Simulation and Stability of Lagrange Point L1

Vinay Kumar

April 17, 2021

Abstract

In this project, the location of the Lagrange Point L1 for a two body system was numerically calculated. A small third body kept at L1 with a suitable initial velocity was simulated using Python and its orbit was observed. The stability of L1 was studied using contour plots.

1 Introduction

For a given two-body system (a Star, S and a Planet, P), there exist special points in space, where, if a small object (a Satellite, Sat) is kept, it exists in a state of equilibrium and orbits the heavier body (S) with the same time period with which the lighter body (P) orbits the heavier body. At Lagrange points, the gravitational pull of two large masses precisely equals the centripetal force required for a small object to move with them.
These points in space can be used by spacecraft to reduce fuel consumption needed to remain in position. These points are called Lagrange Points in honor of Joseph-Louis Lagrange who wrote about them in his paper titled "Essai sur le Problème des Trois Corps" (Essay on the Three-Body Problem)[1].
In totality, there are five such points as shown in the figure below. However, my project will be concerned with only one of them, L1, which lies between the two bodies. I intend to find, numerically, the position of L1, simulate the orbit of a satellite placed at L1 and analyze its stability

2 The Code

To minimize computational errors while working with extremely small numbers, a system of units was chosen in which the value of the Gravitational Constant becomes unity. For the same reason, the mass of the Star, M_S was taken to be 10000 units, that of the planet, M_P was taken as 10 units and that of the satellite, M_{Sat} was taken as 0.01 units. The planet was placed at a distance, $R = 1$ unit from the Star. Circular orbits were considered and using $\omega = \sqrt{\frac{GM_S}{R^3}}$, the angular velocity of the planet, $\omega = 100$ units.

2.1 Location of L1

The magnitude of force was defined to a function.

```
G = 1
def f(r):
    f=G*(10000.0/r**2)-G*(10.0/(1-r)**2)-10000.0*r
    return f
```

Bisection method was used to find its root in between S and P.

```
import numpy as np

def bisect(a,b,tol):
    x1=min(a,b)
    x2=max(b,a)
    c=x1
    while np.abs(f(c))>tol:
        c=(x1+x2)/2.0
        p=f(x1)*f(c)
        if p>=0: x1=c
        elif p<0: x2=c
    return c

r=bisect(0.1,1,1e-11)
print("The Satellite should be placed at r = {0} from the Star.".format(r))
```

So the location of the Lagrange Point L1 was calculated to be as follows.

```
Lagrange Point L1 is at a distance of 0.9323021625868072 units from S.
```

2.2 Simulation

VPython was used to simulate and animate the Star, the Planet and the Satellite.
The gravitational force between two bodies was defined vectorially as below.

```
#define gravitational force
def gforce(p1,p2):
    # Calculate distance vector between p1 and p2.
    r_vec = p1.pos-p2.pos
    # Calculate magnitude of distance vector.
    r_mag = mag(r_vec)
    # Calcualte unit vector of distance vector.
    r_hat = r_vec/r_mag
    # Calculate force magnitude.
    force_mag = G*p1.mass*p2.mass/r_mag**2
    # Calculate force vector.
    force_vec = -force_mag*r_hat

    return force_vec
```

The objects were plotted and their motion was simulated. The forces, momenta and positions were updated in each iteration.

```
#plot initial positions
from vpython import *
scene = canvas(autoscale=False, range=1.2)

star = sphere( pos=vector(0,0,0), radius=0.2, color=color.yellow,
               mass = 10000.0, momentum=vector(0,0,0), make_trail=True )
planet = sphere( pos=vector(1,0,0), radius=0.03, color=color.blue,
               mass = 10.0, momentum=vector(0,1000.0,0), make_trail=True )
sat = sphere( pos=vector(r,0,0), radius=0.02, color=color.white,
               mass = 0.01, momentum=vector(0,r,0), make_trail=True )

Lsat = label(pos=0.9*sat.pos, text='Sat', space=20, height=10, border=4,font='
    sans')
Lp = label(pos=1.1*planet.pos, text='P', space=20, height=10, border=4,font='
    sans')
Ls = label(pos=star.pos, text='S',xoffset=20, yoffset=20, space=20, height=10,
    border=4,font='sans')

dt = 0.00001
t = 0
i=0
while (i<6283):
    rate(500)

    # Calculate forces.
    planet.force = gforce(planet,star)
    sat.force = gforce(sat,star)+gforce(sat,planet)

    # Update momenta.
    planet.momentum = planet.momentum + planet.force*dt
    sat.momentum = sat.momentum + sat.force*dt

    # Update positions.
    planet.pos = planet.pos + (planet.momentum/planet.mass)*dt
    sat.pos = sat.pos + (sat.momentum/sat.mass)*dt
    Lsat.pos=0.9*sat.pos
    Lp.pos=1.1*planet.pos
    Ls.pos=star.pos

    t = t + dt
    i=i+1
```

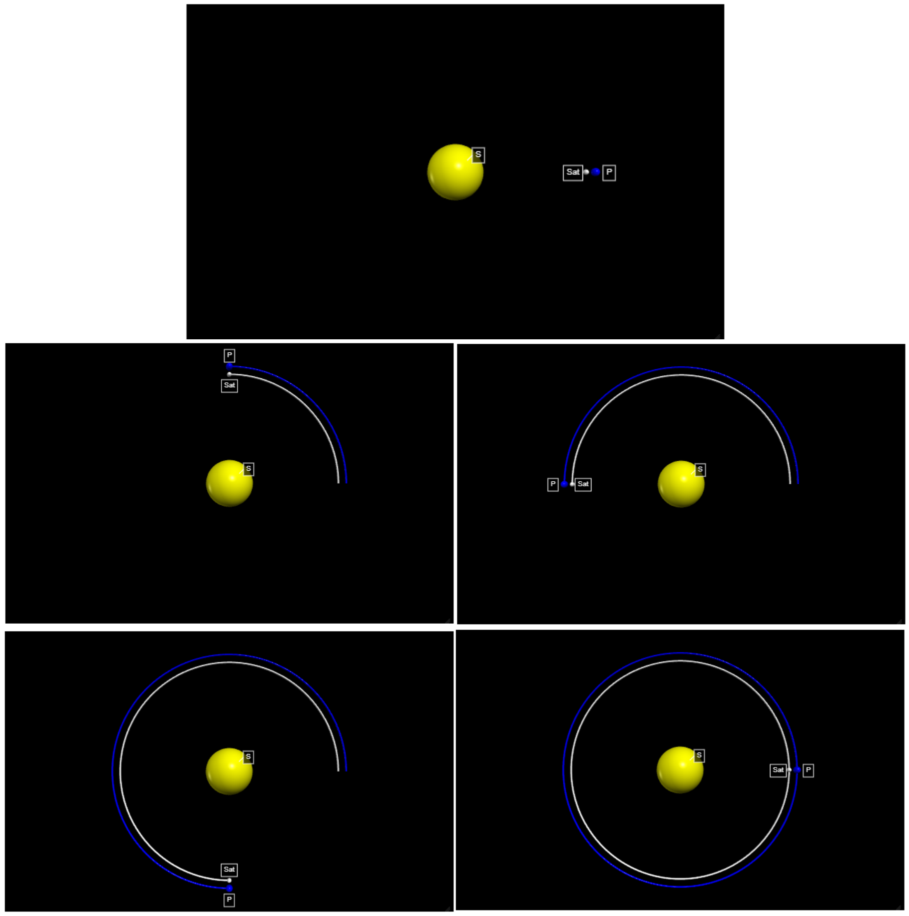

Figure 1: Screenshots of Simulated Orbits of the Planet and the Satellite

2.3 Plotting Effective Potential and Lines of Force

An effective potential was defined.

```
#plot potential
import matplotlib
from mpl_toolkits import mplot3d
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

#define effective potential
def pot(x, y):
    R1=np.sqrt(x**2+y**2)
    R2=np.sqrt((x-1)**2+y**2)
    return -1*(+0.5*10000*R1**2+G*10000/R1+G*10/(R2))
```

The potential was plotted as contour plots with different scales and ranges and corresponding lines of force were also plotted.

```
X = np.linspace(-1.5, 1.5, 250)
Y = np.linspace(-1.5,1.5, 250)
x, y = np.meshgrid(X, Y)
Z = pot(x, y)

plt.figure()
levels_h = np.arange(-20000,-15300,250)
levels_w=np.append(levels_h,pot(r,0))#-15219.788044152605)
levels_l=np.arange(-15200,-15000,250)
levels=np.append(levels_w,levels_l)
C = plt.contour(x, y, Z, levels=(levels))
plt.colorbar()
plt.plot(r,0,'rx', label="L1")
plt.plot(0,0,'gx', label="S")
plt.plot(1,0,'mx', label="P")
plt.xlim(-1.5, 1.5)
plt.ylim(-1.5,1.5)
plt.gca().set_aspect("equal")
plt.legend()
plt.title("Effective Potential Contour Plot")
plt.show()
```

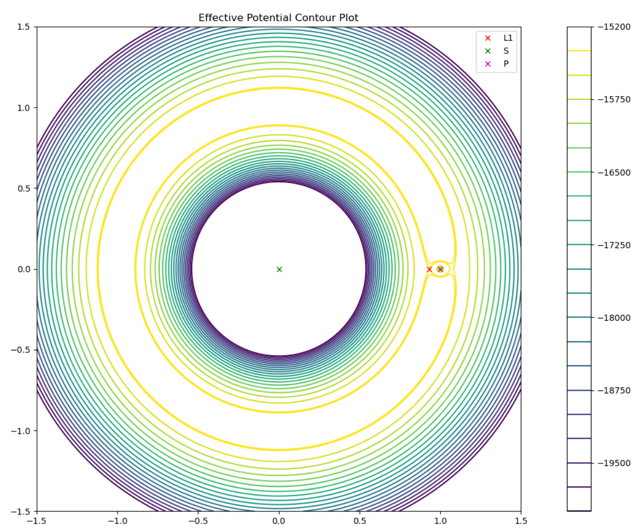

Figure 2: Contour Plot for Effective Potential

```
X = np.linspace(0.8, 1.2, 250)
Y = np.linspace(-0.2,0.2,250)
x, y = np.meshgrid(X, Y)
Z = pot(x, y)
u=-1*((10000*G*x)/(x**2+y**2)**(3/2)-10000*x-(10*G*(1-x))/((1-x)**2+y**2)**(3/2))
v=-1*((10000*G*y)/(y**2+x**2)**(3/2)+(10*G*y)/(y**2+(1-x)**2)**(3/2)-10000*y)
f=np.sqrt(u**2 + v**2)
UN = u/f
VN = v/f

plt.figure()
levels_h = np.arange(-15750,-15300,50)
levels_w=np.append(levels_h,pot(r,0))
levels_l=np.arange(-15200,-15000,50)
levels=np.append(levels_w,levels_l)
cfz = plt.contourf(x, y, Z, levels=(levels), cmap = "autumn")
plt.colorbar()
plt.plot(r,0,'rx', label="L1")
plt.plot(1,0,'mx', label="P")
plt.xlim(0.8, 1.2)
plt.ylim(-0.2,0.2)
plt.gca().set_aspect("equal")
plt.title("Effective Potential Contour Plot with Lines of Force")
plt.streamplot(x, y, UN, VN)
plt.legend()
plt.show()
```

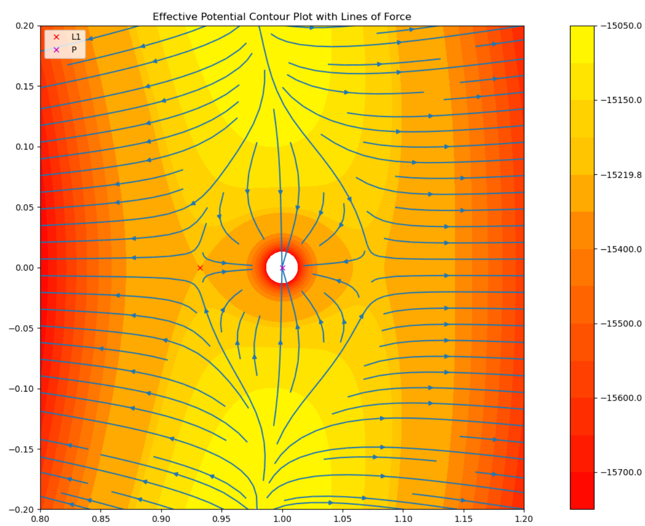

Figure 3: Filled Contour Plot with Lines of Force

```
X = np.linspace(0.8, 1.2, 250)
Y = np.linspace(-0.2,0.2,250)
x, y = np.meshgrid(X, Y)
Z = pot(x, y)
levels_h = np.arange(-15750,-15300,50)
levels_w=np.append(levels_h,pot(r,0))
levels_l=np.arange(-15200,-15000,50)
levels=np.append(levels_w,levels_l)

plt.figure()
cz = plt.contour(X, Y, Z, levels=(levels), cmap = "viridis")
plt.colorbar()
plt.plot(r,0,'rx', label="L1")
plt.plot(1,0,'mx', label="P")
plt.xlim(0.8, 1.2)
plt.ylim(-0.2,0.2)
plt.gca().set_aspect("equal")
plt.legend()
plt.title("Effective Potential Contour Plot - Zoomed In")
plt.show()
```

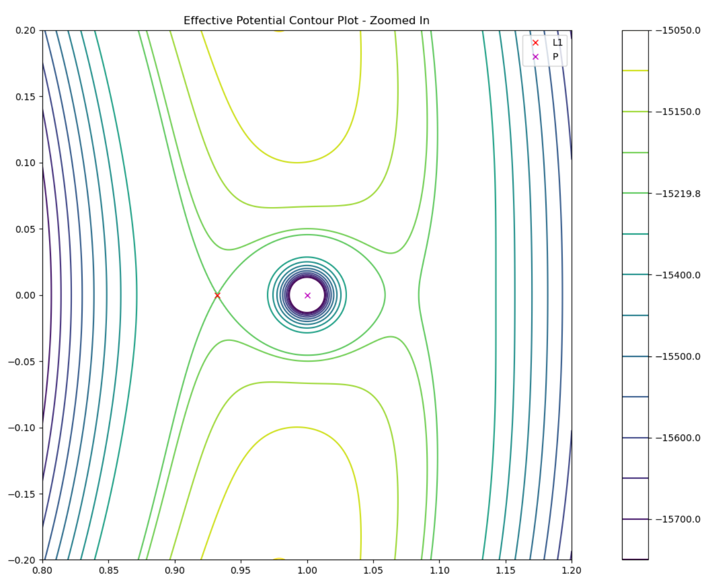

Figure 4: Zoomed in Contour Plot

3 Analysis

- From Figure 1, it can be seen that the Satellite at L1 moves in conjunction with the Planet, P. This verifies that a body placed at L1 is in equilibrium with respect to the Star-Planet system.
- From Figure 4, we see that L1 is a point of intersection for two lines of constant potential. This makes it a Saddle Point. From Figure 3 we see that if displaced vertically, the satellite will oscillate about L1, however, if displaced horizontally, it will not come back but will fall into S or into P.
- This means that L1 is a point of unstable equilibrium.

4 Applications and Scope for Further Study

- For the Earth-Sun system, L1 lies between the Sun and the Earth. As a result it is a highly suitable location for a solar observatory. It has an uninterrupted view of the Sun and can constantly communicate with Earth at high speeds through line-of-sight comm systems. Currently, the Solar and Heliospheric Observatory Satellite, SOHO is placed at L1.
- Knowledge about Lagrange Point L1 is critical in studying Mass Transfer and Supernovae in Binary Star Systems.
- Apart from L1, there are four more Lagrange Points for any two-body system. Those can also be studied in a similar fashion.

References

[1] J. L. Lagrange, "Essai sur le Problème des Trois Corps," Oeuvres de Lagrange, Vol. 6, 1772, pp. 229-332.

[2] L1, the first Lagrangian point. (n.d.). Retrieved April 01, 2021, from https://www.esa.int/Science_Exploration/Space_Science/L1_the_first_Lagrangian_Point