_6 \end{bmatrix} = \begin{bmatrix} \boldsymbol{v} \\ 2\dot{y} + \dfrac{\partial U}{\partial x} \\ -2\dot{x} + \dfrac{\partial U}{\partial y} \\ \dfrac{\partial U}{\partial z} \end{bmatrix} \tag{15}$$

and the state transition matrix $\dot{\Phi} = F\Phi$ where $\Phi(t_0, t_0) = \mathcal{I}_6$ and:

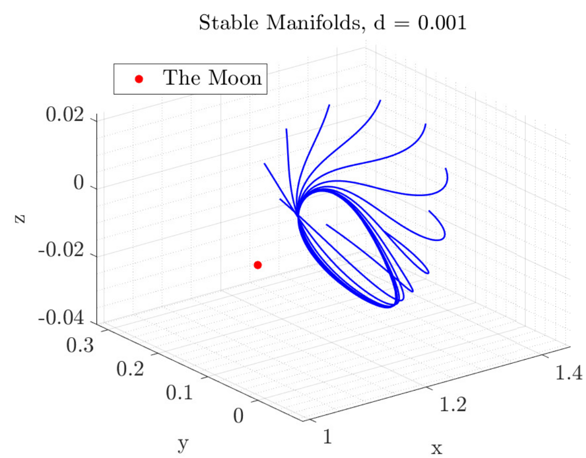
$$F = \begin{bmatrix} \dfrac{\partial f_1}{\partial x} & \dfrac{\partial f_1}{\partial y} & \dfrac{\partial f_1}{\partial z} & \dfrac{\partial f_1}{\partial v_x} & \dfrac{\partial f_1}{\partial v_y} & \dfrac{\partial f_1}{\partial v_z} \\ \dfrac{\partial f_2}{\partial x} & \dfrac{\partial f_2}{\partial y} & & \cdots & & \dfrac{\partial f_2}{\partial v_z} \\ \vdots & \vdots & & \ddots & & \vdots \\ \dfrac{\partial f_6}{\partial x} & \dfrac{\partial f_6}{\partial y} & & \cdots & & \dfrac{\partial f_6}{\partial v_z} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathcal{I}_3 \\ \nabla_{x,y,z} U & \begin{bmatrix} 0 & 2 & 0 \\ -2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \end{bmatrix} \tag{16}$$

$\dot{x}$ and Φ can then be propagated to find any future state.

From the provided initial conditions, a perturbation of magnitude $d = 0.001$ was applied for different points along the orbit. Specifically, orbits were propagated forward for the unstable manifolds, and propagated backwards for the stable manifolds. The following stable and unstable manifolds were generated:



(a) Unstable manifolds.



(b) Stable manifolds.

Figure 4: Stable and unstable manifolds near the Earth-Moon L2 Halo orbit.

References

[1] S. Widnall, "Lecture l18 - exploring the neighborhood: the restricted three-body problem." https://ocw.mit.edu/courses/aeronautics-and-astronautics/16-07-dynamics-fall-2009/lecture-notes/MIT16_07F09_Lec18.pdf Accessed: 3-1-2021.

Appendix A: Code for Problem 1

```
%% AE 502 HW1 P1, SP21
% Tiger Hou

%% initialization
close all
clear;clc
addpath('../../tools')
latexify

G = 1;
mu = 0.012150585609262;

m2 = mu;
m1 = 1 - mu;
R = 1;

alpha = m2 / (m1+m2);
beta  = m1 / (m1+m2);

x1 = -alpha*R; % construct the system with barycenter at the origin
x2 =   beta*R;

w = sqrt(G*(m1+m2)/R^3);
dt = 0.001;

% Lagrange points
% https://ocw.mit.edu/courses/aeronautics-and-astronautics/16-07-dynamics-fall-2009/lecture-notes/MIT16_07F09_Lec18.pdf
L1 = [  R*(1-(alpha/3)^(1/3)), 0, 0];
```

```
L2 = [  R*(1+(alpha/3)^(1/3)), 0, 0];
L3 = [ -R*(1+5*alpha/12), 0, 0];
L4 = [  0.5+x1,  0.5*sqrt(3), 0];
L5 = [  0.5+x1, -0.5*sqrt(3), 0];

% u0 = [ 0.5,  0.5,  0.0,  1.0,  0.5,  0.0]';
u1 = [...
        L1 + [-4e-3,0,0], 0, 0, 0; ... L1 (unstable)
        L4 + [0,0.01,0], 0, 0, 0; ... L4 (perturbed, stable)
        L5 + [0,0.01,0], 0, 0, 0; ... L5 (perturbed, stable)
        L1 + [7e-2,0,0], 0, -0.3, 0; ... lunar orbit
        0.15, 0, 0, 0, 2.25, 0; ... LEO
     ]';
T1 = 40;

u2 = [...
        L2, 0, -2.046483231514e-2, 0; ... L2 (unstable, outbound)
        L2, 0, -2.046483231515e-2, 0; ... L2 (unstable, inbound)
        L3 + [0,2e-3,0], 0, 0, 0; ... L3 (perturbed, unstable)
        L3 + [-1,0,0], 0, 2.7, 0; ... orbit around both bodies, small
        L3 + [-2.8,0,0], 0, 4.31, 0; ... orbit around both bodies, large
     ]';
T2 = 35;
%         0.5,  0.5*sqrt(3),  0.0,  0.0,  0.0,  0.0; ... % L4
%         0.5, -0.5*sqrt(3),  0.0,  0.0,  0.0,  0.0; ... % L5
%         (1+1e-6)*L3, 0.0,  0.0,  0.0,  -1.00002e-1,  0.0; ... % weird L3
%         L3, 1e-3, 0, 0, +6.1e-5, 0; ... % L3 with perturbation + vel adj
%         0, 0.3, 0, -1.5, 0, 0; ... % orbit around m1
% %         1.05, 0, 0, 0, 0.38, 0 ; ...
%         0, 2.5, 0, 3.2, 0, 0 ; ...
%         1.118824382902157, 0.0, ... % L2 from problem 3
%         0.014654873101278, 0.0, ...
%         0.180568501159703, 0.0 ...

u0 = u2;
T = T2;
steps = ceil(T/dt);

```

```matlab
N = size(u0,2);
uu = zeros(6,N,steps);

%% propagation

for jj = 1 : steps

    uu(:,:,jj) = u0;
    % ============= RK4 =============
    k1 = dt * ff( u0,N,m1,m2,x1,x2,w);
    k2 = dt * ff((u0 + k1/2),N,m1,m2,x1,x2,w);
    k3 = dt * ff((u0 + k2/2),N,m1,m2,x1,x2,w);
    k4 = dt * ff((u0 + k3)  ,N,m1,m2,x1,x2,w);
    u0 = u0 + 1/6 * (k1 + 2*k2 + 2*k3 + k4);

end


%% plotting - 3D

dat = permute(uu,[1,3,2]);

figure(1)
hold on
M1 = scatter3(x1,0,0,'r','filled');
M2 = scatter3(x2,0,0,'b','filled');
for i = 1:N
    plot3(dat(1,:,i),dat(2,:,i),dat(3,:,i),'LineWidth',1.25)
end
hold off
xlabel('x')
ylabel('y')
zlabel('z')
legend([M1,M2],'$M_1 = 1-\mu$','$M_2 = \mu$','Location','NorthWest')
axis equal
grid minor

latexify(16,12,12)

%% function definitions
```

```
function rv_dot = ff(rv,N,m1,m2,x1,x2,w)
% takes the 6xN position & velocity vector and computes the acceleration
%   where N is the number of particles to track

rv_dot = zeros(6,N);

for i = 1:N
    x = rv(1,i);
    y = rv(2,i);
    z = rv(3,i);

    dudx = x*w^2 ...
         - (m1*(2*x - 2*x1))/(2*((x - x1)^2 + y^2 + z^2)^(3/2)) ...
         - (m2*(2*x - 2*x2))/(2*((x - x2)^2 + y^2 + z^2)^(3/2));
    dudy = w^2*y ...
         - (m1*y)/((x - x1)^2 + y^2 + z^2)^(3/2) ...
         - (m2*y)/((x - x2)^2 + y^2 + z^2)^(3/2);
    dudz = - (m1*z)/((x - x1)^2 + y^2 + z^2)^(3/2) ...
           - (m2*z)/((x - x2)^2 + y^2 + z^2)^(3/2);

    % velocity
    rv_dot(1:3,i) = rv(4:6,i);
    % acceleration
    rv_dot(4:6,i) = [dudx + 2*w*rv(5,i);...
                     dudy - 2*w*rv(4,i);...
                     dudz];
end

end
```

Appendix B: Code for Problem 2

```

%% AE 502 HW1 P2, SP21
% Tiger Hou

%% initialization
close all
```

```
clear;clc
latexify

G = 1;
% mu = 0.012;  % Earth-Moon
mu = 5.974e24 / 1.989e30;  % Sun-Earth

m2 = mu;
m1 = 1 - mu;
R = 1;

alpha = m2 / (m1+m2);
beta  = m1 / (m1+m2);

x1 = -alpha*R; % construct the system with barycenter at the origin
x2 =   beta*R;

%% create L1-L2 contours
% hw = 0.2;
hw = 0.02;
x = linspace(x2-hw,x2+hw,600);
y = linspace(-hw,hw,600);
[X,Y] = meshgrid(x,y);

R1 = sqrt((X-x1).^2+Y.^2);
R2 = sqrt((X-x2).^2+Y.^2);
Z = X.^2 + Y.^2 + 2*(1-mu)./R1 + 2*mu./R2;

%% plot results for L1-L2
figure(1)
new_bone = bone;
new_bone = new_bone(28:60,:);
colormap(new_bone)
hold on
% set(gca,'ColorScale','log')
% [M,c] = contourf(X,Y,Z,[1,3.12, 3.171, 3.187, 3.24, 3.3]);
[M,c] = contourf(X,Y,Z,[1,2.999, 3.0007, 3.000891, 3.0012]);
clabel(M,c)
xlabel('x')
ylabel('y')
```

```
scatter(x2,0,'b','filled')
hold off
axis equal
setgrid
latexify(16,12,12)

%% create L3-L5 contours
x = linspace(-1,1.5,1500);
y = linspace(-1.25,1.25,1500);
[X,Y] = meshgrid(x,y);

R1 = sqrt((X-x1).^2+Y.^2);
R2 = sqrt((X-x2).^2+Y.^2);
Z = X.^2 + Y.^2 + 2*(1-mu)./R1 + 2*mu./R2;

%% plot results for L3-L5
figure(2)
hold on
% set(gca,'ColorScale','log')
colormap(new_bone)
% [M,c] = contourf(X,Y,Z,[1, 3, 3.1, 3.18, 3.5, 5]);
[M,c] = contourf(X,Y,Z,[1,2.999, 3.00, 3.2, 3.5, 4, 5]);
clabel(M,c)
xlabel('x')
ylabel('y')
scatter(x1,0,'r','filled')
scatter(x2,0,'b','filled')
hold off
axis equal
setgrid
latexify(16,12,12)
```

Appendix C: Code for Problem 3

```

%% AE 502 HW1 P3, SP21
% Tiger Hou

%% initialization
```

```
close all
clear;clc
addpath('../../tools')
latexify

mu = 0.012150585609262;
x0 = [  1.118824382902157, 0.0, ...
        0.014654873101278, 0.0, ...
        0.180568501159703, 0.0]';
P = 1.706067405636607;

G = 1;
m2 = mu;
m1 = 1 - mu;
R = 1;

alpha = m2 / (m1+m2);
beta  = m1 / (m1+m2);

x1 = -alpha*R; % construct the system with barycenter at the origin
x2 =   beta*R;

w = sqrt(G*(m1+m2)/R^3);

%% state transition matrix

phi_t0 = eye(6);
Y0 = [reshape(phi_t0,36,1);x0];

dudx = @(x,y,z) x*w^2 ...
     - (m1*(2*x - 2*x1))/(2*((x - x1)^2 + y^2 + z^2)^(3/2)) ...
     - (m2*(2*x - 2*x2))/(2*((x - x2)^2 + y^2 + z^2)^(3/2));

dudy = @(x,y,z) w^2*y ...
     - (m1*y)/((x - x1)^2 + y^2 + z^2)^(3/2) ...
     - (m2*y)/((x - x2)^2 + y^2 + z^2)^(3/2);

dudz = @(x,y,z) ...
     - (m1*z)/((x - x1)^2 + y^2 + z^2)^(3/2) ...
     - (m2*z)/((x - x2)^2 + y^2 + z^2)^(3/2);
```

```

dudx_dx = @(x,y,z) w^2 ...
        - m2/((x - x2)^2 + y^2 + z^2)^(3/2) ...
        - m1/((x - x1)^2 + y^2 + z^2)^(3/2) ...
        + (3*m1*(2*x - 2*x1)^2)/(4*((x - x1)^2 + y^2 + z^2)^(5/2)) ...
        + (3*m2*(2*x - 2*x2)^2)/(4*((x - x2)^2 + y^2 + z^2)^(5/2));

dudy_dy = @(x,y,z) w^2 ...
        - m2/((x - x2)^2 + y^2 + z^2)^(3/2) ...
        - m1/((x - x1)^2 + y^2 + z^2)^(3/2) ...
        + (3*m1*y^2)/((x - x1)^2 + y^2 + z^2)^(5/2) ...
        + (3*m2*y^2)/((x - x2)^2 + y^2 + z^2)^(5/2);

dudz_dz = @(x,y,z) ...
          (3*m1*z^2)/((x - x1)^2 + y^2 + z^2)^(5/2) ...
        - m2/((x - x2)^2 + y^2 + z^2)^(3/2) ...
        - m1/((x - x1)^2 + y^2 + z^2)^(3/2) ...
        + (3*m2*z^2)/((x - x2)^2 + y^2 + z^2)^(5/2);

dudx_dy = @(x,y,z) ...
          (3*m1*y*(2*x - 2*x1))/(2*((x - x1)^2 + y^2 + z^2)^(5/2)) ...
        + (3*m2*y*(2*x - 2*x2))/(2*((x - x2)^2 + y^2 + z^2)^(5/2));

dudy_dz = @(x,y,z) ...
          (3*m1*y*z)/((x - x1)^2 + y^2 + z^2)^(5/2) ...
        + (3*m2*y*z)/((x - x2)^2 + y^2 + z^2)^(5/2);

dudz_dx = @(x,y,z) ...
          (3*m1*z*(2*x - 2*x1))/(2*((x - x1)^2 + y^2 + z^2)^(5/2)) ...
        + (3*m2*z*(2*x - 2*x2))/(2*((x - x2)^2 + y^2 + z^2)^(5/2));

dudy_dx = @(x,y,z) dudx_dy(x,y,z);
dudz_dy = @(x,y,z) dudy_dz(x,y,z);
dudx_dz = @(x,y,z) dudz_dx(x,y,z);

x_dot = @(x,y,z,vx,vy,vz) ...
        [   vx; vy; vz; ...
            2*vy + dudx(x,y,z); ...
           -2*vx + dudy(x,y,z); ...
            dudz(x,y,z)];
```

```
F = @(x,y,z) ...
    [0 0 0 1 0 0; ...
     0 0 0 0 1 0; ...
     0 0 0 0 0 1; ...
     dudx_dx(x,y,z) dudx_dy(x,y,z) dudx_dz(x,y,z)  0 2 0;...
     dudy_dx(x,y,z) dudy_dy(x,y,z) dudy_dz(x,y,z) -2 0 0;...
     dudz_dx(x,y,z) dudz_dy(x,y,z) dudz_dz(x,y,z)  0 0 0];

phi_dot = @(yy) F(yy(37),yy(38),yy(39)) * reshape(yy(1:36),6,6);

Y_dot = @(yy) ...
        [   reshape(phi_dot(yy),36,1); ...
            x_dot(yy(37),yy(38),yy(39),yy(40),yy(41),yy(42))];

tspan = 2*[0,P];

options = odeset('RelTol',1e-6,'AbsTol',1e-9);
[t,Y] = ode45(@(t,yy) Y_dot(yy), tspan, Y0, options);

phi_tp = reshape(Y(end,1:36)',6,6);

% sanity check - does the orbit complete 1 revolution at t = P?
max(Y(end,37:42)'-Y0(37:42))

%% compute eigenvalues of the STM at time P
%   to find stable and unstable manifolds

[V,D] = eig(phi_tp);

% we only care about the first two eigenvectors
wu = real(V(:,1)); % unstable manifold
ws = real(V(:,2)); %   stable manifold


%% plot manifolds
n = 10;
t_arr = linspace(0,2*P,n+1);
t_arr(1) = [];
du = 0.001; % disturbance to unstable manifold
```

```
ds = 0.001; % disturbance to stable manifold

% plot unstable manifold
for i = 1:n

    % propagate to a certain timestep
    tspan = [0,t_arr(i)];
    Y0 = [reshape(phi_t0,36,1);x0];
    [~,Y] = ode45(@(t,yy) Y_dot(yy), tspan, Y0, options);
    phi_tp = reshape(Y(end,1:36)',6,6);

    % compute direction of manifolds
    wdir_u = phi_tp * wu;
    wdir_s = phi_tp * ws;

    % normalize directions
    wdir_u = wdir_u / norm(wdir_u);
    wdir_s = wdir_s / norm(wdir_s);

    % get the starting position
    xx = Y(end,37:42)';

    % plot unstable manifold
    xw = xx + du*wdir_u;
    tspan = [0,2*P];
    Y0 = [reshape(phi_t0,36,1);xw];
    [~,Y] = ode45(@(t,yy) Y_dot(yy), tspan, Y0, options);
    figure(1)
    hold on
    plot3(Y(:,37),Y(:,38),Y(:,39),'r','LineWidth',1.2);
    hold off

    % plot stable manifold
    xw = xx + ds*wdir_s;
    tspan = [0,2*P];
    tspan = fliplr(tspan);
    Y0 = [reshape(phi_t0,36,1);xw];
    [~,Y] = ode45(@(t,yy) Y_dot(yy), tspan, Y0, options);
    figure(2)
    hold on
```

```
    plot3(Y(:,37),Y(:,38),Y(:,39),'b','LineWidth',1.2);
    hold off

end

figure(1)
view(3)
hold on
moon = scatter3(x2,0,0,'b','filled');
hold off
grid on
grid minor
title(['Unstable Manifolds, d = ' num2str(du)])
xlabel('x')
ylabel('y')
zlabel('z')
legend(moon,'The Moon','Location','Best')
latexify(16,12,16)

figure(2)
view(3)
hold on
moon = scatter3(x2,0,0,'r','filled');
hold off
grid on
grid minor
title(['Stable Manifolds, d = ' num2str(ds)])
xlabel('x')
ylabel('y')
zlabel('z')
legend(moon,'The Moon','Location','Best')
latexify(16,12,16)
```

Appendix D: Derivation Code for the Pseudo-Potential

```

close all
clear;clc

```

```
syms w x y z x1 x2 m1 m2 real

r1 = sqrt((x-x1)^2+y^2+z^2);
r2 = sqrt((x-x2)^2+y^2+z^2);
U = w^2/2*(x^2+y^2) + m1/r1 + m2/r2;

dudx = simplify(diff(U,x))
dudy = simplify(diff(U,y))
dudz = simplify(diff(U,z))

dudx_dx = simplify(diff(dudx,x))
dudx_dy = simplify(diff(dudx,y))
dudx_dz = simplify(diff(dudx,z))

dudy_dx = simplify(diff(dudy,x))
dudy_dy = simplify(diff(dudy,y))
dudy_dz = simplify(diff(dudy,z))

dudz_dx = simplify(diff(dudz,x))
dudz_dy = simplify(diff(dudz,y))
dudz_dz = simplify(diff(dudz,z))
```