Problem Set 7

Dario Loprete

October 29, 2024

GitHub link

https://github.com/dariolop76/phys-ga2000

1 Problem 1

In this problem, we want to find the Lagrange point L_1 between two massive objects of mass M and m. It is defined as the point at which a satellite will orbit around the more massive of the two objects in perfect synchrony with the other, staying always between the two. Assuming circular orbits and supposing that the mass of the satellite and m are much smaller then M, the distance r from the center of the more massive object to L_1 is obtained by equating the total force acting on the satellite to the centripetal force which provides it with the same angular velocity ω of the smaller object. The total force acting on the satellite, whose direction corresponds to the line connecting the three objects, is

$$F_{tot} = G\frac{Mm_s}{r^2} - G\frac{mm_s}{(R-r)^2}, \tag{1}$$

where G is the Newton's gravitational constant and R is the distance between the two objects of masses M and m. The first term is the force on the satellite due to the presence of the more massive object, while the second term is the force on the satellite due to the presence of the less massive object. The centripetal force is given by

$$F_c = m_s\omega^2 r, \tag{2}$$

where the angular velocity ω is the same as that of the less massive object. It is therefore obtained by

$$G\frac{Mm}{R^2} = m\omega^2 R \rightarrow \omega^2 = G\frac{M}{R^3} \tag{3}$$

By equating Eq. 1 and Eq. 2 and using the expression for ω^2 in Eq. 3, we obtain the following equation for r:

$$\frac{M}{r^2} - \frac{m}{(R-r)^2} = \frac{Mr}{R^3}. \tag{4}$$

By defining the mass ratio $m' = \frac{m}{M}$, we can rewrite Eq. 4 as follows:

$$\frac{1}{r^2} - \frac{m'}{(R-r)^2} = \frac{r}{R^3} \tag{5}$$

By defining the ratio $r' = \frac{r}{R}$ and performing some manipulations, we obtain a degree-five polynomial equation in r':

$$f(r', m') = r'^5 - 2r'^4 + r'^3 - r'^2(1-m') + 2r' - 1 = 0 \tag{6}$$

Our objective is to solve Eq. 6 numerically by using Newton's method, thus obtaining a solution for the Lagrange point. This method allows to find the root of $f(r', m')$ using the fact that we know the derivative of this function:

$$f'(r', m') = \frac{df}{dr'} = 5r'^4 - 8r'^3 + 3r'^2 - 2r'(1-m') + 2 \tag{7}$$

Starting with a single guess for r', which we choose to be $\frac{1}{2}$, namely $r = \frac{R}{2}$, we use the slope at that position to extrapolate and make another guess, and we iterate this process until we reach the required accuracy. Two consecutive guesses for the root of $f(r', m')$ are related to each other by

$$r'_{i+1} = r'_i - \frac{f(r'_i)}{f'(r'_i)}. \tag{8}$$

We perform this computation for different inputs, the problem for values appropriate to the Moon and the Earth, the Earth and the Sun, and for the case of a Jupiter-mass planet orbiting the Sun at the distance of the Earth. The values for these parameters and the implementation of Newton's method are in the code `prob_1.py`. The solution is obtained with accuracy up to four significant figures. The obtained results are summarized in Table 1.

Massive objects	**Lagrange point**
Earth - Moon	$3.263 \cdot 10^8$ m
Earth - Sun	$1.481 \cdot 10^{11}$ m
Jupiter-mass planet - Sun	$3.588 \cdot 10^8$ m

Table 1: Results for the Lagrange points.

The Lagrange points between the Earth and the Moon and between the Sun and the Earth match with the values reported online (`https://en.wikipedia.org/wiki/Lagrange_point`).

2 Problem 2

We implement Brent's 1D minimization method and test it on the function

$$f(x) = (x - 0.3)^2 e^x \tag{9}$$

This method consists in applying a Successive Parabolic Interpolation (SPI) and, when this fails, a Golden Section Search (GSS). The SPI consists in performing a parabolic interpolation of the function given three points a, b, c, chosen so that the interval $[a, c]$, called bracket, contains the minimum point. From the parabolic interpolation we obtain a guess for the minimum point given by

$$x = b + \frac{p}{q} = b - \frac{1}{2}\frac{(b-a)^2(f(b)-f(c)) - (b-c)^2(f(b)-f(a))}{(b-a)(f(b)-f(c)) - (b-c)(f(b)-f(a))}. \tag{10}$$

Then, we restrict the bracketing interval based on whether x belongs to $[a, b]$ or $[b, c]$. However, SPI might fail. This happens when:

- the denominator in Eq. 10 is zero;
- x is outside the bracketing interval $[a, c]$;
- the parabolic step is greater than half the step before last:

$$\left|\frac{p}{q}\right|_{i+1} > \frac{1}{2}\left|\frac{p}{q}\right|_i \tag{11}$$

In these cases, we switch to GSS. These conditions are implemented in the code `prob_2.py`. We choose the first bracketing interval noticing that the minimum point must be between -2 and 3, as shown in Fig. 1.

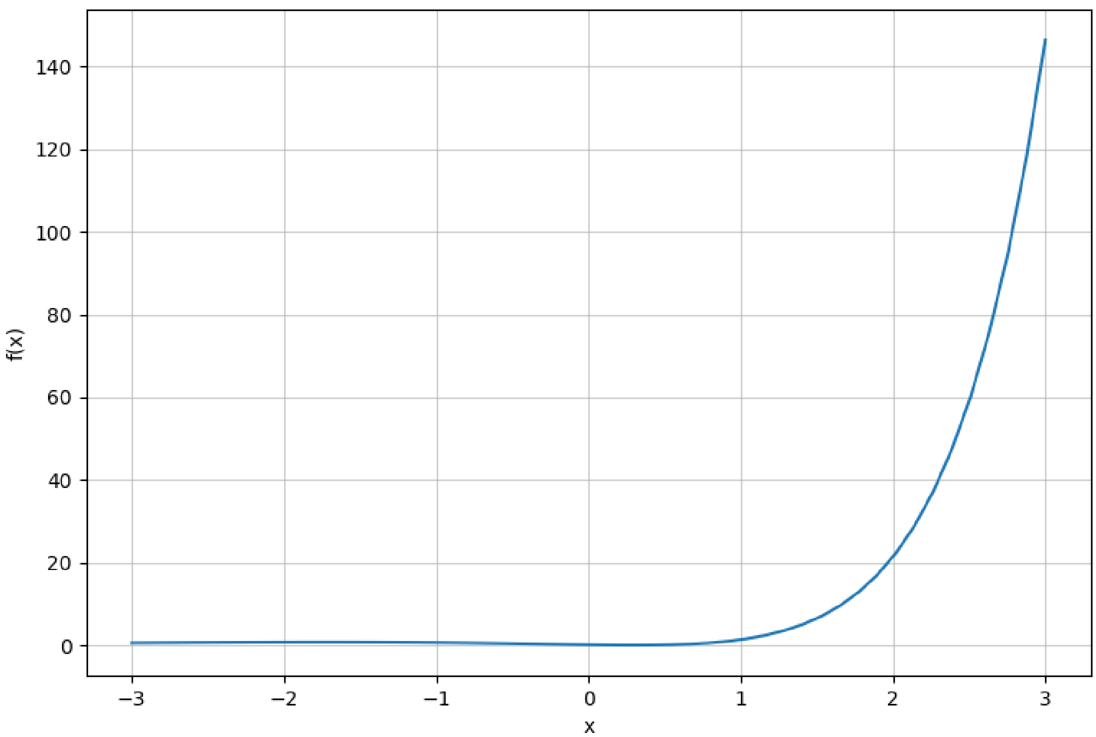

Figure 1: Plot of the function $f(x) = (x - 0.3)^2 e^x$.

The implementation of Brent's method is compared to the application of the method `scipy.optimize.brent`. The obtained results are summarized as follows:

Brent (our implementation): $x_{min} = 0.2999999998$
Brent (scipy): $x_{min} = 0.2999999999$
Difference: $\Delta(x_{min}) = 1 \cdot 10^{-10}$