Influence of Solar Acceleration on the Earth-Moon CR3BP

MANE 4100: Spaceflight Mechanics Term Project

Aaryan Sonawane

1 Abstract

In this project, we aim to deepen our understanding of celestial satellite orbits through the detailed study of periodic orbits around the Earth-Moon system Lagrange points. Utilizing data from the Jet Propulsion Laboratory's Solar System Dynamics (JPL SSD) website [1], we will select a known periodic orbit condition and focus on its propagation within an Earth-Moon Barycentered Reference Frame (EMBRF). This study will not only involve plotting one revolution of this selected orbit but also extend to analyzing the perturbative effects exerted by a third body, the Sun, over a designated orbital period. Through this analysis, we intend to calculate the deviations caused by these perturbations from the original periodic orbit. This will be achieved by propagating the "new" governing equations, which include the third-body perturbation, using the initial conditions of the original orbit.

2 Introduction

The dynamical intermingling of different celestial bodies is essential for the field of astrodynamics. The positions of celestial bodies with respect to each other cause various complex forces on not just each other but also small bodies orbiting around these large bodies. These forces are mostly one sided since the mass ratio of these small bodies (satellites) is so less compared to the large bodies applying force on them. These forces are called perturbating forces and will be the focus of this study. There are several factors that determine the magnitude of perturbating forces on a small body and trying to calculate the exact perturbating forces can be an extremely complex task. Hence, there are ways to simplify the equations of motion for the satellite when given the celestial bodies near it. One of the most popular methods of calculating perturbation is via the Circular Restricted 3 Body Problem (CR3BP).

Within a CR3BP system, there exist 5 Lagrange points which are marked in a way that if an object of negligible mass is added to the system at these points, then it will remain stationary compared to the other two large bodies within the system [2]. They are given by L1, L2, L3, L4, L5 and are labelled in the figure below:

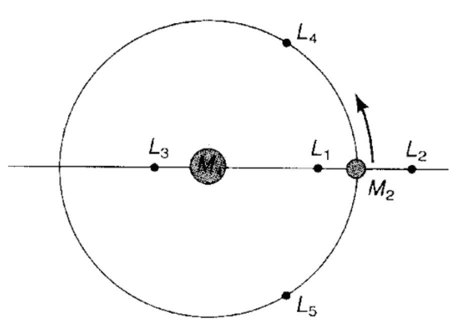


Figure [1]: Illustrates the location of the Lagrange points of an earth-moon system.

L1, L2, and L3 are collinear in nature and are always unstable. The stability of L4 and L5 depends on the mass ratio of the two bodies involved and can be either stable or unstable. NASA's JPL SSD website includes the initial conditions of thousands of orbits, most orbiting around the Lagrange points in the earth-moon system. There are also various types of orbits for the satellite and the type of orbit chosen for the satellite depends on each mission and the goal of it.

The main types of orbits focused on this study were the Northern Halo orbits around L1 and L2, Northern Butterfly and Vertical orbits and are shown in the figure below:

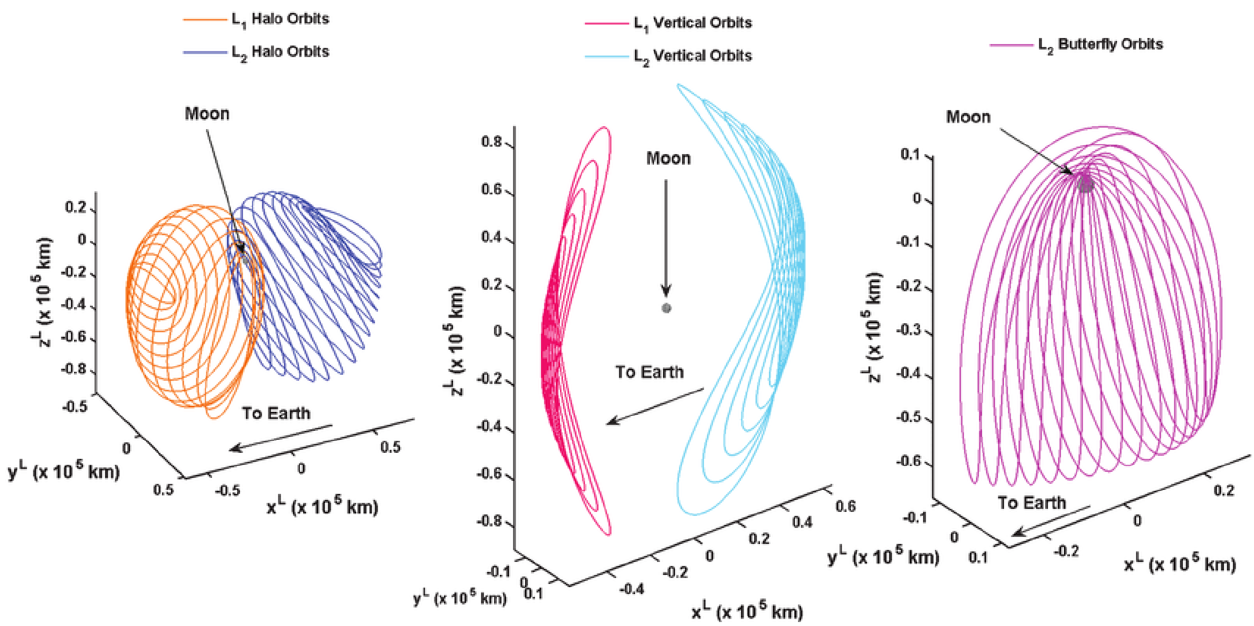


Figure [2]: Illustration of the different orbital families which is the focus of the study.

Halo orbits closely resemble elliptical orbits with their high eccentricity and are extremely useful in long term missions. For example, an "Angelic Halo" orbit was chosen as a base to explore the lunar surface [3]. The shape of this orbit allows for more stable visualization and is great for long term missions. On the other hand, the butterfly and vertical orbits also have certain advantages relating to certain missions due to their shape. For example, butterfly orbits, due to their two lobed shape add design flexibility for insertion and departure maneuvers [4].

3 Theory

The challenging aspect of this project was the analysis of the satellite trajectory along first the earth-moon system and later the perturbating effects of the sun on this orbit. The moon-earth system is modeled as a Circular Restricted 3 Body Problem (CR3BP) in a synodic reference frame with the center being the barycenter of the earth-moon system. The earth and the moon lie on the x axis of this synodic reference frame and the satellite starts with the initial conditions of position and velocity. For the purposes of the CR3BP, it is assumed that the earth and the moon stay stationary at their initial place and only the satellite is moving. The initial conditions of the satellite are given as (x_0, y_0, z_0) and (Vx_0, Vy_0, Vz_0). Note that these initial conditions are taken from the JPL SSD website and are modeled with respect to the moon, not the earth-moon barycenter. They are also normalized using the LU and TU unis to describe length and time respectively. 1 LU is defined as the distance between the earth and the moon equaling 389703km and 1 TU is defined as the inverse of the system's angular frequency equals 382981s. The LU and TU values are usually used to simplify the calculations to track the orbit but, in this case, they are converted back to standard units to fit the synodic frame being used for the problem. This is done by taking the mass ratio of the moon with respect to the earth-moon system, let's call this value μ. Considering the coordinates of the earth-moon barycenter as (0,0,0) then the coordinates of the moon are found out to be $((1 - \mu) * LU, 0, 0)$. Given this, the coordinates of the initial position of the satellite can be found as (x_0+ (1- μ) *LU, y_0, z_0). Since the earth and the moon lie on the x axis of the reference frame used for the problem, the (Vx_0, Vy_0, Vz_0) values remain unchanged except for the conversion of them from LU/TU units to km/s. The given figure illustrates this process.

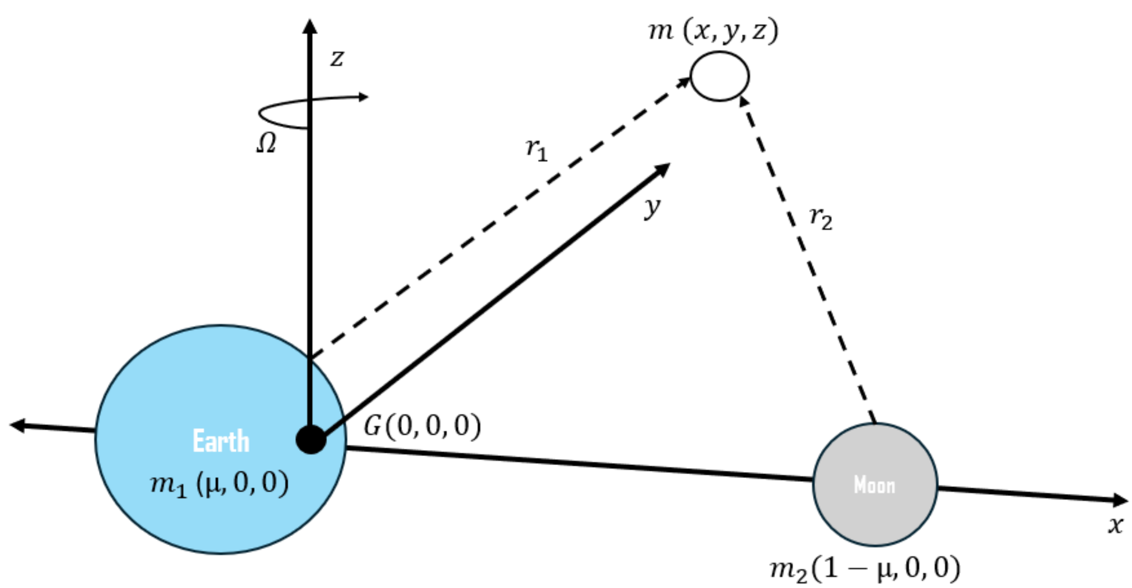


Figure [3]: Visual representation of the Earth-Moon 3 Body Problem

Note: For the purposes of this figure, r_1 & r_2 values are shown from the surface but for the code and in practical purposes, they are always calculated from the barycenter of their respective bodies.

Given the mass and location of both planets and the satellite along with the initial position and velocity of the satellite with respect to the EMBRF, the governing equations of motion for the satellite were found to be [5]:

$$\ddot{x} = 2\dot{y} + x - (1-\mu)\frac{x+\mu}{r_1^3} - (\mu)\frac{x+\mu-1}{r_2^3}$$

$$\ddot{y} = -2\dot{x} + y - (1-\mu)\frac{y}{r_1^3} - (\mu)\frac{y}{r_2^3}$$

$$\ddot{z} = -(1-\mu)\frac{z}{r_1^3} - (\mu)\frac{z}{r_2^3}$$

Equation [1]: Equation of motion for the Satellite in the Earth-Moon system

In the above equations, r_1 describes the distance of the earth from the satellite and r_2 describes the distance of the moon from the satellite, μ is the moon to earth mass ratio given by $\frac{m_2}{m_1+m_2}$. $x, y,\ and\ z$ are the instantaneous coordinates and represent the centrifugal component of the force acting on the satellite due to the rotating frame given by Ω. $\dot{x}\ and\ \dot{y}$ are the instantaneous velocity components of the satellite and represent the Coriolis component of the acceleration. The differential equations are solved using MATLAB's inbuilt ode solver: ode45 and the plot of the orbit is tracked.

The problem complexity increases when you add the sun and its perturbative effects on the satellite. To solve this problem, a new reference frame was introduced to make the problem from a Circular Restricted 3 Body Problem to a Circular Restricted 4 Body Problem (CR4BP). In this new problem, it is assumed that the Earth, Sun and the Moon start in a straight line with the center being the earth-moon barycenter as done previously. Another assumption made is that the earth goes around the Sun in a circular orbit with the period being 365 days (or 3.154e+7s). However, the main difference between the assumptions for the CR3BP and the CR4BP is that the Sun is not assumed to be stationary. Since it is known that one period of an earth orbit around the sun takes 365 days, the sun will be orbiting around the EMBRF in a circular orbit. And the period of this orbit will come out to be 365 days. Using this information, we can find the true anomaly of the Sun with respect to the EMBRF at different time stamps. From the true anomaly, one can find the exact x and y coordinates of the sun using trigonometry in the EMBRF. This logic will be used to find the distance between the sun and the satellite at different time stamps, adding more complexity to the equations of motion and making them more accurate. Keeping this in mind, the new equations of motion for the satellite when the sun perturbating effects are accounted for are given by [6]:

$$a_s = -\mu_s * \frac{\vec{r_3}}{{r^3}_3}$$

Equation [2]: Describes the equation to find total solar acceleration on the satellite.

Where a_s stands for the combined solar acceleration on the satellite, μ_s is the mass ratio of the sun with respect to the earth given by $\frac{m_3}{m_1+m_3}$ and $\overrightarrow{r_3}$ is the distance between the sun and the satellite in the EMBRF given in an array. This is a dynamic quantity, making the solar acceleration a dynamic quantity which changes according to the location of the sun and the satellite. $\overrightarrow{r_3}$ is given by the distance from the Sun to the earth-moon barycenter (given as SunDist) + the location of the satellite with respect to the EMBRF. This means $\overrightarrow{r_3} = (SunDist * \cos(\theta) + x, SunDist * \sin(\theta) + y, z)$ with θ being the true anomaly of the Sun with respect to the EMBRF. Adding all these elements, the new governing equations of motion for the CR4BP become

$$\ddot{x} = \left[2\dot{y} + x - (1-\mu)\frac{x+\mu}{r_1^3} - (\mu)\frac{x+\mu-1}{r_2^3} - \mu_s * \frac{(SunDist * \cos(\theta) + x)}{r_3^3}\right]$$

$$\ddot{y} = \left[-2\dot{x} + y - (1-\mu)\frac{y}{r_1^3} - (\mu)\frac{y}{r_2^3} - \mu_s * \frac{(SunDist * \sin(\theta) + y)}{r_3^3}\right]$$

$$\ddot{z} = \left[-(1-\mu)\frac{z}{r_1^3} - (\mu)\frac{z}{r_2^3} - \mu_s * \frac{z}{r_3^3}\right]$$

Equation [3]: New governing equations of motion for the satellite with added Solar Acceleration

Along with the new equations of motions, the updated diagram for the CR4BP becomes:

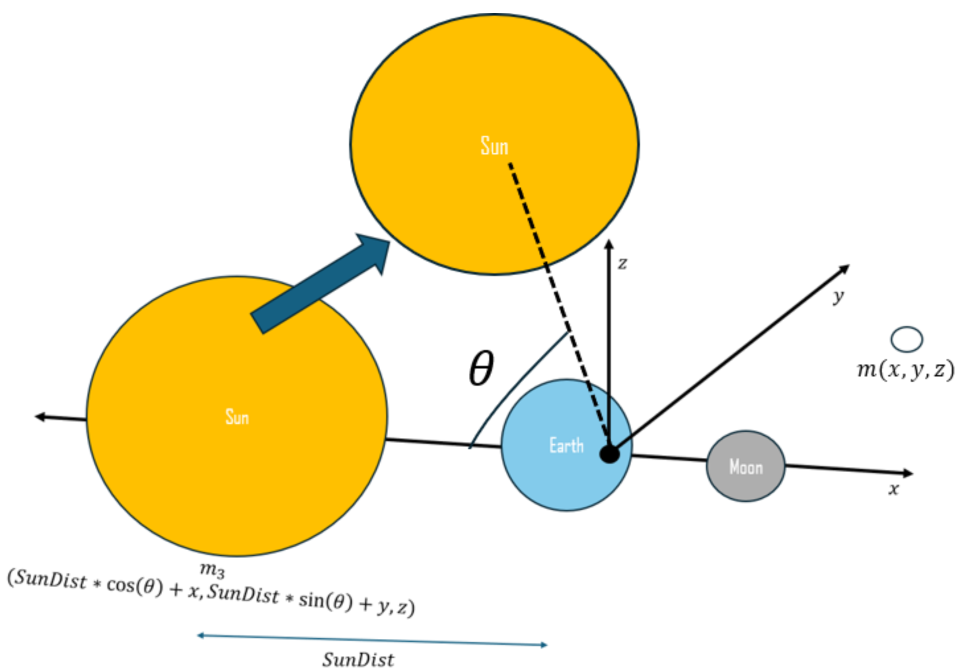


Figure [4]: Visual representation of the Sun-Earth-Moon CR4BP

4 Methodology

To implement the above equations and simulate the orbit of the satellite, two main MATLAB scripts were used. The first script (Original_Orbit.m) tracks the orbit of the CR3BP within the earth-moon system and the (Solar_Perturb.m) script tracks the solar perturbed orbit. Both scripts are fed with the given initial conditions of the satellite and MATLAB's ode solver ode45 is used to integrate the derived equations of motion for varying periods of the satellite using the tspan function in MATLAB.

```
% Initial Conditions
x0 = 8.97E-01; % LU
y0 = -7.61E-27;
z0 = 1.99E-01;
vx0 = -5.23E-14; % LU/TU
vy0 = 1.91E-01;
vz0 = 2.22E-13;

% Earth-Moon System Constants
mu = 1.215058560962404E-2;
LU = 389703;
TU = 382981;)

% Initial state vector
X0 = [x0, y0, z0, vx0, vy0, vz0];

% Time span for the simulation
periodTU = 1.95E+00; % Period in TU
tspan = [0, periodTU * 1 * pi];

% Solve the CR3BP equations using ode45
options = odeset('RelTol',1e-12,'AbsTol',1e-12);
[t, X] = ode45(@(t,X) cr3bpEOM(mu, X), tspan, X0, options);
```

Equation [4]: Shows the initial conditions/arguments of the Original_orbit.m script.

```
% Function to Calculate the CR3BP Equations of Motion
function dxdt = cr3bpEOM(mu, X)
    % Unpack the state vector
    x = X(1);
    y = X(2);
    z = X(3);
    xdot = X(4);
    ydot = X(5);
    zdot = X(6);

    % Distance to the primary body (Earth) and secondary body (Moon)
    r1 = sqrt((x + mu)^2 + y^2 + z^2); % Distance to the Earth
    r2 = sqrt((x - 1 + mu)^2 + y^2 + z^2); % Distance to the Moon

    % Initialize the derivative of the state vector
    dxdt = zeros(6,1);

    % CR3BP Equations of Motion
    dxdt(1) = xdot;
    dxdt(2) = ydot;
    dxdt(3) = zdot;
    dxdt(4) = 2*ydot + x - (1-mu)*(x+mu)/r1^3 - mu*(x-1+mu)/r2^3;
    dxdt(5) = -2*xdot + y - (1-mu)*y/r1^3 - mu*y/r2^3;
    dxdt(6) = -(1-mu)*z/r1^3 - mu*z/r2^3;
end
```

Equation [5]: The differential equation function used to simulate the orbit of the satellite.

When the code is run, the plot of one period of the orbit looks exactly how it is supposed to be on the JPL SSD website.

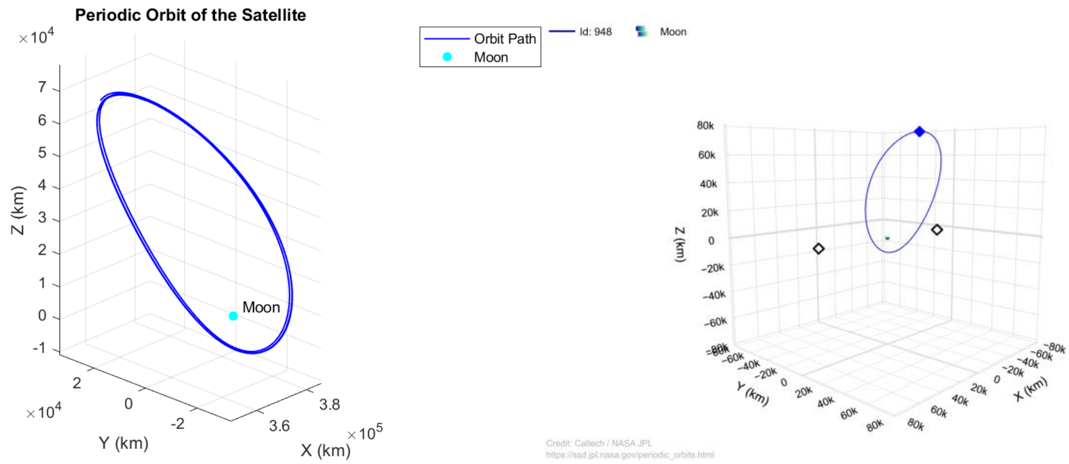


Figure [5]: Comparison of the simulated orbit vs the simulated Northern Halo orbit ID 948 around L1 on the JPL SSD website.

The implementation of the Solar_perturb.m script is slightly more complex with not only the addition of the solar acceleration vector but also because of an extra differential function that calculates the true anomaly of the sun with respect to the earth and finding its dynamic coordinates. Here, the main domain for the true anomalies is going to be $[0, 2\pi]$ and the angular velocity of the sun with respect to the EMBRF is given as $\frac{2\pi}{3.154e+7}$. Using this logic, one can find the true anomaly of the sun and accordingly its coordinates with respect to the EMBRF. The function to calculate the coordinates of the sun is given below:

```
function [sunX, sunY] = calculateSunPosition(t, sunInitialDist, LU)
    % Constants
    orbitalPeriod = 3.154e+7; % Seconds in a year
    angularSpeed = 2*pi / orbitalPeriod; % Radians per second

    % Calculate the true anomaly (angle) of the Sun at time t
    angle = mod(angularSpeed * t, 2*pi);

    % Calculate Sun's position using trigonometry and normalize distance
    sunX = sunInitialDist / LU * cos(angle);
    sunY = sunInitialDist / LU * sin(angle);
end
```

Equation [6]: Function to calculate the dynamic Sun coordinates in the EMBRF.

5 Results & Discussion

Most of the orbits in the NASA JPL SSD database are unstable orbits and deviate from their path after one full period. The orbit ID 948 around the Lagrange point L1 in the earth-moon system is used in the first simulation to test the perturbative effects of the sun. This is an extremely stable orbit with a stability index of 1.000. The perturbative effects on a stable orbit after one period are as shown in the figure below:

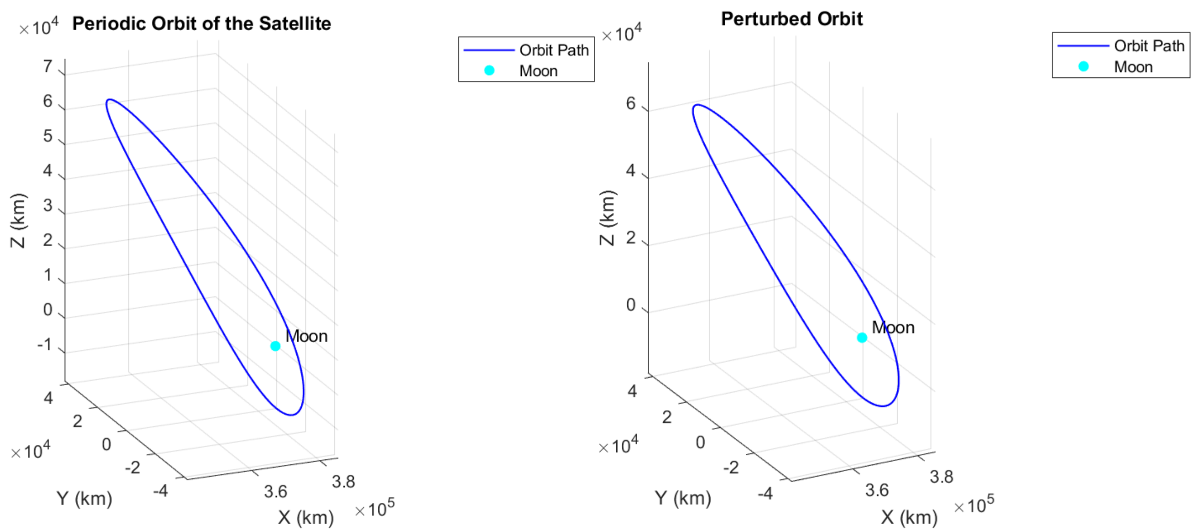


Figure [6]: Perturbative effects on a Northern Halo stable orbit ID: 948 around L1

As seen, the perturbative effects on a stable orbit are extremely negligible after just 1 period of this orbit. But, with more periods, the chaos within the system increases and the perturbative effects are more visible. After 50 periods, the orbit shape for the stable orbit looks extremely different to its original and the solar perturbative effects are clearly visible:

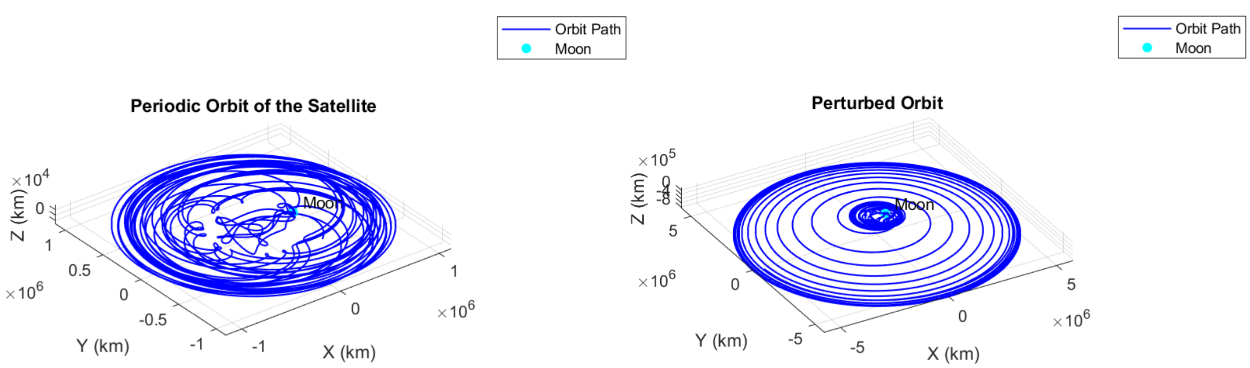


Figure [7]: Perturbative effects on a Northen Halo stable orbit after 50 periods ID: 948 around L1

As seen in the plot, the path of the perturbed orbit follows a more uniform shape than the path of the unperturbed orbit.

Even in the orbits with a high stability index, meaning the orbits are unstable, the perturbative effects on one period are extremely negligible. Below is a comparison of an unstable orbit ID:882 with a stability index of 114.6 which is an extremely unstable orbit.

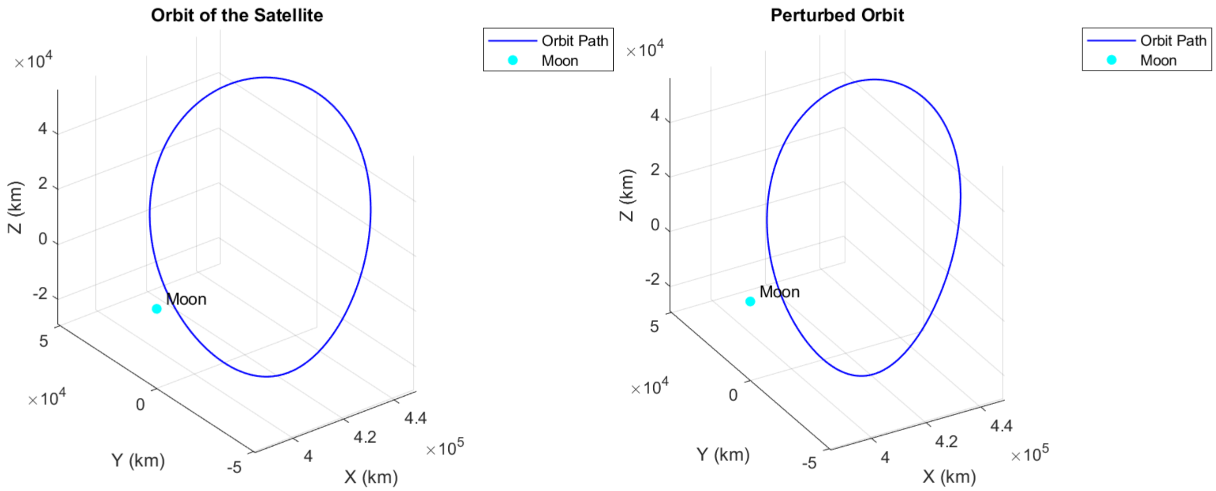


Figure [8]: Perturbative effects on a Norther Halo unstable orbit ID: 882 around L2

As seen in the comparative plots, there are slight changes in the overall Halo shape of the orbit but no significant changes. However, as seen with the previous orbit, the changes are more apparent when the number of periods increase. A noticeable difference between the stable and unstable orbit is that the stable orbit passes near the moon on multiple occasions for both the perturbed and unperturbed orbits whereas the perturbed unstable orbit does not pass close to the moon closely multiple times but only once. However, the unperturbed orbit passes close to the moon multiple times as given in the plot below:

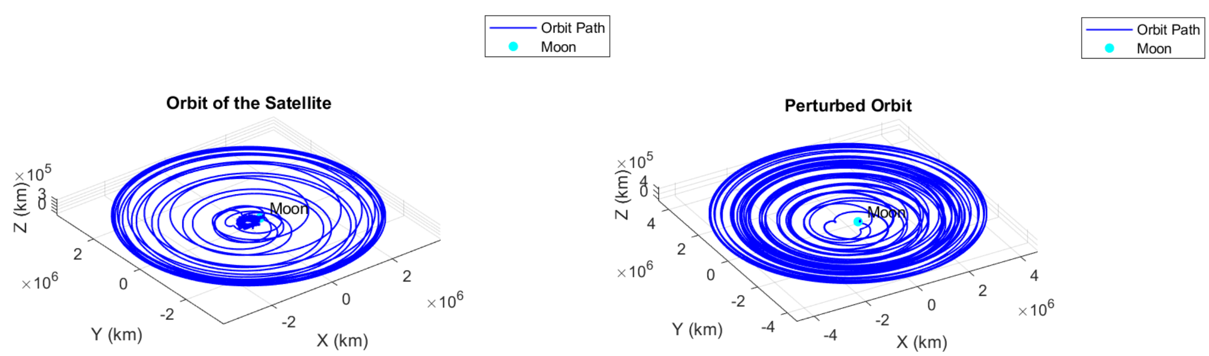


Figure [9]: Perturbative effects of a Northern Halo unstable orbit after 50 periods ID:882 around L2

The perturbative effects on a Northern Butterfly show a different picture compared to the Halo orbits. In this simulation, the orbit has a stability index of 207 which means it is an extremely unstable orbit. But, compared to the Halo orbits where the unperturbed orbits go extremely close to the moon multiple times, the butterfly orbits are always inching close to the moon. As shown in the figure below, the perturbed orbits still try to form an outward spiral from the moon whereas the unperturbed orbits almost get sucked into the earth-moon system.

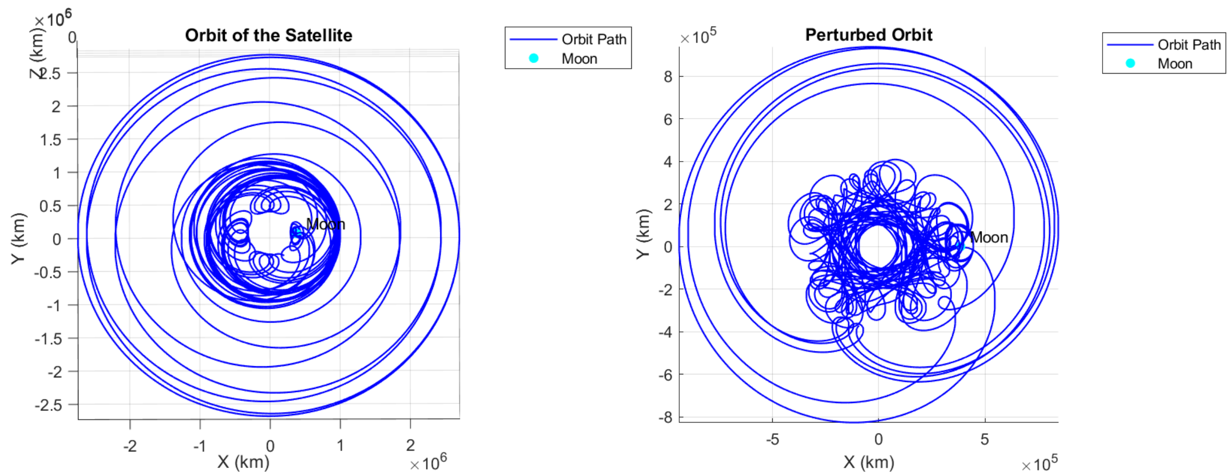


Figure [10]: Perturbative effects on a Northern Butterfly after 60 periods ID:66 shown in a top-down view for better visualization.

6 Conclusion & Future Work

With more time and resources, it would be optimal to track the Sun and moon's exact ephemeris position in a file outside the MATLAB script. NASA's JPL SSD website has a section to track the ephemeris data of different celestial bodies with respect to each other. The position of the Sun and Moon can be tracked on different time stamps. The inclination and eccentricity of the Sun_Earth and Earth_Moon orbits will also be accounted for by the ephemeris data. For simplicity, the earth was assumed to be in a circular orbit around the moon and the position of the moon was assumed to be stationary with respect to the EMBRF. However, this process consumes an incredible amount of data, making the simulation extremely time and resource consuming. This would be unfeasible given the resources and time on hand. But that is something that is of high interest to me.

Modelling a multibody system can be an extremely complex and tedious task. Although not 100% accurate, this project models the CR3BP and CR4BP as done in class. And the results really show how unstable lunar orbits can be. For real world applications, ephemeris data has to be taken into account for but for purposes of a simulation the CR3BP and CR4BP are a great starting point to model the orbit of a satellite.

7 References

[1] NASA Jet Propulsion Laboratory. (n.d.). Periodic orbits tool. Retrieved April 12, 2024, from https://ssd.jpl.nasa.gov/tools/periodic_orbits.html#/periodic

[2] Puckett, C. (2023). Solar Gravity Influences on Earth-Moon Three-Body Periodic Orbits. MANE 4100 Term Project, Spring 2023.

[3] European Space Agency. (2019, July 18). Angelic halo orbit chosen for humankind's first lunar outpost. ESA. Retrieved from https://www.esa.int/Enabling_Support/Operations/Angelic_halo_orbit_chosen_for_humankind_s_first_lunar_outpost.

[4] May, Z. D., Qu, M., & Merrill, R. G. (2020). Enabling global lunar access for human landing systems staged at Earth-Moon L2: Southern Near Rectilinear Halo and Butterfly Orbits. Analytical Mechanics Associates, Hampton, VA, 23666, USA; NASA Langley Research Center, Hampton, VA, 23681, USA.

[5] Curtis, Howard D. Orbital Mechanics for Engineering Students. Fourth edition, Butterworth-Heinemann, 2020

[6] Williams, T., Palmer, E., Hollister, J., Godine, D., Ottenstein, N., & Burns, R. (2019). Lunisolar perturbations of high-eccentricity orbits such as the Magnetospheric Multiscale mission. Paper presented at the AAS/AIAA Space Flight Mechanics Meeting, Portland, ME

8 Appendix

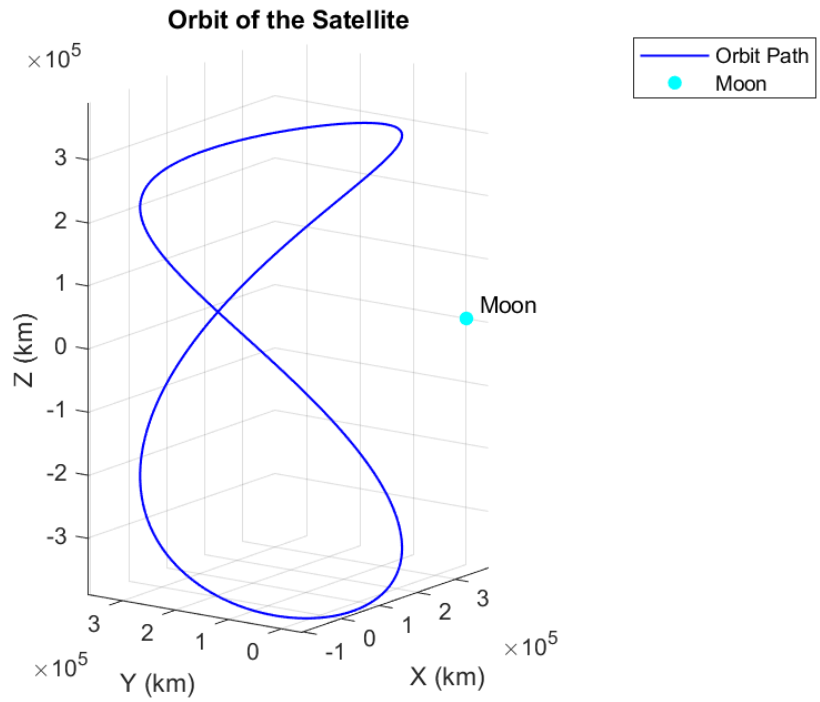


Figure [11]: Depicting a Vertical orbit around L4. It is found that orbits around the stable Lagrange points L4 and L5 are extremely stable in nature.

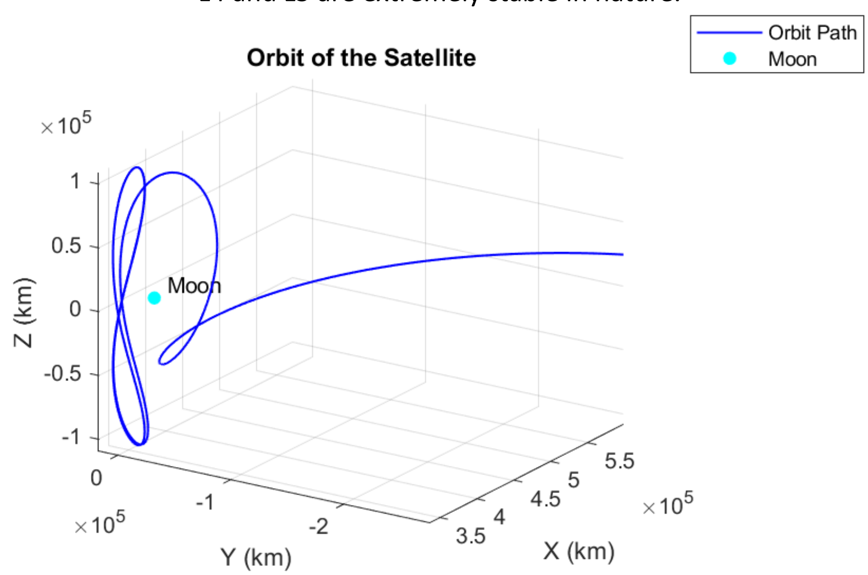
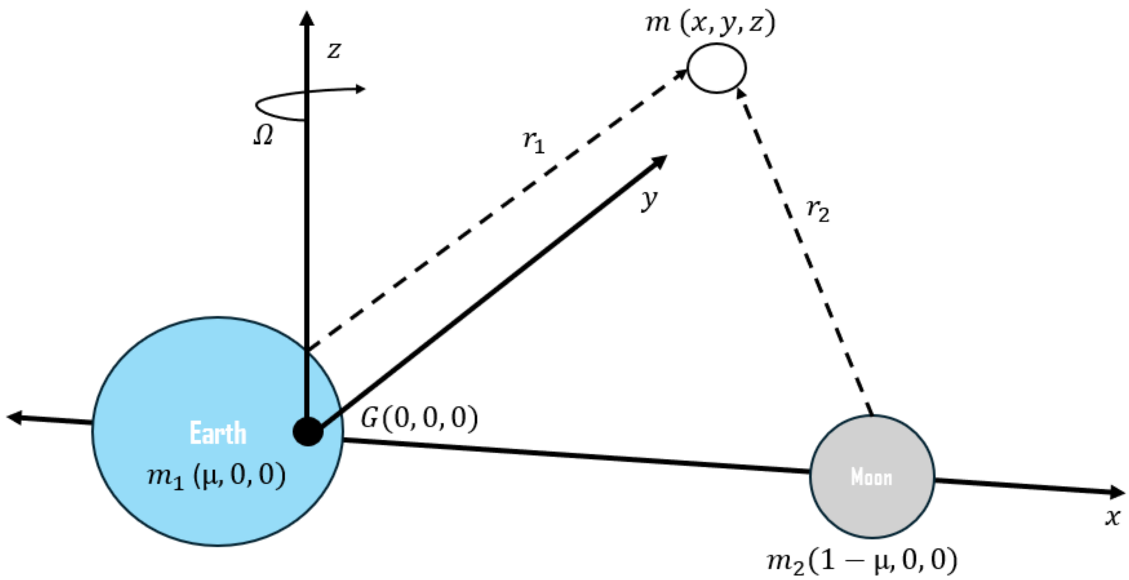
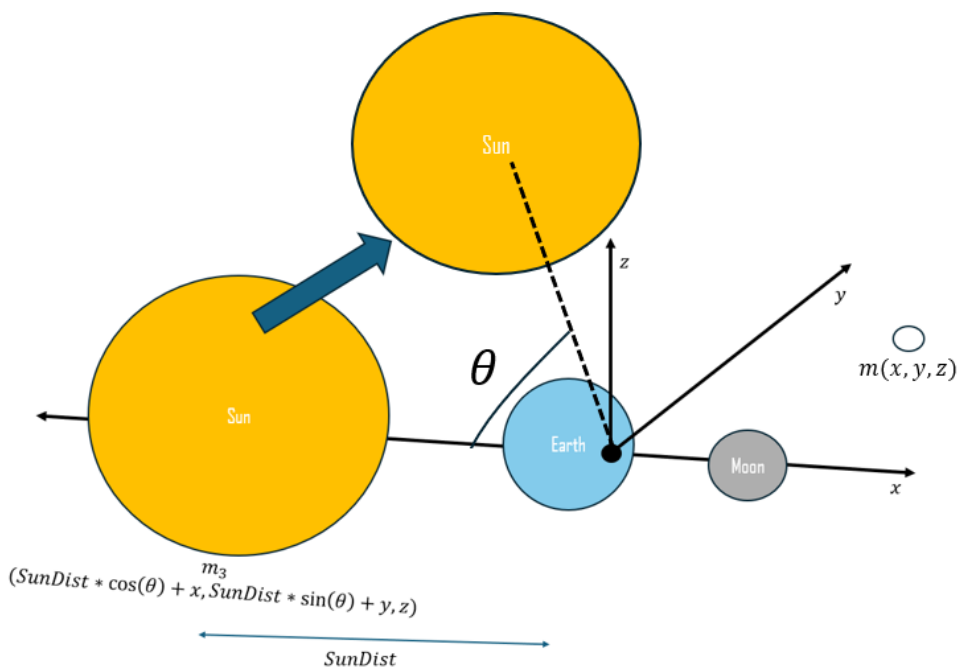


Figure [12]: Depicts another vertical orbit swaying out of its path during the third period.





Figures [3 and 4]: Depict the CR3BP and CR4BP visually.

MATLAB Code

For the scope of this project, three main MATLAB scripts were used. They are given by original_orbits.m, Solar_perturbed.m and Perturbed.m. Original_orbits.m is used to plot the orbit of the satellite in the CR3BP Earth-Moon system, Solar_perturbed.m is used to plot the orbit of the satellite where the sun is assumed to be a moving entity with respect to the EMBRF and perturbed.m is used to plot the perturbed orbit in case solar_perturbed is too data consuming due to its complexity. The scripts are given below:

Original_orbits.m

```
% Updated Initial Conditions
x0 = 7.5681291130804940E-1; % LU
y0 = 4.2544999999999999E-1;
z0 = 4.8666229898485008E-1;
vx0 = 2.3572429286298373E-2; % LU/TU
vy0 = -1.0293772889188169E+0;
vz0 = 8.7170581363331578E-1;

% Earth-Moon System Constants
mu = 1.215058560962404E-2;
LU = 389703;
TU = 382981;

% Initial state vector
X0 = [x0, y0, z0, vx0, vy0, vz0];

% Time span for the simulation
periodTU = 1.95E+00; % Period in TU
tspan = [0, periodTU * 4 * pi];

% Solve the CR3BP equations using ode45
options = odeset('RelTol',1e-12,'AbsTol',1e-12);
[t, X] = ode45(@(t,X) cr3bpEOM(mu, X), tspan, X0, options);

% Define Lagrange Points positions in LU (converted to km)
L1 = [0.83691513, 0, 0] * LU;
L2 = [1.15568217, 0, 0] * LU;

% Plot the orbit and Lagrange points
figure;
plot3(X(:,1)*LU, X(:,2)*LU, X(:,3)*LU, 'b-', 'LineWidth', 1);
hold on;

% Highlight the Moon's position at (1 - mu)
Moonpos = (1 - mu) * LU;
scatter3(Moonpos, 0, 0, 'filled', 'MarkerFaceColor', 'c');
text(Moonpos, 0, 0, '  Moon', 'VerticalAlignment', 'bottom');

% Axes and title
title('Orbit of the Satellite');
xlabel('X (km)');
ylabel('Y (km)');
zlabel('Z (km)');
```

```
grid on;
axis equal;

% Label only for the Moon and Lagrange points L1 and L2
legend('Orbit Path', 'Moon', 'L1', 'L2');

hold off; % Release the plot

% Function to Calculate the CR3BP Equations of Motion
function dxdt = cr3bpEOM(mu, X)
    % Unpack the state vector
    x = X(1);
    y = X(2);
    z = X(3);
    xdot = X(4);
    ydot = X(5);
    zdot = X(6);

    % Distance to the primary body (Earth) and secondary body (Moon)
    r1 = sqrt((x + mu)^2 + y^2 + z^2);
    r2 = sqrt((x - 1 + mu)^2 + y^2 + z^2);

    % Initialize the derivative of the state vector
    dxdt = zeros(6,1);

    % CR3BP Equations of Motion
    dxdt(1) = xdot;
    dxdt(2) = ydot;
    dxdt(3) = zdot;
    dxdt(4) = 2*ydot + x - (1-mu)*(x+mu)/r1^3 - mu*(x-1+mu)/r2^3;
    dxdt(5) = -2*xdot + y - (1-mu)*y/r1^3 - mu*y/r2^3;
    dxdt(6) = -(1-mu)*z/r1^3 - mu*z/r2^3;
end
```

Solar_perturbed.m

```
% Earth-Moon System Constants
mu = 1.215058560962404E-2;
LU = 389703;
TU = 382981;
mus = 3.003e-6; % Mass ratio of the Sun to the Earth
sunInitialDist = 147108099;

% Initial state vector
X0 = [8.97E-01, -7.61E-27, 1.99E-01, -5.23E-14, 1.91E-01, 2.22E-13];

% Time span for the simulation
periodTU = 1.95E+00; % Period in TU
tspan = [0, periodTU * TU]; % Simulate for one period

% Solve the CR3BP equations using ode45 with solar perturbation
options = odeset('RelTol',1e-12,'AbsTol',1e-12);
[t, X] = ode45(@(t,X) cr3bpEOMwithSun(mu, X, LU, t, sunInitialDist, mus), tspan, X0,
options);
```

```
% Plot the orbit and Lagrange points
figure;
plot3(X(:,1)*LU, X(:,2)*LU, X(:,3)*LU, 'b-', 'LineWidth', 2); % Blue line for the
orbit
hold on;

% Highlight the Moon's position at (1 - mu)
Moonpos = (1 - mu) * LU;
scatter3(Moonpos, 0, 0, 'filled', 'MarkerFaceColor', 'c'); % Cyan marker for the Moon
text(Moonpos, 0, 0, '  Moon', 'VerticalAlignment', 'bottom');

% Axes and title
title('Periodic Orbit around the Earth-Moon L1 Lagrange Point with Solar
Perturbation');
xlabel('X (km)');
ylabel('Y (km)');
zlabel('Z (km)');
grid on;
axis equal;

% Label only for the Moon
legend('Orbit Path', 'Moon');

hold off; % Release the plot

function dxdt = cr3bpEOMwithSun(mu, X, LU, t, sunInitialDist, mus)
    % Calculate Sun's position at time t
    [sunX, sunY] = calculateSunPosition(t, sunInitialDist, LU);

    % Unpack the state vector
    x = X(1);
    y = X(2);
    z = X(3);
    xdot = X(4);
    ydot = X(5);
    zdot = X(6);

    % Calculate distances
    r1 = sqrt((x + mu)^2 + y^2 + z^2);
    r2 = sqrt((x - 1 + mu)^2 + y^2 + z^2);
    r3 = sqrt((x - sunX)^2 + (y - sunY)^2 + z^2);

    % CR3BP equations with solar perturbation
    dxdt = zeros(6,1);
    dxdt(1) = xdot;
    dxdt(2) = ydot;
    dxdt(3) = zdot;
    dxdt(4) = 2*ydot + x - (1-mu)*(x+mu)/r1^3 - mu*(x-1+mu)/r2^3 - mus*(x-sunX)/r3^3;
    dxdt(5) = -2*xdot + y - (1-mu)*y/r1^3 - mu*y/r2^3 - mus*(y-sunY)/r3^3;
    dxdt(6) = -(1-mu)*z/r1^3 - mu*z/r2^3 - mus*z/r3^3;
end

function [sunX, sunY] = calculateSunPosition(t, sunInitialDist, LU)
    % Constants
```

```
    orbitalPeriod = 3.154e+7; % Seconds in a year
    angularSpeed = 2*pi / orbitalPeriod; % Radians per second

    % Calculate the true anomaly (angle) of the Sun at time t
    angle = mod(angularSpeed * t, 2*pi);

    % Calculate Sun's position using trigonometry and normalize distance
    sunX = sunInitialDist / LU * cos(angle);
    sunY = sunInitialDist / LU * sin(angle);
end
```

Perturbed.m

```
function earth_moon_satellite_perturbation_plot
    clear all; clc;

    % Earth-Moon System Constants
    mu = 1.215058560962404E-2; % Mass ratio of the Moon
    LU = 389703; % Length Unit (km)
    TU = 382981; % Time Unit (s)


    % Initial Conditions
    x0 = 8.6387487687066056E-1; % LU
    y0 = 3.5177128990586685E-28;
    z0 = -1.6087122416233736E-13;
    vx0 = 1.8717491648877084E-13; % LU/TU
    vy0 = 1.0286165131804773E-1;
    vz0 = -4.8075882510399448E-1;

    X0 = [x0, y0, z0, vx0, vy0, vz0];

    % Time span for the simulation
    periodTU = 1.95E+00; % Period in TU
    tspan = [0, periodTU * 2 * pi];

    % Solve the CR3BP equations using ode45
    options = odeset('RelTol',1e-12,'AbsTol',1e-12);
    [t, X] = ode45(@(t,X) cr3bpEOM(mu, X, LU), tspan, X0, options);

    % Plot the orbit and Lagrange points
    figure;
    plot3(X(:,1)*LU, X(:,2)*LU, X(:,3)*LU, 'b-', 'LineWidth', 1);
    hold on;

    % Highlight the Moon's position at (1 - mu)
    Moonpos = (1 - mu) * LU;
    scatter3(Moonpos, 0, 0, 'filled', 'MarkerFaceColor', 'c');
    text(Moonpos, 0, 0, '  Moon', 'VerticalAlignment', 'bottom');

    % Plot the orbit with cyan color for the path
    plot3(X(:,1)*LU, X(:,2)*LU, X(:,3)*LU, 'b-', 'LineWidth', 1); % Cyan line for the
orbit
    hold on;
```

```
    % Set the view angle similar to the one in the image
    view([-30, 30]);

    % Axes and title
    title('Perturbed Orbit');
    xlabel('X (km)');
    ylabel('Y (km)');
    zlabel('Z (km)');
    grid on;
    axis equal;

    % Legend for orbit and start position
    legend('Orbit Path', 'Moon');

    hold off; % Release the plot
end

function dxdt = cr3bpEOM(mu, X, LU)
    % Unpack the state vector
    x = X(1);
    y = X(2);
    z = X(3);
    xdot = X(4);
    ydot = X(5);
    zdot = X(6);

    % Earth-Moon-Sun System Constants
    mus = 3.003e-6; % Mass ratio of the Sun to the Earth
    sunDist = 149.59e6 / LU; % Approximate SunDist in normalized units

    % Calculate distances
    r1 = sqrt((x + mu)^2 + y^2 + z^2);
    r2 = sqrt((x - 1 + mu)^2 + y^2 + z^2);
    r3 = sqrt((x - sunDist)^2 + y^2 + z^2);

    % Include the solar perturbation in the equations of motion
    dxdt = zeros(6,1);
    dxdt(1) = xdot;
    dxdt(2) = ydot;
    dxdt(3) = zdot;
    dxdt(4) = 2*ydot + x - (1-mu)*(x+mu)/r1^3 - mu*(x-1+mu)/r2^3 - mus*(x-
sunDist)/r3^3;
    dxdt(5) = -2*xdot + y - (1-mu)*y/r1^3 - mu*y/r2^3 - mus*y/r3^3;
    dxdt(6) = -(1-mu)*z/r1^3 - mu*z/r2^3 - mus*z/r3^3;
end
```

With the combination of these three scripts, one can simulate any orbit from the JPL SSD database by simply inputting the initial conditions.