# OPTIMIZATION OF ORBITAL PARAMETERS OF ELISA SPACECRAFT CONFIGURATION

November 25, 2017


*Prepared by*

**Komal Gupta**


*In partial fulfillment of the requirements of study project*

Under the guidance of


**Dr. Kinjal Banerjee**

**Department of Physics, BITS Pilani - K K Birla Goa Campus**


**BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI**

# Abstract

This report highlights the results obtained from multi-body simulations of the eLISA spacecraft configuration, under the gravitational influence of the Sun, Earth, Moon and Jupiter. These bodies are added to the simulation one by one and their effects on a stable triangular configuration of three spacecrafts are studied. The orbits are optimized for minimal variation in arm lengths over a period of two years. The software Mathematica has been used for all simulations.

# Acknowledgments

I want to express my sincerest gratitude to Dr. Kinjal Banerjee, Department of Physics, BITS Pilani - K K Birla Goa Campus, for giving me the opportunity to work on this project and for his immense counsel throughout its course. Without his continual guidance and encouragement, I would not have been able to succeed in this endeavour. I also want to thank my senior and friend Siddharth Paliwal, for helping me learn the basics of Mathematica during the initial stages of this project.

# Contents

# List of Figures

# 1 Introduction

eLISA, or evolved Laser Interferometer Space Antenna is a mission proposed by the European Space Agency to detect and accurately measure gravitational waves. The mission consists of a constellation of three spacecrafts in an equilateral triangular formation, with arm lengths of 5 million kilometers. By precisely monitoring the changes in arm lengths of the triangle, the space based observatory aims to detect gravitational waves of the order of $10^{-5}$ Hz to 1 Hz. These low frequency gravitational waves, created by collisions of massive black hole binaries and compact stellar mass binaries, cannot be detected by even the most sensitive ground based observatories.

The changes in the arm lengths of the eLISA triangle causes a Doppler shift in the laser signals between spacecrafts which introduces a measurements error in the detection of gravitational waves. The following chapters describe the optimization of initial positions and velocities of the three spacecrafts in order to have minimum variations in the arm lengths over a period of two years. The effects of the Sun, Earth, Moon and Jupiter are added one by one to quantify the variations caused by each body on the configurations. Effects of other objects in the solar system (Mars, Venus, etc.) are much smaller in magnitude compared to the effects of these bodies and were ignored in the simulation.

## 2 The Orbits

The eLISA spacecrafts follow heliocentric, keplerian orbits with small eccentricities inclined at a small angle to the ecliptic. The centroid of the triangle trails the earth by an angle of approximately 20º. The plane of the triangle is inclined at an angle of 60º with the ecliptic. According to Bender [2], spacecrafts lying in a plane making an angle of 60º or 150º with the ecliptic perform rigid rotations around the origin; hence the choice of plane. The geometry of the eLISA configuration is shown in the following figure:

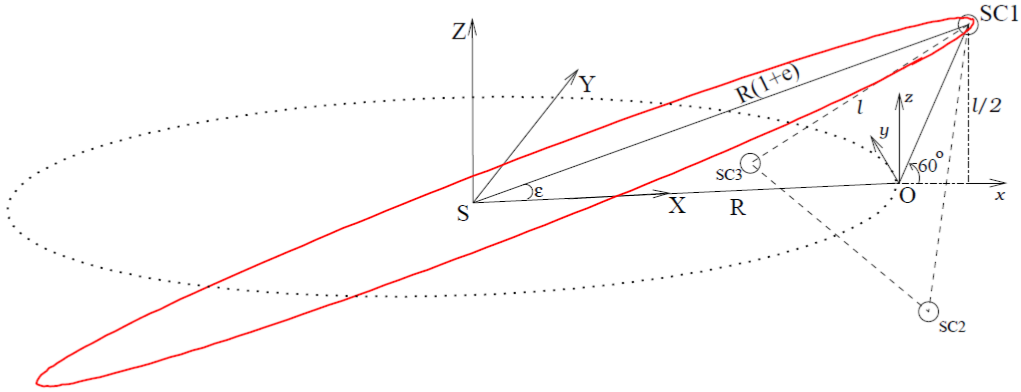



Figure 1: Geometry of eLISA configuration. (Source: [1])

where
l = 5 x $10^6$ kilometers, is the arm length of the triangle;
R = 1.5 x $10^8$kilometers, is the radius of the circular reference orbit on which the centroid of the triangle lies;
e $\sim$ 0.01, is the eccentricity of the orbits of the spacecrafts;
$\varepsilon \sim$ 0.016 radians, is the inclination of the orbits with respect to the ecliptic.

The three spacecrafts are abbreviated as SC1, SC2 and SC3. The coordinate system shown in the figure has its origin at the sun, with X and Y axes lying in the plane of the ecliptic and Z axis perpendicular to it. The direction of X axis is fixed such that initially SC1 lies in the X-Z plane.
Figure 1 shows the orbit of SC1 only (marked in red). The orbits of the other two spacecrafts can be obtained by rotating this orbit by 120º and 240º.

# 3 Simulation of Orbits

The orbits for the three spacecrafts were simulated using the Newton's equations of motion. The positions of the planetary bodies whose effects are to be evaluated were also calculated, in accordance with Kepler's laws of planetary motion. This chapter explains the simulations of orbits of the three spacecrafts as well as the planetary bodies.

## 3.1 Calculation of forces

The gravitational force on a spacecraft due to the sun is

$$\overline{F} = -\frac{GMm}{r^2}\hat{r}$$

where

- M is mass of the sun
- m is mass of the spacecraft
- r is the distance of spacecraft from sun

The components of acceleration in x, y and z directions are

$$a_x = -\frac{GMx}{r^3} \qquad a_y = -\frac{GMy}{r^3} \qquad a_z = -\frac{GMz}{r^3}$$

This leads us to the following set of coupled equations for the motion of spacecrafts under the influence of the sun:

$$\frac{dv_x}{dt} = -\frac{GMy}{r^3} \qquad \frac{dv_y}{dt} = -\frac{GMy}{r^3} \qquad \frac{dv_z}{dt} = -\frac{GMz}{r^3}$$

$$\frac{dx}{dt} = v_x \qquad \frac{dy}{dt} = v_y \qquad \frac{dz}{dt} = v_z$$

These equations were solved using the NDSolve function of Mathematica.

#### Effects of planetary bodies

The gravitational effects of the planets were added after the sun. To do so, the equations were modified as:

$$\overline{F} = -\frac{GM_sm}{r_s^2}\hat{r}_s - \frac{GM_pm}{r_p^2}\hat{r}_p$$

where $M_p$ is the mass of the planetary body and $r_p$ is the distance between it and the spacecraft.

## 3.2 Equations of orbits

### For planets and the moon

The planets are assumed to lie in the ecliptic plane, i.e., their inclination with respect to the ecliptic is ignored. The foci of the elliptical orbits lie at the sun. The position of a planet was calculated using the formula

$$r = \frac{a(1-e^2)}{1+ecos\theta}$$

where a is the semi-major axis and e is the eccentricity of its orbit.

To obtain the orbit of the moon, its positions were first calculated in a geocentric frame using the aforementioned formula, with earth at the focus of the ellipse. The positions in the heliocentric frame were then calculated by translation of coordinates from geocentric to heliocentric coordinate system.

### For spacecrafts

The eLISA spacecrafts, when launched, will follow elliptical orbits around the sun. The orbit equations for SC1 are given by

$$X_1 = R(cos\psi_1 + e)cos\varepsilon$$

$$Y_1 = R\sqrt{1-e^2}sin\psi_1$$

$$Z_1 = R(cos\psi_1 + e)sin\varepsilon$$

The eccentric anomaly $\psi_1$ is given in terms of t by

$$\psi_1 + esin\psi_1 = \Omega t$$

where $\Omega$ is the average angular velocity of the spacecraft.

The orbits of SC2 and SC3 are obtained by rotating the orbit of SC1 by $2\pi/3$ and $4\pi/3$ respectively about the Z axis. The equation for spacecraft k = 2,3 are:

$$X_k = X_1 cos\left[\frac{2\pi}{3}(k-1)\right] - Y_1 sin\left[\frac{2\pi}{3}(k-1)\right]$$

$$Y_k = X_1 sin\left[\frac{2\pi}{3}(k-1)\right] + Y_1 cos\left[\frac{2\pi}{3}(k-1)\right]$$

$$Z_k = Z_1$$

where the phases are given by:

$$\psi_k + e\sin\psi_k = \Omega t - (k-1)\frac{2\pi}{3}$$

The orbits of the spacecrafts are as shown below.

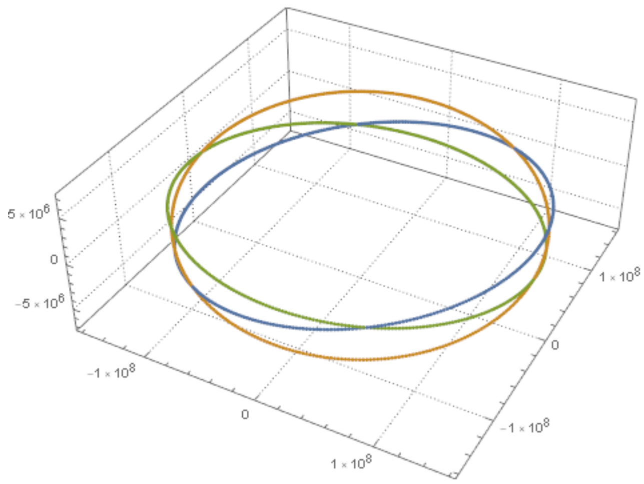



Figure 2: The orbits

# 4 Optimization of Orbits

In order to optimize the orbits, we choose to optimize the initial phases of the spacecrafts. This is because the initial phases can be controlled by suitably choosing the launch dates of the spacecrafts. In the simulation, the initial phase of SC1 was taken to be zero, i.e., $\psi_1(t=0)=0$. The initial positions and velocities of the other two spacecrafts were optimized in order to have minimal variations in arm lengths of the constellation over a period of two years. To do this, a cost function was defined as follows:

$$\Phi = \sum_{t=0}^{2} \left(|d_{12}(t)-l| + |d_{23}(t)-l| + |d_{31}(t)-l|\right)$$

where $d_{ij}(t)$ is the distance between $i^{th}$ and $j^{th}$ spacecrafts at time $t$ (in years).
The results of the optimization are summarized in the following subsections.

## 4.1 Effects of Sun

The variation in arm lengths of eLISA constellation with time when only the effects of the sun are considered is shown in the following figure:

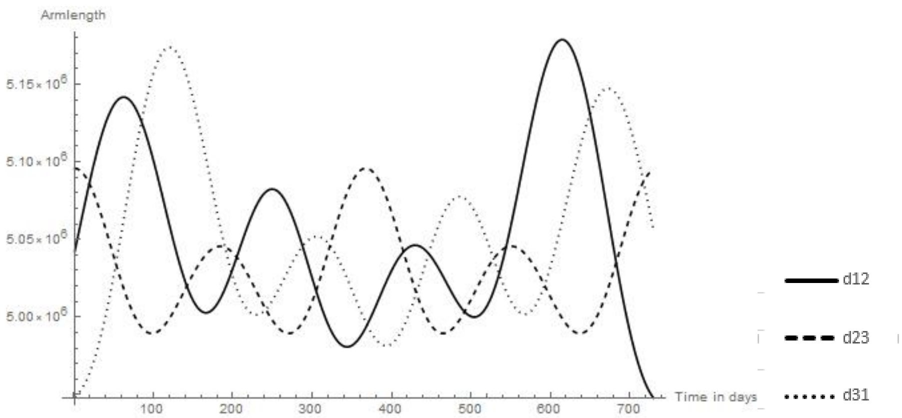



Figure 3: Variations in arm lengths under effects of the Sun

In this case, $\Phi = 1.12771\times10^8$. Optimal initial phases were found to be $\psi_2 = 4.19791$ and $\psi_3 = 2.08652$.

## 4.2 Effects of Earth

In order to measure the perturbation in spacecrafts' orbits caused by the earth's gravity, the variations in arm lengths were first plotted using the optimal initial phases for the Sun only case.

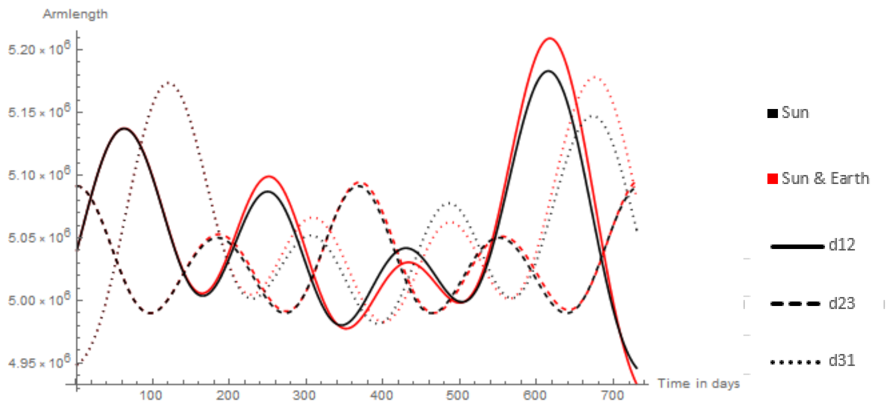



Figure 4: Variations in arm lengths under effects of the Sun and the Earth (unoptimized)

It can be observed from the plot that the perturbation due to earth is negligible for the first 200 days after the start of the mission, but grows substantially over time, which is undesirable. In this case, $\Phi = 1.2058 \times 10^8$.
After optimization, the variation in arm lengths was as follows:

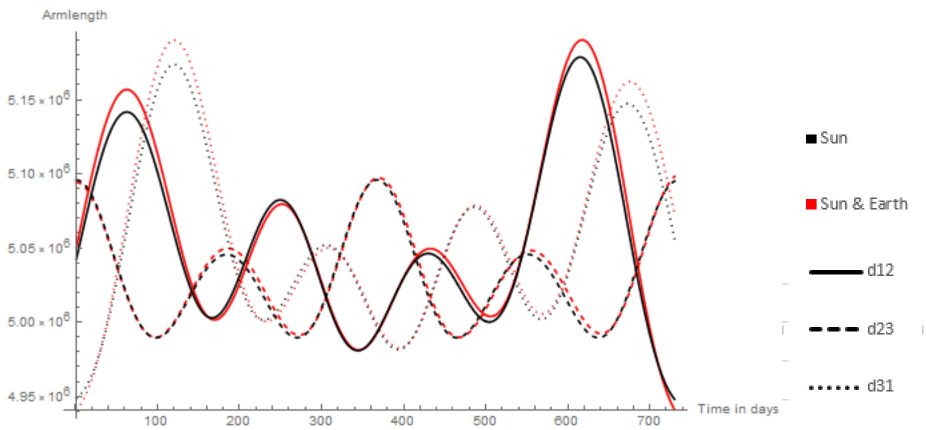



Figure 5: Variations in arm lengths under effects of the Sun and the Earth (optimized)

In this case, $\Phi = 1.20347 \times 10^8$. Optimal initial phases were found to be $\psi_2 = 4.19801$ and $\psi_3 = 2.08663$. The optimization has reduced the cost function even further, and the perturbation due to earth does not grow over time.

## 4.3 Effects of Moon

Next, the effects of the moon were added and the phases were optimized again. The variations are shown in the following figure.

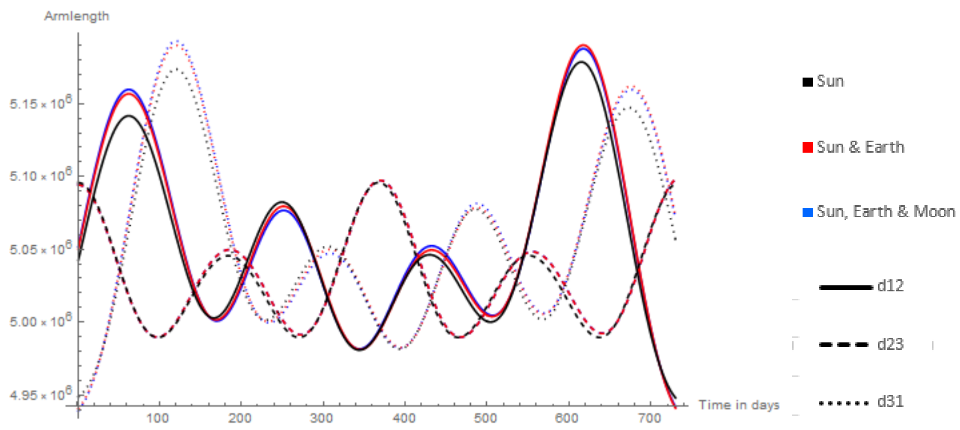



Figure 6: Variations in arm lengths under effects of the Sun, the Earth and the Moon

In this case, $\Phi = 1.20438 \times 10^8$. Optimal initial phases were found to be $\psi_2 = 4.19798$ and $\psi_3 = 2.08657$. Due to its lesser mass compared to the earth, the perturbation in spacecrafts' orbits caused by the moon are small, as can be seen in the figure.

## 4.4 Effects of Jupiter

In order to find the effects of Jupiter on the eLISA configuration, its position at the time of launch needs to be determined. The mission is supposed to be launched in 2034. It is assumed that the mission starts on 1 January 2034. Taking this into account, the variation in arm lengths after optimization are as shown in the following figure.

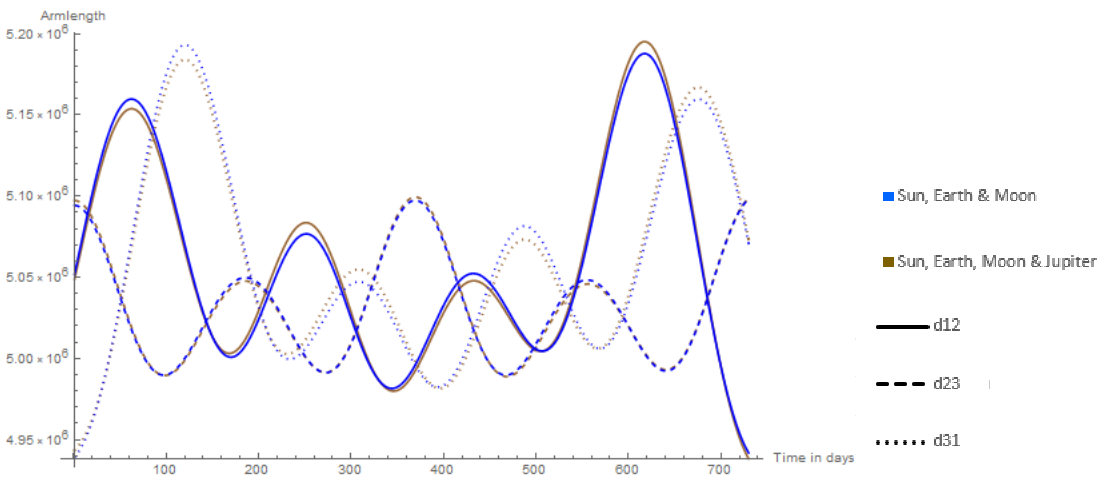



Figure 7: Variations in arm lengths under effects of the Sun, the Earth, the Moon and Jupiter

In this case, $\Phi = 1.21023 \times 10^8$. Optimal initial phases were found to be $\psi_2 = 4.19799$ and $\psi_3 = 2.08659$.

The cost functions for the four cases can now be plotted as follows.

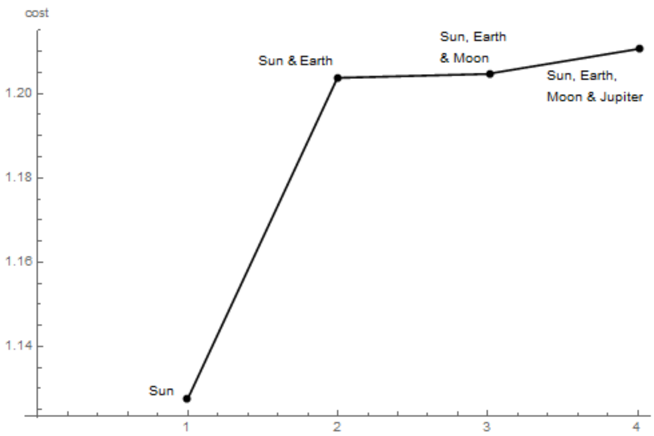



Figure 8: Change in optimum value of cost functions

By looking at the changes in the optimum value of cost function, we can say that the Earth causes the largest perturbation in the orbits of the spacecrafts, followed by Jupiter, which is followed by the Moon. Jupiter causes a larger perturbation in the orbits than the Moon despite being farther away because of its much larger mass.

# 5 Conclusion

The previous chapters have presented an analysis of the orbits of the eLISA spacecraft configuration. However, in this report, only the phases have been optimized. In reality, one can optimize as many parameters as required for improving the stability of the configuration, such as arm length of the triangle, eccentricity and inclination of orbits with the ecliptic, trailing angle behind the Earth, internal angles of the constellation, etc.

It can be safely concluded that the overall variation in the arm lengths of the constellation increases as the gravitational effects of more bodies are considered. Knowing these variations accurately will allow one to account for them while detecting gravitational waves, which will reduce the error in measurements. Further research needs to be conducted to account for and optimize more parameters for the stability of the configuration.

# References


1. Dhurandhar S V, Nayak K R, Koshti S, Vinet J-Y. January 2005. “Fundamentals of the LISA Stable Flight Formation”. Classical and Quantum Gravity. Volume 22(3).

2. Folkner W M, Hechler F, Sweetser T H, Vincent M A, Bender P L. 1997. “LISA orbit selection and stability”. Classical and Quantum Gravity. Volume 14(6).

3. Yan Xia, GuangYu Li, Gerhard Heinzel, Albrecht Rüdiger, YongJie Luo. January 2010. “Orbit design for the Laser Interferometer Space Antenna (LISA)”. Science China Physics, Mechanics & Astronomy. Volume 53(1):179-186.

# Appendix

```
(*the following code gives the variations with Sun, Earth,
Moon and Jupiter. To switch off any particular variation,
put the mass of the body = 0 *)

In[1]:= omega = 6.2832; (*rad per year*)
l = 5 * 10^6; (*km*)
R = 1.5 * 10^8; (*km*)
alpha = l / (2 * R);
time = Range[1, 365] / 365;
G = (6.67408 * 10^(-20)) * (31557600^2); (* km^3 kg^-1 yr^2 *)
M = 2 * 10^30; (*kg*)
Me = 6 × 10^24; (*kg*)
Ae = 1.496 * 10^8;
eps = ArcTan[(l / 2) / (R + l / (2 * Sqrt[3]))];
e = (1 + 2 * alpha / Sqrt[3] + 4 * alpha^2 / 3)^0.5 - 1;
theta0 = Pi / 9;

In[13]:= (*orbital parameters of moon*)

In[14]:= Mm = 7.35 × 10^22; (*kg*)
em = 0.0549; (*eccentricity*)
am = 384400; (*major axis*)
epsm = 0; (* 5.145*Pi/180;rad*)
thetam = Pi;
omegam = 13.176 * 365.25 * Pi / 180 (*rad/year*);

(*coordinates of earth*)

In[20]:= re[t_] := Ae * (1 - 0.0167^2) / (1 + 0.0167 * Cos[omega * t + theta0]);
xe[t_] :=
  Ae * (1 - 0.0167^2) * Cos[omega * t + theta0] / (1 + 0.0167 * Cos[omega * t + theta0]);
ye[t_] := Ae * (1 - 0.0167^2) * Sin[omega * t + theta0] / (1 + 0.0167 * Cos[omega * t + theta0]);

(*coordinates of moon*)

In[23]:= rm[t_] := am (1 - em^2) / (1 + em * Cos[omegam t + thetam]);
xm[t_] := rm[t] Cos[epsm] Cos[omegam t + thetam] + xe[t] ;
ym[t_] := rm[t] Cos[epsm] Sin[omegam t + thetam] + ye[t];
zm[t_] := rm[t] Sin[epsm];

(*orbital parameters of jupiter*)
```

```
In[27]:= Mj = 1.9 × 10^27; (*kg*)
         ej = 0.0489;
         aj = 7.7857 × 10^8;
         omegaj = 0.52969; (*rad/year*)

In[31]:= rj[t_] := aj (1 - ej^2) / (1 + ej * Cos[omegaj t + thetaj]);
         xj[t_] := rj[t] Cos[omegaj t + thetaj];
         yj[t_] := rj[t] Sin[omegaj t + thetaj];

         (*coordinates of jupiter*)

         (*on 1/1/2017*)

In[34]:= xcoord = 4.776242; (*AU*)
         ycoord = -1.4239822; (*AU*)

In[36]:= Clear[thetaj]

In[37]:= eqa = xj[0] == xcoord * Ae;
         eqb = yj[0] == ycoord * Ae;

In[39]:= thetaj = thetaj /. Flatten[Solve[eqb, thetaj]]

         Solve: Inverse functions are being used by Solve, so some solutions may not be found; use Reduce for complete solution
           information.

Out[39]= -2.8772

In[40]:= thetaj = thetaj - (3.5 / 12) * 2 Pi;
```

# Minimization

```
In[41]:= eq1 = x''[t] == -G M x[t] / Sqrt[(x[t])^2 + (y[t])^2 + (z[t])^2]^3 -
      G Me (x[t] - xe[t]) / Sqrt[(x[t] - xe[t])^2 + (y[t] - ye[t])^2 + (z[t])^2]^3 -
      G Mm (x[t] - xm[t]) / Sqrt[(x[t] - xm[t])^2 + (y[t] - ym[t])^2 + (z[t] - zm[t])^2]^3 -
      G Mj (x[t] - xj[t]) / Sqrt[(x[t] - xj[t])^2 + (y[t] - yj[t])^2 + (z[t])^2]^3;

    eq2 = y''[t] == -G M y[t] / Sqrt[(x[t])^2 + (y[t])^2 + (z[t])^2]^3 -
      G Me (y[t] - ye[t]) / Sqrt[(x[t] - xe[t])^2 + (y[t] - ye[t])^2 + (z[t])^2]^3 -
      G Mm (y[t] - ym[t]) / Sqrt[(x[t] - xm[t])^2 + (y[t] - ym[t])^2 + (z[t] - zm[t])^2]^3 -
      G Mj (y[t] - yj[t]) / Sqrt[(x[t] - xj[t])^2 + (y[t] - yj[t])^2 + (z[t])^2]^3;

    eq3 = z''[t] == -G M z[t] / Sqrt[(x[t])^2 + (y[t])^2 + (z[t])^2]^3 -
      G Me (z[t]) / Sqrt[(x[t] - xe[t])^2 + (y[t] - ye[t])^2 + (z[t])^2]^3 -
      G Mm (z[t] - zm[t]) / Sqrt[(x[t] - xm[t])^2 + (y[t] - ym[t])^2 + (z[t] - zm[t])^2]^3 -
      G Mj (z[t] ) / Sqrt[(x[t] - xj[t])^2 + (y[t] - yj[t])^2 + (z[t])^2]^3;
```

## SC1

```
In[44]:= x10 = R (Cos[si10] + e) Cos[eps];
    y10 = R Sqrt[1 - e^2] Sin[si10];
    z10 = R (Cos[si10] + e) Sin[eps];
    vx0 = - R Sin[si10] omega Cos[eps] / (1 + e Cos[si10]);
    vy0 = R Sqrt[1 - e^2] Cos[si10] omega / (1 + e Cos[si10]);
    vz0 = R (-Sin[si10]) Sin[eps] omega / (1 + e Cos[si10]);
```

## SC2

```
In[50]:= x20 = - R (Cos[si20] + e) Cos[eps] / 2 - R Sqrt[1 - e^2] Sin[si20] Sqrt[3] / 2;
    y20 = - R Sqrt[1 - e^2] Sin[si20] / 2 + R (Cos[si20] + e) Cos[eps] Sqrt[3] / 2;
    z20 = R (Cos[si20] + e) Sin[eps];
    vx20 = (R (Sin[si20]) Cos[eps] / 2 - R Sqrt[1 - e^2] Cos[si20] Sqrt[3] / 2) omega / (1 + e Cos[si20]);
    vy20 =
      (- R Sqrt[1 - e^2] Cos[si20] / 2 - R Sin[si20] Cos[eps] Sqrt[3] / 2) omega / (1 + e Cos[si20]);
    vz20 = R (-Sin[si20]) Sin[eps] omega / (1 + e Cos[si20]);
```

## SC3

```
In[56]:= x30 = - R (Cos[si30] + e) Cos[eps] / 2 + R Sqrt[1 - e^2] Sin[si30] Sqrt[3] / 2;
y30 = - R (Cos[si30] + e) Cos[eps] Sqrt[3] / 2 - R Sqrt[1 - e^2] Sin[si30] / 2;
z30 = R (Cos[si30] + e) Sin[eps];
vx30 =
  (R (Sin[si30]) Cos[eps] / 2 + R Sqrt[1 - e^2] Cos[si30] Sqrt[3] / 2) omega / (1 + e Cos[si30]);
vy30 = (R Sin[si30] Cos[eps] Sqrt[3] / 2 - R Sqrt[1 - e^2] Cos[si30] / 2) omega / (1 + e Cos[si30]);
vz30 = R (-Sin[si30]) Sin[eps] omega / (1 + e Cos[si30]);
```

## Initial Conditions

```
In[62]:= ic1 = {x[0] == x10, y[0] == y10, z[0] == z10, x'[0] == vx0, y'[0] == vy0, z'[0] == vz0};
ic2 = {x[0] == x20, y[0] == y20, z[0] == z20, x'[0] == vx20, y'[0] == vy20, z'[0] == vz20};
ic3 = {x[0] == x30, y[0] == y30, z[0] == z30, x'[0] == vx30, y'[0] == vy30, z'[0] == vz30};
```

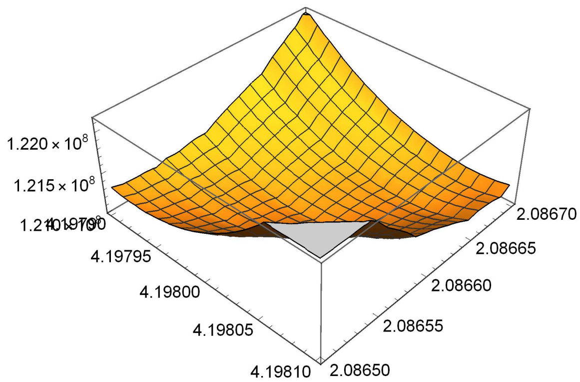
## Find optimal phases

```
In[65]:= cost = {};
 (*the starting and end points of the loops are chosen
  such that they bracket the minima of the cost function*)
```

In[66]:=
```
For[phi2 = 4.1979, phi2 ≤ 4.1981, phi2 = phi2 + 0.00001,
 For[phi3 = 2.0865, phi3 ≤ 2.0867, phi3 = phi3 + 0.00001,
  {

sol1 = Flatten[
     NDSolve[{eq1, eq2, eq3, ic1 /. si10 → 0}, {x[t], y[t], z[t]}, {t, 0, 2, 1 / 365}], 1];
   sol2 = Flatten[NDSolve[{eq1, eq2, eq3, ic2 /. si20 → phi2},
      {x[t], y[t], z[t]}, {t, 0, 2, 1 / 365}], 1];
   sol3 = Flatten[NDSolve[{eq1, eq2, eq3, ic3 /. si30 → phi3},
      {x[t], y[t], z[t]}, {t, 0, 2, 1 / 365}], 1];
   S1 = Table[{x[t] /. sol1, y[t] /. sol1, z[t] /. sol1}, {t, 0, 2, 1 / 365}];
   S2 = Table[{x[t] /. sol2, y[t] /. sol2, z[t] /. sol2}, {t, 0, 2, 1 / 365}];
   S3 = Table[{x[t] /. sol3, y[t] /. sol3, z[t] /. sol3}, {t, 0, 2, 1 / 365}];
   D12[t_] := Abs[Sqrt[(S1[[t, 1]] - S2[[t, 1]])^2 +
        (S1[[t, 2]] - S2[[t, 2]])^2 + (S1[[t, 3]] - S2[[t, 3]])^2]];
   D23[t_] := Abs[Sqrt[(S2[[t, 1]] - S3[[t, 1]])^2 + (S2[[t, 2]] - S3[[t, 2]])^2 +
        (S2[[t, 3]] - S3[[t, 3]])^2]];
   D31[t_] := Abs[Sqrt[(S1[[t, 1]] - S3[[t, 1]])^2 + (S1[[t, 2]] - S3[[t, 2]])^2 +
        (S1[[t, 3]] - S3[[t, 3]])^2]];
   func[t_] := Abs[D12[t] - 1] + Abs[D31[t] - 1] + Abs[D23[t] - 1];
cost = Append[cost, { Sum[func[t], {t, 1, 730}], phi2, phi3}];

  }]]

(*the following plots will help the reader in putting the correct start and end
 points in the above loops. If the minima occurs at the edge in the graphs,
another range for phases should be put in the loop. This
 can be repeated until desired precision is achieved*)
```

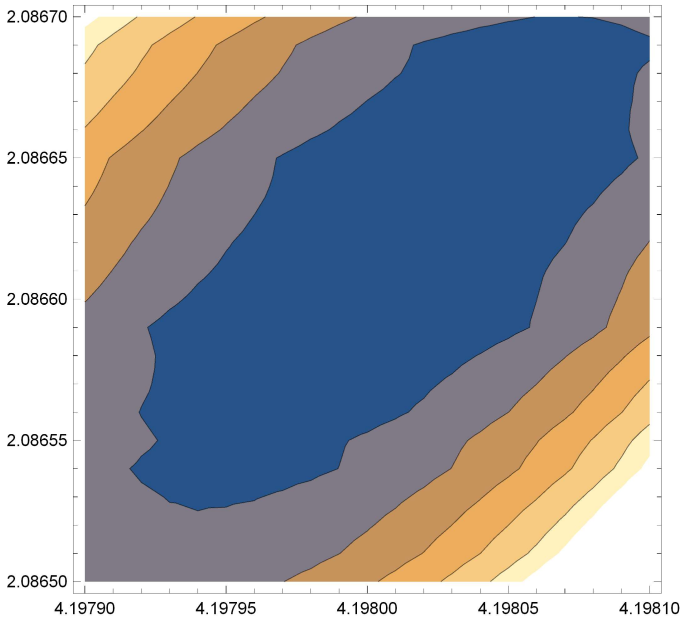
In[67]:=
```
ListPlot3D[Table[{cost[[i, 2]], cost[[i, 3]], cost[[i, 1]]}, {i, 1, Length[cost]}]]
```

Out[67]=



In[68]:=
```
ListContourPlot[Table[{cost[[i, 2]], cost[[i, 3]], cost[[i, 1]]}, {i, 1, Length[cost]}]]
```

Out[68]=




In[69]:=
```
sortedcost = Sort[cost];

(*finally the optimal phases are*)
```

In[127]:=
```
ph2 = sortedcost[[1, 2]];
ph3 = sortedcost[[1, 3]];
```

In[72]:=
```
sol1 = Flatten[
   NDSolve[{eq1, eq2, eq3, ic1 /. si10 → 0}, {x[t], y[t], z[t]}, {t, 0, 2, 1 / 365}], 1];
sol2 = Flatten[NDSolve[{eq1, eq2, eq3, ic2 /. si20 → ph2},
     {x[t], y[t], z[t]}, {t, 0, 2, 1 / 365}], 1];
sol3 = Flatten[NDSolve[{eq1, eq2, eq3, ic3 /. si30 → ph3},
     {x[t], y[t], z[t]}, {t, 0, 2, 1 / 365}], 1];
```

In[129]:=
```
S1 = Table[{x[t] /. sol1, y[t] /. sol1, z[t] /. sol1}, {t, 0, 2, 1 / 365}];
S2 = Table[{x[t] /. sol2, y[t] /. sol2, z[t] /. sol2}, {t, 0, 2, 1 / 365}];
S3 = Table[{x[t] /. sol3, y[t] /. sol3, z[t] /. sol3}, {t, 0, 2, 1 / 365}];
ListPointPlot3D[{S1, S2, S3},
  PlotRange → {{-1.1 R, 1.1 R}, {-1.1 R, 1.1 R}, {-R / 20, R / 20}}];

D12[t_] := Abs[Sqrt[
     (S1[[t, 1]] - S2[[t, 1]])^2 + (S1[[t, 2]] - S2[[t, 2]])^2 + (S1[[t, 3]] - S2[[t, 3]])^2]];
D23[t_] := Abs[Sqrt[(S2[[t, 1]] - S3[[t, 1]])^2 +
      (S2[[t, 2]] - S3[[t, 2]])^2 + (S2[[t, 3]] - S3[[t, 3]])^2]];
D31[t_] := Abs[Sqrt[(S1[[t, 1]] - S3[[t, 1]])^2 + (S1[[t, 2]] - S3[[t, 2]])^2 +
      (S1[[t, 3]] - S3[[t, 3]])^2]];
```

In[124]:=
```
D1j =
  DiscretePlot[D12[t], {t, 1, 730}, PlotStyle → Purple, Joined → True, Filling → None];
D2j = DiscretePlot[D23[t], {t, 1, 730}, PlotStyle → Red, Joined → True, Filling → None];
D3j = DiscretePlot[D31[t], {t, 1, 730}, PlotStyle → Green, Joined → True, Filling → None];
```

In[134]:=
```
Show[D1j, D2j, D3j, PlotRange → All]
```

Out[134]=