UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

MASTER THESIS

Computation of Libration point orbits and manifolds using collocation methods

Kirill Rogov

SUPERVISED BY

Prof. Josep J. Masdemont

Universitat Politècnica de Catalunya
Master in Aerospace Science & Technology
July 2016

This Page Intentionally Left Blank

Computation of Libration point orbits and manifolds using collocation methods

BY

Kirill Rogov

DIPLOMA THESIS FOR DEGREE

Master in Aerospace Science and Technology

AT

Universitat Politècnica de Catalunya

SUPERVISED BY:

Prof. Josep J. Masdemont

Departament de Matemàtiques

This Page Intentionally Left Blank

ABSTRACT

This thesis contains a methodology whose aim is to compute trajectories describing natural motion of the phase space in a neighborhood of Libtation points and stable/unstable manifolds which correspond to these orbits in the Restricted Three Body Problem. There are two models the Circular Restricted Three Body Problem and Elliptic Restricted Three Body Problem which are special cases of RTBP. In this paper we pay attention to CRTBP which is autonomous (depending on time). The CRTBP is the most easily understood and well-analysed in a coordinate system rotating with two large bodies. The method is based on the collocation method implemented in AUTO – 07p software and must provide an isolated periodic solution. The paper includes explanation of the collocation method, its application in case of CRTBP, numerical and graphical results of its implementation.

This Page Intentionally Left Blank

Table of Contents

INTRODUCTION **1**

Chapter 1 BACKGROUND **7**

1.1. Dynamical Model **7**

1.1.1. Three Body Problem 7

1.1.2. Reference frame 8

1.1.3. The Circular Restricted Three Body Problem 9

1.2. Libration points **17**

1.2.1. How to compute the position of L-points 18

1.2.2. Libration point orbits 22

1.2.3. Invariant manifolds and connecting orbits 25

1.2.4. Libration point missions 26

1.3. The method of collocation **27**

Chapter 2 METHODOLOGY **31**

2.1. Lindstedt – Poicaré method **31**

2.1.1. Lindstedt – Poincaré procedure for Lissajous orbits 32

2.1.2. Lindstedt – Poincaré procedure for Halo orbits 35

2.2. Collocation method for solving non-linear BVP **38**

2.2.1. Computing periodic solution and period by collocation 45

2.2.2. Computing periodic solution with fixed period by collocation 50

2.2.3. Continuation approach 52

2.3. Numerical integrator **53**

2.4. Stable and unstable manifolds **55**

2.4.1. Homoclinic and Heteroclinic trajectories 60

Chapter 3 NUMERICAL RESULTS **63**

3.1. Lindstedt – Poincaré procedure and Collocation Method **63**

3.2. Evolution of orbit families **68**

3.3. Continuation approach **78**

3.4. Stable and unstable manifolds **82**

Chapter 4 Conclusion **91**

Bibliography **93**

Appendix A **95**

Appendix B **103**

Appendix C **107**

This Page Intentionally Left Blank

List of Figures

Figure 0.1 Libration points in the Sun-Earth system (http://press.cosmos.ru) 1
Figure 0.2 Different families of orbits [12] 2
Figure 0.3 Example of Halo orbit 3
Figure 0.4 Low energy transfer and orbit of Genesis mission (http://www.hindawi.com/journals/mpe/2012/351759/fig1/) 4
Figure 1.1 The illustration of the Newton's law of the universal gravitation 7
Figure 1.2 The Synodic Reference Frame of the RTBP 9
Figure 1.3 Coordinate systems *CXYZ* and *Cxyz* 10
Figure 1.4 Libration points of the Sun-Earth system 17
Figure 1.5 Libration point locations 18
Figure 1.6 Collinear points of the Sun-Earth system 19
Figure 1.7 Gravitational acceleration at the L4 in the Earth-Moon system 20
Figure 1.8 Planar Lyapunov family 22
Figure 1.9 Top-left: Vertical Lyapunov orbit, top-right: Lissajous quasi-orbit, ... 22
Figure 1.10 Periodic orbits around L2 24
Figure 1.11 Quasi - Periodic orbits around L2 24
Figure 1.12 (a) Unstable invariant manifold (b) Stable invariant manifold 25
Figure 1.13 WMAP mission (http://map.gsfc.nasa.gov) 26
Figure 2.1 Intervals and collocation points t_{ij} inside $[t_i, t_{i+1}]$ 39
Figure 2.2 Lagrange polynomials for k = 3 41
Figure 2.3 Illustration of orthogonally conditions to fix periodic orbit 47
Figure 2.4 The basic concept of the continuation procedure 52
Figure 2.5 State Transition Matrix concept 55
Figure 2.6 Concept of choosing points for manifold local linear approximation 58
Figure 2.7 New initial conditions 59
Figure 2.8 Cross section 61
Figure 3.1 Vertical Lyapunov orbits obtained semi-analytically (blue) and numerically (red). 64
Figure 3.2 Vertical Lyapunov orbits' positions related to the Earth 64
Figure 3.3 Planar Lyapunov orbits obtained semi-analytically (blue) and numerically (red) .. 65
Figure 3.4 Planar Lyapunov orbits' positions related to the Earth 65
Figure 3.5 Halo orbits obtained semi-analytically (blue) and numerically (red) 66
Figure 3.6 Halo orbits' positions related to the Earth 67
Figure 3.7 Lissajous orbit with β = 0.05 67
Figure 3.8 Lissajous orbit with β = 0.15 68
Figure 3.9 Vertical Lyapunov orbit's state vector over the time (null deviation) 69
Figure 3.10 Vertical Lyapunov orbit around L1 (null deviation) 69

Figure 3.11 Vertical Lyapunov orbit's state vector over the time (negative deviation) 70
Figure 3.12 Vertical Lyapunov orbit around L1 (negative deviation) 70
Figure 3.13 Vertical Lyapunov orbit's state vector over the time (positive deviation) 71
Figure 3.14 Vertical Lyapunov orbit around L1 (positive deviation) 71
Figure 3.15 Evolution of Vertical Lyapunov orbit over mass parameter 72
Figure 3.16 Comparison of the orbits 72
Figure 3.17 Halo orbit's state vector over the time (null deviation) 73
Figure 3.18 Halo orbit around L2 (null deviation) 73
Figure 3.19 Halo orbit's state vector over the time (negative deviation) 74
Figure 3.20 Halo orbit around L2 (negative deviation) 74
Figure 3.21 Halo orbit's state vector over the time (positive deviation) 75
Figure 3.22 Halo orbit around L2 (positive deviation) 75
Figure 3.23 Evolution of Halo orbit over the mass parameter 76
Figure 3.24 Comparison of three orbits when 10^{-7} deviation to the period is applied 76
Figure 3.25 Comparison of three orbits when 10^{-5} deviation to period is applied 77
Figure 3.26 Comparison of three orbits when 10^{-3} deviation to period is applied 77
Figure 3.27 Comparison of three orbits when 10^{-1} deviation to period is applied 78
Figure 3.28 Halo orbit computed by the collocation procedure 79
Figure 3.29 Evolution of state vector of given orbit over the time 79
Figure 3.30 Family of Halo orbits around L2 (3-D and X-Y) 79
Figure 3.31 Family of Halo orbits around L2 (X-Z and Y-Z) 80
Figure 3.32 Planar Lyapunov orbit computed by collocation 80
Figure 3.33 State vector of given orbit over the time 81
Figure 3.34 Family of Planar Lyapunov orbits 81
Figure 3.35 Halo and Planar Lyapunov families 82
Figure 3.36 Stable and unstable manifolds of one orbit around L1 83
Figure 3.37 Intersections needed to obtain Y - components 83
Figure 3.38 Homoclinic trajectory 84
Figure 3.39 Stable and unstable manifolds of the orbits around L1 and L2 85
Figure 3.40 Intersections needed to obtain Y - components for heteroclinic trajectories 85
Figure 3.41 Heteroclinic trajectory 86
Figure 3.42 Stable and Unstable manifolds of Halo orbits (Isometric view) 87
Figure 3.43 Stable and Unstable manifolds of Halo orbits (X-Y view) 87
Figure 3.44 Stable and Unstable manifolds of Halo orbits (X-Z view) 88
Figure 3.45 Intersection of Y and Z profiles 88

List of tables

Table 2.1 Coefficients up to order 3 for Lissajous orbits around L_1 34
Table 2.2 Coefficients up to order 3 for Lissajous orbits around L_2 34
Table 2.3 Coefficients up to order 3 for Halo orbits around L_1 37
Table 2.4 Coefficients up to order 3 for Halo orbits around L_2 37
Table 2.5 Butcher-array 42
Table 3.1 State vector of initial conditions (Vertical Lyapunov) 63
Table 3.2 State vector of initial conditions (Planar Lyapunov) 65
Table 3.3 State vector of initial conditions (Halo) 66
Table 3.4 State vector of Vertical Lyapunov orbit 69
Table 3.5 State vector of Halo orbit 73
Table 3.6 State vectors and periods of orbits 78
Table 3.7 The state vector of orbit to obtain homoclinic trajectory 83
Table 3.8 Initial conditions for Homoclinic trajectory 84
Table 3.9 The state vectors of orbits to obtain heteroclinic trajectory 84
Table 3.10 Initial seeds for heteroclinic trajectory 86
Table 3.11 State vectors of Halo orbits 87

This Page Intentionally Left Blank

INTRODUCTION

The objective of this project is to study the orthogonal collocation method implemented in AUTO – 07p software package and to apply the method of collocation for studying the phase space, natural motion, refining periodic orbits located in the neighborhood of libration points L1 and L2 and computing their stable and unstable manifolds.

Space programs of many countries include not only active exploration of the near-Earth space but also far-space exploration. Planning every space mission takes years and many details should be taken into consideration and many subjects such as space weather, communications, orbital dynamics, attitude control and heat transfer must be thoroughly studied from different angles. In our work we pay attention to a discipline called orbital dynamics.

Study of orbital dynamics is critical for space mission design. It is necessary both from a mathematical and an engineering points of view. Orbital dynamics is irreplaceable for computing trajectories of spacecrafts, asteroids, comets etc. The trajectory of a body moving in a gravitational field can be represented in wide varieties of models. The ease of integrations in a model depends on its simplicity. The most basic model uses Newton‘s laws of gravitation and integrations performance depends on this model. The model which we use, is the circular restricted three body problem. The CR3BP is the partial case of the three body problem. This problem is well described in many works, for example ‘The foundation of astrodynamics’ by Archie E. Roy [1].

In mission design the focus is done on practical usage of studied and proven capabilities which are obtained, for instance, by placing spacecrafts on periodic orbits around libration points.

Libration points, also called Lagrangian points, are special points in restricted three body problem where the third body with a negligible mass may stay fixed relative to primaries. There are 5 libration points. libration points L1, L2, L3 lay on the straight line connecting primaries and called collinear points, L4 and L5 are triangular points which form equilateral triangles with primaries on edges.

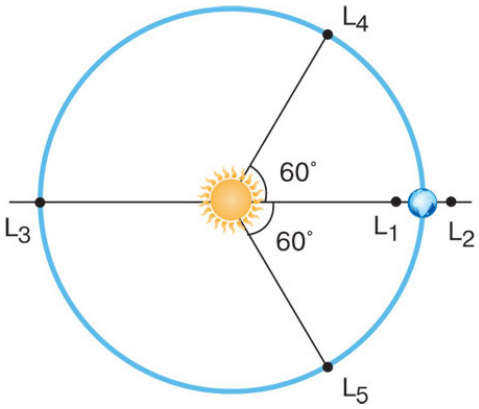

Figure 0.1 Libration points in the Sun-Earth system (http://press.cosmos.ru)

It is known that libration points can be orbited by small body, such as asteroid or spacecraft. There are five families of orbits existing around collinear libration points illustrated by Figure 0.2.

The movement in the neighborhood of collinear libration points, which are solutions of the circular restricted three body problem, may be considered a complex of two oscillators, one in-plane and another out-of-plane, and also some hyperbolic behavior. This means that orbits are unstable and even small adjustments can lead to the abandonment of the periodic orbit.

Orbits described by one of the oscillations (in-plane or out-of-plane), are usually called planar (horizontal) Lyapunov orbits or vertical Lyapunov orbits. Lyapunov was the first person who theoretically proved that there are periodical solutions (orbits) around points of equilibria with certain conditions.

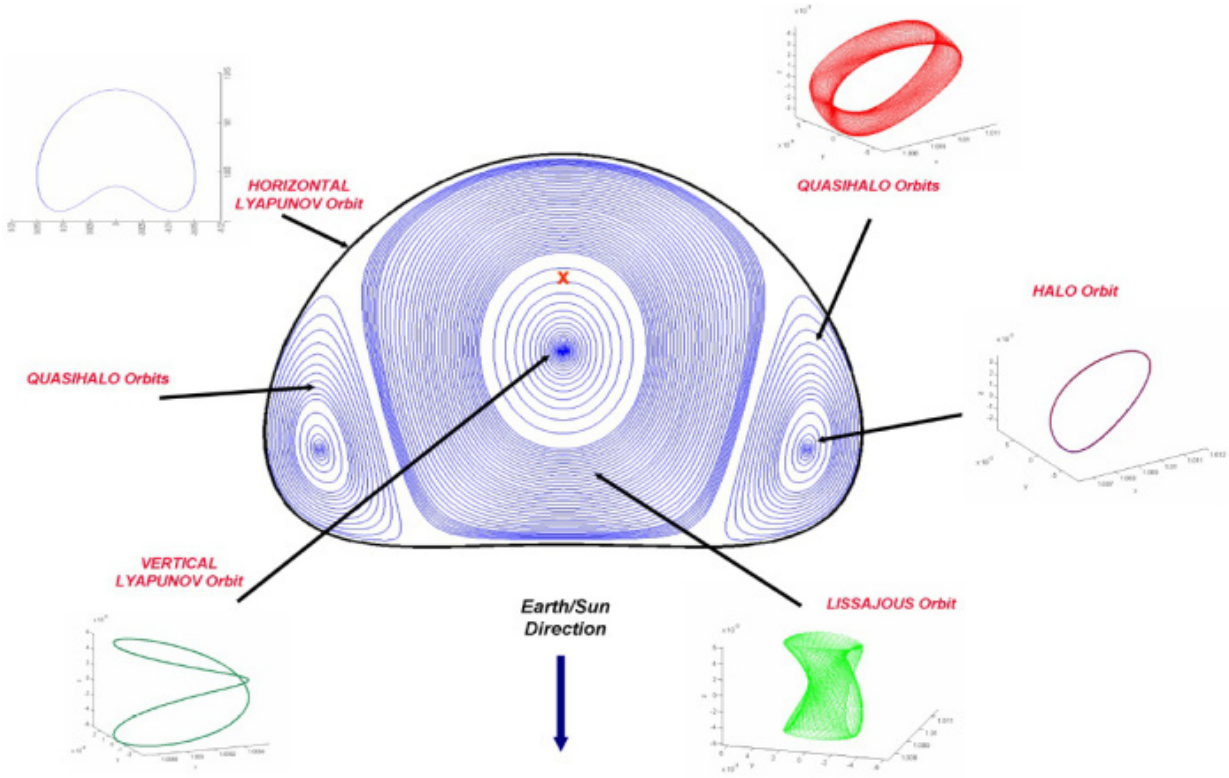

Figure 0.2 Different families of orbits [12]

If the difference in frequencies is incommensurable, the movement is not periodical and its trajectory is called Lissajous orbit. Lissajous orbits are quasi-periodic orbits.

Frequencies of oscillators change depending on amplitudes (non-linear problem) and equal frequencies result in Halo orbits.

Halo orbits is a family of periodic, three-dimensional orbits in the neighborhood of collinear libration points L1, L2 and L3. They can be computed semi analytically and numerically by using the circular restricted three body problem model.

Using orbits in the neighborhood of collinear libration points allows us not only to carry out astronomical and space research, monitor the state of the space, but also warn in time Geo services about an unusual solar activity and power of solar storms.

Launching a swarm of satellites, such us interferometers, is conceded for the study of far-space objects.

The implementation of all these projects requires a deep study of dynamics and phase space around collinear libration points, increase in the accuracy of the trajectory computation in order to minimize ΔV which is needed for keeping the orbit. According to this, our work, whose aim is to study phase space and natural motion (passive motion in Russia) in the neighborhood of libration points by collocation methods, looks of current interest.

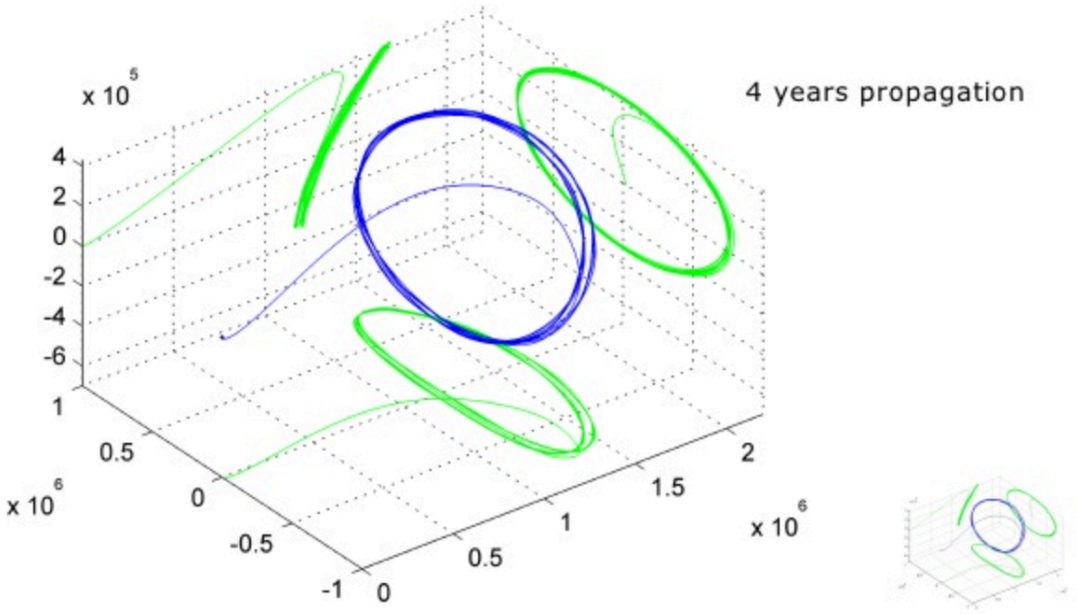

Figure 0.3 Example of Halo orbit

Libration point orbits can be defined by 2 parameters according to [2], in the case of the periodic Halo orbits α and β are in a relation:

- α – the degree of remoteness of the negligible mass (spacecraft) from libration point in the ecliptic plane (in-plane component);

- β – the degree of remoteness of the negligible mass (spacecraft) from libration point in the plane which is orthogonal to the ecliptic plane (out-of-plane component);

Halo orbits were coined by Robert Farquar in 1968. He described them in his dissertation. Farquar suggested placing a spacecraft on Halo orbit in the neighborhood of collinear libration point L2 in the Earth – Moon system to use it as a retranslator to establish a connection with Apollo located on the far-side of the Moon. The spacecraft

taking this kind of orbit can be simultaneously and continuously observed from the Earth and the Moon. Apollo mission to the far-side of the Moon was not fulfilled.

Farquar's idea was implemented later and several missions used Halo orbits around L1 and L2 in the Sun-Earth system. The first mission was ISEE – 3 launched in 1978. It took the Halo orbit around L1 and orbited L1 for several years. The following mission which used Halo orbit was NASA and ESA project, which aimed at solar observation – SOHO. It parked at L1 point in 1996. Although many other missions have used orbits around libration points since 1996, usually they have taken quasi-periodic Lissajous orbits. For example, spacecraft Genesis launched in 2001 was the first mission using low energy transfer for reaching and leaving the orbit (Figure 0.3).

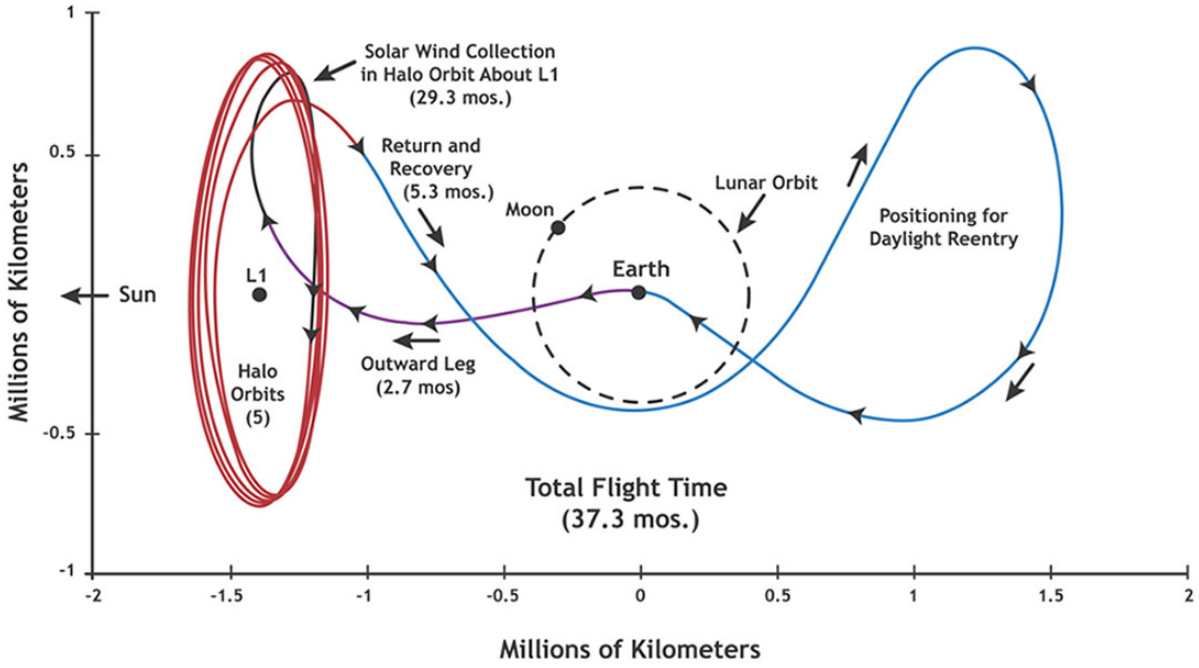

Figure 0.4 Low energy transfer and orbit of Genesis mission (http://www.hindawi.com/journals/mpe/2012/351759/fig1/)

The possibility of usage of low energy transfers was obtained as a result of studying invariant manifolds and connecting orbits. There are many works focused on these topics, for example Jobra and Masdemont (1999) [3, 4].

There are several methods to describe orbits around libration points. Talking about semi-analytical representation of orbits, it is necessary to refer to high order expansion which is based on the Lindstedt – procedure. For more accurate results, we need to use numerical methods. The most popular method is "shooting method" and its better version "multishooting method". These methods solve initial problem and apply corrections to initial conditions. The method of collocation provides high accuracy and requires small computation time and resources. It applies corrections to the whole trajectory. For the numerical study of the long-term (more than one period) behavior of an orbit we use numerical integrator which propagates given initial seed for a chosen period of time. As a numerical integrator we use function ode45 which which is based on Runge-Kutta method of order 4 and 5.

The paper is organized as follows. In chapter 1 we provide theoretical background about three body problem, its restricted version, reference frame used for the circular restricted three body problem, libration points and their orbits. Also explanation of the basic collocation method is given in chapter 1. Chapter 2 introduces the method of collocation applied for solving boundary value problem (BVP), representation of circular restricted three body problem as BVP, linear analysis of the orbits to obtain initial seeds numerical integrations and continuation of the family of orbits. In chapter 3 numerical results and plots of computations are presented. All families except quasihalo are shown.

This Page Intentionally Left Blank

Chapter 1

BACKGROUND

1.1. Dynamical Model

In this sub-section, we introduce a mathematical model which we decided to use for studying the neighborhood of the collinear libration points. Mathematical model is a description of some system or object by means of the mathematical language and equations. The process of building and developing of a mathematical model is called mathematical modelling. All natural (such as physics, biology), engineering (computer science) and even social (economics, linguistics and sociology) sciences are based on mathematical modelling: In them an object or a system of study is replaced by a mathematical model which closely describes the behavior of the object or the system. The model which we use in this work is The Circular Restricted Three Body Problem (CR3BP). The more detailed information is provided in the sections below. We show how we arrive at the CR3BP and start with a description of the main problem named The Three Body Problem.

1.1.1. Three Body Problem

In physics and celestial mechanics, the three body problem is a problem of taking three bodies with specific state vectors (positions and velocities) and masses in some chosen moment of time and then computing the motions of these three bodies according to the laws of classical mechanics, such as Newton's laws of motion and universal gravitation. In 1687 Newton discovered the law which describes the interaction between bodies in the gravitational fields. Now we know this law as the Newton's law of the universal gravitation (1.1).

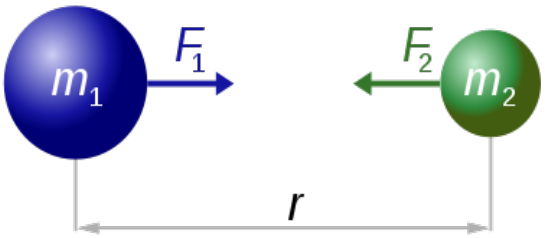

Figure 1.1 The illustration of the Newton's law of the universal gravitation

$$F = G\frac{m_1 m_2}{r^2} \tag{1.1}$$

Here F is the force between bodies, G is the gravitational constant which is equal to 6.674×10^{-11} N · $(m/kg)^2$, m_i is the mass of one of the bodies, r is the distance between the bodies.

The general statement for the three body problem is as follows. At an instant of time, for vector positions x_i and masses m_i, three coupled second-order differential equations exist:

$$\begin{aligned}
\ddot{x}_1 &= -\frac{Gm_2}{(x_1 - x_2)^3}(x_1 - x_2) - \frac{Gm_3}{(x_1 - x_2)^3}(x_1 - x_3) \\
\ddot{x}_2 &= -\frac{Gm_3}{(x_2 - x_3)^3}(x_2 - x_3) - \frac{Gm_1}{(x_2 - x_1)^3}(x_2 - x_1) \\
\ddot{x}_3 &= -\frac{Gm_1}{(x_3 - x_1)^3}(x_3 - x_1) - \frac{Gm_2}{(x_3 - x_2)^3}(x_3 - x_2)
\end{aligned} \tag{1.2}$$

The three body problem is a special case of the main problem which is termed The N-Body Problem, where n – is a free number of bodies. It is well known that taking a bigger and bigger number of bodies tragically increases the complexity of the n-body problem. It is not possible to find a general solution even for the three body problem. Many famous scientists (Lagrange, Jacobi, Poincaré, Birkhoff and etc.) spent plenty of years trying to look for and form a general solution of the three body problem.

Bruns and Poincaré proved that the system of differential equations of the motion of the three bodies cannot be reduced to an integrable problem, expanding it into independent equations.

According to the fact that the three body problem does not have a general solution we need to transform this problem into a simplified problem: The Restricted Three Body Problem. There are two problems to describe: The Circular Restricted Three Body Problem which is the one that we use and The Elliptical Restricted Three Body Problem which is very interesting to be mentioned but is non-autonomus and thus more complex. Before introducing CR3BP it is necessary to present the reference frame applied for the model.

1.1.2. Reference frame

A frame of reference is a set which contains a reference body, the associated to this body coordinate system and the reference time of the system, with respect to which, the motion of some body/bodies is studied. In modern physics, any movement is considered as a relative movement, and a movement of the body should be considered only in relation to any other body (body of reference) or a system of bodies. We cannot specify, for example, that the Moon moves in general, we can only determine its motion, for example, in relation to the Earth, the Sun, stars, and so on. The right choice of a reference frame can simplify the computation of a problem. In our project we use the Synodic Dimensionless Reference System. To simplify the equations of motion we need to transform SI units to convenient units. Let us take units of mass, length and time such that the sum of the masses of the primary and the secondary body, the

gravitational constant and the period of the primaries is 1, 1 and 2π, respectively. With these units the distance between the primaries is also 1 according to Kepler's third law. Mass ratio μ needed for the problem can be obtained by Equation 1.3 (for the Sun-Earth + Moon system μ = 3.040423398444176 x 10^{-6} according to JPL ephemeris DE403).

$$\mu = \frac{m_2}{m_1 + m_2} \tag{1.3}$$

The synodic reference frame which is centered at the barycenter of mass of the primary and the secondary body and is rotating in a way that they keep their positions on the X-axis is used. The orientation of the X-axis is provided by the line going from the secondary body to the primary body. The Z-axis has a direction given by the angular motion of m_1 and m_2 and Y-axis is orthogonal to the previous ones to establish a positively orientated coordinate system.

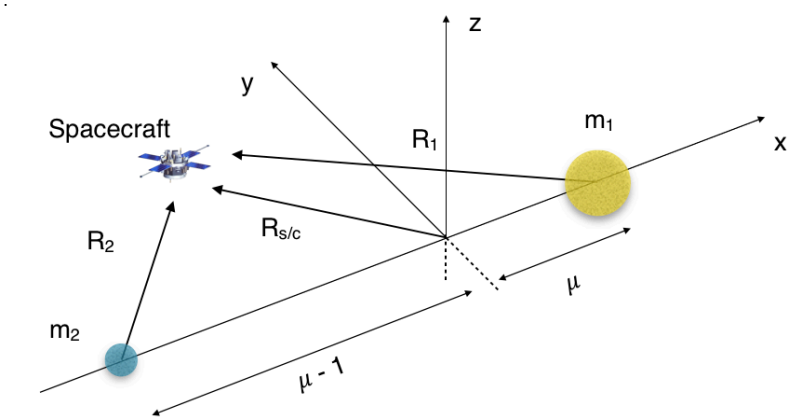

Figure 1.2 The Synodic Reference Frame of the RTBP

In this reference frame, the secondary body is located at (μ - 1, 0, 0) and the primary body is at (μ, 0, 0). The masses can be represented as 1 - μ for the primary body and as μ for the secondary body.
The described above reference system is easily applied for the circular restricted three body problem. Our model – CR3BP is described in the next sub-section.

1.1.3. The Circular Restricted Three Body Problem

The aim of this sub-section is to introduce the circular restricted three body problem. The CR3BP is the most analyzed model in a coordinate system with rotating two large bodies (m_1 and m_2). We can consider that there is a motion of a small mass (a spacecraft) under gravitational attraction of the two large bodies. In the Sun - Earth system, the Sun is a primary body and the Earth is a secondary body. Assume that the primary and the secondary bodies are orbiting circular orbits. The attraction of the spacecraft on the primaries is so small that they are rotating around their center of mass. The study of the motion of this small mass means that it is necessary to solve

the circular restricted three body problem. The structure of the circular restricted three body problem in the Synodic reference frame is illustrated in the Figure 1.2.

To have our model perfectly clear we start describing it from the very beginning [2,5,6]. Consider that there are two gravitational points with large masses *(A_1, m_1) (A_2, m_2)* which are called primary and secondary bodies, and they are rotating around their barycenter *C*. We are interested in the passive motion of a point with a small mass *(P, m)* in the gravitation field which is produced by the primary and the secondary bodies (or primaries).

Firstly, we define the right-orientated inertial coordinate system *CXYZ* (Figure 1.3) with the origin located in the barycenter *C.*

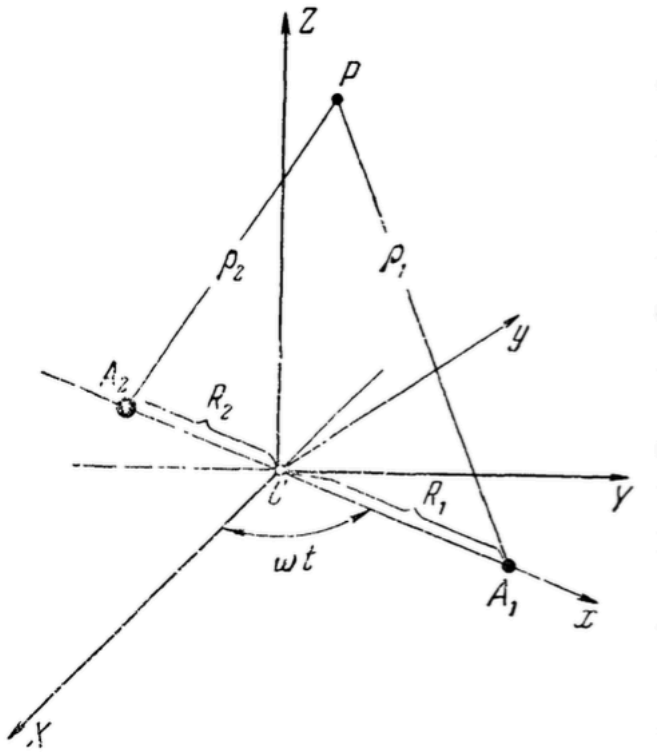

Figure 1.3 Coordinate systems *CXYZ* and *Cxyz*

The X – axis is taken in such a way that it is collinear with *CX* at the moment of time t = 0. The plane *CXY* is the plane where the primaries move. The positive direction of the rotation of the primaries is the anticlockwise direction.

Now we consider some terms and start describing the system using mathematical language. Assume that:

$$\overrightarrow{CA_1} = R_1 \,, \overrightarrow{CA_2} = R_2 \,, \overrightarrow{CP} = R, \qquad \textbf{(1.4)}$$

$$\left.\begin{aligned} &\overrightarrow{A_1P} = p_1, \overrightarrow{A_2P} = p_2, a = A_2A_1, \\ &M = m_1 + m_2, \mu = m_1 / M. \end{aligned}\right\} \quad \textbf{(1.5)}$$

Since C is the barycenter of points *(A₁, m₁)* and *(A₂, m₂)* we have $m_1R_1 = m_2R_2$. Moreover, $R_1 + R_2 = a$. And from here we obtain:

$$R_1 = \mu a, \quad R_2 = (1-\mu)a. \quad \textbf{(1.6)}$$

Forces F_1 and F_2, which are created by primaries and attracting the small body, are:

$$Gm_1m(R_1 - R)/p_1^3 \quad and \quad Gm_2m(R_2 - R)/p_2^3. \quad \textbf{(1.7)}$$

According to the second Newton's Law:

$$m\frac{d^2R}{dt^2} = F_1 + F_2, \quad \textbf{(1.8)}$$

whence

$$\frac{dR^2}{dt^2} = G\frac{m_1}{p_1^3}(R_1 - R) + G\frac{m_2}{p_2^3}(R_2 - R). \quad \textbf{(1.9)}$$

This is (1.9) differential equation which describes motion of the point with negligible mass *(P, m)* in the coordinate system *CXYZ*.

Above we have mentioned that we choose the other reference system for our model. The famous German mathematician Jacobi discovered that it would be possible to refine valuable information about motion of the small body *(P, m)*, if the rotating reference system has been applied: *CZ* – axis rotates together with A_2A_1 - axis.

Let us take a new non-inertial reference system which is called *Cxyz* (Figure 1.3). Let *Cx* – axis be the same as CA_1 at each moment of time, the plane *Cxy* and *Cz* – axis coincide with the plane *CXY* and *CZ* – axis, respectively. It is known that ω is angular velocity of the primaries orbiting the barycenter *C*. According to this our new system rotates around *CZ* – axis with angular velocity ω.

Before deriving the differential equations of the motion of *P,* we need to consider one subsidiary equation. Denote that the unit vectors of the coordinate system *CXYZ* as ***I, J, K*** and the unit vectors of the coordinate system *Cxyz* as ***i, j, k***. Coordinates of the point *P*, in this case, are X, Y, Z and x, y, z, respectively.

Then

$$R = XI + YJ + ZK. \tag{1.10}$$

Assume that

$$r = xi + yj + zk. \tag{1.11}$$

That brings us at

$$R = r.$$

Acceleration of the point *P* is given by the second derivative of the equation (1.10) in the coordinate system *CXYZ*:

$$\ddot{R} = \ddot{X}I + \ddot{Y}J + \ddot{Z}K, \tag{1.12}$$

and velocity and acceleration of the same point, but in the coordinate system *Cxyz,* are given by:

$$\dot{r} = \dot{x}i + \dot{y}j + \dot{z}k, \quad \ddot{r} = \ddot{x}i + \ddot{y}j + \ddot{z}k. \tag{1.13}$$

Rewrite $\ddot{R}$ by using $\ddot{r}, \dot{r}, r \; and \; \omega$:

$$\begin{aligned} X &= x\cos\omega t - y\sin\omega t, \\ Y &= x\sin\omega t + y\cos\omega t, \\ Z &= z. \end{aligned} \tag{1.14}$$

and

$$\begin{aligned} I &= i\cos\omega t - j\sin\omega t, \\ J &= i\sin\omega t + j\cos\omega t, \\ K &= k. \end{aligned} \qquad \textbf{(1.15)}$$

If we use the equations (1.14) to obtain $\ddot{X}, \ddot{Y}, \ddot{Z}$ and then put the obtained equations together with ***I, J, K*** into the equation (1.12), after simplification we will obtain:

$$\ddot{R} = \ddot{x}i + \ddot{y}j + \ddot{z}k + 2\omega(-\dot{y}i + \dot{x}j) - \omega^2(xi + yj). \qquad \textbf{(1.16)}$$

Consider the angular velocity vector:

$$\omega = \omega k.$$

Then, using the rule $\bar{a}\times(\bar{b}\times\bar{c}) = \bar{b}(\overline{ac}) - \bar{c}(\overline{ab})$, equation (1.16) can be rewritten as:

$$\ddot{R} = \ddot{r} + 2(\omega\times\dot{r}) + \omega(\omega\times r). \qquad \textbf{(1.17)}$$

That is the subsidiary equation which we looked for.

Now we have all the necessary information to derive the equation of the motion of the point *P* in the rotating coordinate system *Cxyz*. Consider that primaries A_1 and A_2 have coordinates (X_1, Y_1, 0), (X_2, Y_2, 0) and (x_1, 0 , 0), (x_2, 0 ,0) in the coordinate systems *CXYZ* and *Cxyz*, respectively, where $x_1 = \mu a$ and $x_2 = (\mu - 1)a$.
It is obvious that:

$$R_1 = X_1 I + Y_1 J, \quad R_2 = X_2 I + Y_2 J.$$

Assume that:

$$r_1 = x_1 i, \quad r_2 = x_2 i.$$

From here we have:

$$\overrightarrow{CA}_1 = R_1 = r_1, \quad \overrightarrow{CA}_2 = R_2 = r_2. \tag{1.18}$$

Now we rewrite the equation (1.9) by using equations (1.17) and (1.18), and obtain:

$$\ddot{r} = -2(\omega \times \dot{r}) - \omega \times (\omega \times r) + \frac{Gm_1}{p_1^3}(r_1 - r) + \frac{Gm_2}{p_2^3}(r_2 - r), \tag{1.19}$$

where

$$p_k = |r_k - r| = \sqrt{(x_k - x)^2 + y^2 + z^2}, k = 1,2.$$

This equation (1.19) is the vector differential equation of the motion of the point *P* in the rotating coordinate systems *Cxyz*. There is another way to write the equation (1.19) using the (1.16):

$$\ddot{r} = -2\omega(\dot{y}i - \dot{x}j) + \omega^2(yi + xj) + \frac{Gm_1}{p_1^3}(r_1 - r) + \frac{Gm_2}{p_2^3}(r_2 - r). \tag{1.20}$$

The last vector differential equation (1.20) is equivalent to the system of three scalar differential equations:

$$\begin{cases} \ddot{x} = 2\omega\dot{y} + \omega^2 x + \dfrac{Gm_1}{p_1^3}(x_1 - x) + \dfrac{Gm_2}{p_2^3}(x_2 - x), \\ \ddot{y} = -2\omega\dot{x} + \omega^2 y + \dfrac{Gm_1}{p_1^3} y + \dfrac{Gm_2}{p_2^3} y, \\ \ddot{z} = -\dfrac{Gm_1}{p_1^3} z + \dfrac{Gm_2}{p_2^3} z. \end{cases} \tag{1.21}$$

In case when m_1 and m_2 are well known values, it is possible to study numerically the behavior of the system for some small period of time. The system can be approximately solved by applying well-developed integration methods.

Now we apply the reference frame to simplify the equations of the motion as it is possible. The name of the system is the Synodic Dimensionless Reference System. In this reference frame the sum of two masses m_1 and m_2 equals one, the distance between A_1 and A_2 is also equal to one. In this system period T is equal to 2π. Now we need to obtain the value of f. For this we apply the formula $\frac{a^3}{T^2} = f(m_1 - m_2)/4\pi^2$ (2.9.10) from Lidov M.L.[5]. If $a = 1, T = 2\pi, m_1 + m_2 = 1$, so $f = 1$. In the Synodic Dimensionless Reference System, we can replace m_1 with 1-μ and m_2 with μ, $f = 1$ and $\omega = 1$ in the equations (1.21).

$$\begin{cases} \ddot{x} = 2\dot{y} + x + \dfrac{1-\mu}{p_1^3}(x_1 - x) + \dfrac{\mu}{p_2^3}(x_2 - x), \\ \ddot{y} = -2\dot{x} + y - \dfrac{1-\mu}{p_1^3}y - \dfrac{\mu}{p_2^3}y, \\ \ddot{z} = -\dfrac{1-\mu}{p_1^3}z - \dfrac{\mu}{p_2^3}z. \end{cases} \quad \textbf{(1.22)}$$

It is necessary to note, that the equations (1.22) can be represented as a form which is much easier to memorize. For this we need to consider a new function:

$$\Omega(x,y,z) = \frac{1}{2}(x^2 + y^2) + \frac{1-\mu}{p_1} + \frac{\mu}{p_2} + \frac{1}{2}\mu(1-\mu). \quad \textbf{(1.23)}$$

It is easy to see that the system (1.22) can be rewritten as:

$$\begin{cases} \ddot{x} = 2\dot{y} + \dfrac{\partial\Omega}{\partial x}, \\ \ddot{y} = -2\dot{x} + \dfrac{\partial\Omega}{\partial y}, \\ \ddot{z} = \dfrac{\partial\Omega}{\partial z}. \end{cases} \quad \textbf{(1.24)}$$

The system (1.24) is the set of equations which we use as our mathematical model. To complete the introduction of the model we need to present the Jacobi constant (named after Carl Gustav Jacobi) [7] which is the only conserved quantity in the CR3BP. The Jacobi constant is used as an effective tool to control the correctness of the integrations.

We multiply the first, the second and the third equation of (1.24) by $2\dot{x}, 2\dot{y}$ and $2\dot{z}$, respectively, then obtain the equation by adding all multiplied results:

$$2(\dot{x}\ddot{x} + \dot{y}\ddot{y} + \dot{z}\ddot{z}) = 2(\frac{\partial \Omega}{\partial x}\dot{x} + \frac{\partial \Omega}{\partial y}\dot{y} + \frac{\partial \Omega}{\partial z}\dot{z}),$$

or

$$\frac{d}{dt}(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) = 2\frac{d\Omega}{dt}.$$

From here we can obtain:

$$(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) = 2\Omega - C, \qquad \textbf{(1.25)}$$

where C – is the Jacobi constant.

This equation (1.25) is the first integral for the system (1.24) and named the Jacobi integral. It is easy to see that:

$$C = 2\Omega(x, y, z) - (\dot{x}^2 + \dot{y}^2 + \dot{z}^2). \qquad \textbf{(1.26)}$$

In this section of chapter 1 we thoroughly described the mathematical model which is used in our project. In the next section we focus on describing the five partial solutions of the CR3BP and their application.

1.2. Libration points

The objective of this section is to provide the information about equilibrium points, their positions and practical meaning. It is well known that the CR3BP has five equilibrium points named libration or Lagrangian points which are named after Josef Lagrange who was the first person who proved the existence of L_4 and L_5 points in 1772 (Euler discovered L_1,L_2 and L_3 in 1768). Figure 1.4 shows locations of libration points in the Sun - Earth system. A libration point is a location in Space where the mix of gravitational forces of two big bodies, like the Sun and the Earth, equals the centrifugal force which affects the point *P* with a small mass, such as a spacecraft. The combination of forces produces special places in the space where a spacecraft can be parked to make observations [1,2,8]. There are five points, three of them lie along the line connecting the primary and the secondary bodies. These points are called the collinear libration points L_1, L_2 and L_3. The remaining two, known as L_4 and L_5, are also in the plane Z = 0 forming an equilateral triangle with the primary and the secondary body.

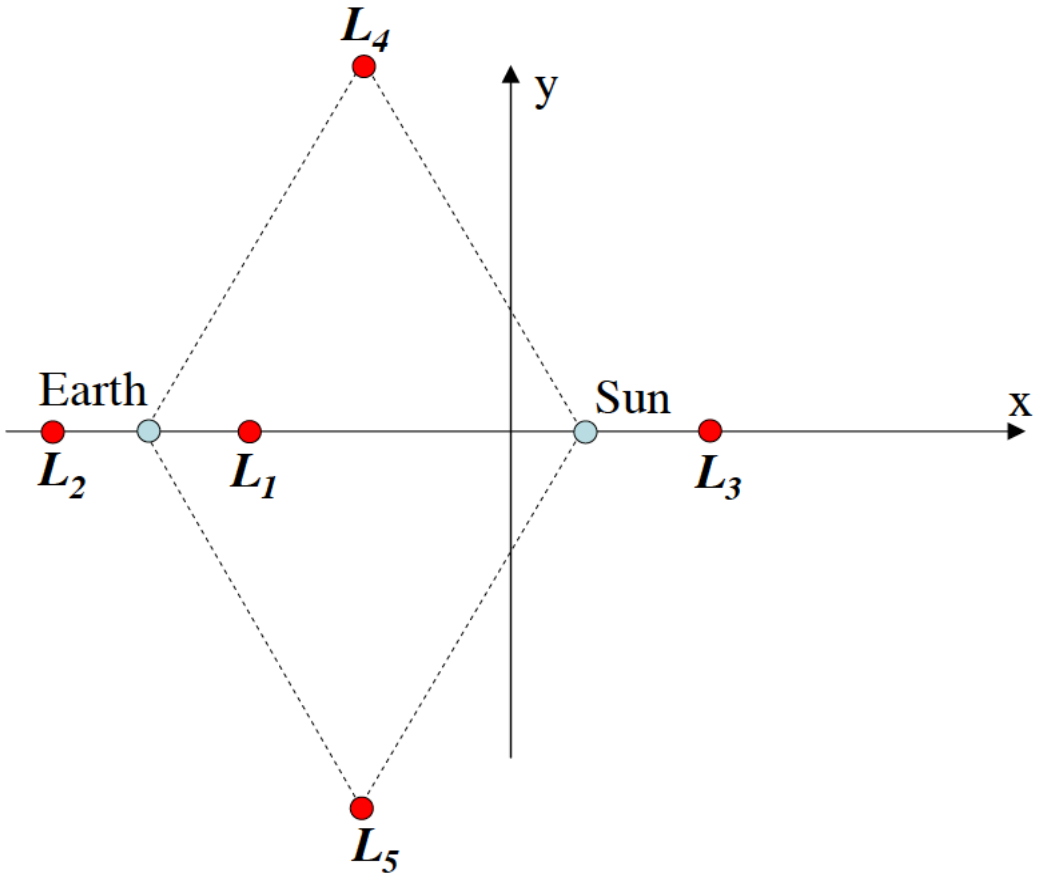

Figure 1.4 Libration points of the Sun-Earth system

Figure 1.4 shows locations of the all equilibrium points of the Sun-Earth system. The L_1 point lies on the line defined by two large masses m_1 and m_2, and between them. At this point gravitational attraction of m_2 partially cancels m_1's gravitational attraction that is why an object (a spacecraft) which is located closer to the Sun does not rotate around it faster than the Earth. At the L_1 point, the orbital period of the object becomes exactly equal to Earth's orbital period. L_1 is about 1.5 million kilometers from the Earth.

The L_2 point lies on the same line, beyond the smaller mass (the Earth). On the opposite side of the Earth from the Sun, the orbital period of an object would normally be greater than that of the Earth. The extra pull of Earth's gravity decreases the orbital period of the object, and at the L_2 point in such a way that the orbital period becomes equal to Earth's. L_1, L_2 are about 1.5 million kilometers from Earth. The L_3 point lies on the line defined by the two large masses, beyond the larger of the two. L_3 is located beyond the primary body of the system. The location of the point is a little bit closer than the Earth's orbit and the attraction of combined pull of the Earth and the Sun provides the point with the period which is equal to the Earth's one. The L_4 and L_5 points lie in the third corners of the two equilateral triangles in the plane of orbit whose common base is the line between the centers of the two masses, such that the point lies behind or ahead of the smaller mass with regard to its orbit around the larger mass. L_4 and L_5 are stable equilibria if mass ratio $m_1/(m_1+m_2)$ is close to 0.0397 for L_4 and for L_5. In contrast to L_4 and L_5, where stable equilibrium exists, the points L_1, L_2, and L_3 are positions of unstable equilibria. Any object orbiting at one of L_1–L_3 will have a tendency to fall out of an orbit.

1.2.1. How to compute the position of L-points

Above we have provided the information about libration points positions, in this section we present how the positions are computed. Figure 1.4 illustrates a schematic view of the Sun-Earth system and its equilibrium points. Figure 1.5 provides a more detailed overview.

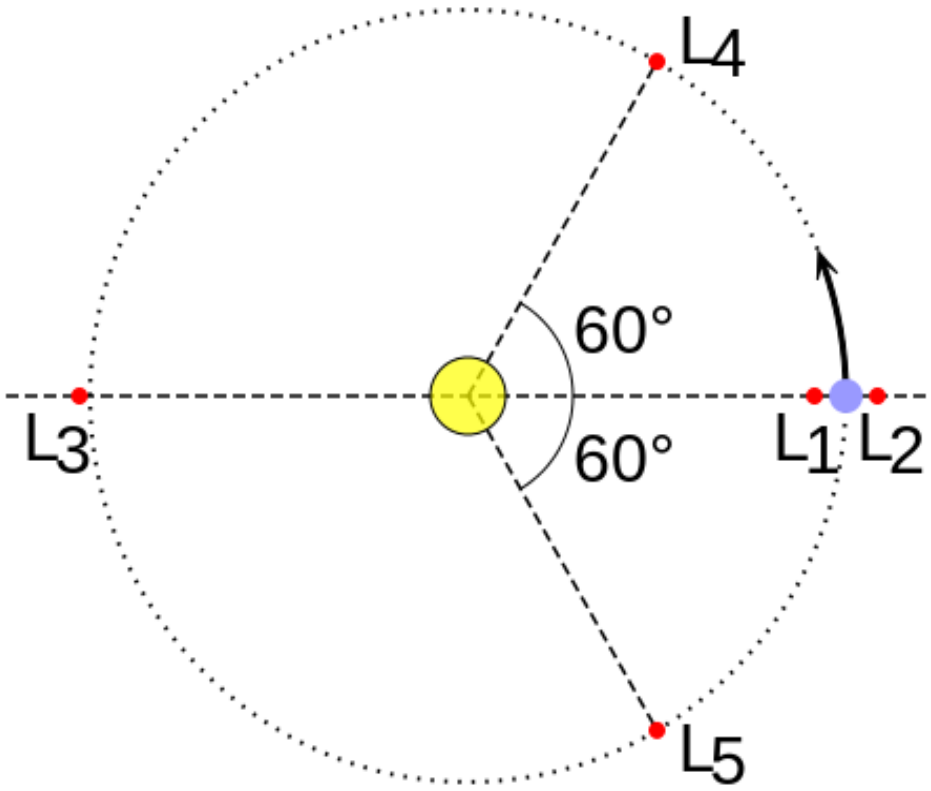

Figure 1.5 Libration point locations

There are two types of the libration points: collinear (such as L_1, L_2, L_3) and triangular (such as L_4, L_5). According to this fact the ways of the computation of their locations are not the same. Let us start with calculations of the locations of the collinear group of the points. It is well known that L_1 is located between the primaries closer to the less massive body, L_2 is behind the less massive body and L_3 is behind the large mass. The approximate distances to these points from the center of masses can be computed by:

$$r_1 \simeq (1-(\frac{\mu}{3})^{1/3}, \quad 0),$$
$$r_2 \simeq (1+(\frac{\mu}{3})^{1/3}, \quad 0), \qquad \textbf{(1.27)}$$
$$r_3 \simeq (1+\frac{5}{12}\mu, \quad 0).$$

but the true values come from the solving Euler's quintic equation.

It is interesting to note, that in the Sun-Earth system L_3 is located approximately at the same distance as the Earth. (for the Sun-Earth + Moon system μ = 3.040423398444176 x 10^{-6} according to JPL ephemeris DE403)

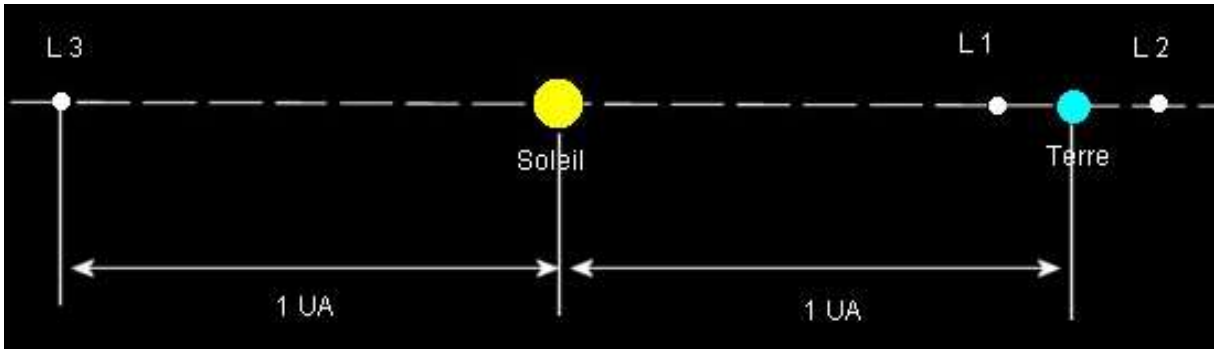

Figure 1.6 Collinear points of the Sun-Earth system

The L_3 point in the Sun - Earth is located on the opposite side of the Earth's orbit. However, despite its small gravity (compared to the Sun's gravity), the Earth still has a little impact on the Sun, so L3 point is not on the Earth's orbit, but a little farther from the Sun than the Earth, since the rotation is not around the Sun but around the barycenter. The combination of gravity of the Sun and the Earth at the point L_3 results in that the object moves with the same orbital period as our planet.

The existence of these points and their high instability is caused by the facts that the distance to the two bodies at these points is the same, the force of gravity of two mass bodies is in the same relation as their masses, and thus the resulting force is directed

to the center of mass of the system; In addition, the geometry of the triangle of the forces proves that the resultant acceleration is associated with the distance to the center of mass in the same relation as the two massive bodies do.

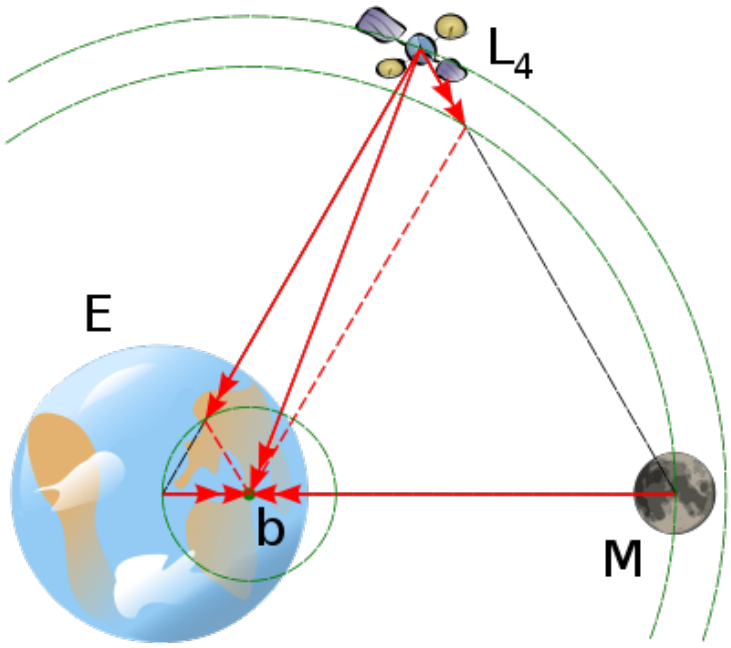

Figure 1.7 Gravitational acceleration at the L4 in the Earth-Moon system

Since the center of mass is also a center of rotation of the system, the resultant force is exactly the same which is necessary to keep the body in the neighborhood of Lagrange point in orbital equilibrium with the rest of the system. (In fact, the third weight body should not be negligible). This triangular configuration was discovered by Lagrange and was described the first time in his work “The Three-Body Problem”. L_4 and L_5 points are called triangular points. The locations of these points from the center of masses of the system are given by:

$$r_4 \simeq (\frac{m_1 - \mu}{2}, \quad \frac{\sqrt{3}}{2}),$$
$$r_5 \simeq (\frac{m_1 - \mu}{2}, \quad -\frac{\sqrt{3}}{2}). \qquad \textbf{(1.28)}$$

In out project, we focus on the neighborhood around two collinear Lagrangian points L_1 and L_2 which are more interesting in terms of space mission design. Above the method to obtain approximate positions of the libration points has been described, but to obtain true positions of the two collinear points we need to solve an equation which can be used for computation of the distance between the libration point and the closest

primary body. A unique positive root of the Euler's quintic equation solution is the distance γ between the one of L_1, L_2 points and the nearest primary, in our case the Earth [11]:

$$\gamma^5 \mp (3-\mu)\gamma^4 + (3-2\mu)\gamma^3 - \mu\gamma^2 \pm 2\mu\gamma - \mu = 0, \quad \textbf{(1.29)}$$

with upper signs for L_1 and the lower sings for the L_2.

Solution of the equation is used for the linear analysis of the neighborhood of the libration points. The semi-analytical procedure is presented in chapter two.

Above we have mentioned that points L_1 and L_2 are interesting in terms of space mission design, it is true because these points have high application capabilities. Talking about L_1 point we have to say that in the Sun-Earth system L_1 point can be a perfect site to host the space observatory to monitor the Sun, which at this point is never blocked by the Earth or the Moon. Moon Point L_1 (in the Earth – Moon system) can be an ideal place to build manned space orbital station, which, being located in the "halfway" between the Earth and the Moon, would allow easy access to the Moon with the minimum cost of fuel, and to become a key hub of freight traffic between the Earth and its satellite. The L_2 point has a completely different application. The L_2 point in the Sun – Earth system is an ideal place for the construction of orbital space observatories and telescopes. Since the object at the point L_2 is capable maintaining for a long time its orientation relative to the Sun and the Earth, it becomes much easier to calibrate its position and produce shielding. However, this point is located slightly on the Earth's shadow (in the penumbra), so that the solar radiation is not completely blocked. The L_2 point in the Earth-Moon system can be used for placing communication satellite to establish contact with objects on the far side of the Moon, as well as being a convenient location for the fuel station to ensure traffic between the Earth and the Moon. The L_3 point in the Sun-Earth system is also worth to be mentioned. Orbital spacecrafts and satellites, located around the L_3 point can constantly monitor the various forms of activity on the Sun's surface, in particular, the appearance of new spots or flashes, and efficiently transmit information to the Earth (for example, within the framework of an early warning system for space weather NOAA Space Weather Prediction Center). In addition, information from these satellites can be used to ensure the safety of long-distance manned space flights, such as to Mars or asteroids. In 2010, several options for the launch of this satellite were studied.

It is obvious that the libration point missions require deep study of orbital dynamics in the neighborhood of the collinear libration points and orbits existing around them. The next sub-section introduces libration point orbits.

1.2.2. Libration point orbits

The motion of the negligible mass in the neighborhood around two collinear libration points L_1 and L_2 is an important thing to study in the case of libration point mission design. There are 5 different kinds of orbits around the equilibrium points. They have very different shapes but all of them can be described as a set of two oscillators, one in-plane and another out-of-plane, and also some hyperbolic behavior. That hyperbolic component means that orbits are unstable and even small adjustments can lead to the abandonment of the periodic orbit. Figures bellow illustrate all existing families of the libration point orbits.

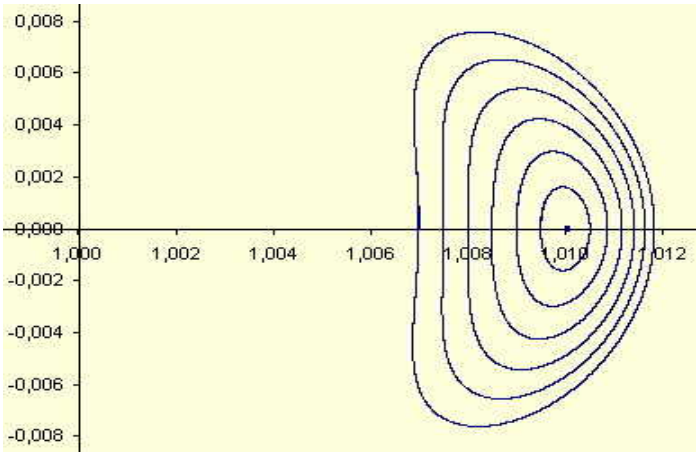

Figure 1.8 Planar Lyapunov family

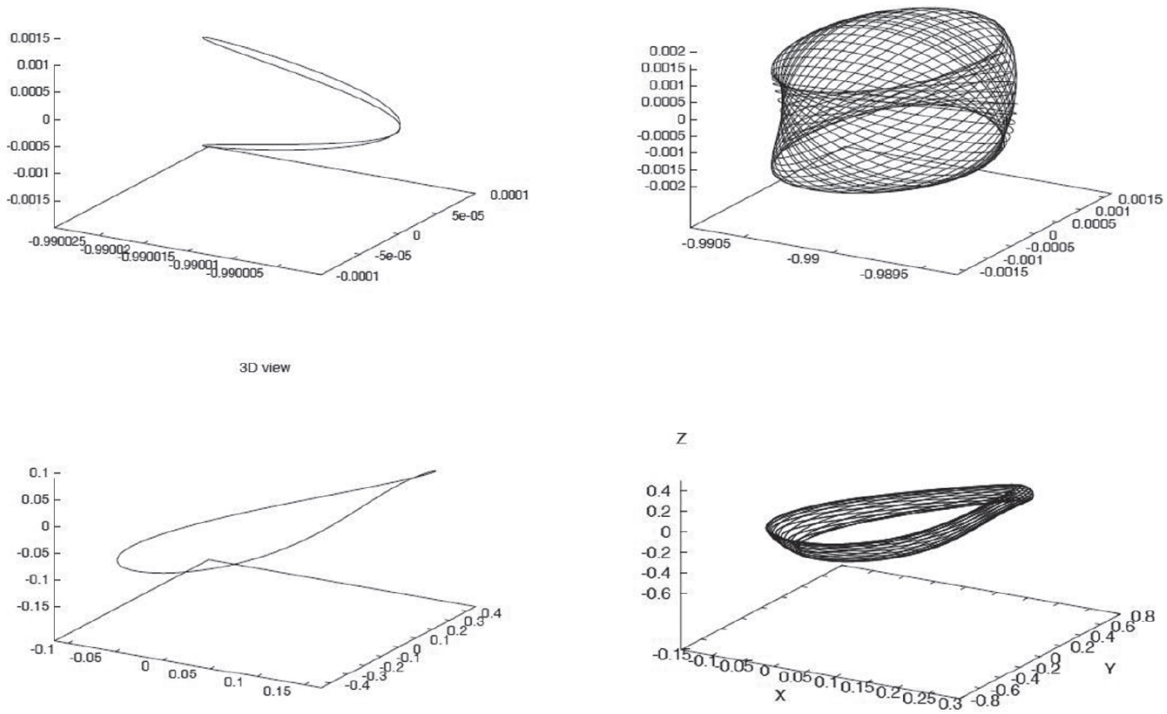

Figure 1.9 Top-left: Vertical Lyapunov orbit, top-right: Lissajous quasi-orbit, bottom-left: Halo orbit, bottom-right: Halo quasi-orbit [4]

In a mission design two families of orbits are more interesting than others: Halo orbits and Lissajous quasi – periodic orbits. But refering to the aim of our project, which is deep study of natural motion in the neighborhood of the libration points, we describe all families of the orbits. The motion around the libration points L_1 and L_2 can be separated by two classes: a periodic motion and a quasi – periodic motion.

We suggest starting with the description of periodic orbits. As an example we take L_2 point and the linear six-dimensional phase space around it which is a *center* x *center* x *saddle* [4]. If the energy is close to that at L_2, there exist two families of periodic orbits; the planar Lyapunov orbits, which are in the ecliptic plane, and the symmetric figure-eight-shaped vertical Lyapunov orbits. These orbits were named after a famous Russian mathematician Aleksandr Mikhailovich Lyapunov whose theorem says that there is a family of periodic orbits surrounding each of these points; one can think of this meaning that one can “go into orbit about these points”. Refering to the fact that motion in the neighborhood of a libration point can be considered as two oscillators and hyperbolic component Lyapunov orbits can be described by one of the oscillations in-plane or out-of-plane and depending on oscillation we have planar Lyapunov orbit or vertical Lyapunov orbit, respectively. An increase in energy (bigger amplitudes) leads to several effects: non-linear terms become important, the phase space is broken, a new family of periodic orbits bifurcate from the planar Lyapunov orbit family. This family, which is three-dimensional and asymmetric about the ecliptic plane, is called Halo orbits. The name of Halo orbits was coined by Robert Farquar in 1968. He described them this way in his dissertation

As we have mentioned in the introduction, all libration point orbits can be represented by two parameters:

- α – the degree of remoteness of the negligible mass (spacecraft) from libration point in the ecliptic plane (in-plane component);
- β – the degree of remoteness of the negligible mass (spacecraft) from libration point in the plane which is orthogonal to the ecliptic plane (out-of-plane component);

For the case of Halo orbits α and β are depended. Different combinations of these parameters can result in different types of orbits. It is important, that by manipulating the amplitudes it is possible change the type of an orbit. Having α and β not equal to zero we obtain a Lissajous orbit, but if α or β is equal to zero, we have a vertical Lyapunov orbit or planar Lyapunov orbit, respectively. In the case of Halo orbits α depends on the value of β. The method how the Halo orbits are defined by these parameters is described in chapter two.

Figure 1.10 shows three existing families of periodic orbits around the libration point 2.

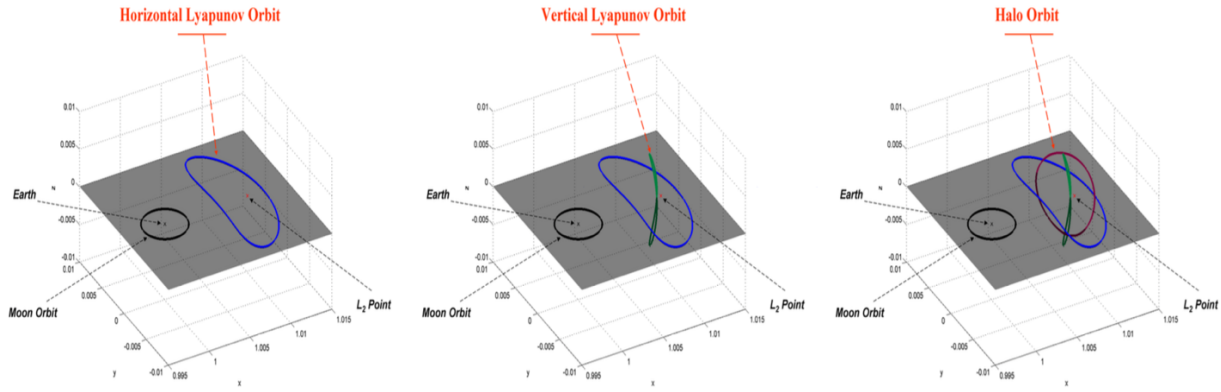

Figure 1.10 Periodic orbits around L2

In the neighborhood of the libration point (still use L_2 as the example) there are two families of quasi-periodic families: The Lissajous family around the Vertical Lyapunov orbits, and the Quasi-Halos around the Halo orbits. If the ratio of the oscillation frequencies in the different planes is irrational, the motion takes quasi-periodic character. In the case of close frequencies and amplitudes of the oscillations of the spatial periodic solution - a Halo orbit - a quasi-periodic trajectory lies in the vicinity of the periodic Halo orbit to the surface of the two-dimensional torus. Such a trajectory family is called Quasi-Halo orbits. If the frequencies vary significantly, the trajectory is called Lissajous orbit (named after Julies Antoine Lissajous). Figure 1.11 illustrates Quasi – Halo and Lissajous orbits.

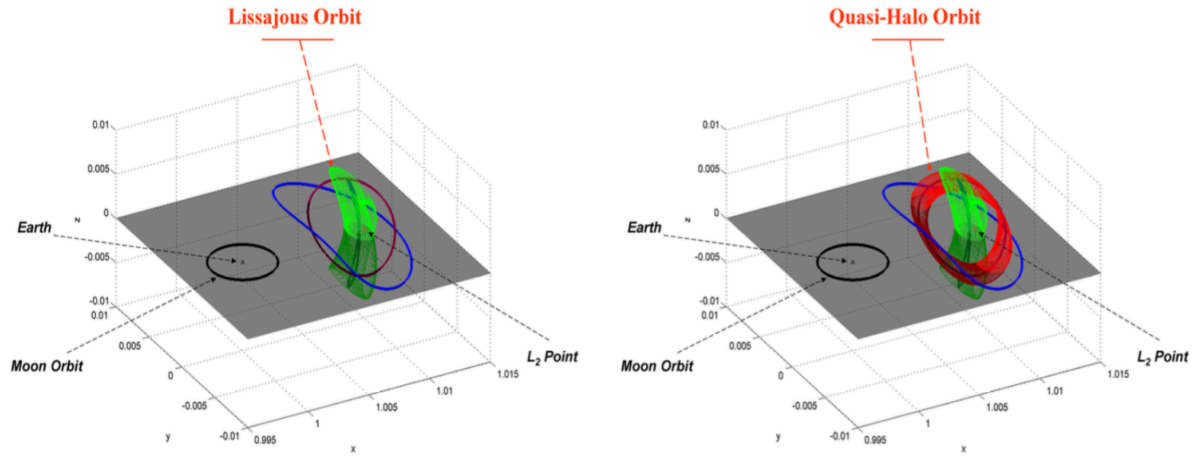

Figure 1.11 Quasi - Periodic orbits around L2

So there are five families of the orbits in the neighborhood of each collinear libration point. It is well known that the collinear libration points are unstable equilibria, so there exist transfer and return trajectories which can be considered as manifolds. The following sub-section gives basic information about the manifold and connecting orbits.

1.2.3. Invariant manifolds and connecting orbits

In this sub-section we go over the basics of stable and unstable manifolds associated to the fixed points of a dynamical system. According to the fact that our problem – CR3BP is the dynamical system and the libration points are fixed, we can consider that stable and unstable manifolds exist in our system. In the section 1.2 we have mentioned that the Lagrangian points are unstable equilibria, so periodic solutions (orbits) in the neighborhood of the libration points are classified as unstable[14]. According to this, libration point orbits have a set of invariant manifolds. Invariant manifolds are natural trajectories in the phase space which show departure from or approach to an orbit. Figure 1.12 illustrates invariant manifolds and the direction of the motion.

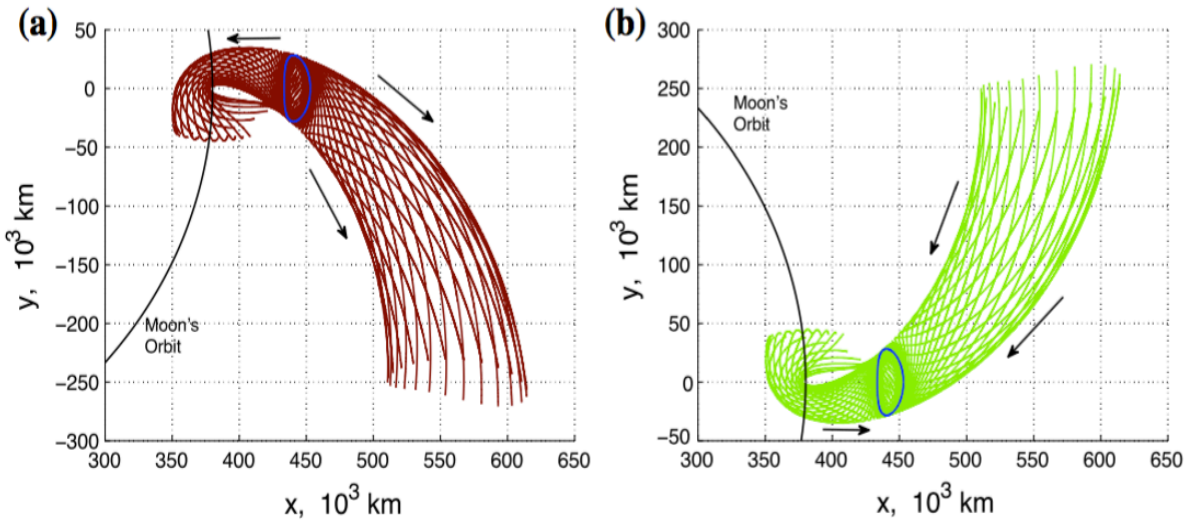

Figure 1.12 (a) Unstable invariant manifold (b) Stable invariant manifold

The unstable manifold is a set of all possible trajectories, which asymptotically depart from a nominal orbit. The stable manifold includes the set of all possible trajectories that a particle could take to asymptotically arrive onto the nominal orbit.

Talking about connecting orbits, it is worth mentioning two researchers Conley and McGehee who proved the existence of homoclinic connecting orbits [15, 16]. These connecting orbits establish the way from an orbit to the same orbit. From a practical point of view, heteroclinic orbits are more interesting. They provide a transfer from one libration point orbit to another one. This kind of connection is described by Koon [17]. These kind of orbits can be used for “Lagrange stairway”, for low cost transfers and as return trajectories.

Choosing a point, an orbit and a transfer/return trajectory for a mission totally depends on the objectives of each mission. The following sub-chapter provides information about the mission around the Lagrangian points.

1.2.4. Libration point missions

Libration point missions are of grate interesa in terms of space science, astronomy, astrophysics and fundamental science. Every mission is carefully planned and thoroughly studied from different angles. The first mission to Lagrangian points was organized by NASA in 1978 and spacecraft was named The International Sun Earth Explorer 3 (ISEE - 3). The ISEE – 3 was placed in a Halo orbit around the libration point L_1 in the Sun – Earth system. Then another two spacecrafts (WIND and SOHO) were placed at the L_1 point in 1994 and 1995. For example, SOHO (Solar and Heliospheric Observatory) orbited a Halo orbit with a period which is equal to 178 days (it circles L_1 approximately twice a year). The Halo orbit provided SOHO with the site for constantly monitoring of the Sun and comets which flew by the Sun. Not only Halo orbits are used around the libration point L_1, for instance, the spacecraft named Advanced Composition Explorer (ACE) used Lissajous orbit. It was launched in 1997. The spacecraft is still in generally good condition in 2016, and is projected to have enough fuel to maintain its orbit until 2024.

The libration point behind the Earth (from the Sun) is used for scientific needs, telescopes and observatories. The Wilkinson Microwave Anisotropy Probe (WMAP) was a spacecraft operating from 2001 to 2010 which measured differences across the sky in the temperature of the cosmic microwave background (CMB) – the radiant heat remaining from the Big Bang. It took the Lissajous orbit around L_2 point by using swingby to approach (figure 1.12).

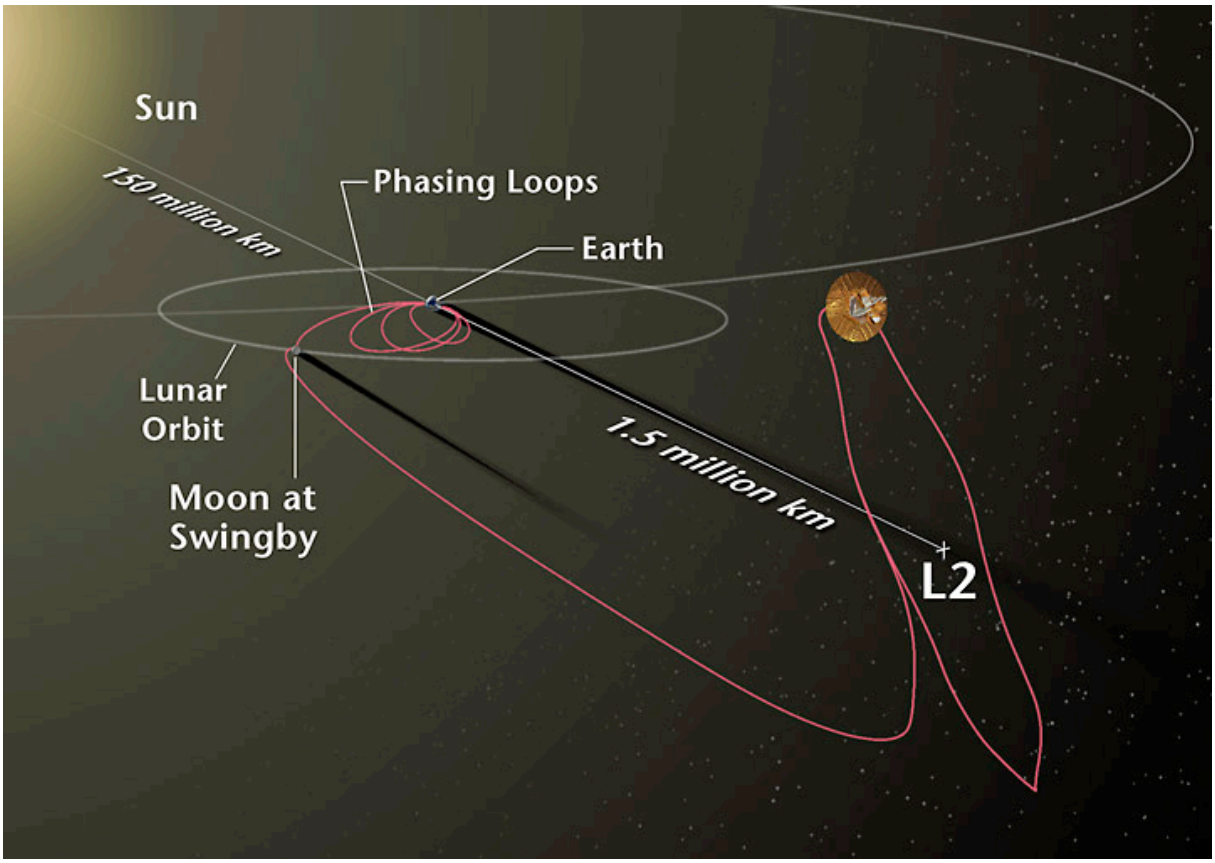

Figure 1.13 WMAP mission (http://map.gsfc.nasa.gov)

Another spacecraft which used a Lissajous orbit around L_2 was Herschel Space Observatory. This satellite was launched together with PLANK which was placed in a Halo orbit around L_2 in 2009. There are some missions which are being studied now and planned to fulfill in the future. For example, space telescope "James Webb" coming to replace the telescope "Hubble". The launch is planned for 2018. In 2017, Lavochkin and Roscosmos intend to place the point L2 space observatory Spektr-RG at the point L_2. Spektr-RG instrumentation consists of 5 telescopes spanning the energy range from the far ultraviolet to the hard X-ray, plus an all-sky monitor.

Although there are no satellites and spacecraft which are orbiting L_4 of L_5 points, there were two spacecraft of STEREO mission which used them as transit points in 2009. Moreover, there was the project named JIMO (Jupiter Icy Moons Orbiter). JIMO - the canceled NASA project to study the moons of Jupiter, which was supposed to actively use the Lagrange points of the system to transfer from one satellite to another with minimal fuel costs. This maneuver is called "Lagrange staircase".

Now we have provided all theoretical background about the mathematical model and the system which this model describes. The goal of this project is to apply the method of collocation to the CR3BP for the study of a natural motion. The next section introduces basics of the method of collocation.

1.3. The method of collocation

In the last section of chapter one the method which was chosen for the project is introduced. The method of collocation and its variations are widely used for different kinds of computations in a large number of sciences. In engineering the method is used for, for instance, some computations of nuclear plant reactors, studying the pressure on roads [18] etc. In this section we give the basics of collocation.

Suppose we want to define a function $y = y(x)$ that satisfies the linear differential equation (note that *p(x)*, *q(x)* and *f(x)* are known):

$$L(y(x)) \equiv y'' + p(x)y' + q(x)y = f(x) \tag{1.30}$$

subjected to the linear boundary conditions:

$$\left.\begin{aligned} \Gamma_a &\equiv a_0 y(a) + a_1 y'(a) = A \\ \Gamma_b &\equiv \beta_0 y(a) + \beta_1 y'(a) = B \end{aligned}\right\}, \tag{1.31}$$

where $|a_0| + |a_1| \neq 0,\ |\beta_0| + |\beta_0| \neq 0.$

It is necessary to note that A, B, α_0 and β_0 are given preselected values.

We choose a set of linearly independent functions:

$$U_0(x), U_1(x), ..., U_n(x), \tag{1.32}$$

which we call the system of basis functions.

Suppose that the function $U_0(x)$ satisfies the inhomogeneous boundary conditions:

$$\Gamma_a(U_0) = A, \Gamma_b(U_0) = B, \tag{1.33}$$

and other functions $U_i, i = 1{:}n$ satisfy the corresponding homogeneous boundary conditions:

$$\Gamma_a(U_i) = 0, \Gamma_b(U_i) = 0, \quad i = 1, 2, ..., n. \tag{1.34}$$

In case the boundary conditions (1.31) are homogeneous (A = B = 0), then we can suppose $U_0(x) = 0$ and consider only the system of functions $U_i(x),\ i = 1, 2, \dots, n.$

We seek for an approximate solution to the problem (1.30), (1.31) as a linear combination of basis functions:

$$y \simeq \hat{y} = U_0(x) + \sum_{i=1}^{n} c_i U_i(x). \tag{1.35}$$

Then the function y satisfies the boundary conditions (1.31). In fact, because of the linearity of the boundary conditions we have:

$$\Gamma_a(y) = \Gamma_a(U_0) + \sum_{i=1}^{n} c_i \Gamma_a(U_i) = A + \sum_{i=1}^{n} c_i 0 = A,$$

and the same for the $\Gamma_b(y) = B$.

Compose residual function $R = L(y) - f(x)$. Substituting y for the expression (1.35), we obtain:

$$R(x, c_1, ..., c_n) \equiv L(y) - f(x) = L(U_0) - f(x) + \sum_{i=1}^{n} c_i L(U_i). \tag{1.36}$$

If for some choice of the c_i coefficients following equation is true

$$R(x, c_1, ..., c_n) \equiv 0 \quad for \quad a \le x \le b,$$

that means the function y is an exact solution to the problem (1.30), (1.31). However, it is generally not possible to successfully choose U_i function and the c_i coefficients. According to the previous sentence, it is required that the $R(x, c_1, \dots, c_n)$ function becomes zero in the given system of points $x_1, x_2, \dots, x_n$ in the interval [a, b], which are called collocation points. The function R is called residual of the equation (1.30). So the collocation method leads to a system of linear equations:

$$\left.\begin{aligned} R(x_1, c_1, \dots, c_n) &= 0 \\ \dots & \\ R(x_n, c_1, \dots, c_n) &= 0 \end{aligned}\right\}. \qquad \textbf{(1.37)}$$

In the case of compatibility, we can determine the coefficients from the system (1.37), then an approximate solution to the boundary value problem can be computed by using the (1.35).

In this chapter we have provided a detailed overview of the theoretical background which is necessary to know to understand how the problem is defined and what the collocation method is. The mathematical model of CR3BP and the equations of motion are thoroughly described. The following chapter introduces the methods which are used for the study and computations in our project.

This Page Intentionally Left Blank

Chapter 2

METHODOLOGY

The aim of this chapter is to describe analytical and numerical methods which are applied to our work. The semi - linear analysis of the neighborhood of a libration point is presented. The method of collocation can refine a periodic solution from an initial seed which is an initial guess of a period and an initial state vector (positions and velocities). Having the isolated solution with the exact period we can propagate one of the solution points for a longer time than the computed period in order to see how the system behaves.

2.1. Lindstedt – Poicaré method

In this section we deal with the Lindstedt – Poincaré procedure which is a semi – analytical method which can be used for obtaining an approximate representation of the libration point orbits and, what is much more important for our work, for a computation of an initial seed which is needed to initiate the collocation procedure. In order to perform the computations using the Lindstedt – Poincaré method, we need the equations of motion (1.24) which are expanded in power series. To describe exact orbits, the high-order expansions are usually used. But in our case, it is enough to obtain only an approximate representation and to collect initial seeds for the collocation procedure, so we use the expansions of order three. This method is well described in a large number of works and we refer to [3, 11]. According to the papers for applying the Lindstedt – Poincaré method we need to rewrite the equations of motion (1.24):

$$\left.\begin{aligned} \ddot{x} - 2\dot{y} - (1+2c_2)x &= \sum_{n\geq 2} c_{n+1}(n+1)T_n \\ \ddot{y} + 2\dot{x} + (c_2-1)y &= y\sum_{n\geq 2} c_{n+1}R_{n-1} \\ \ddot{z} + c_2 z &= z\sum_{n\geq 2} c_{n+1}R_{n-1} \end{aligned}\right\}, \qquad \textbf{(2.1)}$$

where

$$c_2(\mu) = \frac{1}{\gamma^3}\left((1)^2 \mu + (-1)^2 \frac{(1-\mu)\gamma^3}{(1 \mp \gamma)^3} \right). \tag{2.2}$$

with upper signs for L_1 and the lower sings for the L_2.

It is possible to compute T_0 by the given equation and use it as initial guess of the period in the collocation method

$$T_0 = \frac{2\pi}{\omega_0} \tag{2.3}$$

Talking about the analysis of the neighborhood of the libration points number 1 and 2, it is necessary to define two systems: one with origin in a libration point (*x, y, z*) and the other in CR3BP coordinates (*X, Y, Z*) which corresponds to the equations in chapter one. For changing coordinates, we use these equations

$$X = -\gamma x + \beta, \quad Y = -\gamma y, \quad Z = \gamma z \tag{2.4}$$

where

$$\beta = \mu - 1 \pm \gamma. \tag{2.5}$$

Note that this scaling have to be applied to the equations of motion (1.24).

The Lindstedt – Poincaré procedure can be used for both the analysis of the periodic solutions in the neighborhood of the libration points and the analysis of quasi – periodic orbits around Lagrangian points.

2.1.1. Lindstedt – Poincaré procedure for Lissajous orbits

Looking for 2-D – invariant tori as a series expansion in two frequencies is the idea of the Lindstedt – Poincaré method. This expansion formally satisfies the equation of motion. The procedure is recursive and depends on the previous results. In our work we use the already obtained coefficients up to order three. As a starting point we need the librating solutions to the linear part of (2.1)

$$\ddot{x} - 2\dot{y} - (1 + 2c_2)x = 0, \quad \ddot{y} + 2\dot{x} + (c_2 - 1)y = 0, \quad \ddot{z} + c_2 z = 0, \tag{2.6}$$

these equations can be rewritten as:

$$x(t) = \alpha\cos(\omega_0 t + \phi_1), \quad y(t) = k\alpha\cos(\omega_0 t + \phi_1), \quad z(t) = \beta\cos(\upsilon_0 t + \phi_2), \tag{2.7}$$

where

$$\omega_0 = \sqrt{\frac{2 - c_2 + \sqrt{9c_2^2 - 8c_2}}{2}}, \ \upsilon_0 = \sqrt{c_2} \quad and \quad k = \frac{-(\omega_0^2 + 1 + 2c_2)}{2\omega_0} \tag{2.8}$$

and from here the period of time can be calculated with (2.3).

The parameters α and β, which are in plane and out of plane amplitudes, respectively, and ϕ_1, ϕ_2, which are phases, give us all solutions to the equations.

Considering nonlinear terms of the equations, we try to find expansions in powers of the amplitudes α and β, which look like

$$x(t) = \sum_{i,j=1}^{\infty}\left(\sum_{|k|\le i,|m|\le j} x_{ijkm}\cos(k\theta_1 + m\theta_2)\right)\alpha^i\beta^j, \quad y(t) = \sum_{i,j=1}^{\infty}\left(\sum_{|k|\le i,|m|\le j} y_{ijkm}\sin(k\theta_1 + m\theta_2)\right)\alpha^i\beta^j,$$
$$z(t) = \sum_{i,j=1}^{\infty}\left(\sum_{|k|\le i,|m|\le j} z_{ijkm}\cos(k\theta_1 + m\theta_2)\right)\alpha^i\beta^j, \tag{2.9}$$

where $\theta_1 = \omega t + \phi_1$ and $\theta_2 = \upsilon t + \phi_2$ and *ijk* will be expanded to 3 in our wale $(i + j \le 3)$. Since we have nonlinear terms of the frequencies, ω and υ cannot be constants longer, so we have to expand them as well in powers of amplitudes:

$$\omega = \sum_{i,j=0}^{\infty}\omega_{ij}\alpha^i\beta^j, \quad \upsilon = \sum_{i,j=0}^{\infty}\upsilon_{ij}\alpha^i\beta^j. \tag{2.10}$$

As it has been mentioned above we use the expansions of order three in our work. All necessary coefficients are provided in the tables below. The coefficients were calculated in the work by Jobra and Masdemont [3]. The coefficients are true when the mass parameter μ = 0.3040357143000000E-05 and γ = 0.1001090475489518E-01 for the libration point number 1 and γ = 0.1007816698993660E-01 for L_2.

Table 2.1 Coefficients up to order 3 for Lissajous orbits around L_1

i	*j*			ω_{ij}	υ_{ij}
0	0			0.2086453455276053E+01	0.2015210551475634E+01
2	0			-0.1720615798629176E+01	0.2227433602201768E+00
0	2			0.2581845049892893E-01	-0.1631915163450506E+00
i	*j*	*k*	*m*	x_{ijkm} *or* y_{ijkm}	z_{ijkm}
1	0	1	0	0.1000000000000000E+01	-0.3229268096183804E+01
0	1	0	1	0.1000000000000000E+01	
2	0	0	0	0.2092695580695452E+01	-0.4778923164284034E+01
2	0	2	0	-0.9059647953966100E+00	-0.4924458751013138E+00
0	2	0	0	0.2482976702637036E+00	0.0000000000000000E+00
0	2	0	2	0.1108251912938997E+00	-0.6776374300069993E-01
1	1	1	-1	-0.1116868337925509E+01	
1	1	1	1	0.3549453066222734E+00	
3	0	1	0	0.0000000000000000E+00	0.2845081147294533E+01
3	0	3	0	-0.7938201951138280E+00	-0.8857007762159341E+00
1	2	1	-2	-0.1499994708837557E+01	-0.4841967094812451E+01
1	2	1	0	0.0000000000000000E+00	0.2875530871063365E+00
1	2	1	2	0.8387778032395528E-01	0.2082880779824395E-01
2	1	0	1	0.0000000000000000E+00	
2	1	2	-1	0.1216565483254493E+02	
2	1	2	1	0.4060792886575456E+00	
0	3	0	1	0.0000000000000000E+00	
0	3	0	3	-0.1952722079096668E-01	

Table 2.2 Coefficients up to order 3 for Lissajous orbits around L_2

i	j			ω_{ij}	υ_{ij}
0	0			0.2057014295696122E+01	0.1985074963711547E+01
2	0			-0.1531347696560955E+01	0.2943988259478343E+00
0	2			0.3525050019435665E-01	-0.1474823682728311E+00
i	j	k	m	x_{ijkm} or y_{ijkm}	z_{ijkm}
1	0	1	0	0.1000000000000000E+01	-0.3187229438133198E+01
0	1	0	1	0.1000000000000000E+01	
2	0	0	0	-0.2053038911972471E+01	0.4844772537508761E+01
2	0	2	0	0.8962835946742865E+00	0.4913574320833921E+00
0	2	0	0	-0.2516462771258266E+00	-0.0000000000000000E+00
0	2	0	2	-0.1132975889914482E+00	0.7016433879412928E-01
1	1	1	-1	0.1135799090644544E+01	
1	1	1	1	-0.3605240591354948E+00	
3	0	1	0	0.0000000000000000E+00	0.2720780779594278E+01
3	0	3	0	-0.7806464279381702E+00	-0.8553045853428568E+00
1	2	1	-2	-0.1451359308311418E+01	-0.4591171666445184E+01
1	2	1	0	0.0000000000000000E+00	0.2493806449729320E+00
1	2	1	2	0.8490080672754975E-01	0.1794119790942379E-01
2	1	0	1	0.0000000000000000E+00	
2	1	2	-1	0.1129777481199042E+02	

2	1	2	1	0.4021703922115226E+00	
0	3	0	1	0.0000000000000000E+00	
0	3	0	3	-0.1926879636938705E-01	

In the analysis of quasi – periodic Lissajous orbit around the libration points we have two input parameters. They are amplitudes. It is necessary to note, that by manipulating the amplitudes we can change the type of an orbit. Having α and β not equal to zero we obtain a Lissajous orbit, but if α or β is equal to zero, we have a vertical Lyapunov orbit or planar Lyapunov orbit, respectively. After integrating the expansions for some chosen time we obtain an approximate profile of an orbit and can take any point as an initial condition for the collocation procedure.

2.1.2. Lindstedt – Poincaré procedure for Halo orbits

It has been mentioned above that Halo orbits bifurcate from the planar Lyapunov orbits and the existence of the Halo orbits occurs when in-plane and out-of-plane frequencies are equal. Since we have a 1:1 resonance we can look for 1-D invariant tori as some series expansions with a single frequency. For applying the Lindstedt – Poincaré procedure, we need to modify the equations of motion (2.1) by adding the components Δ and z to the third equation,

$$\left.\begin{aligned} \ddot{x} - 2\dot{y} - (1+2c_2)x &= \sum_{n\geq 2} c_{n+1}(n+1)T_n \\ \ddot{y} + 2\dot{x} + (c_2 - 1)y &= y\sum_{n\geq 2} c_{n+1}R_{n-1} \\ \ddot{z} + c_2 z &= z\sum_{n\geq 2} c_{n+1}R_{n-1} + \Delta z \end{aligned}\right\}, \qquad \textbf{(2.11)}$$

where, referring to [3], we can say that the condition Δ must be equal to zero. In the Lindstedt – Poincaré method, we expand the factor Δ as a frequency series:

$$\Delta = \sum_{i,j=0}^{\infty} d_{ij}\alpha^i \beta^j \qquad \textbf{(2.12)}$$

Now we can write the liberating solution to the linear part of the (2.11):

$$\ddot{x} - 2\dot{y} - (1+2c_2)x = 0, \quad \ddot{y} + 2\dot{x} + (c_2 - 1)y = 0, \quad \ddot{z} + c_2 z = d_{00}z. \qquad \textbf{(2.13)}$$

These equations can be represented as:

$$x(t) = \alpha\cos(\omega_0 t + \phi), \quad y(t) = \kappa\alpha\cos(\omega_0 t + \phi), \quad z(t) = \beta\cos(\omega_0 t + \phi), \tag{2.14}$$

where $d_{00} = c_2 - \omega_0^2$ and

$$\omega_0 = \sqrt{2 - c_2 + \sqrt{9c_2^2 - 8c_2}\,/\,2}, \quad \kappa = \frac{-(\omega_0^2 + 1 + 2c_2)}{2\omega_0}, \tag{2.15}$$

and ϕ is an arbitrary phase.

As it has been shown in the previous sub – section, α and β are termed in-plane and out-of-plane amplitudes. It is obvious that the Halo orbits depend on only one frequency or one amplitude because they are 1-D invariant tori. So the relation between α and β must be established. The relationship is contained in the factor $\Delta = 0$. When we consider the nonlinear terms of (2.11), we look for formal expansions in powers of the amplitudes α and β of the type

$$x(t) = \sum_{i,j=1}^{\infty}\left(\sum_{|k|\le i+j} x_{ijk}\cos(k\theta)\right)\alpha^i\beta^j, \quad y(t) = \sum_{i,j=1}^{\infty}\left(\sum_{|k|\le i+j} y_{ijk}\sin(k\theta)\right)\alpha^i\beta^j,$$
$$z(t) = \sum_{i,j=1}^{\infty}\left(\sum_{|k|\le i+j} z_{ijk}\cos(k\theta)\right)\alpha^i\beta^j, \tag{2.16}$$

where $\theta = \omega t + \phi$ and the frequency ω and Δ must be extended as

$$\omega = \sum_{i,j=0}^{\infty}\omega_{ij}\alpha^i\beta^j, \quad \Delta = \sum_{i,j=0}^{\infty} d_{ij}\alpha^i\beta^j = 0. \tag{2.17}$$

We have mentioned that to get an approximate representation of an orbit and to obtain an initial condition we do expansion of the equations (2.16) up to order three. All coefficients have been computed many times so we refer to [3] and use the coefficients provided in the paper. According to the fact that we focus our project on the study of the neighborhood of the libration points L_1 and L_2, it is necessary to provide some constants which are used for the computations: the mass parameter μ = 0.3040357143000000E-05 and γ = 0.1001090475489518E-01 and γ = 0.1007816698993660E-01 for L_1 and L_2, respectively.

Table 2.3 Coefficients up to order 3 for Halo orbits around L_1

i	j		ω_{ij}	d_{ij}
0	0		0.2086453455276053E+01	-0.2922144542546494E+00
2	0		-0.1720615798629176E+01	0.1596559878224751E+02
0	2		0.2526666105275838E+00	-0.1740900545935829E+01
i	j	k	x_{ijkm} *or* y_{ijkm}	z_{ijkm}
1	0	1	0.1000000000000000E+01	-0.3229268096183804E+01
0	1	1	0.1000000000000000E+01	
2	0	0	0.2092695580695452E+01	-0.4778923164284034E+01
2	0	2	-0.9059647953966100E+00	-0.4924458751013138E+00
0	2	0	0.2482976702637036E+00	0.0000000000000000E+00
0	2	2	0.1044641164074320E+00	-0.6074646717326299E-01
1	1	0	-0.1040596381457606E+01	
1	1	2	0.3468654604858687E+00	
3	0	1	0.0000000000000000E+00	0.2845081147294533E+01
3	0	3	-0.7938201951138280E+00	-0.8857007762159342E+00
1	2	1	0.0000000000000000E+00	0.4316925766873952E+00
1	2	3	0.8268538529318073E-01	0.2301982737772552E-01
2	1	1	0.0000000000000000E+00	
2	1	3	0.3980954251673118E+00	
0	3	1	0.0000000000000000E+00	
0	3	3	-0.1904387005265064E-01	

Table 2.4 Coefficients up to order 3 for Halo orbits around L_2

i	j		ω_{ij}	d_{ij}
0	0		0.2057014295696122E+01	-0.2907852011438137E+00
2	0		-0.1531347696560955E+01	0.1482882483635615E+02
0	2		0.2572237584422004E+00	-0.1673691626171837E+01
i	j	k	x_{ijkm} *or* y_{ijkm}	z_{ijkm}
1	0	1	0.1000000000000000E+01	-0.3187229438133198E+01
0	1	1	0.1000000000000000E+01	
2	0	0	-0.2053038911972471E+01	0.4844772537508761E+01
2	0	2	0.8962835946742865E+00	0.4913574320833921E+00
0	2	0	-0.2516462771258266E+00	-0.0000000000000000E+00
0	2	2	-0.1065995583353226E+00	0.6271902291776088E-01
1	1	0	0.1056355182043019E+01	
1	1	2	-0.3521183940143398E+00	
3	0	1	0.0000000000000000E+00	0.2720780779594278E+01
3	0	3	-0.7806464279381702E+00	-0.8553045853428568E+00
1	2	1	0.0000000000000000E+00	0.3693512915944317E+00
1	2	3	0.8369602467327969E-01	0.2043544134245290E-01
2	1	1	0.0000000000000000E+00	
2	1	3	0.3940279801458219E+00	
0	3	1	0.0000000000000000E+00	
0	3	3	-0.1882903780587591E-01	

In the case of the analysis of the Halo orbits, there is only one input parameter β. In order to obtain a whole state vector, we need to take derivatives of expansions. Integrating the expansions and the derivatives for the chosen period of time we obtain a profile of a Halo orbit. Then taking one of the moments of time we obtain an initial seed which is needed to initiate the collocation procedure, but it is necessary to note that the expansions provide coordinates and velocities in coordinate system which is centered in the libration point, so before using a state vector as the initial conditions it is necessary to transform coordinates to CR3BP by using the equations (2.4) and (2.5). An initial guess of the period can be obtained by the (2.3). The way how the collocation method can be applied is presented in the following sections.

2.2. Collocation method for solving non-linear BVP

The description of the method of collocation is given in chapter one. Here we describe how the collocation method can be used for solving a non-linear boundary value problem (BVP). The behavior of non-linear dynamical systems can be described by solving a set of non-linear differential equations. A large number of works describe the method of collocation. As we have motioned before, one of the objectives is to study the method of collocation presented in the AUTO – 07p software package. In order to optimize, simplify a computational process and save resources we decided to avoid the integral equations which are widely used in the edition of the collocation method which is the base of the AUTO – 07p package algorithm. That is why we refer to the algorithm which is represented in the work by Mallon [19]. Our objectives are to apply the algorithm which is described below for the two problems: refining of the isolated solution and the period and refining the isolated solution with the exact period. The representation of how the method of collocation can be used for solving BVP is given as follows.

We consider a set of ordinary non-linear autonomous differential equations

$$x' = f(x) \tag{2.18}$$

Here *x* is the n (in our case 6 - dimensional) dimensional state vector and *f(x)* a set of n first order differential equations. Due to the non-linear nature of (2.18), it is not feasible to find a closed form solution *x(t)* by analytical methods. According to this we have to consider approximations $x_\pi(t)$ for the solution (2.18). Numerical methods can be applied to the problem's integration for computing these approximations and they lead to a discretization of the solution. A discretization is a representation of the solution by function values on so called mesh points which are mutually connected by interpolation functions. The method of collocation requires that the approximate solution satisfies the ordinary differential equations at certain preselected points which are termed the collocation points. Since we look for the periodic solutions with *x(0)* = *x(T)* our problem is a non-linear boundary value problem. The solution to this kind of problems can be computed by a Newton-type method known as quasi-linearization. Consider the BVP form of the problem (2.18)

$$x' = f(x), \qquad g(x(a), x(b)) = 0. \tag{2.19}$$

An approximate $X_\pi(t)$ solution to the problem (2.19) with the collocation method on a subinterval in a ≤ t ≤ b, denoted by $[t_i, t_{i+1}]$, satisfies the following interpolation conditions:

$$x_\pi(t_i) = x_i \quad and \quad x'(t_{ij}) = f(x_{ij}) \quad for \quad 1 \le i \le N, 1 \le j \le k, \tag{2.20}$$

with t_{ij} the j^{th} collocation point in subinterval *i* and $x_\pi(t)$ being a polynomial of order *k-1* inside $[t_i, t_{i+1}]$. Figure 2.1 shows the locations of the intervals and the collocation points.

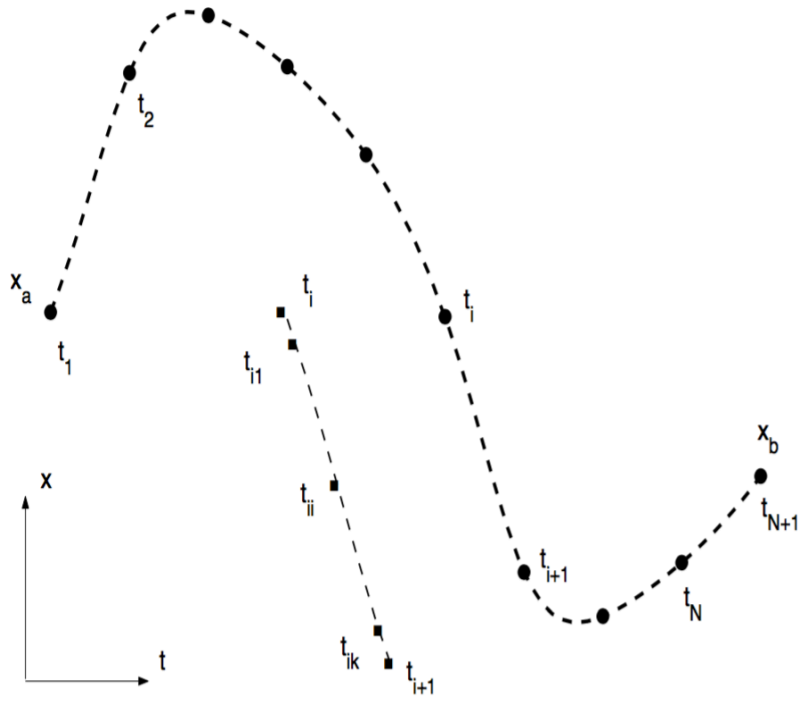

Figure 2.1 Intervals and collocation points t_{ij} inside $[t_i, t_{i+1}]$

The approximation $x_\pi(t)$ can be computed as a sum of its initial conditions and its first derivative as shown in (2.21)

$$x_\pi(t) = x_i + \int_{t_i}^{t} x'_\pi(\xi)\, d\xi. \tag{2.21}$$

The derivative *f(x)* equals the derivative $x'_\pi(t)$ at the *k* collocation point if the approximation satisfies the interpolation conditions (2.20). According to the given fact

we can define an interpolation polynomial of degree *k−1* for the state equations. The Lagrange form is a usually used way to define the interpolation polynomial:

$$p(t) = \sum_{l=1}^{k} f(x_{il}) L_l(p) \; with \; p = \frac{t - t_i}{h_i}, x_{il} = x_\pi(t_{il}) \; and \; p \in [0,1] \qquad \textbf{(2.22)}$$

and

$$L_l(p) = \prod_{\substack{j=1 \\ j \neq l}}^{k} \frac{p - p_j}{p_l - p_j} \; where \; p(l) = \frac{t_{il} - t_i}{h_i}, h_i = t_{i+1} - t_i . \qquad \textbf{(2.23)}$$

For the polynomials $L_l(p)$ time must be rescaled to the domain [0,1] for each interval *i*. Since the polynomials satisfy

$$L_i = \delta_{ij} \; with \; \delta_{ij} = \begin{cases} 1 & for \; j = i \\ 0 & for \; j \neq i \end{cases}, \qquad \textbf{(2.24)}$$

it is easy to see that the polynomial $p(t)$ satisfies $f(x_{ij}) = p(t_{ij})$. The state equations with the $p(t)$ cease to be a function of the state variables *x* so the integration in (2.21) can be done straightforward. The substitution of (2.22) for (2.21) results in

$$x_\pi(t) = x_i + \int_{t_i}^{t} \sum_{l=1}^{k} f(x_{il}) L_l(p) dp. \qquad \textbf{(2.25)}$$

Substituting the integral for a numerical quadrature method and the use of the scaled integration points as collocation points, we arrive at two equations for the local unknowns x_{ij} (2.26) and for the global unknowns x_{i+1} (2.27).

$$x_\pi(t_{ij}) = x_i + h_i \sum_{l=1}^{k} \alpha_{jl} f(x_{il}) = x_{ij} \; with \; \alpha_{jl} = \int_0^{p_l} L_l(p) dp \qquad \textbf{(2.26)}$$

$$x_\pi(t_{i+1}) = x_i + h_i \sum_{l=1}^{k} \beta_l f(x_{il}) = x_{i+1} \; with \; \beta_l = \int_0^{1} L_l(p) dp \qquad \textbf{(2.27)}$$

Extending the polynomial to the subinterval $[t_{i+1}, t_{i+2}]$ by using (2.27) with $i+1$ instead of *i*, results in the piecewise polynomial being continuously matched at t_{i+1}. Extending to

all $1 \le i \le N$, we obtain a continuous piecewise polynomial function of the order $k-1$ over $[a, b]$ which satisfies the vector field at the collocations points t_{ij}.
The equation (2.27) has the identical form as the *Runge-Kutta* scheme. The final scheme has an implicit character because the function is based on a numerical quadrature method and according to these facts the collocation scheme is exactly alike a one-step implicit *Runge-Kutta* scheme. We apply Simpson's quadrature which approximates integration over the interval [0, 1], and has the following set of parameters k = 3, ρ1 = 0, ρ2 = 0.5 and ρ3 = 1 to show this implicitness. The Lagrange interpolation functions for this parameters are defined by

$$L_1(\rho) = \frac{(\rho - 0.5)(\rho - 1)}{0.5}, \quad L_2(\rho) = \frac{(\rho - 0)(\rho - 1)}{-0.25}, \quad L_3(\rho) = \frac{(\rho - 0)(\rho - 0.5)}{0.5}.$$

Figure 2.2 shows three Lagrange polynomials which satisfy properties (2.24). Having these polynomials, we can compute α_{il} and β_l using the equations (2.26) and (2.27), respectively. The numerical values for α and β, together with the (scaled) locations of

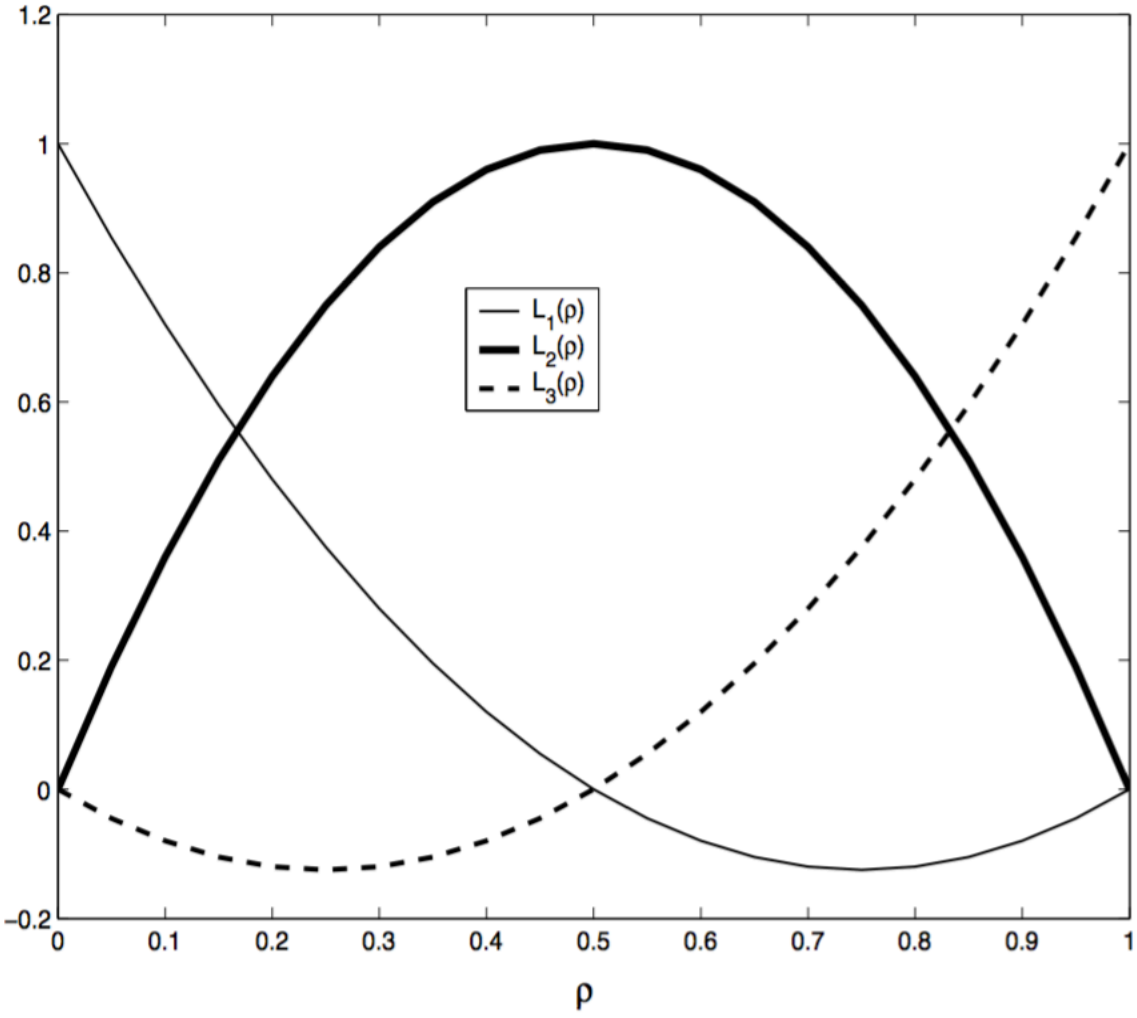

Figure 2.2 Lagrange polynomials for k = 3

the collocation points ρ_l are presented in a so called *Butcher-array*, which is for this scheme shown in Table 1.1.

Table 2.5 Butcher-array

p_l	α_{11}	…	α_{13}	
0	*0*	*0*	*0*	α_{13}
0.5	*5/24*	*1/3*	*-1/24*	…
1	*1/6*	*2/3*	*1/6*	α_{33}
β_l	*1/6*	*2/3*	*1/6*	

The problem that we can see in Table 2.5 appears if we want to solve the equation (2.26). For the computation of, for instance, $f(x_{i2})$ we first need $f(x_{i3})$ and vice versa. Solving this non-linear equation would require an expensive Newton method for each subinterval. According to this we arrive at the point showing how the equations (2.26) and (2.27) are solved. There are two methods which could be applied, firstly, the method of quasi-linearization (Newton's method for non-linear problems) for rewriting the solution to the non-linear BVP into a sequence of solutions to a linear BVP, and secondly, a parameter condensation procedure which can be applied to linear problems to eliminate the local unknowns.

As we know, if we expand the functions in Taylor series and apply a correction for the non-linear terms at each iteration, we obtain Newton's method for algebraic equations. In terms of the quasi-linearization method, it does the same for the non-linear boundary value problem (2.26). Now we rewrite the problem for the correction ω from the initial guess $x^m(t)$

$$(x^m(t)+\omega)' = f(x^m(t)) + \frac{\partial f}{dx}(x^m(t))\omega \tag{2.28}$$

$$\omega' = A(t)\omega + q(t) = f^*(t,\omega) \tag{2.29}$$

where

$$A(t) = \frac{\partial f}{\partial x}(x^m(t)) \quad and \quad q(t) = f(x^m(t)) - (x^m(t))'. \tag{2.30}$$

And the boundary conditions as

$$g(x(a),x(b)) = 0 \rightarrow B_a\omega_1 + B_b\omega_N = -g(x^m(a),x^m(b)) \tag{2.31}$$

with

$$B_a = \frac{\partial g(x(a),x(b))}{\partial x}(x^m(a)) \; and \; B_b = \frac{\partial g(x(a),x(b))}{\partial x}(x^m(b)). \qquad \textbf{(2.32)}$$

The method has identical form as the Newton one, so a residual q(t) is a form of difference between the ODE and the derivative of solution and a Jacobian operator acting on a correction ω. In the case of collocation method, the residual at each collocation point is equal to zero. The solution to the linearized BVP is the solution to the collocation equations for the new set of variables ω_i which are analogous to (2.26) and (2.27).

$$\omega(t_{ij}) = \omega_i + h_i \sum_{l=1}^{k} \alpha_{jl} f^*(t_{il}, \omega_{il}) = \omega_{ij} \qquad \textbf{(2.33)}$$

$$\omega(t_{i+1}) = \omega_i + h_i \sum_{l=1}^{k} \beta_l f^*(t_{il}, \omega_{il}) = \omega_{i+1} \qquad \textbf{(2.34)}$$

In the linear problem the coefficients depend on the previous iteration $x^m(t)$. The quasi-linearization procedure therefore defines a sequence of linear BVPs whose solutions will converge to the solution to the non-linear problem, given that the initial guess was close enough to it [20]. Thus, solving these equations, leads us to a method for solving non-linear BVP as well.

In order to simplify the computational process we present a method to eliminate the local unknowns in the equation (2.34), the derivatives $f^*(t_{il}, \omega_{il})$ where the $(*)$ denotes the derivatives for the set of variables ω_i. This method is known as a parameter condensation in finite element literature. The substitution of (2.34) for (2.29) yields

$$\begin{aligned} f_{ij}^* &= f^*(t_{ij}, \omega_i + h_i \sum_{l=1}^{k} \alpha_{jl} f^*(t_{il}, \omega_{il})) \\ &= A(t_{ij})\omega_i + A(t_{ij}) h_i \sum_{l=1}^{k} \alpha_{jl} f^*(t_{il}, \omega_{il})) + q(t_{ij}) \end{aligned} \qquad \textbf{(2.35)}$$

And a vector form for $1 \le j \le k$ as

$$\begin{bmatrix} f_{i1}^* \\ \vdots \\ f_{ik}^* \end{bmatrix} = \begin{bmatrix} A(t_{i1}) \\ \vdots \\ A(t_{i1}) \end{bmatrix} \omega_i + h_i \begin{pmatrix} a_{11}A(t_{i1}) & \dots & a_{1k}A(t_{i1}) \\ \vdots & \ddots & \vdots \\ a_{k1}A(t_{ik}) & \cdots & a_{kk}A(t_{ik}) \end{pmatrix} \begin{bmatrix} f_{i1}^* \\ \vdots \\ f_{ik}^* \end{bmatrix} + \begin{bmatrix} q(t_{i1}) \\ \vdots \\ q(t_{ik}) \end{bmatrix} \qquad \textbf{(2.36)}$$

Isolating $f^* = [f^*_{i1}, \dots, f^*_{ik}]$ from (2.36) results in an expression for the local unknowns, which is only dependent on the global unknowns ω_i and function values of the previous iteration $x^m(t)$

$$f_i^* = W_i^{-1} V_i \omega_i + W_i^{-1} q_i, \tag{2.37}$$

where $W_i \in R^{nk \times nk}$, $V_i \in R^{nk \times n}$, f_i^* and $q_i \in R^{nk}$ are defined by

$$W_i = I - h_i \begin{pmatrix} a_{11}A(t_{i1}) & \cdots & a_{1k}A(t_{i1}) \\ \vdots & \ddots & \vdots \\ a_{1k}A(t_{ik}) & \cdots & a_{kk}A(t_{ik}) \end{pmatrix}, \quad V_i = \begin{bmatrix} A(t_{i1}) \\ \vdots \\ A(t_{ik}) \end{bmatrix} \quad and \quad q_i = \begin{bmatrix} q(t_{i1}) \\ \vdots \\ q(t_{ik}) \end{bmatrix}. \tag{2.38}$$

It is easy to see that W_i inverted is enough for h_i small.
We have presented the strategy and described two procedures that we are going to use for solving the resulting set of equations. The final collocation equations (2.39) come from the substitution of (2.26) for (2.34)

$$\begin{aligned} \omega_{i+1} &= \omega_i + h_i \sum_{l=1}^{k} \alpha_{jl} f^*(t_{il}, \omega_{il}) \\ &= \omega_i + h_i [\beta_1 I \cdots \beta_k I\ ][W_i^{-1} V_i \omega_i + W_i^{-1} q_i] \\ &= \Gamma_i \omega_i + r_i \end{aligned} \tag{2.39}$$

where

$$D = [\beta_1 I \cdots \beta_k I\ ], \Gamma_i = I + h_i D W_i^{-1} V_i \ and \ r_i = h_i D W_i^{-1} q_i. \tag{2.40}$$

The equation (2.39) together with the boundary conditions (2.31) form a set of n(N+1) equations which need to be solved in each iteration until convergence or another stop criterion has occurred

$$\begin{bmatrix} -\Gamma_1 & I & 0 & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & -\Gamma_N & I \\ \mathrm{B}_a & 0 & 0 & \mathrm{B}_a \end{bmatrix} \begin{bmatrix} \omega_1 \\ \vdots \\ \vdots \\ \omega_{N+1} \end{bmatrix} = \begin{bmatrix} r_1 \\ \vdots \\ r_N \\ -g(x^m(a), x^m(b)) \end{bmatrix}. \tag{2.41}$$

The initial solution profile $x_\pi(t)$ from where (2.41) the values $x_1,...,x_{N+1}$ and $f_1,...,f_N$ come is needed for the startup of the procedure of collocation

$$f_i = \begin{bmatrix} f(x_{i1}) \\ \vdots \\ f(x_{ik}) \end{bmatrix}. \tag{2.42}$$

With the x_i and f_i the values for x_{il} can be obtained by the (2.26). Subsequently the values for $A(x_{il})$ and $f(x_{il})$ can be computed and the set of equations (2.40) can be formed. After solving the linear system, the approximate solution $x^m(t)$ and the derivatives at the collocation points f_i must be updated to finish an iteration according to (2.43) and (2.44)

$$x_\pi^{m+1}(t_i) = x_\pi^m(t_i) + \omega_i \tag{2.43}$$

$$f_i^{m+1} = f_i^m + W_i^{-1} V_i \omega_i + W_i^{-1} q_i \tag{2.44}$$

from which the new values of x_{il} can be determined.

Described above strategy illustrates how the collocation method is applied for solving BVP. The following sub-sections illustrate two ways of using the collocation method, the first one to refine both a periodic orbit and a period and the second one to refine a periodic orbit with the exact given period. Also the process of continuation in a family of orbits is described below. Theoretically the method of collocation provides high accuracy and requires a low capability of recourses (time and CPU). The increase in accuracy can be reached by choosing more accurate initial conditions. Anyway, the rate of convergence will not be constantly decreasing due to the fact that the energy is not fixed and there are many of the orbits with very similar parameters and states.

2.2.1. Computing periodic solution and period by collocation

One of the options to use the method of collocation is to refine a periodic orbit and a period of the orbit. Computing isolated periodic solutions of a set of autonomous non-linear ordinary differential equations requires to solve the two points boundary value problem, which is defined as follows

$$x' = f(x) \; with \; x(0) = x(T). \tag{2.45}$$

This BVP is similar to the problem as presented in (2.19) with the difference that the period T becomes an extra unknown parameter in the problem according to the fact that we look for a closed solution. Here the way how this problem can be transformed into an identical form as (2.19) to apply the method of collocation for the procedure is presented. For transformation we need to write the problem in a dimensionless form with the time scaled to *[0, 1]* domain

$$\frac{dx}{d\tau} = Tf(x) \; with \; x(0) = x(1), \tau = \frac{t}{T} \; and \; \tau \in [0,1]. \tag{2.46}$$

Next the set of n ODEs is rewritten for both small corrections *w* on the variables x_i and for a small perturbation *dT* on the period time *T* from an initial guess $[x^m(\tau), T^m]^T$ as described for the method of quasi-linearization.

$$\frac{dw}{d\tau} = A(\tau)w + f(x^m(\tau))dT + q(\tau) \tag{2.47}$$

with

$$A(\tau) = T^m \frac{\partial f}{\partial x}(x^m(\tau)) \; and \; q(\tau) = T^m f(x^m(\tau)) - (x^m(\tau))' \tag{2.48}$$

and the boundary conditions as

$$x(0) - x(1) = 0 \rightarrow w_1 - w_{N+1} = x_{N+1}^m - x_1^m . \tag{2.49}$$

Now we can rewrite the equation (2.37) for the elimination of the local unknowns with the (2.47)

$$f_i = W^{-1}Vw_i + W^{-1}UdT + W^{-1}q_i . \tag{2.50}$$

with f_i, W_i, V_i and q_i analogous to (2.38) and $U_i \in R^{nk}$ which is defined by

$$U_i = \begin{bmatrix} f(x^m(\tau_{i1})) \\ \vdots \\ f(x^m(\tau_{ik})) \end{bmatrix}. \tag{2.51}$$

Substituting (2.50) for (2.34) results in a new set of *n*×*N* collocation equations for computation of the periodic solution.

$$\begin{aligned} w_{i+1} &= w_i + h_i \sum_{l=1}^{k} \alpha_{jl} f(t_{il}, w_{il}) \\ &= w_i + h_i [\beta_1 I \cdots \beta_k I][W_i^{-1}V_i w_i + W_i^{-1}U_i dT + W_i^{-1}q_i] \\ &= \Gamma_i w_i + \Lambda_i dT + r_i . \end{aligned} \tag{2.52}$$

with D, Γ_i and r_i analogous to (2.40) and

$$\Lambda_i = h_i D W_i^{-1} U_i . \tag{2.53}$$

Since the (2.53) and the (2.49) are a system of *n(N + 1)* equations with *n(N + 1) + 1* unknowns, it cannot be solved uniquely because the phase of the periodic solution is not yet fixed. To overcome this problem a phase condition, also known as an anchor equation, is added to the system of equations to avoid the rotation of the periodic solution.

A phase condition can be represented as a fixed point at the periodic solution but this has a couple of drawbacks (we have to know that the initial point is really one of the orbit ones) and this is not applicable to our problem. A better anchor equation is given as the orthogonality conditions (2.54). Figure 2.3 illustrates geometrical meaning of the orthogonality conditions.

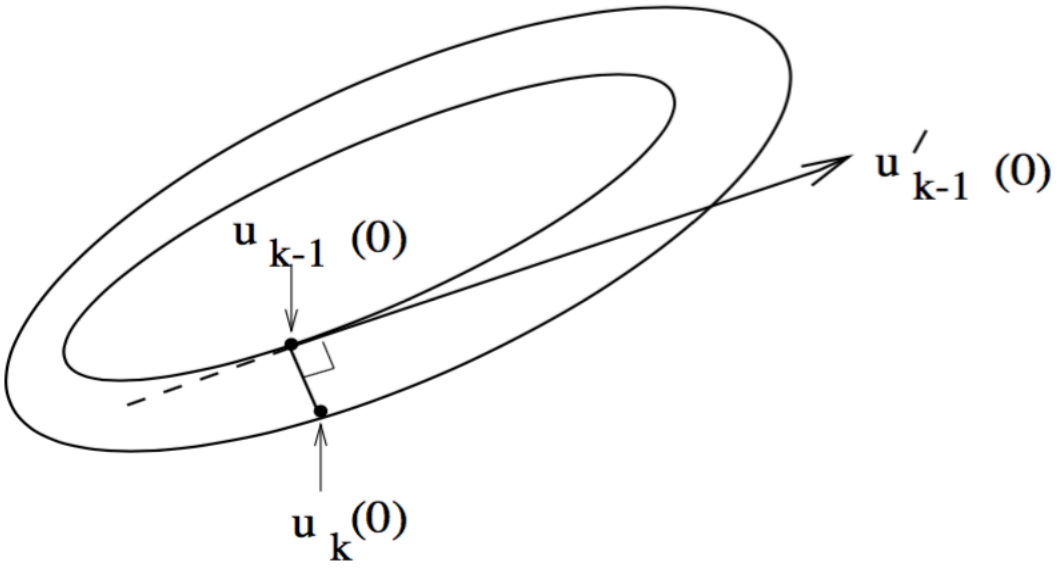

Figure 2.3 Illustration of orthogonally conditions to fix periodic orbit

$$f(x_1) w_1 = 0. \tag{2.54}$$

Using this popular anchor equation [23] gives us some benefits, firstly, it makes our solution fixed in space and secondly, it provides a good rate of convergence despite the problem which has been mentioned above. The linear set of equations which must be solved sequentially is analogous to (2.41) which follows from (2.49), (2.51) and (2.54) now becomes

$$\begin{bmatrix} -\Gamma_1 & I & 0 & 0 & -\Lambda_1 \\ 0 & \ddots & \ddots & 0 & \vdots \\ 0 & 0 & -\Gamma_N & I & -\Lambda_N \\ I & 0 & 0 & -I & 0 \\ f(x_1^m) & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ \vdots \\ w_{N+1} \\ dT \end{bmatrix} = \begin{bmatrix} r_1 \\ \vdots \\ r_N \\ x_{N+1}^m - x_1^m \\ 0 \end{bmatrix}. \quad \textbf{(2.55)}$$

After each iteration the approximate solution $x^m(t)$ must be updated as in (2.43) and f_i and the period time T as

$$f_i^{m+1} = f_i^m + W_i^{-1} V_i w_i + W_i^{-1} U_i dT + W_i^{-1} q_i \quad \textbf{(2.56)}$$

$$T^{m+1} = T^m + dT \quad \textbf{(2.57)}$$

The last step is to present a new vector field which is needed to obtain $A(\tau)$ in (2.54). Since x is the state vector which includes [x, y, z, v_x, v_y, v_z] and $f(x)$ is the set of differential equations in the (2.49) we obtain a matrix which is needed to be added to the vector field (2.58)

$$\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{\partial \Omega_x}{\partial x} & \frac{\partial \Omega_x}{\partial y} & \frac{\partial \Omega_x}{\partial z} & 0 & 2 & 0 \\ \frac{\partial \Omega_y}{\partial x} & \frac{\partial \Omega_y}{\partial y} & \frac{\partial \Omega_y}{\partial z} & -2 & 0 & 0 \\ \frac{\partial \Omega_z}{\partial x} & \frac{\partial \Omega_z}{\partial y} & \frac{\partial \Omega_z}{\partial z} & 0 & 0 & 0 \end{bmatrix} \quad \textbf{(2.58)}$$

where partials are computed with (1.23), (2.49) and (2.50)

$$
\begin{aligned}
\frac{\partial \Omega_x}{\partial x} &= 1+\frac{3(1-\mu)(X-\mu)^2}{p_1^5}+\frac{3\mu(X-\mu+1)^2}{p_2^5}-\frac{(1-\mu)}{p_1^3}-\frac{\mu}{p_2^3},\\
\frac{\partial \Omega_x}{\partial y} &= \frac{3Y(1-\mu)(X-\mu)}{p_1^5}+\frac{3\mu Y(X-\mu+1)}{p_2^5},\\
\frac{\partial \Omega_x}{\partial z} &= \frac{3Z(1-\mu)(X-\mu)}{p_1^5}+\frac{3\mu Z(X-\mu+1)}{p_2^5},\\
\\
\frac{\partial \Omega_y}{\partial x} &= \frac{3Y(1-\mu)(X-\mu)}{p_1^5}+\frac{3\mu Y(X-\mu+1)}{p_2^5},\\
\frac{\partial \Omega_y}{\partial y} &= 1+\frac{3Y^2(1-\mu)}{p_1^5}+\frac{3\mu Y^2}{p_2^5}+\frac{(1-\mu)}{p_1^3}-\frac{\mu}{p_2^3},\\
\frac{\partial \Omega_y}{\partial z} &= \frac{3YZ(1-\mu)}{p_1^5}+\frac{3\mu YZ}{p_2^5},\\
\\
\frac{\partial \Omega_z}{\partial x} &= \frac{3Z(1-\mu)(X-\mu)}{p_1^5}+\frac{3\mu Z(X-\mu+1)}{p_2^5},\\
\frac{\partial \Omega_z}{\partial y} &= \frac{3YZ(1-\mu)}{p_1^5}+\frac{3\mu YZ}{p_2^5},\\
\frac{\partial \Omega_z}{\partial z} &= \frac{3Z^2(1-\mu)}{p_1^5}+\frac{3\mu Z^2}{p_2^5}-\frac{(1-\mu)}{p_1^3}-\frac{\mu}{p_2^3}.
\end{aligned}
\qquad \textbf{(2.59)}
$$

Now the entire algorithm to compute a solution to our problem is described. Using this algorithm, a periodic solution to the mentioned problem can be obtained. To start up the procedure it is necessary to set up a set of initial conditions which includes an initial state vector, an initial guess of the period, a number of mesh points, a number of collocation points and a criterion of convergence. In the case when the initial seed is well computed the procedure takes from 3 to 7 fast iterations. The next sub-section introduces the same algorithm with the only difference that the following one does not have the period T as the extra unknown parameter.

2.2.2. Computing periodic solution with fixed period by collocation

The second way to use the method of collocation for the study of the natural motion in the neighborhood of the libration points is to refine an isolated periodic solution with the exact period of time. This kind of problem looks interesting due to the specifics of some space missions which require some specific orbital period. We consider the same problem as (2.45)

$$x' = f(x) \; with \; x(0) = x(T). \tag{2.60}$$

We ought to note that *T* is not an unknown, now it is some known number. Although the period is a known number, to set up the system and initiate the algorithm the scaling procedure must be done. We rescale the time to *[0, 1]* domain

$$\frac{dx}{d\tau} = Tf(x) \; with \; x(0) = x(1), \tau = \frac{t}{T} \; and \; \tau \in [0,1]. \tag{2.61}$$

In the case when we have the period known, the only corrections *w* on the variables x_i must be applied as

$$\frac{dw}{d\tau} = A(\tau)w + q(\tau), \tag{2.62}$$

with

$$A(\tau) = T\frac{\partial f}{\partial x}(x^m(\tau)) \quad and \quad q(\tau) = Tf(x^m(\tau)) - (x^m(\tau))' \tag{2.63}$$

and the same boundary conditions as the (2.49). Now we can rewrite the equation (2.37) for the elimination of the local unknowns with the (2.61) but without the part which corresponds to corrections for *dT*.

$$f_i = W^{-1}Vw_i + W^{-1}q_i, \tag{2.64}$$

where f_i, W_i, V_i and q_i are analogous to (2.38).

Substituting (2.64) for (2.34) yields a new set of *n*×*N* collocation equations for the computation of the periodic solution with fixed period.

$$
\begin{aligned}
w_{i+1} &= w_i + h_i \sum_{l=1}^{k} \alpha_{jl}\ f(t_{il}, w_{il}) \\
&= w_i + h_i [\beta_1 I \cdots \beta_k I][W_i^{-1} V_i w_i + W_i^{-1} q_i] \\
&= \Gamma_i w_i + r_i .
\end{aligned}
\tag{2.65}
$$

where D, Γ_i and r_i are analogous to (2.40).

Due to the fact that the system is still over defined, the phase conditions must be used. The phase conditions to fix the solution in a phase-space are the same and are given by the equation (2.54). The linear set of equations now looks like

$$
\begin{bmatrix}
-\Gamma_1 & I & 0 & 0 \\
0 & \ddots & \ddots & 0 \\
0 & 0 & -\Gamma_N & I \\
I & 0 & 0 & -I \\
f(x_1^m) & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
w_1 \\ \vdots \\ \vdots \\ w_{N+1}
\end{bmatrix}
=
\begin{bmatrix}
r_1 \\ \vdots \\ r_N \\ x_{N+1}^m - x_1^m
\end{bmatrix}.
\tag{2.66}
$$

Now we have the system of the collocation equations which need to be solved each iteration. And after each iteration the approximate solution $x^m(t)$ and f_i must be updated the same way as in (2.43) and (2.44) as follows:

$$
x_\pi^{m+1}(t_i) = x_\pi^m(t_i) + \omega_i \tag{2.67}
$$

$$
f_i^{m+1} = f_i^m + W_i^{-1} V_i \omega_i + W_i^{-1} q_i \tag{2.68}
$$

It is easy to see that all changes are done only in terms of algorithm, so there is no need to apply any modifications to the vector field. We have 6 first order differential equations for our state vector and the matrix (2.58) with the equations which are described by (2.59). Now the whole strategy of the computation of a periodic orbit with a fixed period is presented. The accuracy of this method critically depends on the accuracy of the given period. The next sub-section introduces us the method of continuation approach which is interesting in terms of following the family of a chosen orbit.

2.2.3. Continuation approach

Talking about a continuation procedure we mean that we follow a family/branch of some orbits which is located in the neighborhood of the collinear Lagragian points. It is well known that several kinds of periodic orbits exsist around the collinear libration points. These kinds of orbits form families or branches which contain a large number of orbits in each family. It is good to be informed about the neighborhood of the libration points for planning space missions at the Lagragian points. The continuation procedure can be represented by different methods (such as arc-length continuation). In our work we use the “step” method with a possibility of choosing parameters: a step and a number of orbits. Technically the continuation procedure is as follows:

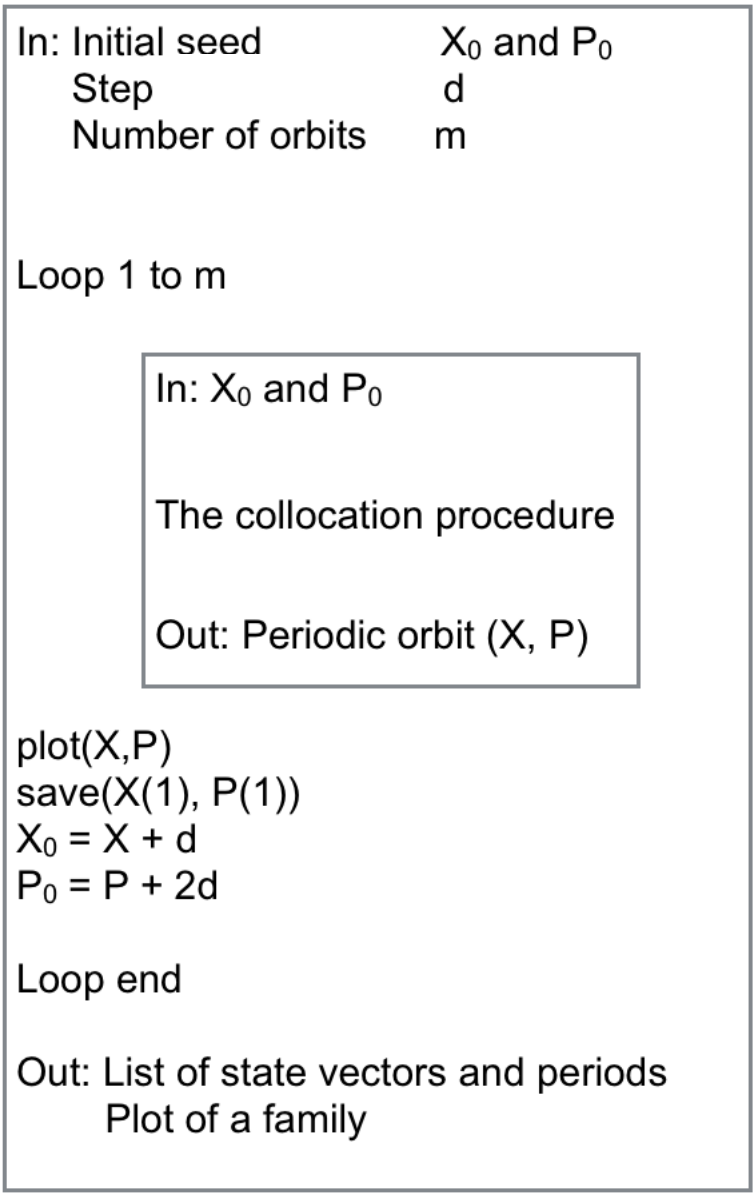

Figure 2.4 The basic concept of the continuation procedure

Figure 2.4 illustrates the basic concept of the procedure. We choose an initial seed for the collocation procedure and set up the step and the number of orbits. The step can be both: positive and negative. The collocation procedure is repeated for *m* iteration and each next iteration we add the step to the initial state vector and double it for the initial period of the orbit. The initial conditions for the iteration are the state vector and the period which are obtained in the previous iteration. As a result of the procedure we have a plot of a family of orbits and a list with the state vector and the period of each refined orbit. It is necessary to note that it is possible to obtain orbits from another family. It happens because the method which is used does not have the adaptive algorithm for the step, so after some iteration we can arrive at a bifurcation point and another family can bifurcate from ours. The continuation approach has the same features as the collocation procedure due to its basement. The accuracy of the method depends on the accuracy of taken initial seed and computational time is equal to the time of the collocation procedure multiplied by *m*. Having the list of state vectors and periods of the orbits of a family it is possible to study each orbit separately by using the collocation procedure or one of numerical integrators. For our needs it is enough to use fast low-order methods. The numerical integrator that we use for the project is described below.

2.3. Numerical integrator

A numerical integrator which is also called a propagator integrates a differential equation or a set of differential equations depending on a model for selected interval of time with a step. There exists a well-known *Runge-Kutta* scheme with different accuracy orders to propagate. The *Runge-Kutta* algorithm lets us solve a differential equation numerically (that is, approximately).

$$\dot{x} = f(t, x), \quad x(t_0) = x_0. \tag{2.69}$$

The *Runge-Kutta* scheme can have high accuracy and fast propagation. It is known to be very accurate and well-behaved for a wide range of problems. According to the fact that our work is done in the MATLAB software package, we apply ODE45 function as a propagator which is based on *Runge-Kutta* scheme.

$$[T, X] = ode45(@RTBP, [t_0\ t_{end}], x_o, options, \mu) \tag{2.70}$$

where x_0 - an initial state vector, $[t_0\ t_{end}]$ - time interval selected for the integration, *options* - set of preferences for the procedure, μ - mass parameter needed for a model.

The CR3BP is the model which we use for the propagation and we have thoroughly described in chapter one. The model includes six first-order ordinary differential equations obtained from the (1.24). The model is a vector field which describes phase

space around the libration points in the CR3BP. The structure of the vector field is provided as follows

$$\dot{X}(t) = f(X(t)). \tag{2.71}$$

More detailed form is

$$\begin{aligned} \dot{X} &= V_x, \\ \dot{Y} &= V_y, \\ \dot{Z} &= V_z, \\ \dot{V}_x &= 2V_y + \Omega_x, \\ \dot{V}_y &= -2V_x + \Omega_y, \\ \dot{V}_z &= \Omega_z. \end{aligned} \tag{2.72}$$

where Ω_x, Ω_y, and Ω_z are the partial derivative of (1.23) and can be determined by (2.73)

$$\begin{aligned} \Omega_X &= X - \frac{(1-\mu)(X-\mu)}{p_1^3} - \frac{\mu(X+1-\mu)}{p_2^3}, \\ \Omega_Y &= Y - \frac{(1-\mu)Y}{p_1^3} - \frac{\mu Y}{p_2^3}, \\ \Omega_Z &= -\frac{(1-\mu)Z}{p_1^3} - \frac{\mu Z}{p_2^3}. \end{aligned} \tag{2.73}$$

Here p_1 and p_2 are provided by

$$\begin{aligned} p_1 &= \sqrt{(X-\mu)^2 + Y^2 + Z^2}, \\ p_2 &= \sqrt{(X+1-\mu)^2 + Y^2 + Z^2}. \end{aligned} \tag{2.74}$$

The use of this numerical propagator gives us opportunities to approximately predict the time of refined orbit being stable and to see how the orbit behaves in the course of time longer than a period. It can be used for planning maneuvers and correction for missions. Also this method can be used as the test tool for the collocation procedure and for the Lindstedt-Poincaré expansions. Now we have presented all methods which we apply to our work to calculate initial seeds, to refine periodic solutions and to propagate the trajectory in the vector field. The following section describes a method which is used to study stable and unstable manifold, heteroclinic and homoclinic trajectories.

2.4. Stable and unstable manifolds

The study of stable and unstable manifolds is critical in terms of low energy transfer trajectories. As it has been mentioned above stable and unstable manifolds are sets of possible trajectories with very interesting applications. The unstable manifold is the set of all possible trajectories, which asymptotically depart from a nominal orbit, they can be approximated by a negligible mass in case a small adjustment directed by an unstable eigenvector is applied to the mass. The stable manifold includes the set of all possible trajectories that a particle could take to asymptotically arrive onto the nominal orbit along the orbit's stable eigenvector. The manifolds can be easily computed by using the following algorithm. Firstly, we need to compute a monodromy matrix which is a State Transition Matrix after an orbit become closed, after one period. The State Transition Matrix (STM), so-called vector flow, illustrates changes in the vector field (2.72) over the time and the concept and the computational process of it must be explained. So the state of a dynamical system can be represented by STM. Let us term $\phi_t(x_0)$ as a parameter of the state of chosen system at some moment of time t. It is necessary to note that at the time t = 0 the initial point is x_0. This can be stated as,

$$\phi_0(x_0) = x_0 . \tag{2.75}$$

The following figure illustrates the concept of STM.

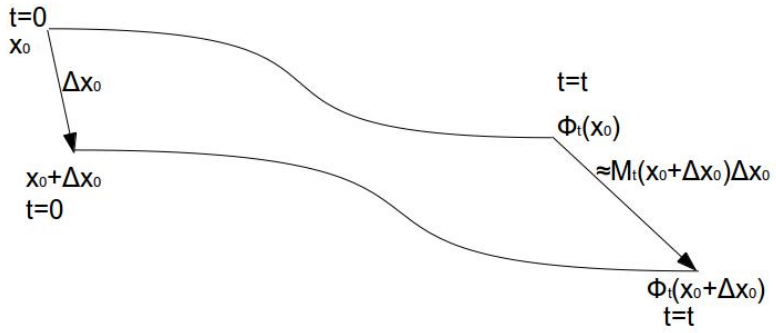

Figure 2.5 State Transition Matrix concept

We can add a small deviation Δx_0 to the x_0 and then the state at the moment of time t is $\phi_t(x_0 + \Delta x_0)$. And this can be expanded as

$$\phi_t(x_0 + \Delta x_0) = \phi_t(x_0) + D_x\phi_t(x_0)\Delta x_0 + \cdots \quad \textbf{(2.76)}$$

By cutting the high order terms, we can approximate (2.76) as

$$\phi_t(x_0 + \Delta x_0) - \phi_t(x_0) \approx D_x\phi_t(x_0)\Delta x_0 \quad \textbf{(2.77)}$$

The matrix *M* is termed the State Transition Matrix and is given by

$$M_t(x_0) = D_x\phi_t(x_0) \quad \textbf{(2.78)}$$

The concept of the equation (2.77) is represented in Figure 2.5. It is easy to see that

$$M_0(x_0) = I, \quad \textbf{(2.79)}$$

where *I* represents the identity matrix of order six. In our case we are more interested in the differential of the STM. It is well known that our dynamical system is represented by

$$\dot{X} = f(X) \quad \textbf{(2.80)}$$

It is clear that the state of the dynamical system $\phi_t(x)$ will be a solution to (2.80). Thus,

$$\frac{d(\phi_t(x))}{dt} = f(\phi_t(x)). \quad \textbf{(2.81)}$$

Taking the differential of (2.80) with respect to *x* we obtain

$$D_x\left(\frac{d(\phi_t(x))}{dt}\right) = D_x(f(\phi_t(x)))D_x(\phi_t(x)), \quad \textbf{(2.82)}$$

but we need to rewrite this equation to another form.

$$\frac{d}{dt}(D_x\phi_t(x)) = D_x(f(\phi_t(x)))D_x(\phi_t(x)), \tag{2.83}$$

where $D_x(\phi_t(x))$ is the State Transition Matrix, which is the same that is given by (2.78). Thus the equation (2.83) can be rewritten as

$$\dot{M} = D_x(f(\phi_t(x)))M\,. \tag{2.84}$$

The matrix D is called Variational matrix and it is represented by

$$D = D_x(f(\phi_t(x)))\,. \tag{2.85}$$

Due to the fact that our dynamical system is autonomous, the state of the system is given by vector X and its derivative with respect to time:

$$\begin{aligned} X &= (x, y, z, \dot{x}, \dot{y}, \dot{z})^T \\ \dot{X} &= (\dot{x}, \dot{y}, \dot{z}, \ddot{x}, \ddot{y}, \ddot{z})^T\,. \end{aligned} \tag{2.86}$$

For our model which is CR3BP the Variational matrix is given by the equation (2.87)

$$D = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{\partial \ddot{x}}{\partial x} & \frac{\partial \ddot{x}}{\partial y} & \frac{\partial \ddot{x}}{\partial z} & \frac{\partial \ddot{x}}{\partial \dot{x}} & \frac{\partial \ddot{x}}{\partial \dot{y}} & \frac{\partial \ddot{x}}{\partial \dot{z}} \\ \frac{\partial \ddot{y}}{\partial x} & \frac{\partial \ddot{y}}{\partial y} & \frac{\partial \ddot{y}}{\partial z} & \frac{\partial \ddot{y}}{\partial \dot{x}} & \frac{\partial \ddot{y}}{\partial \dot{y}} & \frac{\partial \ddot{y}}{\partial \dot{z}} \\ \frac{\partial \ddot{z}}{\partial x} & \frac{\partial \ddot{z}}{\partial y} & \frac{\partial \ddot{z}}{\partial z} & \frac{\partial \ddot{z}}{\partial \dot{x}} & \frac{\partial \ddot{z}}{\partial \dot{y}} & \frac{\partial \ddot{z}}{\partial \dot{z}} \end{pmatrix} \tag{2.87}$$

Having the method that describes computational process of the State Transition Matrix we can obtain an important for us Monodromy matrix (6 x 6) which is represented as

$$D = D(t), \quad t = T. \tag{2.88}$$

The next step is to compute eigenvector and eigenvalues of the monodromy matrix.

$$D_T \rightarrow \begin{matrix} \upsilon^s(\lambda^s) \\ \upsilon^u(\lambda^u) \end{matrix} \tag{2.89}$$

Since we have stable and unstable manifolds we need two eigenvectors which correspond to stable and unstable states. It is necessary to note that the eidenvectors have size (6 x 1). It is possible to choose the number of trajectories which to be computed. For this we need to take the State Transition Matrix which corresponds to the point from where we depart or to where we approach. For choosing the points we use time as it is illustrated bellow.

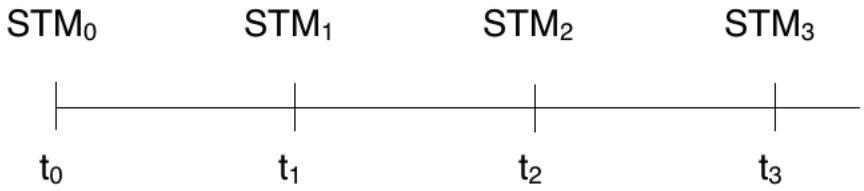

Figure 2.6 Concept of choosing points for manifold local linear approximation

As we have mentioned above the directed adjustment is required to abandon an orbit. We form new initial conditions and they must be directed. To set a needed direction we need, firstly, to normalize STM at chosen points:

$$\begin{aligned} D(t_i) &= STM_i \\ w_i^{s/u} &= STM_i \upsilon^{s/u}. \end{aligned} \tag{2.90}$$

The second step is the computation of direction vectors which are needed to form new initial conditions. It is interesting to note that the normalization factor is computed only with coordinates whereas velocities are skipped.

$$\upsilon_i^{s/u} = \frac{w_i^{s/u}}{L_i}, \tag{2.91}$$

where

$$L_i = (w_{i1}^{s/u})^2 + (w_{i2}^{s/u})^2 + (w_{i3}^{s/u})^2. \tag{2.92}$$

Now we need to choose the last parameter which is needed to form new initial conditions. The ε means a distance which is added to the state vector at a chosen point.
We decided to use 200 kilometers. Due to the fact that our system is dimensionless, we divide 200 by 1 AU. The new initial conditions are given as follows.

$$W_i^{s/u} = x_i \pm \varepsilon \upsilon_i^{s/u}. \tag{2.93}$$

The following figure illustrates the sense of the new initial conditions that have been computed on the local linear approximation of the manifold.

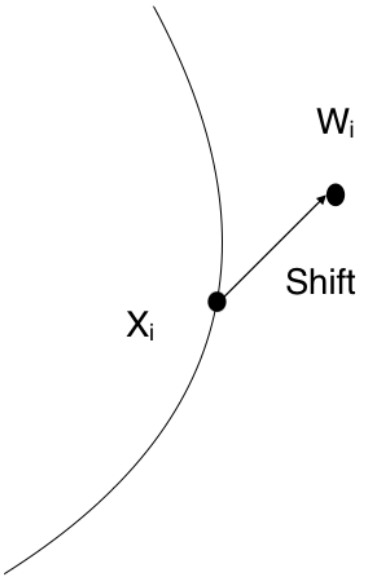

Figure 2.7 New initial conditions

Having initial conditions for each point, we can use the technique which we have described above and which is named numerical propagation. Due to the fact that we start from very different positions it is not possible to choose one propagation time for all trajectories, so we set up the period of time that 2-3 times longer than we usually need to reach the Earth. Also the Poincaré map must be established. We create a function which cuts the phase space in chosen plane and stops integration when the Poincaré map's section is crossed first or second time (depending on objective) by trajectory. The interesting point is that in the case of stable manifold it is necessary to propagate backward in time due to the fact that stable manifold's trajectories approach an orbit, so formally the new initial conditions are the last points of the propagated trajectories.

Now the strategy of the computation of the stable and unstable manifolds has been thoroughly described. In our case the most interesting states are trajectories that form heteroclinic or homoclinic orbits. The following sub-section describes the methodology for obtaining homoclinic and heteroclinic types of trajectories.

2.4.1. Homoclinic and Heteroclinic trajectories

The point of homoclinic and heteroclinic trajectories is to provide free maneuver nearby the closest primary body or free transfer from one libration point orbit which is located at libration point 1, for example, to another libration point orbit located at another point. The secret of searching these trajectories lies in the sense of Poincaré's map. Due to the fact that we placed cross section of the map in the Earth's position at x-axis, we obtain one of six values which is needed to build up a state-vector when one of trajectories crosses the plane (first of second time). The simplest and interesting application of homoclinic and heteroclinic appears when we study manifolds of planar libration point orbits. In this case we can obtain free transfer from one orbit to another. The algorithm of the calculation of these trajectories is as follows. Suppose that we have to planar orbits with the same energy which are located at libration point 1 and LIbration point 1, respectively. The first step is to compute unstable manifold which corresponds to the orbit around libration point 2 and stable manifold corresponded to the orbit located in the neighborhood of libration point 1. From here we obtain x – coordinate for our state vector.

$$\begin{aligned} & x_E = x, \\ & z, \dot{z} = 0, \\ & X = (x_E, y, 0, \dot{x}, \dot{y}, 0)^T . \end{aligned} \qquad \textbf{(2.94)}$$

According to (2.94) we see that we need to find three parameters in the case of heteroclinic and homoclinic orbits, it is easy to see that $z, \dot{z} = 0$ since orbits are planar

and x is equal to the Earth position. The figure below illustrates how y – coordinate and y – component of speed can be obtained.

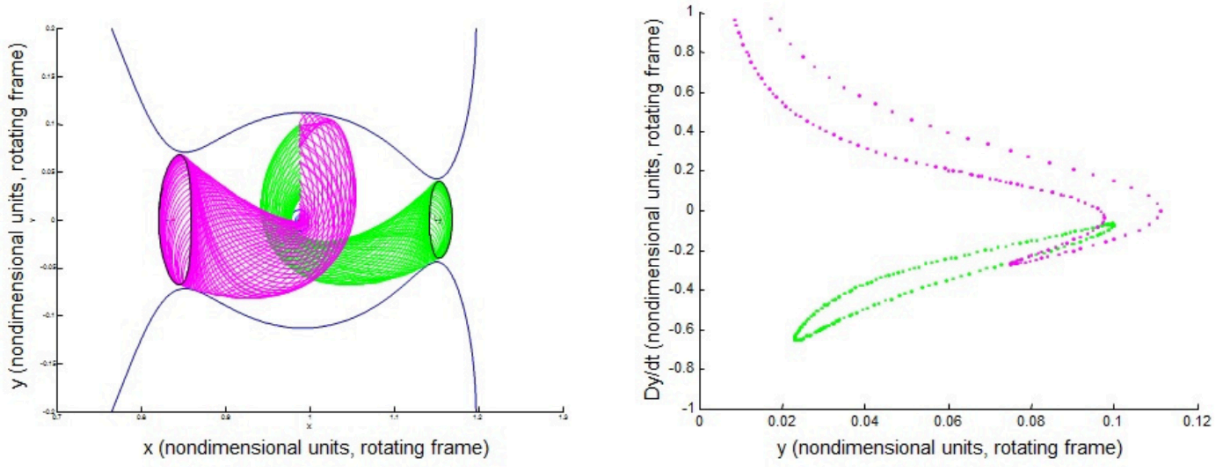

Figure 2.8 Cross section

It is easy to see that there are some intersection points in the second picture. These points mean that the y – coordinate and y – component of speed are the same for both manifolds stable and unstable. So all intersection points show the existence of the connecting orbits. After analyzing the obtained data, we have state vector with only one unknown. This unknown $\dot{x}$ can be computed by means of Jacobi constant. Due to the fact that we have natural motion and do not apply any maneuver we can say that Jacobian constant is the same for the connecting trajectories and orbits which correspond to the trajectories. Using the equation (1.26), it is possible to compute the *x* – component of speed. Having all components of the state vector, we can propagate forward and backwards in time with the initial conditions which are the state vector. The same method can be used for the computation of the homoclinic trajectories with the only difference that stable and unstable manifold for one orbit is computed. We have considered the case when libration points orbits have the same energy. According to the fact that the energy of connecting orbits is the same energy that orbits have, we can successfully suppose that connecting orbits are free transfer trajectories. It is not possible to fulfil a free transfer from one orbit to another one if they have different energy levels. Although, the usage of manifolds can be very efficient even in this situation. In this case we compute the manifolds of two orbits, which now could be not only planar orbits, and use the Poincaré map to stop the integration. The *x* – position is equal to the Earth coordinate. Now we plot *y* – *z* components instead of $y/\dot{y}$ and search for the intersection points. In these points we have state vectors as follows

$$X_1 = \begin{pmatrix} x_1 \\ y_1 \\ z_1 \\ \dot{x}_1 \\ \dot{y}_1 \\ \dot{z}_1 \end{pmatrix}, \quad X_2 = \begin{pmatrix} x_2 \\ y_2 \\ z_2 \\ \dot{x}_2 \\ \dot{y}_2 \\ \dot{z}_2 \end{pmatrix}, \tag{2.95}$$

where

$$\begin{matrix} x_1 & & x_2 \\ y_1 & = & y_2 \\ z_1 & & z_2 \end{matrix} . \tag{2.96}$$

It is easy to see that it is possible to fulfil a maneuver when a spacecraft is located at these interaction points. A maneuver is given by

$$\begin{pmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{z}_1 \end{pmatrix} - \begin{pmatrix} \dot{x}_2 \\ \dot{y}_2 \\ \dot{z}_2 \end{pmatrix} = \begin{pmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{z}_m \end{pmatrix}. \tag{2.97}$$

The statement (2.97) can be represented as

$$\Delta V = \begin{pmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{z}_m \end{pmatrix}. \tag{2.98}$$

Now we have presented all methodology which we use to study the neighborhood of libration points orbits and their manifolds in the Sun – Earth system. The next chapter presents numerical results and their graphical interpretation which are computed in our work.

Chapter 3

NUMERICAL RESULTS

In this chapter we present numerical results of our computations and its graphical interpretation. The chapter includes results of several different computations in a specific order: The first part is the Lindstedt – Poincaré procedure which provides an initial seed for the collocation method and refined orbits by collocation procedure which are computed using the initial seed; The second one is the evolution of orbits over the mass parameter change and some variations in initial guess of the period; Then continuation approach procedure's results are represented; And the fourth part is the stable/unstable manifolds and connecting orbits. As we have mentioned above we work in the Sun-Earth/Moon system with the mass parameter μ = $3.040423398444176 \times 10^{-6}$. It is necessary to remind that all calculations are done in the dimensionless reference system. It has to be noted that all values in the following tables are cut to provide better looking.

3.1. Lindstedt – Poincaré procedure and Collocation Method

We have mentioned many times in our work that the Lindstedt – Poincaré procedure can be used for the computation of initial conditions which are needed to initiate the collocation procedure and refine an isolated solution. In this sub-section the results of the Lindstedt – Poincaré procedure and the method of collocation are presented. The difference between semi-analytical method and numerical is shown. We represent three types of orbits: Vertical Lyapunov orbit, Planar Lyapunov orbit and Halo orbit. The following tables, which are provided in this sub-section, contain the initial seeds which are obtained by the Lindstedt – Poincaré procedure and state vectors of the refined orbit by the collocation collocation. We suggest starting with Vertical Lyapunov orbit. All computations are done for both collinear points L1 and L2 but due to the fact that in the Sun – Earth/Moon system the orbits around the opposite points are very similar, we present orbits around L1.

Table 3.1 State vector of initial conditions (Vertical Lyapunov)

Method	X	Y	Z	V_x	V_y	V_z	Period
Lindstedt-Poincaré	-9.9002192e-01	2.8423650e-05	-2.0436054e-03	1.8682814e-04	8.7872059e-05	1.96421650e-03	3.12040725
Collocation	-9.9014980e-01	2.7851450e-05	-2.0436054e-03	1.8094694e-04	8.7323887e-05	1.9911658e-03	3.1311087

The next two figures 3.1 and 3.2 show the difference in the orbits. It is easy to see that the computed numerically orbit (red) which is refined by collocation method in the CR3BP model is bigger that the orbit which is computed semi – analytically. This happens because the blue orbit is an approximate solution and does not exist in the CR3BP model, whereas the red one is numerically refined using the vector field of the model.

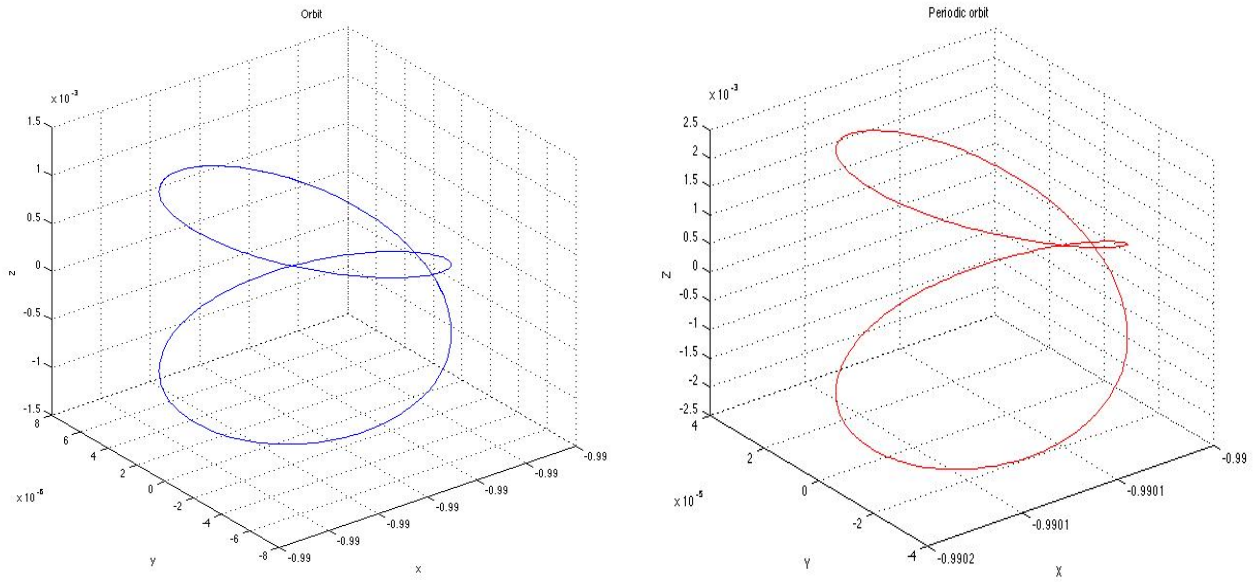

Figure 3.1 Vertical Lyapunov orbits obtained semi-analytically (blue) and numerically (red)

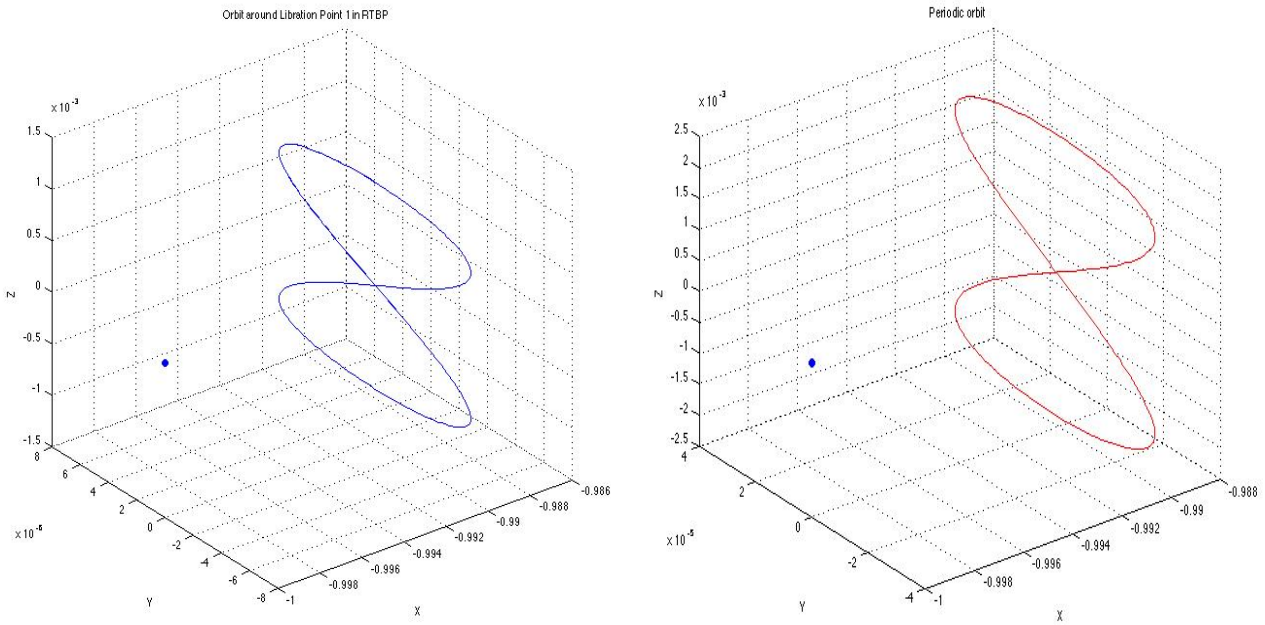

Figure 3.2 Vertical Lyapunov orbits' positions related to the Earth

The second represented type is Planar Lyapunov orbits. This family contains 2D orbits and there are free transfers from one planar orbit to another with the same energy level. Table 3.2 illustrates the state vectors of Planar Lyapunov orbits which are computed by the Lindstedt – Poincaré semi - analytical procedure and the method of collocation which is a numerical method using vector field. The orbits are shown in the figures 3.3 and 3.4.

Table 3.2 State vector of initial conditions (Planar Lyapunov)

Method	X	Y	Z	V_x	V_y	V_z	Period
Lindstedt-Poincaré	-9.9237570e-01	1.96066966e-04	0	-1.8653083e-05	1.39724833e-02	0	3.1153750615600
Collocation	-9.9233197e-01	1.7482299e-04	0	-1.3630561e-05	1.37406317e-02	0	3.1118363929997

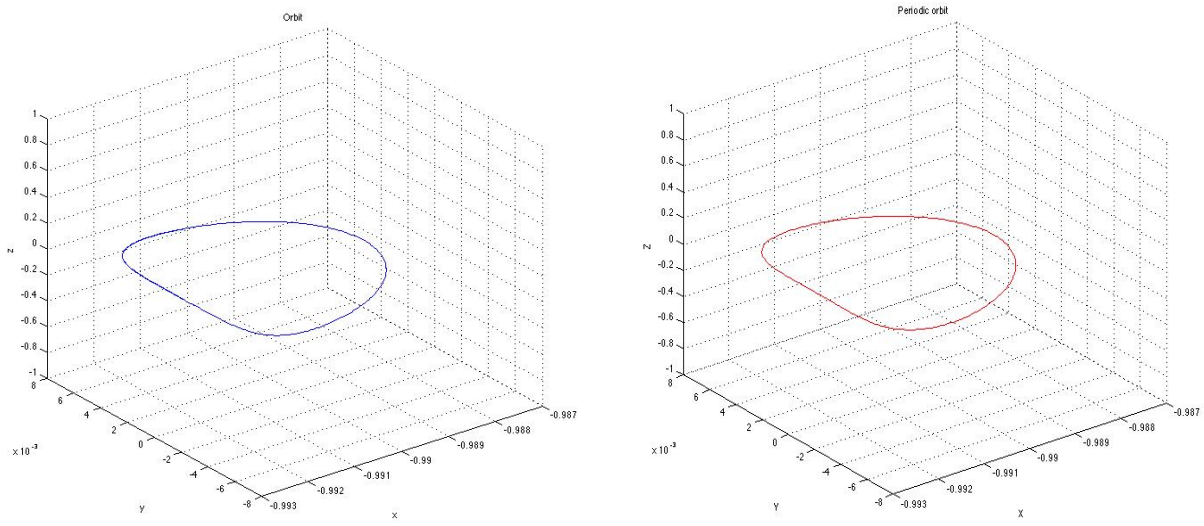

Figure 3.3 Planar Lyapunov orbits obtained semi-analytically (blue) and numerically (red)

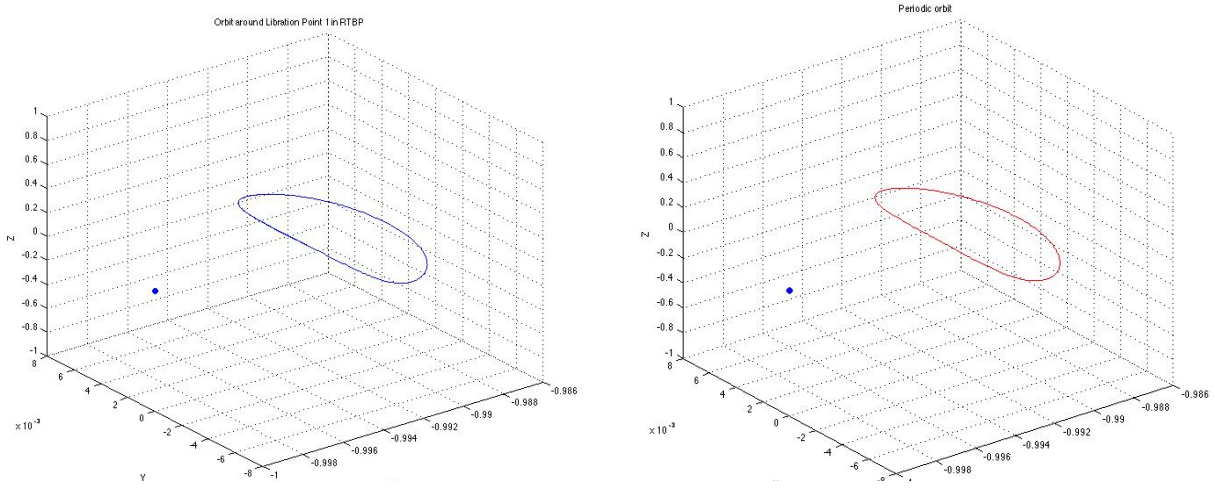

Figure 3.4 Planar Lyapunov orbits' positions related to the Earth

In the case of Planar Lyapunov orbit we see that the approximate profile which is obtained by the Lindstedt – Poincaré expansion is not so different from the collocated orbit. Anyway, the orbit which is refined by collocation procedure is more accurate and exists in the CR3BP model therefore can be used for further computations.

The last family that we provide as the comparison is the Halo family. This type of orbits which exists in the neighborhood of the collinear libration points is the most popular in the case of the space mission design. Table 3.3 illustrates the state vectors of Halo orbits which are obtained by the Lindstedt – Poincaré procedure and the method of collocation.

Table 3.3 State vector of initial conditions (Halo)

Method	X	Y	Z	V_x	V_y	V_z	Period
Lindstedt-Poincaré	-9.9002192e-01	2.8423650e-05	-2.0436054e-03	1.8682814e-04	8.7872059e-05	2.77748151e-04	3.000746366511
Collocation	-9.9197452e-01	1.75536960e-10	1.8842046e-03	2.25424406e-11	1.09696432e-02	1.62919982e-10	3.055724464866

The following two figures 3.5 and 3.6 illustrate the orbits. It is easy to see that the computed numerically orbit (red) which is refined by collocation method in the CR3BP model is bigger that the orbit which is computed semi – analytically (blue). The difference appears because the blue orbit is an approximate solution and is computed using semi – analytical algorithm, whereas the red one is refined using the vector field of the model.

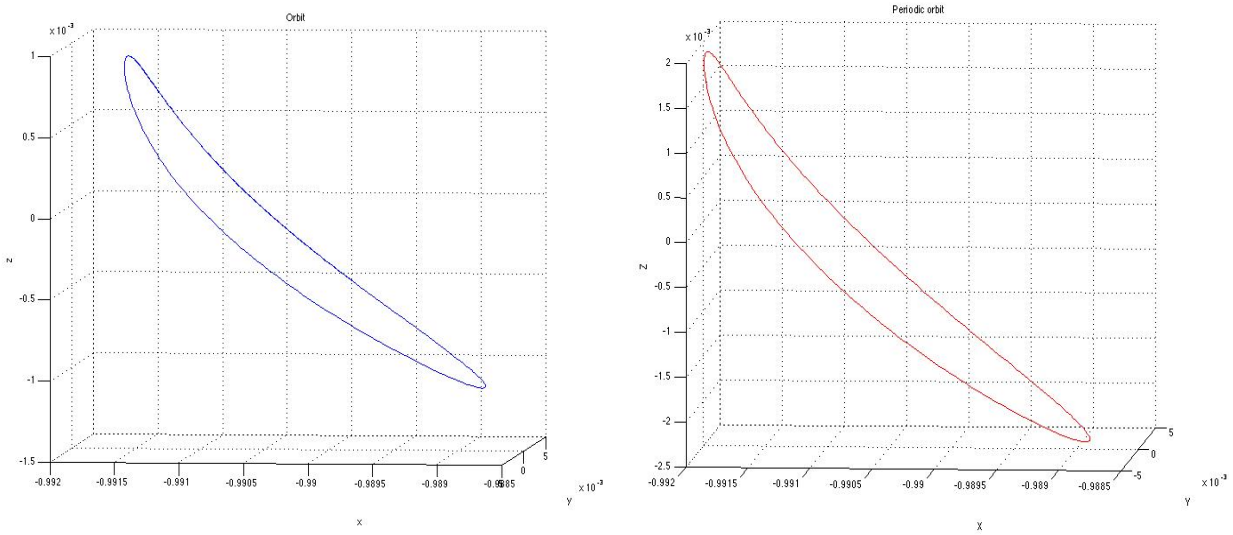

Figure 3.5 Halo orbits obtained semi-analytically (blue) and numerically (red)

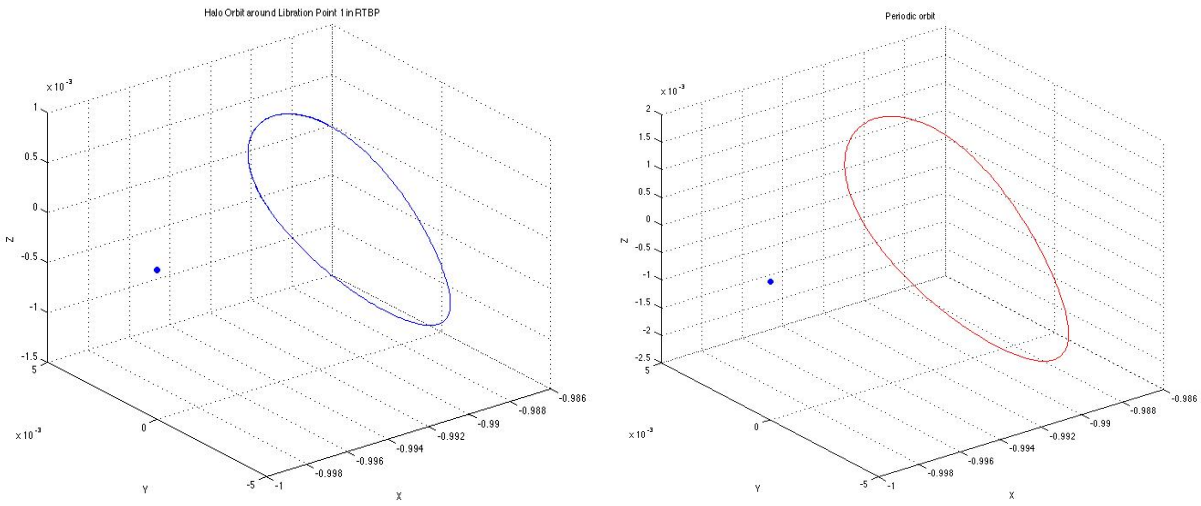

Figure 3.6 Halo orbits' positions related to the Earth

Due to the fact that we look for closed solution and represent our problem as BVP, it is not possible to obtain quasi – periodic orbits without modifying the algorithm. We can study the approximate quasi – periodic solutions which are computed by the Lindstedt – Poincaré procedure. The figures 3.7 and 3.8 below provide two Lissajous orbits with $\alpha = 0.01$ and $\beta = 0.05$ and $\beta = 0.15$, respectively.

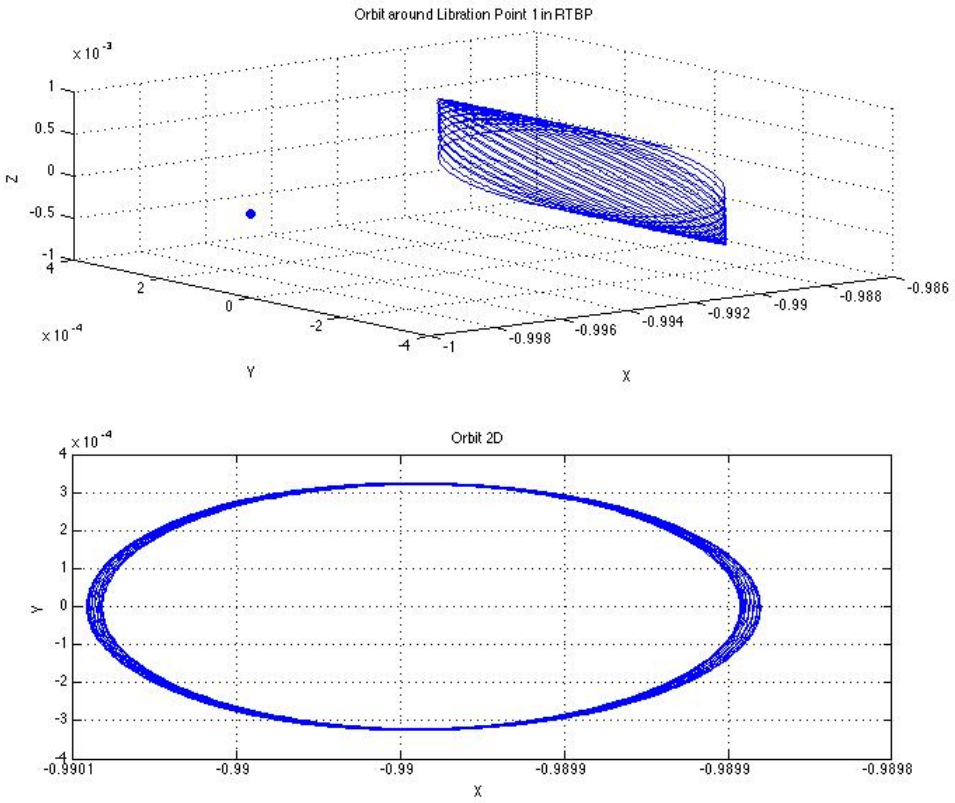

Figure 3.7 Lissajous orbit with β = 0.05

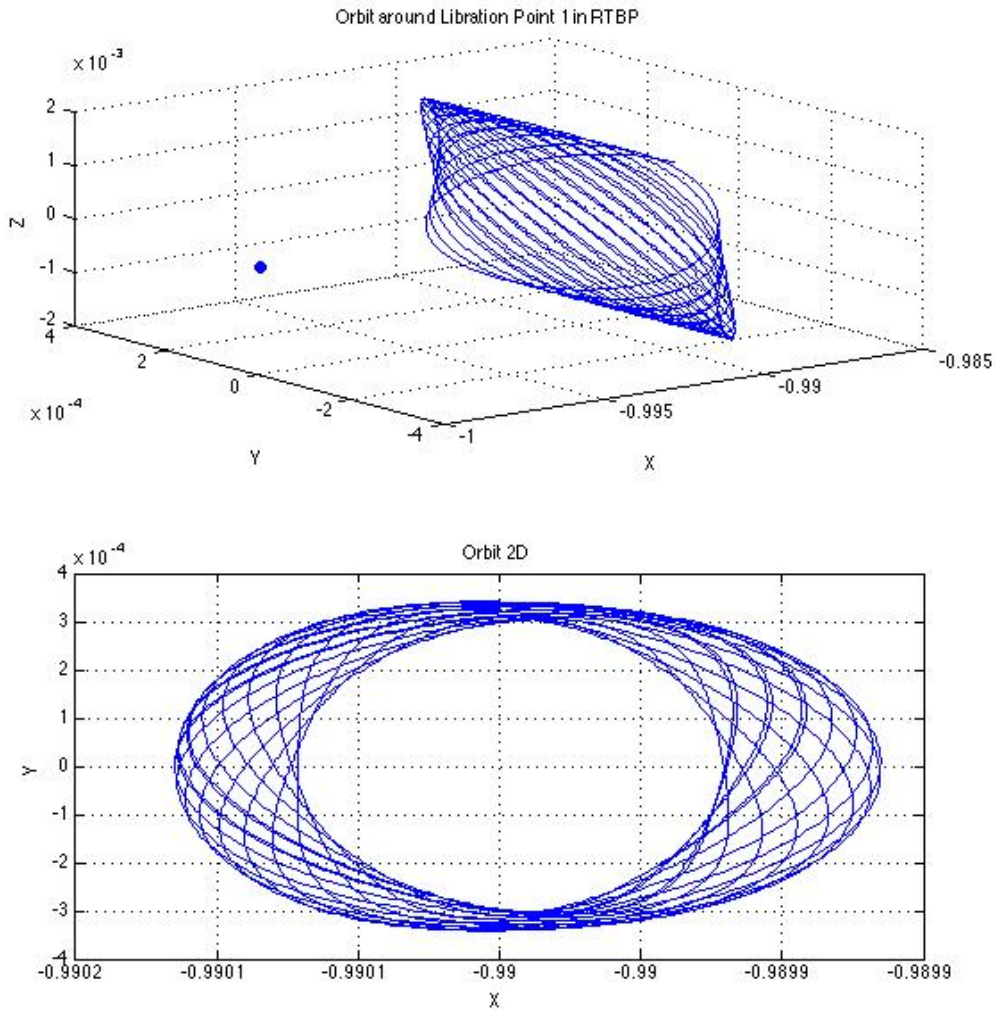

Figure 3.8 Lissajous orbit with β = 0.15

We can see that using the Lindstedt – Poincaré procedure to calculate the initial conditions for the collocation procedure is efficient. Also the expansions can be used for an approximate study of quasi – periodic orbits. The next sub-section shows changes in orbits when A small deviation in some parameters of system is applied.

3.2. Evolution of orbit families

There is a reason why we need to have high accuracy in the case of the computation of the libration point orbits. The reason is that even some small deviations can have critical effects on orbits. That's why this sub - section shows an evolution of orbit in the case when we apply some small deviations for different parameters such as: mass parameter and initial period. Lyapunov vertical orbit around L1 point and Halo orbit

around Lagragian point number two are used for the demonstration due to the fact that these orbits are 3D orbits and it is easy to see any changes. The tables which appear in this section provide the state vectors and the periods of the initial orbits.

Table 3.4 State vector of Vertical Lyapunov orbit

Libration Point	X	Y	Z	V_x	V_y	V_z	Period
L1	-9.9014980e-01	2.7851450e-05	-2.0436054e-03	1.8094694e-04	8.7323887e-05	1.9911658e-03	3.1311087

Using the data from table 3.4 for the collocation procedure we obtain isolated periodic solution which corresponds to the family of Lyapunov vertical orbits. The family of Lyapunov vertical orbits has very a specific shape. It looks like a circle or ellipse in the case of X-Y view and like a 'eight' in case of Y-Z view.

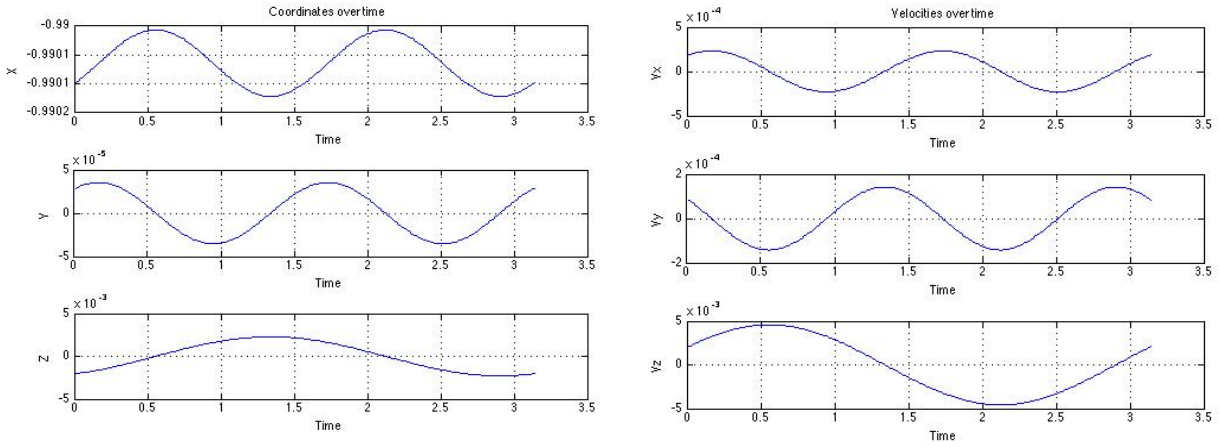

Figure 3.9 Vertical Lyapunov orbit's state vector over the time (null deviation)

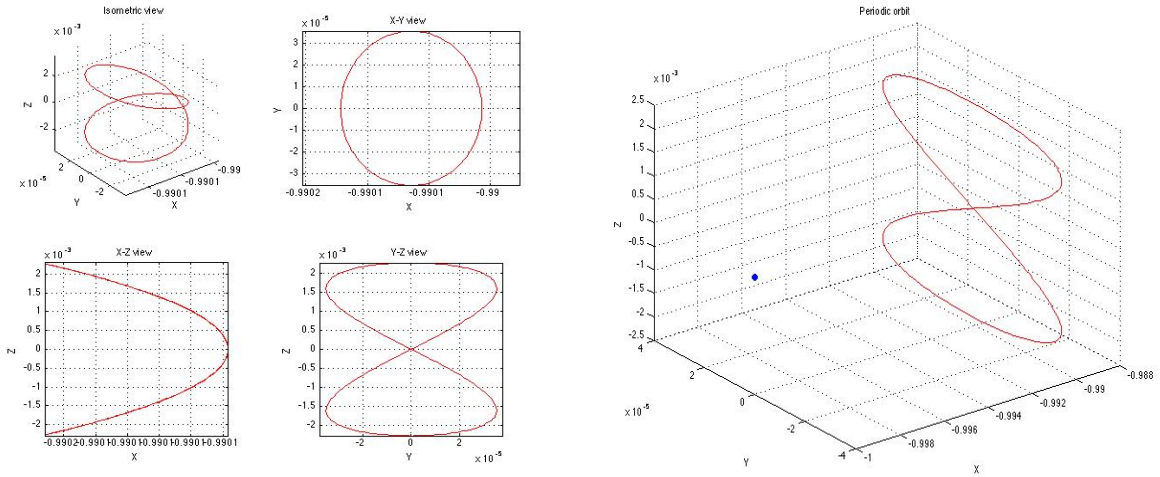

Figure 3.10 Vertical Lyapunov orbit around L1 (null deviation)

We have mentioned above that the mass parameter – the most important parameter of the system in CR3BP. According to this fact, it looks interesting and reasonable to see an evolution of orbits over the mass parameter changes. Now we apply small deviation to mass parameter μ = 3.040423398444176x10^{-6}. The value of the deviation is equal to ±10^{-9}. Applying the negative deviation, we obtain

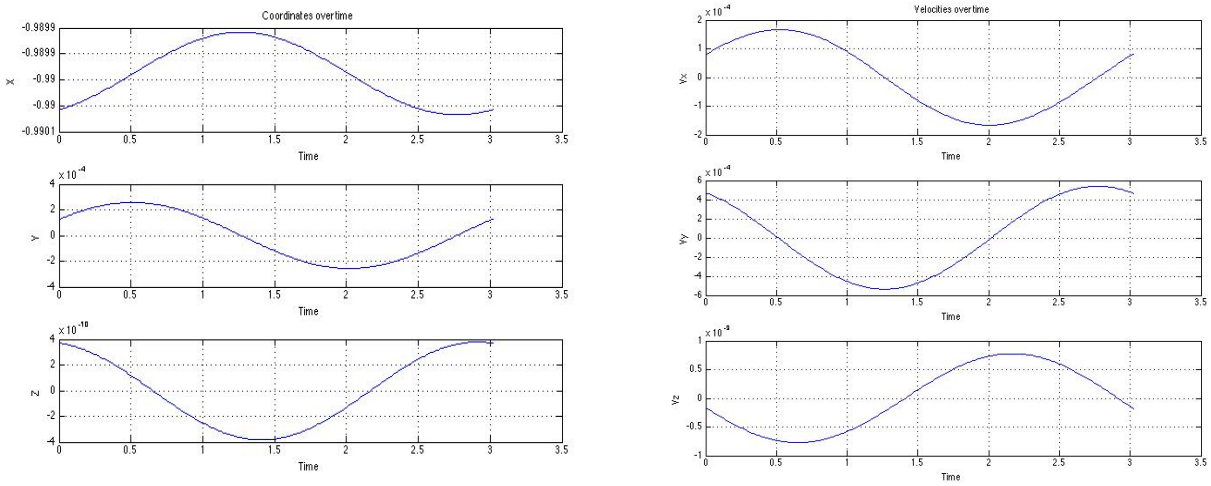

Figure 3.11 Vertical Lyapunov orbit's state vector over the time (negative deviation)

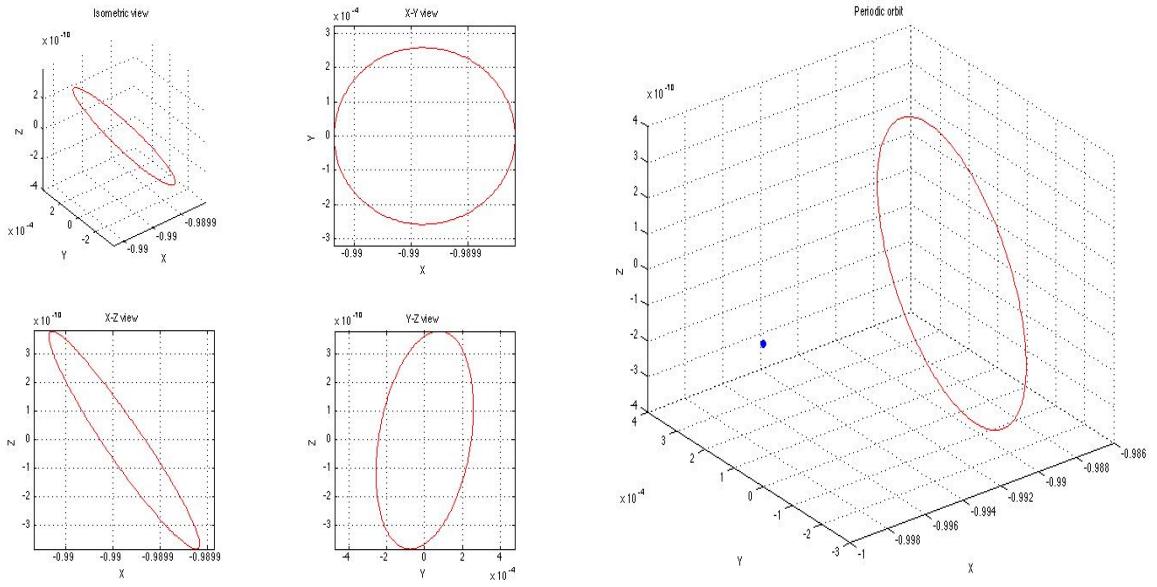

Figure 3.12 Vertical Lyapunov orbit around L1 (negative deviation)

We can see that changes, even very small changes, in mass parameter can deform an orbit significantly. Size and shape can be changed in different ways.

Applying a positive deviation, the orbit results in

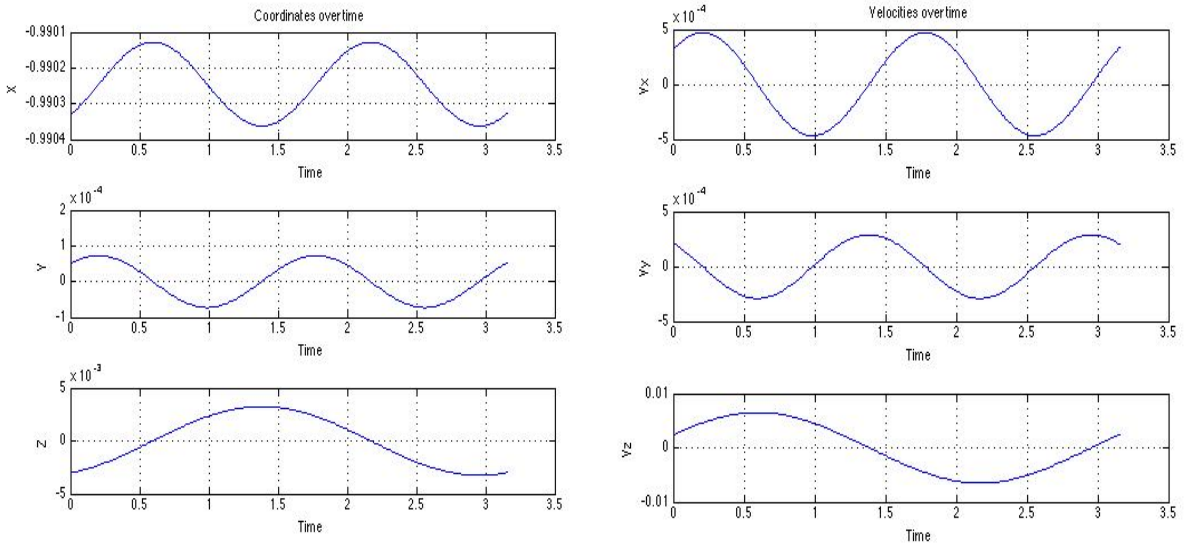

Figure 3.13 Vertical Lyapunov orbit's state vector over the time (positive deviation)

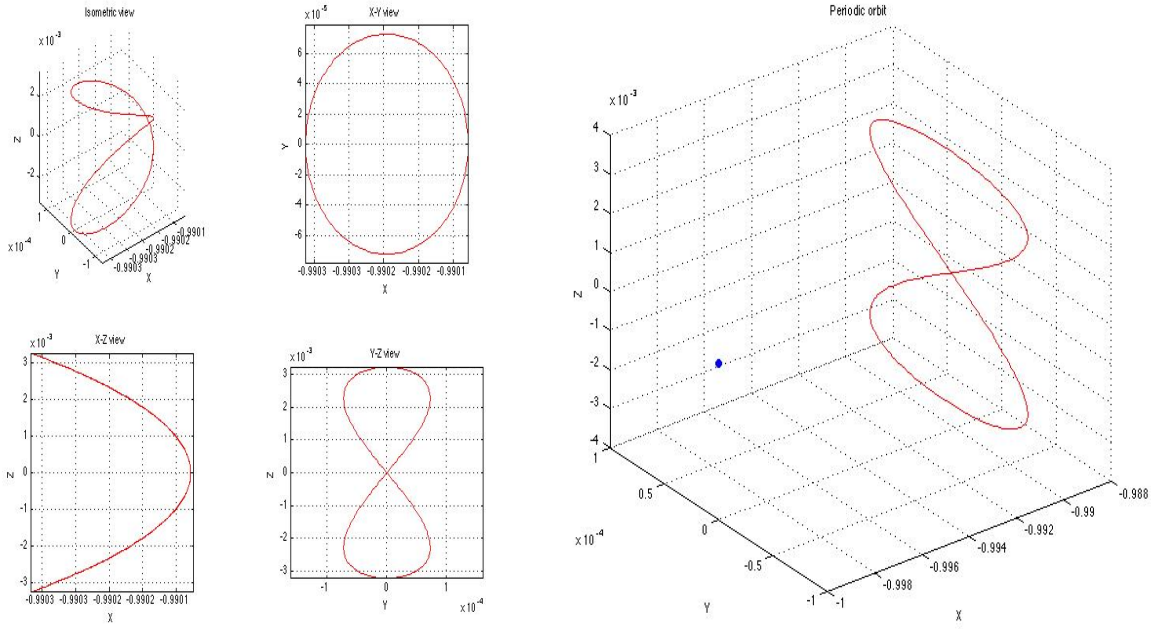

Figure 3.14 Vertical Lyapunov orbit around L1 (positive deviation)

In the case of positive deviation and this particular orbit we see a considerable increase in size and period. The following figures illustrate the calculated orbits together to show the difference.

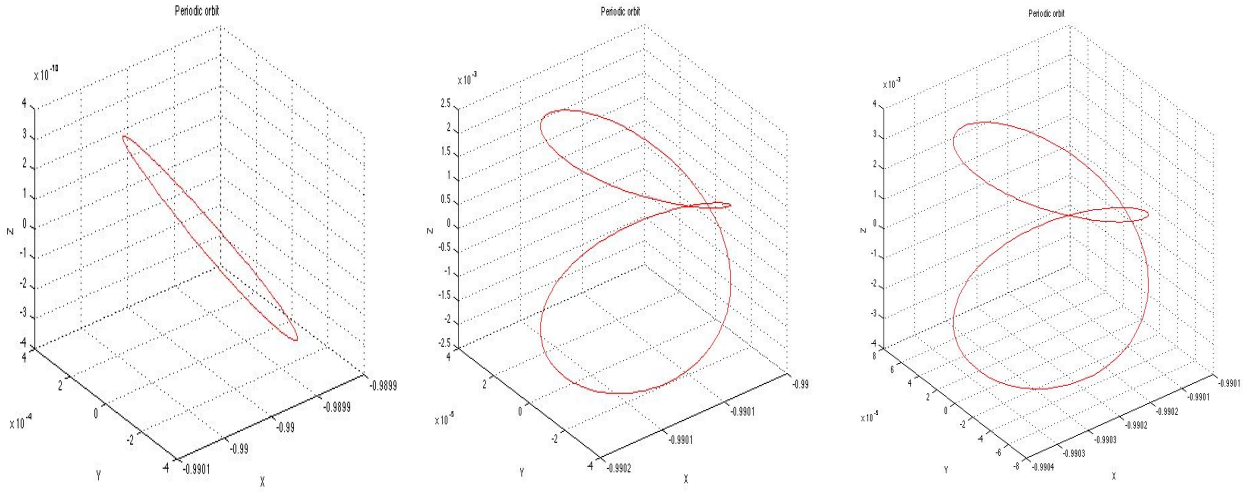

Figure 3.15 Evolution of Vertical Lyapunov orbit over mass parameter

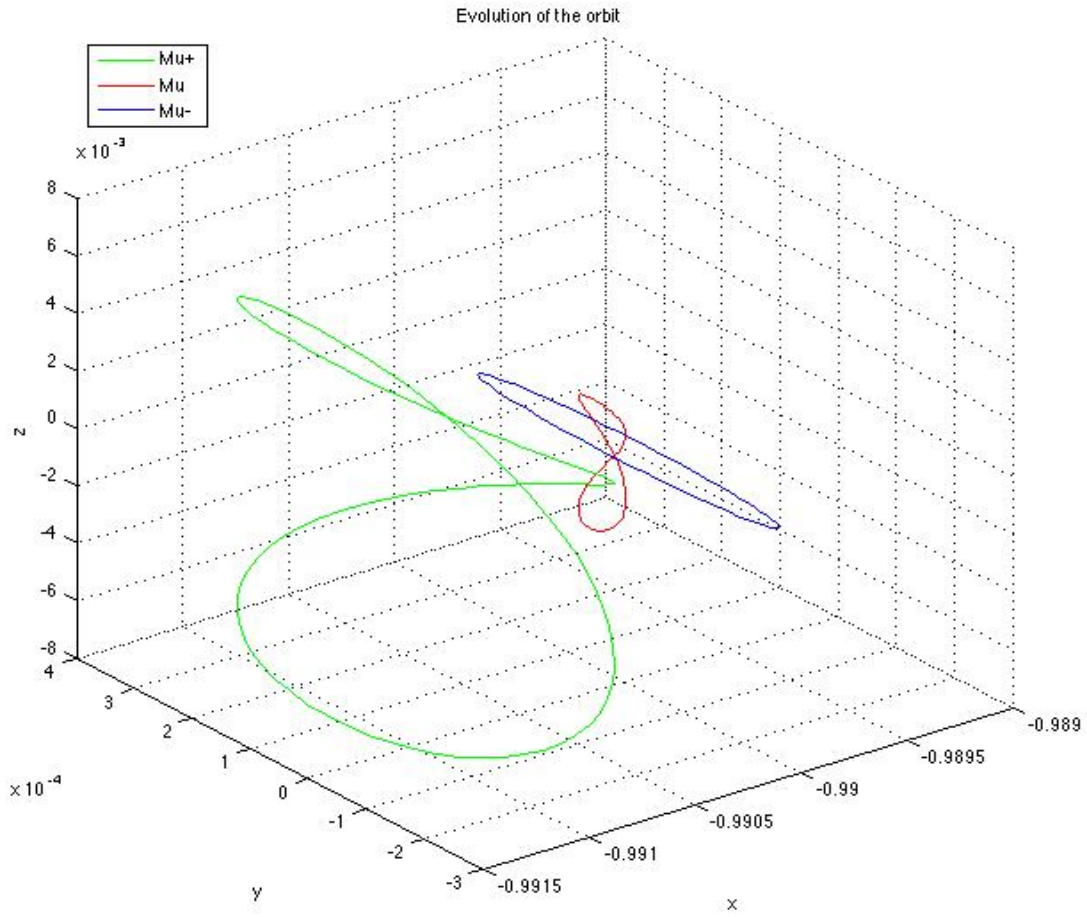

Figure 3.16 Comparison of the orbits

The next step is to fulfil the same test but using the Halo orbit located in the neighborhood of libration point 2. This test is not successful with the deviation value which is equal to $\pm 10^{-9}$. Since the solutions do not converge, so the deviation value decreases to $\pm 10^{-11}$. Having smaller deviation, the solutions converge and Halo orbits are refined. The table 3.5 below provides the initial state vector of the Halo orbit.

Table 3.5 State vector of Halo orbit

Libration Point	X	Y	Z	V_x	V_y	V_z	Period
L2	-1.0080192e+00	-1.2708061e-10	-1.8743063e-03	-1.4769194e-11	-1.1101702e-02	-1.2026598e-10	3.096804317210

We start the collocation procedure three times, once without deviation and once with positive and negative deviation respectively. As the initial seed data from the table 3.5 is used.

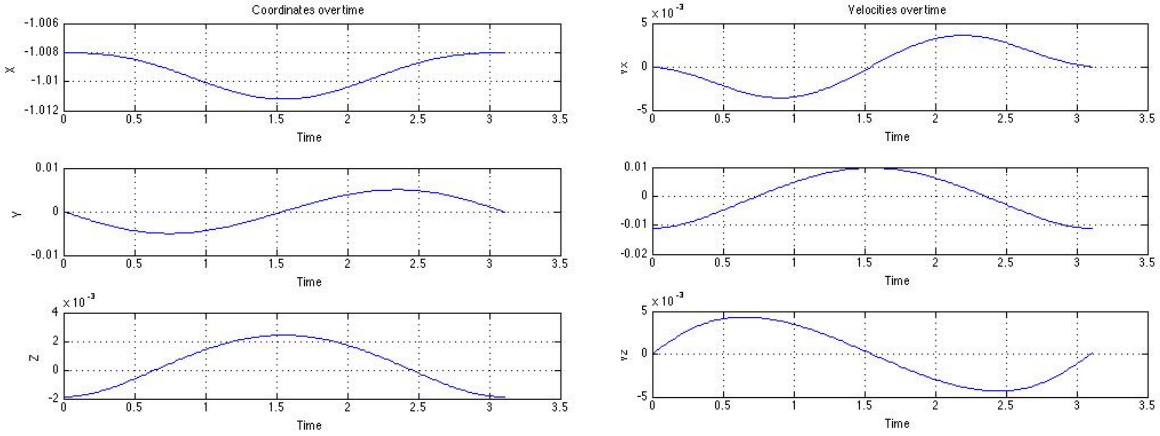

Figure 3.17 Halo orbit's state vector over the time (null deviation)

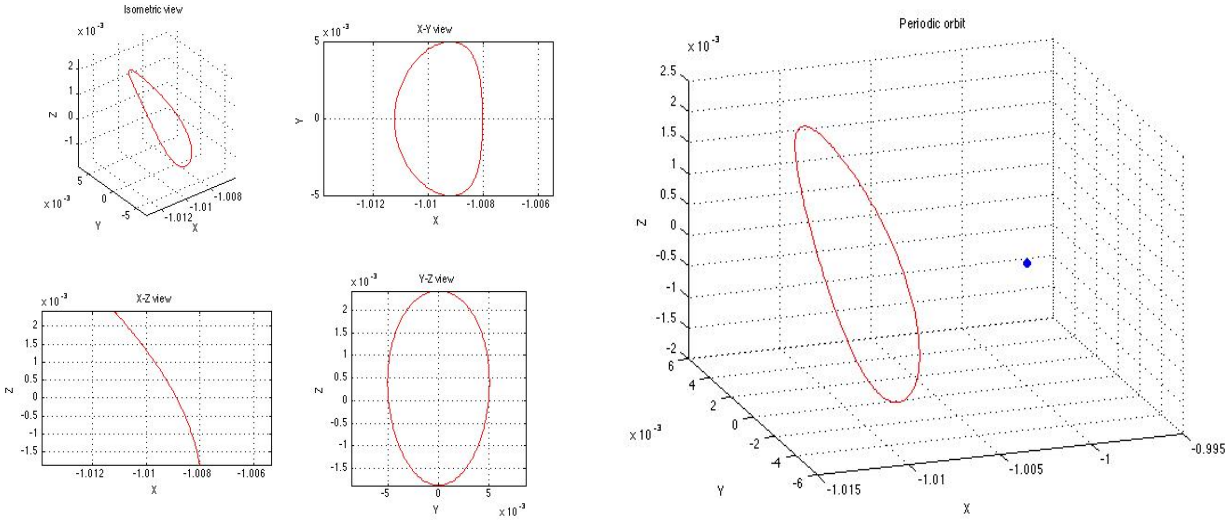

Figure 3.18 Halo orbit around L2 (null deviation)

Now we apply the negative deviation and can see that the size increases significantly. The following two figures show a refined by the collocation process orbit with a negative deviation added to the mass parameter.

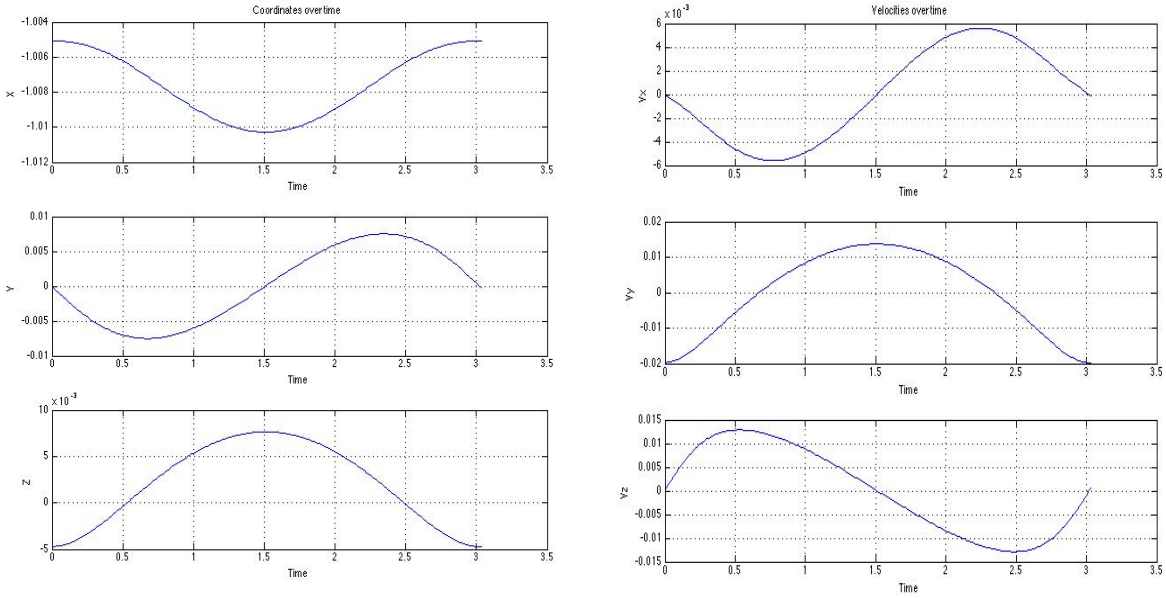

Figure 3.19 Halo orbit's state vector over the time (negative deviation)

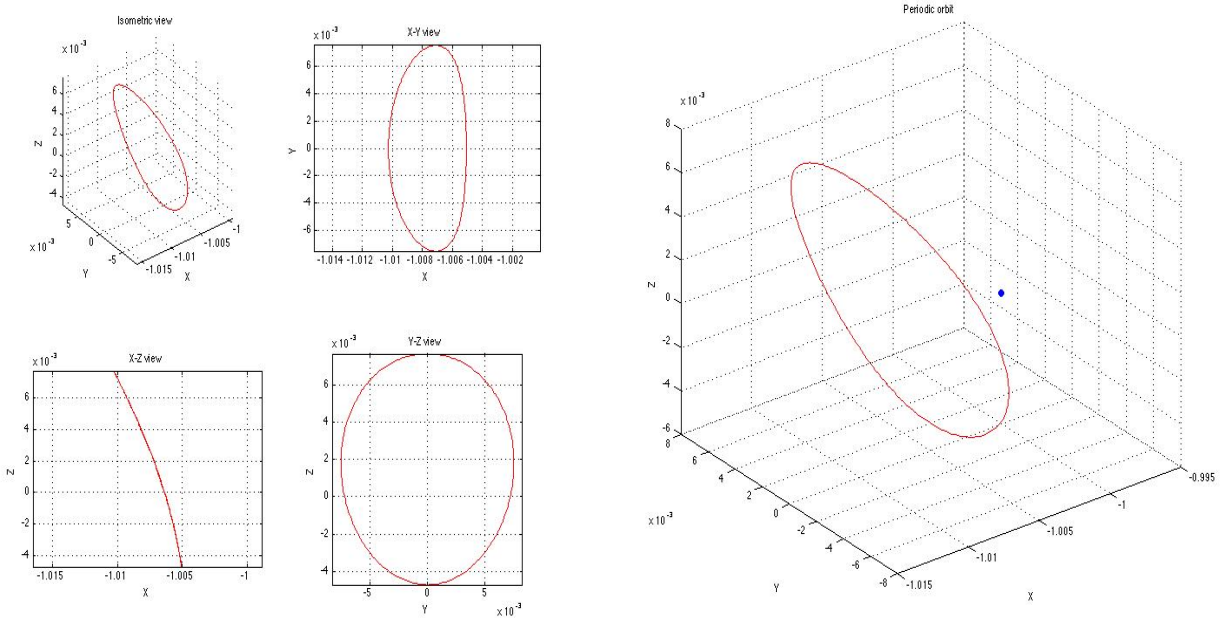

Figure 3.20 Halo orbit around L2 (negative deviation)

Applying the positive deviation to the initial conditions of the collocation procedure, the orbit results in

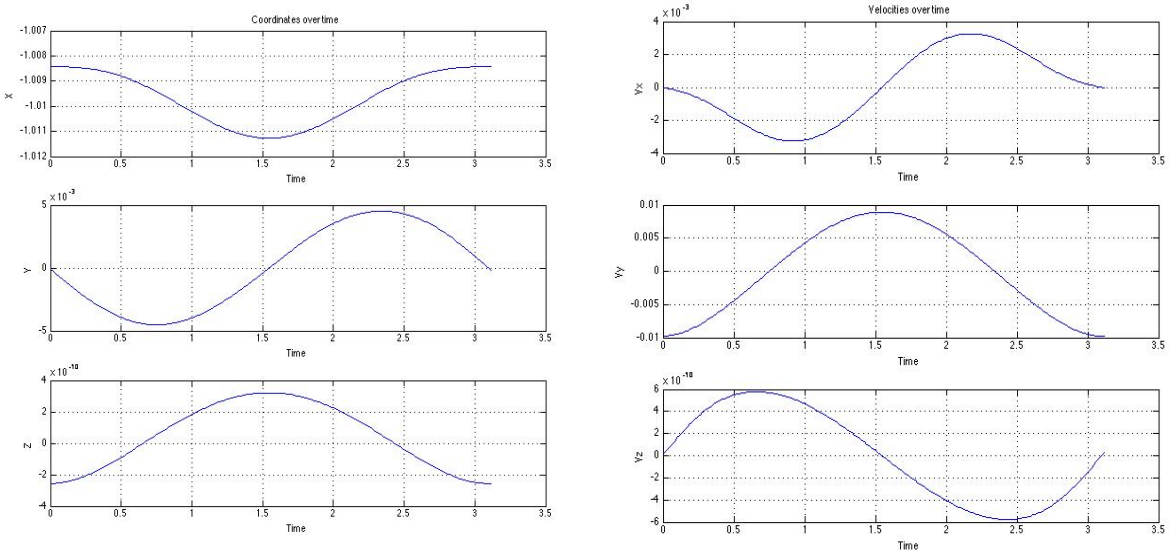

Figure 3.21 Halo orbit's state vector over the time (positive deviation)

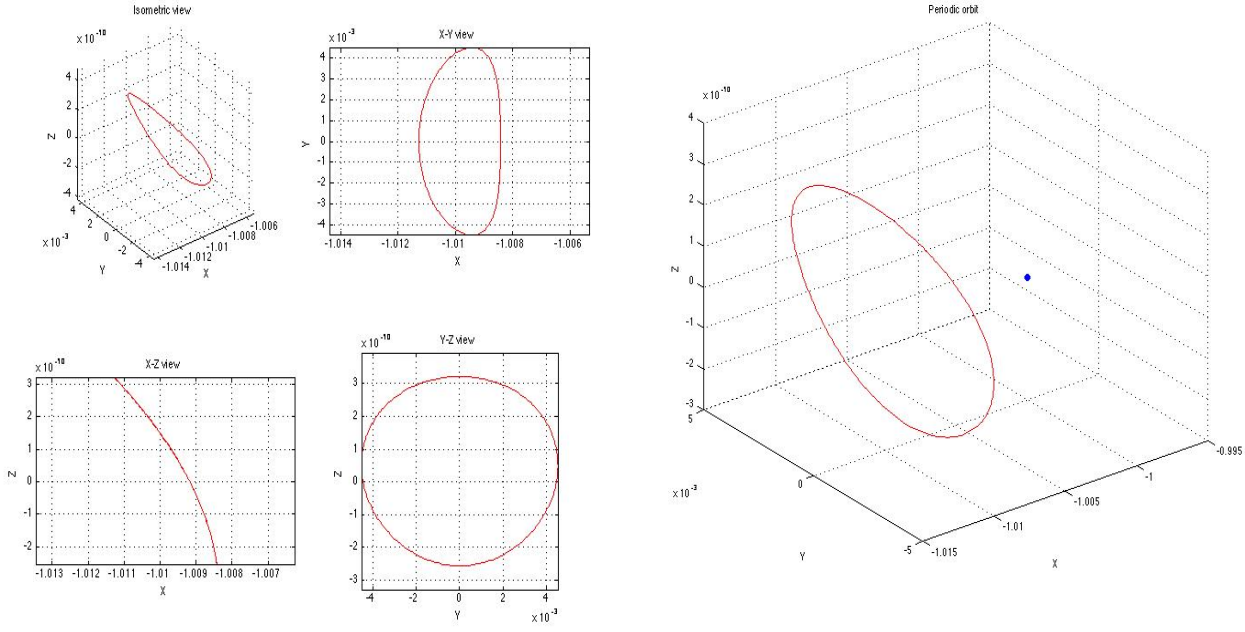

Figure 3.22 Halo orbit around L2 (positive deviation)

It is interesting to note that we can see tremendous decrease in *z* – components of the orbit. The following figure illustrates an evolution of the orbit over the mass parameter.

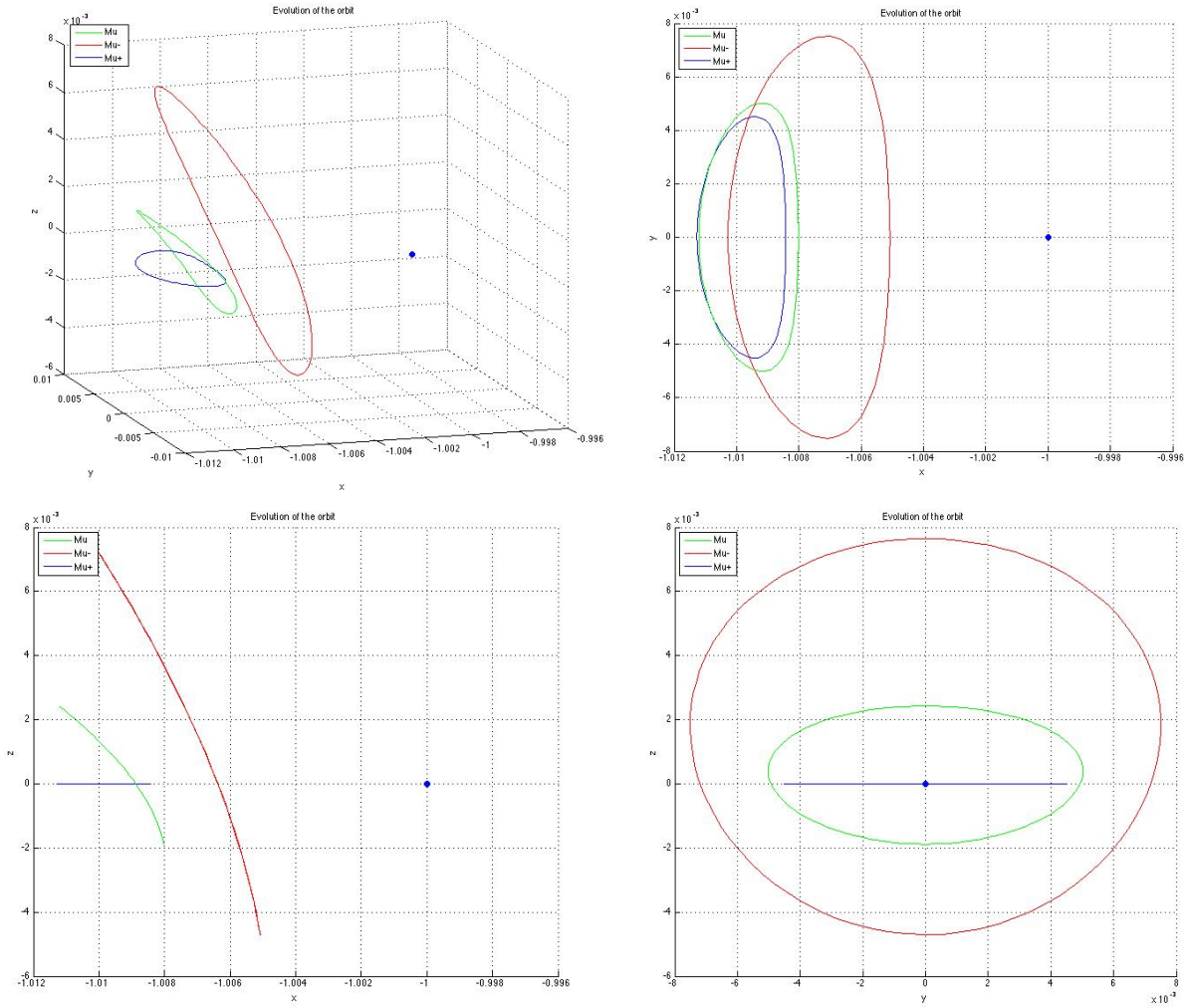

Figure 3.23 Evolution of Halo orbit over the mass parameter

Another test is to apply different deviations to the initial guess of the period. Applying a small deviation 10^{-7} to the period gives us very small changes of orbits

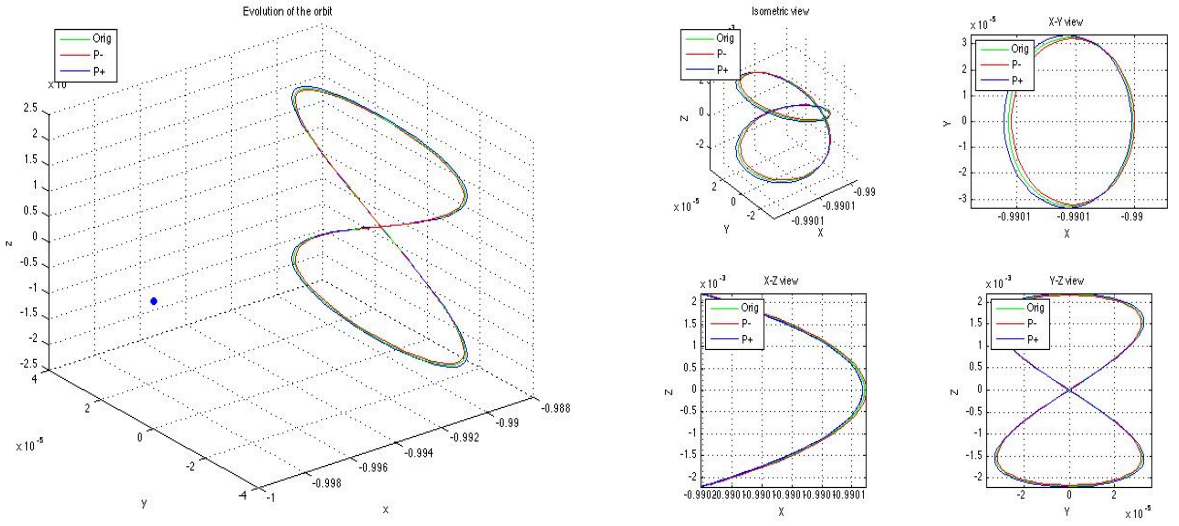

Figure 3.24 Comparison of three orbits when 10^{-7} deviation to the period is applied

Now we apply a bigger deviation to see what happens. The value of deviation equals 10^{-5}.

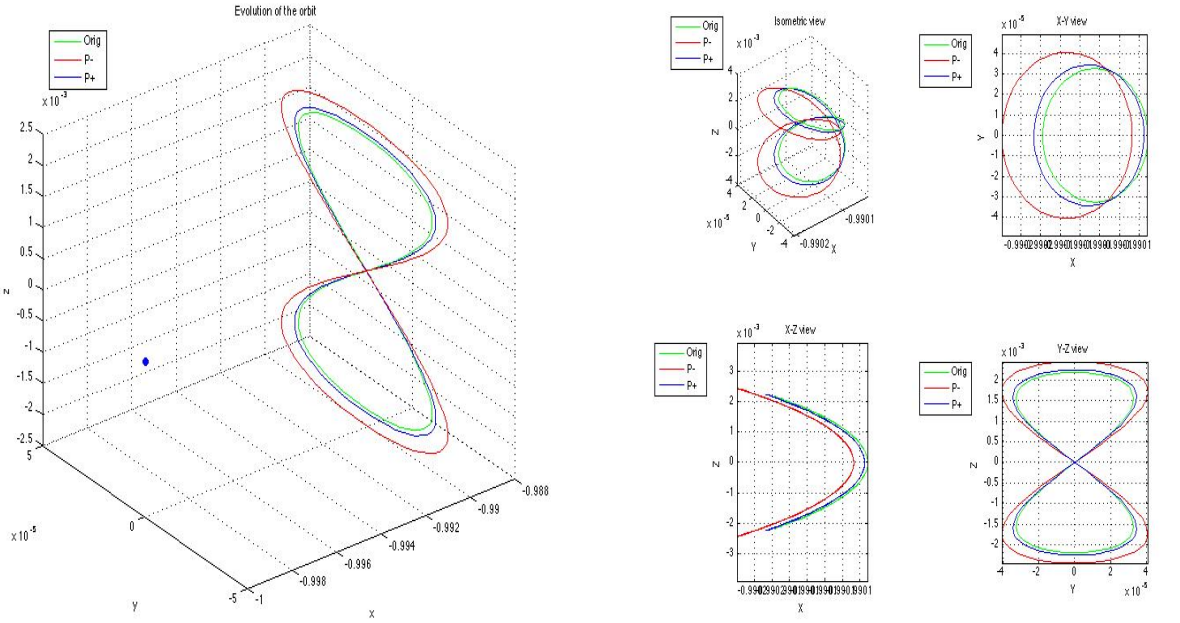

Figure 3.25 Comparison of three orbits when 10^{-5} deviation to period is applied

Now we increase the deviation applied to the initial guess of period up to 10^{-3} and 10^{-1}.

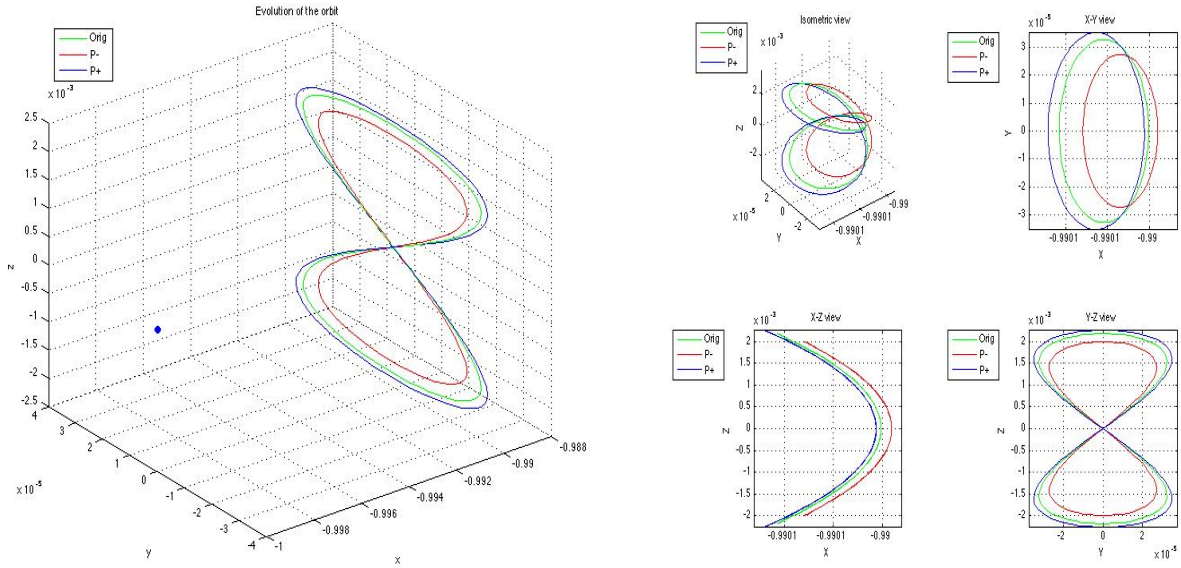

Figure 3.26 Comparison of three orbits when 10^{-3} deviation to period is applied

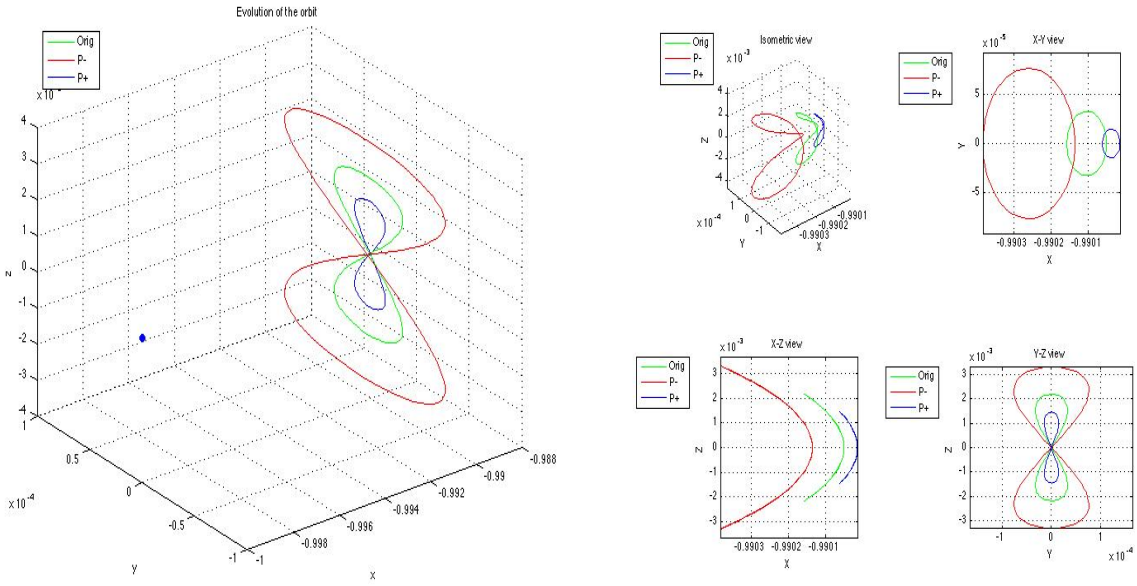

Figure 3.27 Comparison of three orbits when 10^{-1} deviation to period is applied

3.3. Continuation approach

It is well known that there are several kinds of orbits. Some of them have already been presented in the previous sections, for instance: Lissajous orbits and vertical Lyapunov orbits. This section focuses to Halo and Planar Lyapunov families of orbits. The following table provides state vectors and periods for the presented orbits and this is enough to refine the orbit with the collocation procedure. The continuation of an orbit is implemented without any adaptive step and stopping point, so the number of orbits and step must be selected by a user.

Table 3.6 State vectors and periods of orbits

Class	X	Y	Z	V_x	V_y	V_z	Period
Halo	-1.00801920e+00	-1.2708061e-10	-1.8743063e-03	-1.4769194e-11	-1.1101702e-02	-1.2026598e-10	3.096804
Planar Lyapunov	-1.00667003e+00	-2.8401638e-04	0	1.53946454e-04	-1.8992240e-02	0	3.243072

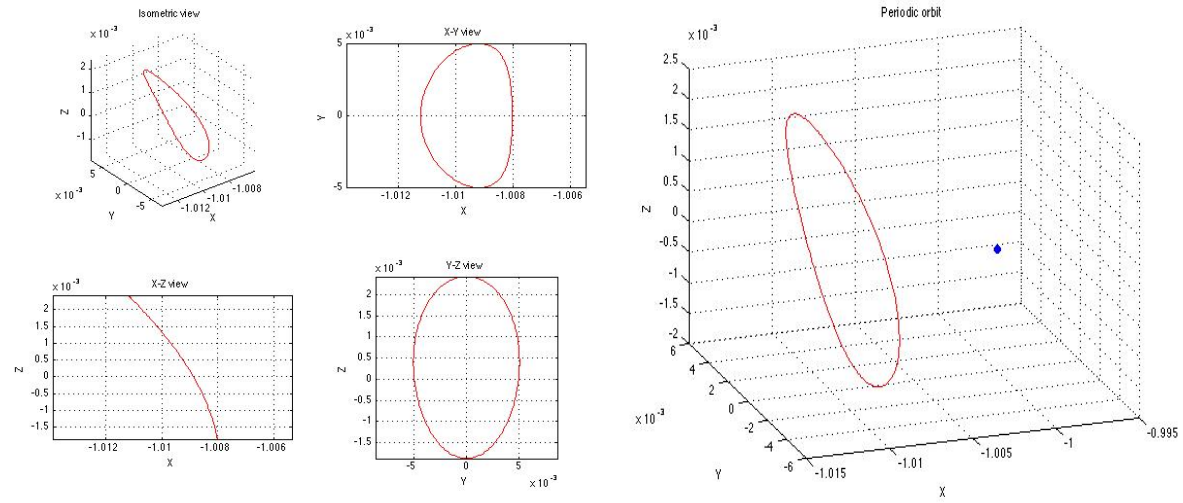

Figure 3.28 Halo orbit computed by the collocation procedure

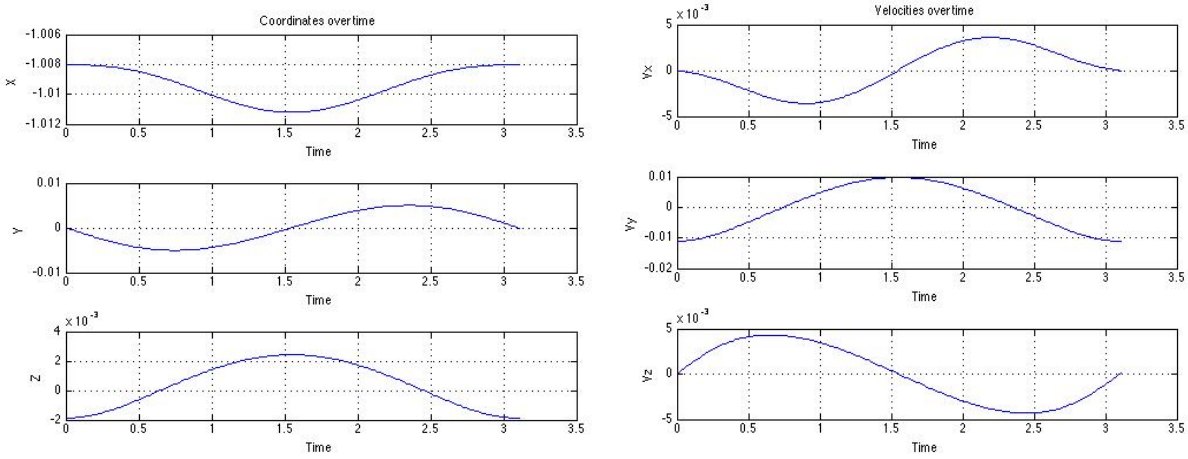

Figure 3.29 Evolution of state vector of given orbit over the time

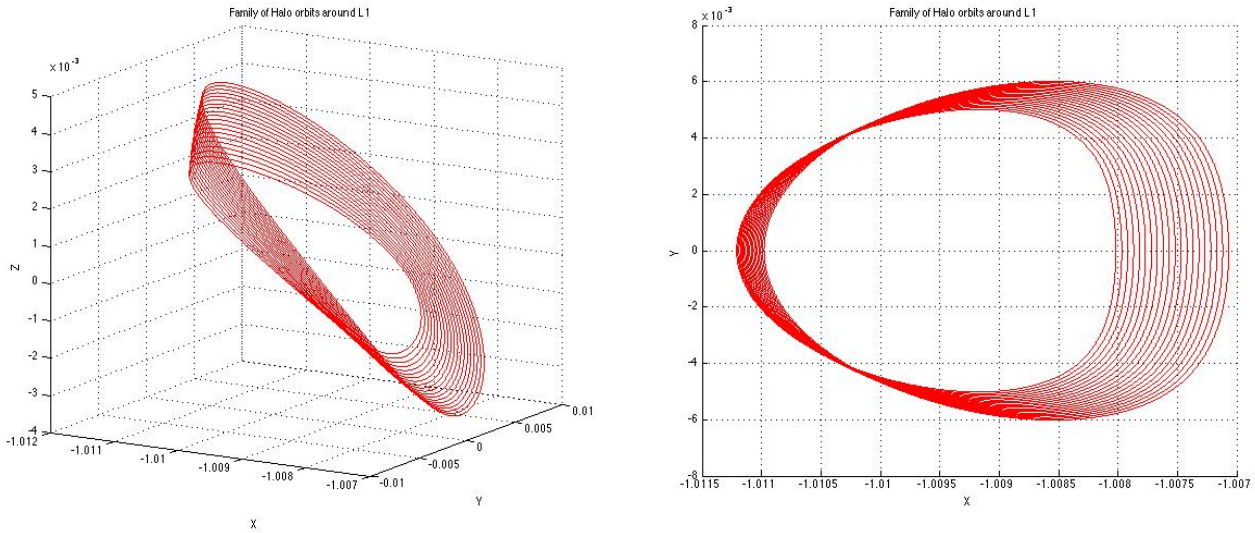

Figure 3.30 Family of Halo orbits around L2 (3-D and X-Y)

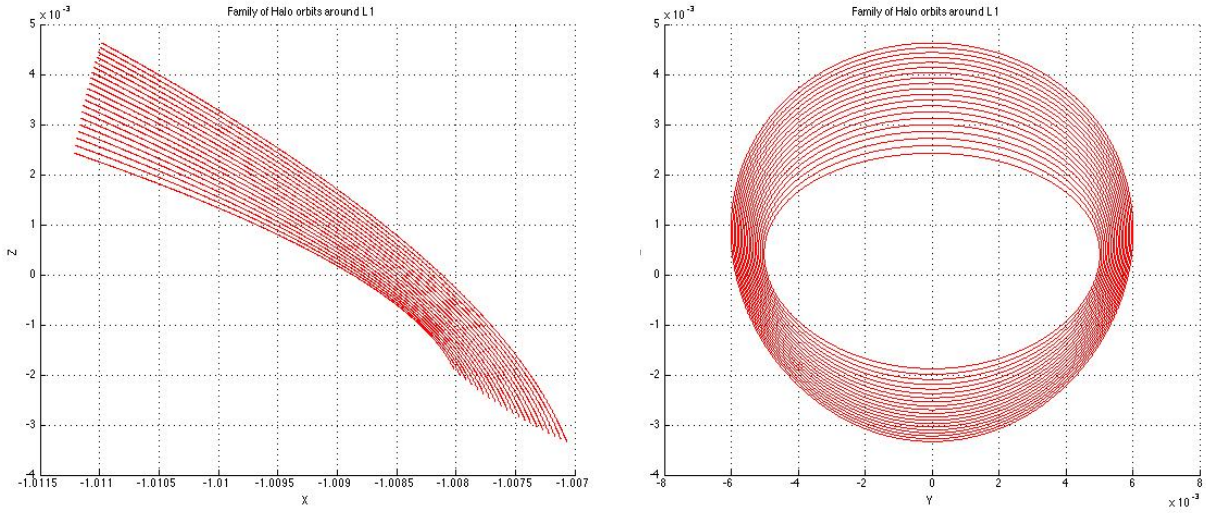

Figure 3.31 Family of Halo orbits around L2 (X-Z and Y-Z)

This family of Halo orbits contains 20 orbits and for the continuation process is performed with step = 0.00005 in dimensionless reference system.

Now the family of Planar Lyapunov orbits in the neighborhood of libration point 2 is presented. The family contains 20 orbits and the step between orbits is equal to 0.00006. The Figure below shows the initial orbit.

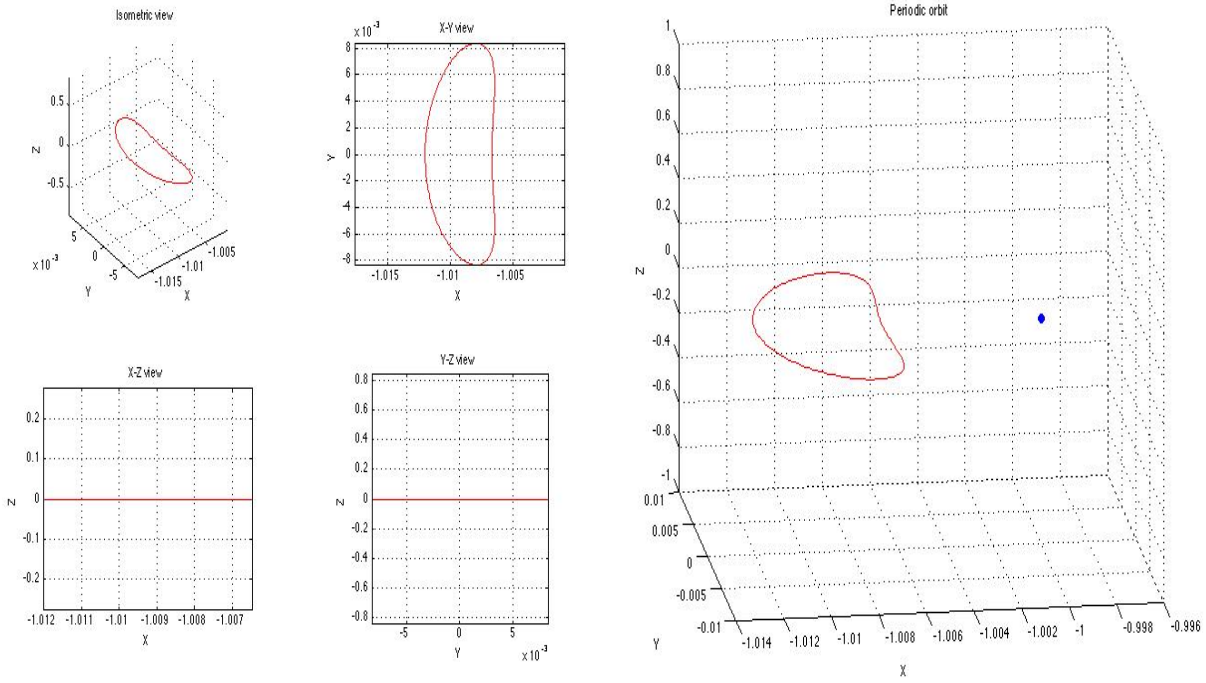

Figure 3.32 Planar Lyapunov orbit computed by collocation

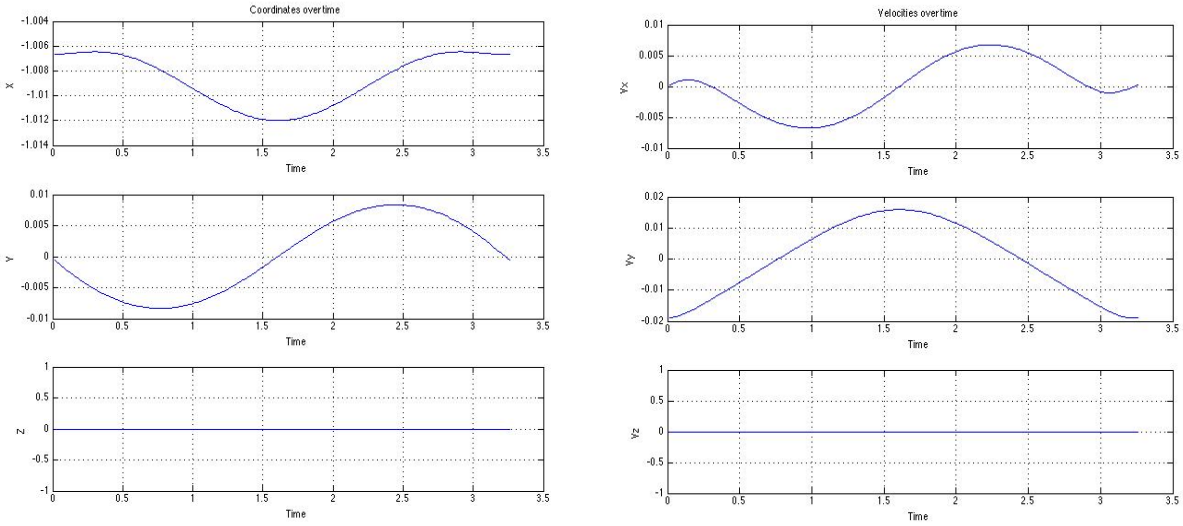

Figure 3.33 State vector of given orbit over the time

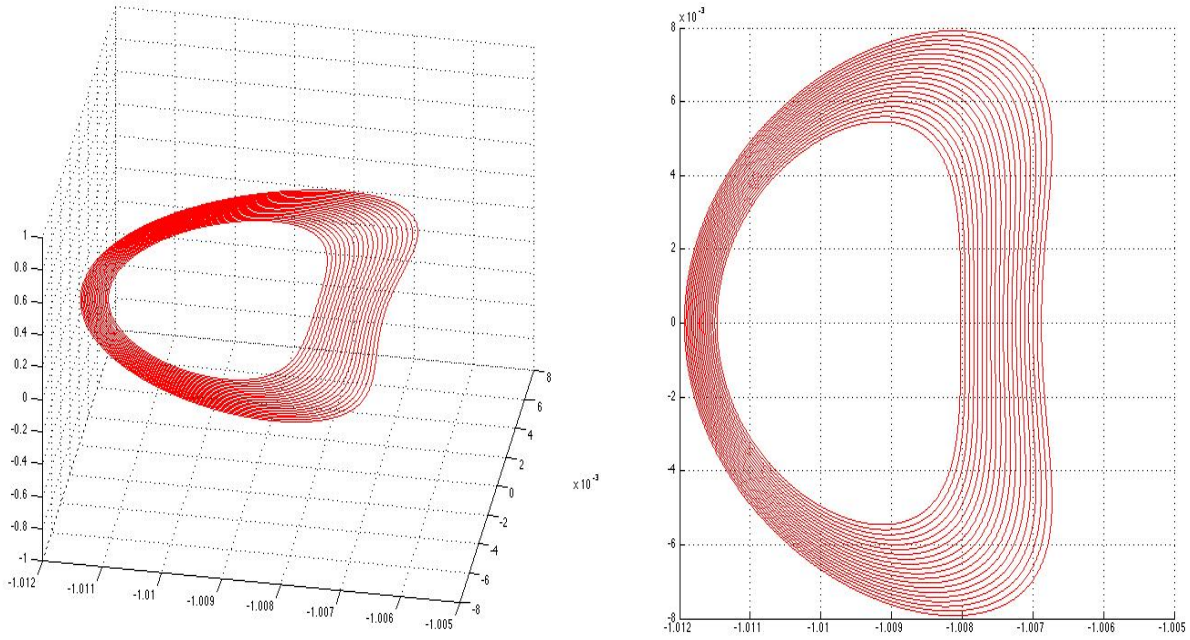

Figure 3.34 Family of Planar Lyapunov orbits

We introduce only isometric and X-Y view according to the fact that *z* – components equal to zero.

There is a possibility that one family of orbits bifurcates from another one and when this situation occurs we can see two types of orbit's branches during one continuation process. The following figure illustrates two different families of orbits which are obtained by one continuation procedure with 30 orbits and step = 0.00007.

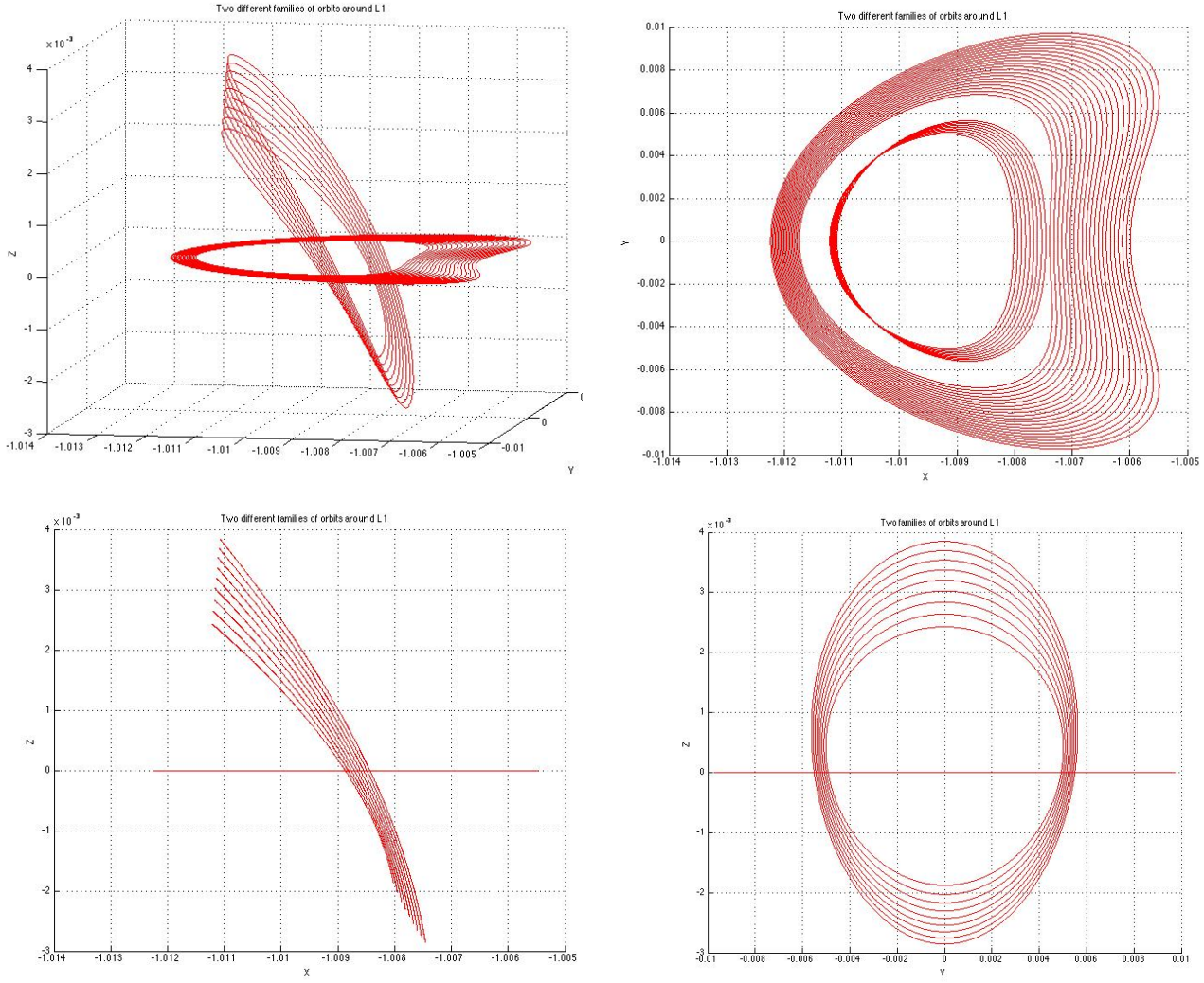

Figure 3.35 Halo and Planar Lyapunov families

3.4. Stable and unstable manifolds

Stable and unstable manifolds and connecting orbits which are provided by manifolds is an important subject for thorough study. This sub-section provides results of manifolds' computations, heteroclinic and homoclinic orbits and maneuvers calculations. It starts with homoclinic trajectories, then provide the results of computations of heteroclinic trajectories which exist between two planar orbits which are located around opposite collinear points and the final part of this sub-section is the representation of stable and unstable manifolds of Halo orbits. Table 3.7 contains the state vector of the orbit which is used to show homoclinic trajectories.

Table 3.7 The state vector of orbit to obtain homoclinic trajectory

Libration Point	X	Y	Z	V_x	V_y	V_z	Period
L1	-9.9179544e-01	1.4999947e-04	0	1.7918310e-05	1.0836423e-02	0	3.0733640063177092

Now we use algorithm which we have described in the previous chapter to calculate stable and unstable manifolds of given orbit.

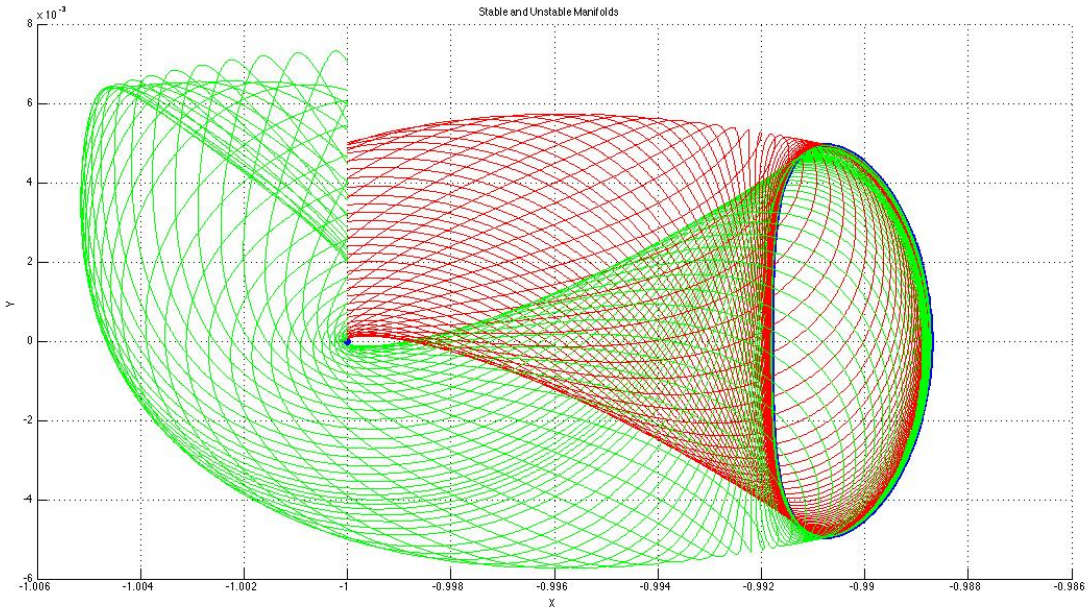

Figure 3.36 Stable and unstable manifolds of one orbit around L1

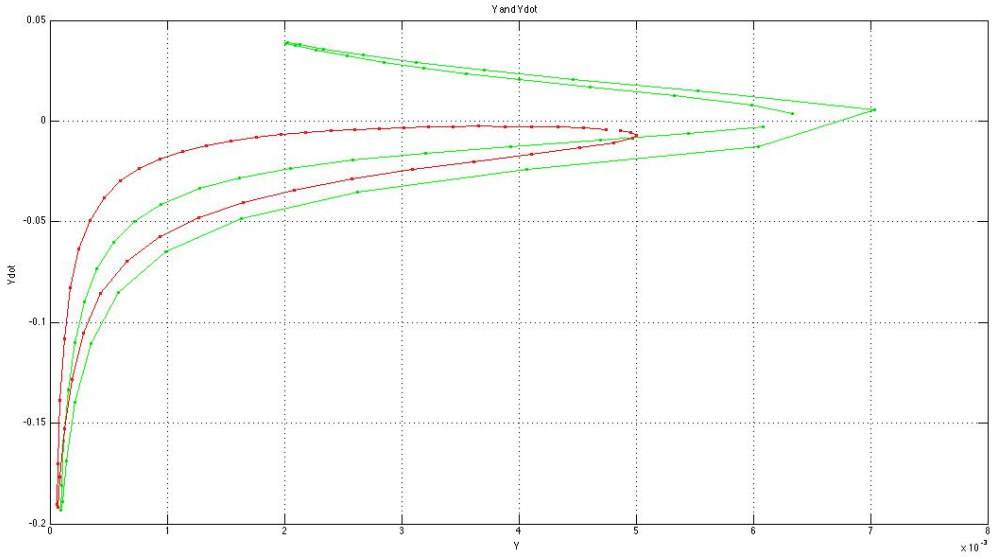

Figure 3.37 Intersections needed to obtain Y - components

Figure 3.36 illustrates 50 (the number that can be chosen by operator) arrival (green) and departure (red) trajectories which correspond respectively to stable and unstable manifolds of this particular orbit. Figure 3.37 presents profiles of *y* – components of each trajectory of both manifolds. They are not boundary due to the feature of the computational process. It will be correct if we simply connect the first and the last points. The intersections of red and green curves mean that there are connecting trajectories. Here, it is easy to see that there are two trajectories. The figure below shows homoclinic trajectory.

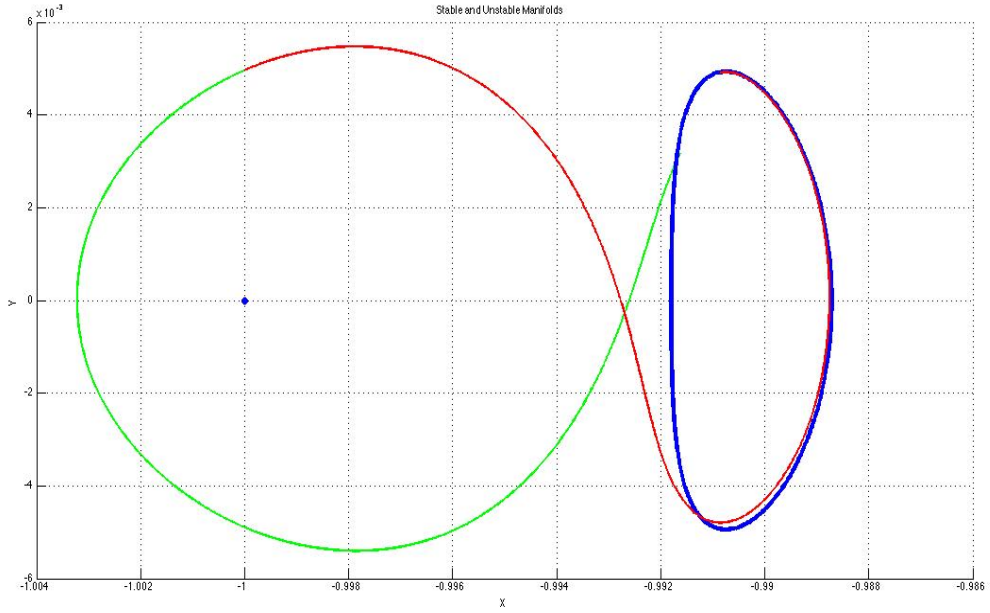

Figure 3.38 Homoclinic trajectory

Table 3.8 Initial conditions for Homoclinic trajectory

Libration Point	X	Y	Z	V_x	V_y	V_z	Period
L1	-0.9908599626	0.004921758301120	0	0.003161076322996	0.000642750481497	0	4.497347857684

Now we represent manifolds of two orbits which are located in different collinear points in the Sun – Earth/Moon system. The vector states can be found in the table below.

Table 3.9 The state vectors of orbits to obtain heteroclinic trajectory

Libration Point	X	Y	Z	V_x	V_y	V_z	Period
L1	-9.9179544e-01	1.4999947e-04	0	1.7918310e-05	1.0836423e-02	0	3.0733640063177
L2	-1.008018e+00	0	0	0	-1.1938388e-02	0	3.1262900575331

We apply the same algorithm with the only difference that we have two orbits, one around L_1 and another around of L_2. The stable manifold is calculated for the orbit around L_1 and the unstable manifold is obtained for the orbit around L_2.

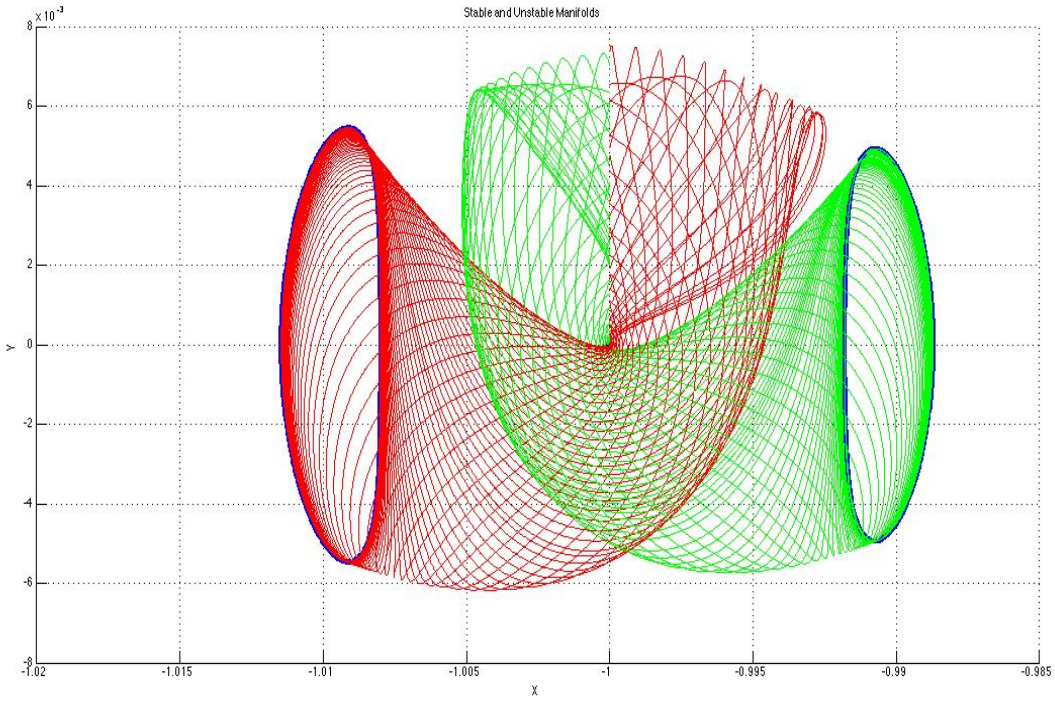

Figure 3.39 Stable and unstable manifolds of the orbits around L1 and L2

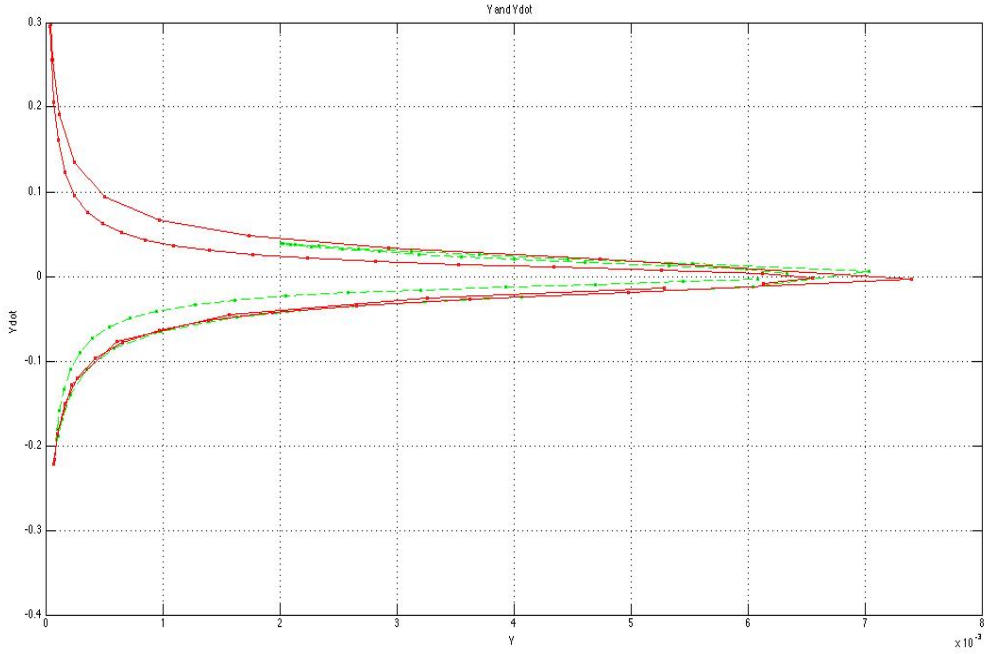

Figure 3.40 Intersections needed to obtain Y - components for heteroclinic trajectories

The following table presents initial conditions of the heteroclinic trajectory which connects two planar orbits located around libration point number one and two. A heteroclinic trajectory can be propagated from both libration points.

Table 3.10 Initial seeds for heteroclinic trajectory

Libration Point	X	Y	Z	V_x	V_y	V_z	Period
L2	-1.0080048578	-0.0012955670044	0	-0.00003643689332	-0.01155341399799	0	4.45233415873
L1	-0.9887330247	-0.00082288614890	0	0.001487719169338	-0.01059113024800	0	4.45233415873

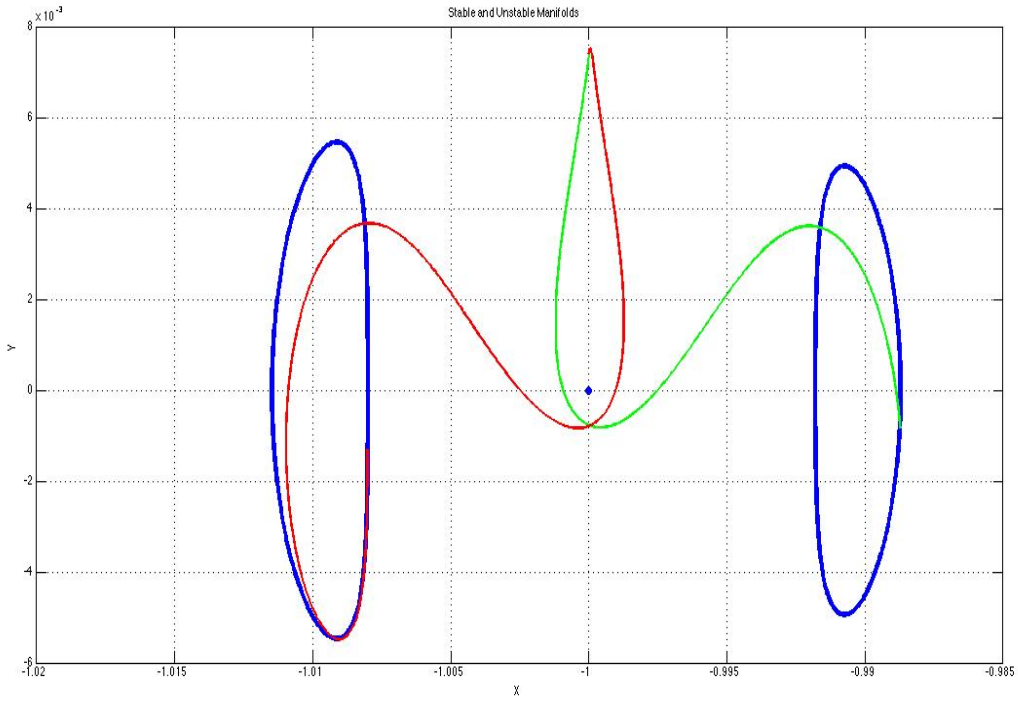

Figure 3.41 Heteroclinic trajectory

It has been mentioned above that heteroclinic and homoclinic trajectories can be computed only for planar orbits. In the case when we have 3D orbits we can provide maneuvers to reach one orbit from another. The following table provides state vector of Halo orbits in the neighborhood of collinear libration points.

Table 3.11 State vectors of Halo orbits

Libration Point	X	Y	Z	V_x	V_y	V_z	Period
L1	-9.9197556e-01	0	-1.886612e-03	0	1.0973081e-02	0	3.0557121784474464
L2	-1.0080185894	-1.5479509e-04	-1.875113e-03	-3.4376537e-05	-1.1098152e-02	1.677541e-04	3.0967911641348698

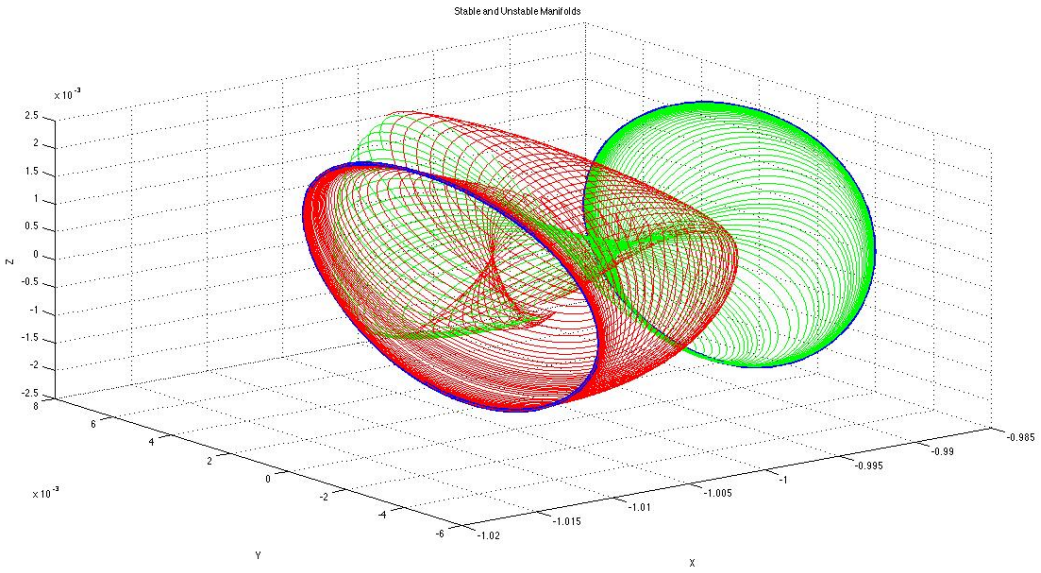

Figure 3.42 Stable and Unstable manifolds of Halo orbits (Isometric view)

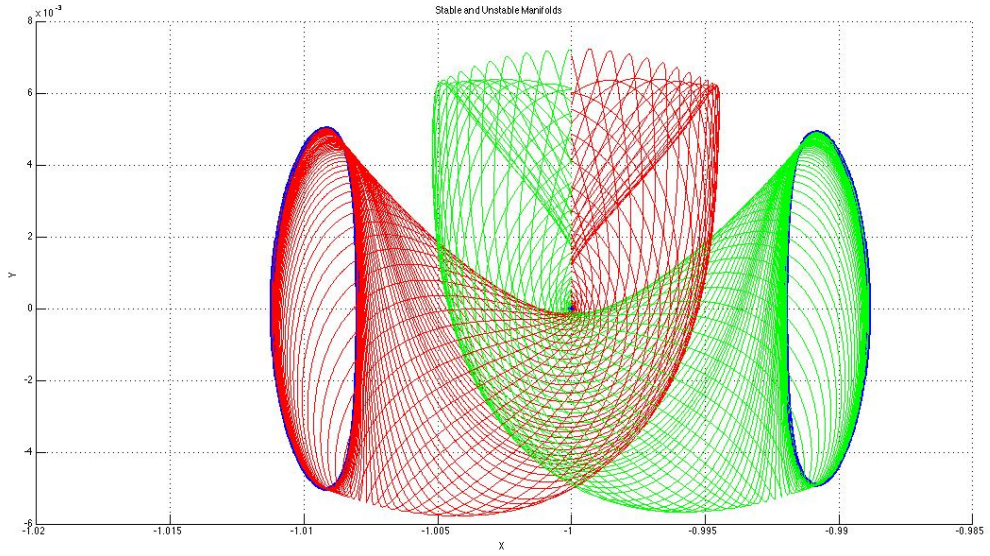

Figure 3.43 Stable and Unstable manifolds of Halo orbits (X-Y view)

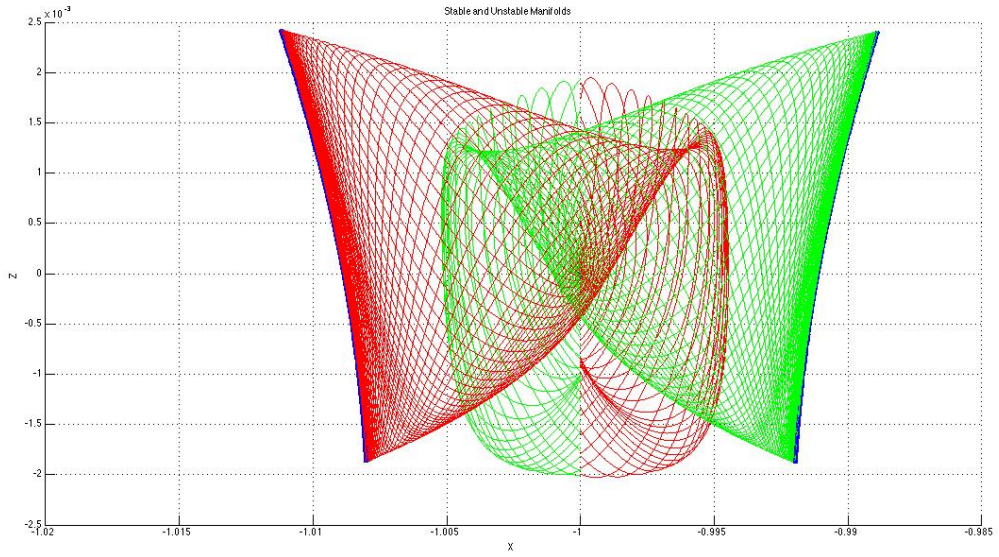

Figure 3.44 Stable and Unstable manifolds of Halo orbits (X-Z view)

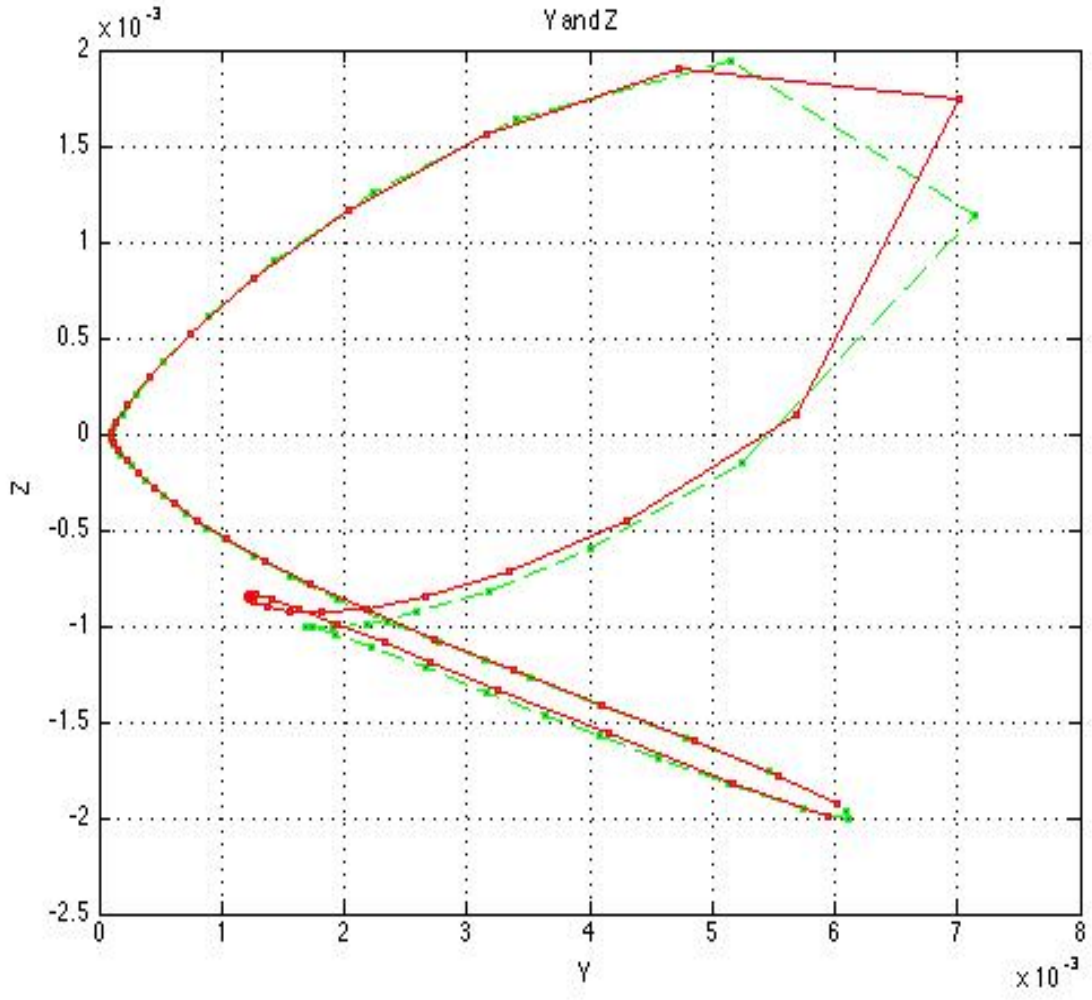

Figure 3.45 Intersection of Y and Z profiles

From figure 3.45 we can see that there are a big number of possibilities to maneuver. We can reach one orbit from another by adding computed delta V to a spacecraft velocity vector.

Now all results of our work are presented. The next chapter contains the summary of the project.

This Page Intentionally Left Blank

Chapter 4

Conclusion

Summarizing all data which has been obtained during the research we can say that the collocation method is very efficient for studying the neighborhood of the collinear libration points. The procedure is very cheap in terms of computational time from 1.5 to 5 seconds depending on criterion of convergence and accuracy of an initial seed. According to the fact that the accuracy of the collocation procedure depends on initial conditions, we can say that using the Lindstedt – Poincaré method is a good option. Looking at the results we can see that:

- the orbits which are computed using the collocation method describe natural motion in phase space around an equilibrium point very well;

- a state vector obtained with the collocation method gives very accurate data for propagation and prediction of the system behavior;

- the procedure can be used for following the family of orbits;

- orbits obtained by collocation procedure can be used for studying stable and unstable manifolds;

- heteroclinic and homoclinic trajectories have been computed.

This method can be used for the mission design when, for example, you have only the state vector of an object entering a neighborhood of a libration point of the chosen system and you need to know how a spacecraft or a probe will orbit the libtation point for an obtained period of time. Having this information, it is possible to propagate the trajectory to see stability of the orbit for a longer period of time and compute the point where it will be necessary to apply corrections to eliminate the tendency to fall out of the periodic orbit.

As a result of the work we provide several MATLAB scripts, which are located in appendix section of the paper. There are two scripts of collocation algorithm. The first script for the refining an orbit and a period and the second one for finding orbit with selected period of time. There is also a plotter with an implemented interpolation part and saver and loader to/from txt formatted file.

As a further research I can propose applying the method of collocation to refine homoclinic and heteroclinic trajectories. Using the method of collocation for computing connecting trajectories provides much higher accuracy results and saves computational time and resources.

This Page Intentionally Left Blank

Bibliography

[1] Roy Archie E., *The foundations of astrodynamics*, 1965, Macmillian, 385 p.

[2] Markeev A. P., *Libration points in celestial mechanics and astrodynamics,* 1974, Science, Moscow.

[3] Masdemont J. J., Jobra A., *Dynamics in the center manifold of the collinear points of the restricted three body problem*, Physica D 132 (1999), p 189 – 213.

[4] Canalias E., Gomez G., Marcote M. and Masdemont, J. J., *Assessment of Mission Design Including Utilization of Libration Points and Weak Stability Boundaries,* UPC.

[5] Lidov M.L., *Theoretical Mechanics (1974),* Phismalit, Moscow, 2001, 478 p.

[6] Bruno A.D., *Restricted Three Body Problem*, Nauka, Moscow, 1990, 295 p.

[7] Murray C.D. and Dermot F., *Solar System Dynamics*, Cambrige, 1999, 68-71 pp.

[8] Kononenko A., “Five wonderful points”, *Science and Lif*, 1973, №1, 42-46 pp.

[9] Szebehely, V., 1967, *Theory of Orbits* (New York: Academic Press).

[10] Folta, D. and Beckman, M., 2003, Libration Orbit mission design: applications of numerical and dynamical methods. *Libration Point Orbits and Applications* (Singapore: World Scientific), pp. 85–113.

[11] Masdemont, J. J., 2005, 'High-order expansions of invariant manifolds of Libration point orbits with applications to mission design', *Dynamical Systems*, 20: 1, 59 — 113

[12] Egemen Kolemen, N. Jeremy Kasdin and Pini Gurfil, *Quasi-Periodic Orbits of the Restricted Three-Body Problem Made Easy,* Mechanical and Aerospace Engineering, Princeton.

[13] Howell K. C., 1984, ‘Three-Dimensional, Periodic, 'Halo' Orbits', Celestial Mechanics, Vol. 32, Issue 1, p. 53.

[14] Strogatz, S.H., 1994 *Nonlinear Dynamics and Chaos*, Perseus Books Publishing, L.L.C.

[15] Conley, C., 1968, *Low energy transit orbits in the restricted three body problem*, SIAM J. Appl. Math. 16(4), pp: 732–746.

[16] McGehee, R.P., 1969 *Some homoclinic orbits for the restricted three body problem*, Ph.D. thesis, University of Wisconsin, Madison, Wisconsin.

[17] Koon, W.S., Lo, M.W., Marsden, J.E., Ross, S.D., 2000, *Heteroclinic connections between periodic orbits and resonance transitions in celestial mechanics*, Chaos 10(2), pp: 427–469.

[18] Golushko S.K., Shapeev V.P., 2013, *The collocation method in the mechanics of isotropic plates*, RSA, Novosibirsk, Russia.

[19] Mallon N.J., 2002, *Collocation – a method for computing periodic solutions of ordinary differential equations*, Department of mechanical engineering Eindhoven University of Technology, the Netherlands.

[20] Uri M. Ascher, Robert M. Mattheij, Robert D. Russel. *Numerical solution of boundary value problems for ordinary differential equations*, Prentice Hall 1988, 224-226.

[21] Doedel E.J., Champneys, A.R., Fairgrieve, T.F., Kuznetsov, Y.A., Sandstede, B. and Wang, X. AUTO97: Continuation and bifurcation software for ordinary differential equations, User Manual, *ftp://ftp.cs.concordia.ca/pub/doedel/auto*, 1998, 13.

[22] Richardson, D., L.,1980, Analytic Construction of Periodic Orbits About the Collinear Points. *Celestial Mechanics* 22: 241-253.

[23] Doedel E. J., Tuckerman L. S., Numerical Methods for Bifurcation Problems and Large-Scale Dynamical Systems (The IMA Volumes in Mathematics and its Applications), Springer, 1st Edition, 2000, 471 pp.

Appendix A

Here we attach the collocation algorithms (Refining of the trajectory and period and fixed period refining).

```
function [x, t, T, JC, Si, Ti] = GLcollocation(x0, T0, N, QUAD, mu, ANCH, diffeps)

tol = 1e-12;
fi=[];
ffun = @CollRTBP;
%setting some variabeles
n = size(x0,2);             %Dimenson of the system
I = eye(n);                 %Identity matrix of size [n,n]
T = T0;                     %initial guess for period time
x = [];                     %variable for state nodes [N,n]
h = 1/N;                    %time step for mesh (scaled)
t = linspace(0,1,N+1)';  %time nodes (scaled)
Norm=1e99; Normmax = 40; stop=0; conv=1;
options=odeset('Reltol',tol,'Abstol',tol);

[rho,alpha,beta]=Gquad(QUAD);

%setting arrays
np=max(size(rho));      %Number of points
D=[];               %D matrix for local parameter elimination
B=[];               %B matrix for determine xij
B2=[];

for i=1:np
   D=[D I*beta(i)];
   br=[];
   for j=1:np
      br=[br alpha(i,j)*I];
   end
   B=[B; br];
   B2=[B2; I];
end

%Colocation points tij
tij=[];
 for i=1:max(size(t))-1
   tij=[tij t(i)+h*rho];
 end

 %Integration from x0 for initial guess
 tic
 disp('* Integration from x0 for initial guess')
```

```
[ti,xi] = ode45(ffun,[0,T],x0,options, 0,mu);

%Scale time (tau=t/T)

  ti=ti./T;
%determine values xij for initial guess
  for i=1:n
     xij(:,i) = interp1(ti,xi(:,i),tij','spline');
  end
 xij=xij.';

 %setting up the initial mesh xi
  for i=1:n
     x(:,i) = interp1(ti,xi(:,i),t,'spline');
  end

  x=x.';

 %derivatives fi for initial guess
  fi=[];
  for i=1:max(size(tij))
      fi  = [fi; T*feval(ffun,0,xij(:,i), 0, mu)]; %derivative
  end

%collocation

NM=[];  %empty array for norm
i=0;    %Iteration counter
Si =[];
Ti =[];
while (Norm>diffeps) & (stop == 0)

    i=i+1;              %Iteration counter
    H = zeros(n*(N+1)+1); %variable for set of equations
    R  = [];            %rhs
    FP=[];               %variable for update data for fi

  for pp=1:N

      E  = zeros(np*(n+1));
      V  = [];
      U  = [];
      qi = [];
      Q  = [];
      s1 = n*np*(pp-1);

      %Determine xij
      xij=B2*x(:,pp)+h*B*fi(s1+1:s1+np*n);
```

```
    %determine f(xij), df(xij)/dx
    fxi=[];
    dfxi=[];
    for j=1:np
      fxi  = [fxi  feval(ffun,0,xij(((j-1)*n+1):j*n,:), 0, mu)]; %derivative
      dfxi = [dfxi feval(ffun,0,xij(((j-1)*n+1):j*n,:), 1, mu)]; % Jacobian(6x6)
    end

    %Collocation equations
    for qq=1:np                  %for each coll. point
      s2 = n*(qq-1)+1;
      A  = T*dfxi(:,s2:n+s2-1);   %A(tij,xij)
      V  = [V; A];
      U  = [U; fxi(:,qq)];
      qi = [qi; T*fxi(:,qq) - fi(s1+s2:s1+s2+n-1,:)];

      for rr=1:np                %column
        s3 = n*(rr-1)+1;
        Q(s2:s2+n-1,s3:s3+n-1) = alpha(qq,rr)*A;
      end
    end

    %Local parameter elimination
    W=eye(np*n)-h*Q;
    iW=inv(W);
    GAMMA=I+h*D*iW*V;
    LAMBDA=h*D*iW*U;
    R=[R; h*D*iW*qi];          %column with ri;
    FP=[FP; iW*V iW*U iW*qi];   %storage for update fi
    rowH=(pp-1)*n+1:pp*n;
    H(rowH,rowH)=-GAMMA;
    H(rowH,rowH+n)=I;
    H(rowH,(N+1)*n+1)=-LAMBDA;
  end

  %Boundary equations
  H(N*n+1:(N+1)*n,1:n)=I;
  H(N*n+1:(N+1)*n,N*n+1:(N+1)*n)=-I;

  %Residual
  R=[R; x(:,N+1)-x(:,1); 0];

%Anchor equation (orthogonality condition on x0')
  if ANCH==1
    f1=feval(ffun,0, x(:, 1), 0, mu);
    H((N+1)*n+1,1:n)=f1';
   elseif ANCH==2
    H((N+1)*n+1,1)=1;
  end
 %Create sparse matrix
```

```
  H=sparse(H);

  %Solve linear system
  w = H\R;

  wi = w(1:n*(N+1));
  wi = reshape(wi,n,N+1);
  dT = w(n*(N+1)+1);

  %Update variables

   x = x + wi;
   T = T + dT;

  for rr=1:N       %Update fi with eq.
    s=np*n*(rr-1);

fi(s+1:s+n*np,:)=fi(s+1:s+n*np,:)+FP(s+1:s+n*np,1:n)*wi(:,rr)+FP(s+1:s+n*np,n+1)*dT
+FP(s+1:s+n*np,n+2);
  end

  %Store values of each iterations Si and Ti
  Sii=x.';
  Si = [Si, Sii];
  Tii = T;
  Ti = [Ti; Tii];

  %Determine |w| for converge check
  NormOld = Norm;
  Norm = norm(w);
  NM=[NM Norm];

  disp(sprintf('* Iteration : %d, T = %0.7g, Norm = %0.7g',i,T,Norm))

 if   Norm > Normmax
     disp('* not convergent, norm is higher than 40')
       stop = 1;
       conv = 0;
 end
end                 %end of while

%rescale t
t=t*T; x=x.';

xJ = x(length(x),:);
xJ = xJ.';
JC = computeJacobi (xJ, mu);
disp(sprintf('* Elapsed time = %0.4g',toc))

end
```

```
function [x, t, T, JC, EV] = FPcollocation(x0, T, N, QUAD, mu, ANCH, diffeps)

tol = 1e-12;
fi=[];
EV=[];
ffun = @CollRTBP;
%setting some variabeles
n = size(x0,2);          %Dimenson of the system
I = eye(n);              %Identity matrix of size [n,n]
x = [];                  %variable for state nodes [N,n]
h = 1/N;                 %time step for mesh (scaled)
t = linspace(0,1,N+1)';  %time nodes (scaled)
Norm=1e99; Normmax = 40; stop=0; conv=1;
options=odeset('Reltol',tol,'Abstol',tol);

[rho,alpha,beta]=Gquad(QUAD);

%setting arrays
np=max(size(rho));      %Number of points
D=[];                   %D matrix for local parameter elimination
B=[];                   %B matrix for determine xij (2.12)
B2=[];

for i=1:np
   D=[D I*beta(i)];
   br=[];
   for j=1:np
      br=[br alpha(i,j)*I];
   end
   B=[B; br];
   B2=[B2; I];
end

%Colocation points tij
tij=[];
 for i=1:max(size(t))-1
   tij=[tij t(i)+h*rho];
 end

 %Integration from x0 for initial guess
 tic
 disp('* Integration from x0 for initial guess')

 [ti,xi] = ode45(ffun,[0,T],x0,options, 0,mu);

 %Scale time (tau=t/T)

   ti=ti./T;
 %determine values xij for initial guess
   for i=1:n
```

```
    xij(:,i) = interp1(ti,xi(:,i),tij','spline');
  end
 xij=xij.';

 %setting up the initial mesh xi
  for i=1:n
    x(:,i) = interp1(ti,xi(:,i),t,'spline');
  end

  x=x.';

 %derivatives fi for initial guess
  fi=[];
  for i=1:max(size(tij))
     fi  = [fi; T*feval(ffun,0,xij(:,i), 0, mu)]; %derivative
  end


%collocation

NM=[];  %empty array for norm
i=0;    %Iteration counter
Si =[];
Ti =[];
while (Norm>diffeps) & (stop == 0)

    i=i+1;                    %Iteration counter
    H = zeros(n*(N+1)+1, n*(N+1)); %variable for set of equations
    R  = [];                  %rhs
    FP=[];                     %variable for update data for fi

  for pp=1:N

     E  = zeros(np*(n+1));
     V  = [];
     U  = [];
     qi = [];
     Q  = [];
     s1 = n*np*(pp-1);


     %Determine xij
     xij=B2*x(:,pp)+h*B*fi(s1+1:s1+np*n);

     %determine f(xij), df(xij)/dx
     fxi=[];
     dfxi=[];
     for j=1:np
       fxi  = [fxi  feval(ffun,0,xij(((j-1)*n+1):j*n,:), 0, mu)]; %derivative
       dfxi = [dfxi feval(ffun,0,xij(((j-1)*n+1):j*n,:), 1, mu)]; % Jacobian(6x6)
```

```
    end

    %Collocation equations
    for qq=1:np                 %for each coll. point
      s2 = n*(qq-1)+1;
      A  = T*dfxi(:,s2:n+s2-1);   %A(tij,xij)
      V  = [V; A];
      U  = [U; fxi(:,qq)];
      qi = [qi; T*fxi(:,qq) - fi(s1+s2:s1+s2+n-1,:)];

      for rr=1:np               %column
        s3 = n*(rr-1)+1;
        Q(s2:s2+n-1,s3:s3+n-1) = alpha(qq,rr)*A;
      end
    end

    %Local parameter elimination
    W=eye(np*n)-h*Q;
    iW=inv(W);
    GAMMA=I+h*D*iW*V;
    %LAMBDA=h*D*iW*U;
    R=[R; h*D*iW*qi];           %column with ri;
    FP=[FP; iW*V iW*U iW*qi];   %storage for update fi
    rowH=(pp-1)*n+1:pp*n;
    H(rowH,rowH)=-GAMMA;
    H(rowH,rowH+n)=I;
    %H(rowH,(N+1)*n+1)=-LAMBDA;
  end

  %Boundary equations
  H(N*n+1:(N+1)*n,1:n)=I;
  H(N*n+1:(N+1)*n,N*n+1:(N+1)*n)=-I;

  %Residual
  R=[R; x(:,N+1)-x(:,1);0];

  %Anchor equation (orthogonality condition on x0')
  if ANCH==1
    f1=feval(ffun,0, x(:, 1), 0, mu);
    H((N+1)*n+1,1:n)=f1';
   elseif ANCH==2
    H((N+1)*n+1,1:n)=x0;
  end

  %Create sparse matrix
%   H=sparse(H);

  %Solve linear system
  w = H\R;
```

```
 wi = w(1:n*(N+1));
 wi = reshape(wi,n,N+1);


 %Update variables

  x = x + wi;       %


 for rr=1:N       %Update fi
   s=np*n*(rr-1);
   fi(s+1:s+n*np,:)=fi(s+1:s+n*np,:)+FP(s+1:s+n*np,1:n)*wi(:,rr)+FP(s+1:s+n*np,n+2);
 end

 %Store values of each iterations Si and Ti

 Sii=x.';
 Si = [Si, Sii];
 Tii = T;
 Ti = [Ti; Tii];

 %Determine |w| for converge check
 NormOld = Norm;
 Norm = norm(w);
 NM=[NM Norm];

 disp(sprintf('* Iteration : %d, T = %0.7g, Norm = %0.7g',i,T,Norm))

 if   Norm > Normmax
    disp('* not convergent, norm is higher than 40')
      stop = 1;
      conv = 0;
 end
end                 %end of while

%rescale t

t=t*T; x=x.';

xJ = x(length(x),:);
xJ = xJ.';
JC = computeJacobi (xJ, mu);

disp(sprintf('* Elapsed time = %0.4g',toc))

end
```

Appendix B

Here the functions used for collocation procedures and our vector field are presented.

```
function [rho,alpha,beta]=Gquad(n)
%returns scaled Gauss-Legendre quadrature butcher array
%rho   = vector with points
%alpha = [nxn] matrix
%beta = vector with weights
%first get the roots and weights
[rts,beta] = GLNodeWt(n);
%Next determine matrix alpha
syms x
a=-1; alpha=[];
%beta=[];
for i=1:n
   L=1;
   for ii=1:n
      if ii~=i
         L=L*(x-rts(ii))/(rts(i)-rts(ii));
      end
   end
    for iii=1:n
      alpha(iii,i)=int(L,x,a,rts(iii));
    end
   %beta(i)=int(L,x,a,1);
end
%scale integration limits from [-1,1] to [0,1]
rho=(rts'+1)/2; alpha=alpha*0.5; beta=beta*0.5;
end
```

```
function [x,w] = GLNodeWt(n)
% GLNodeWt Nodes and weights for Gauss-Legendre quadrature of arbitrary
%          order obtained by solving an eigenvalue problem
%
% Synopsis:  [x,w] = GLNodeWt(n)
%
% Input:
%
% Output:
%
%  Algorithm based on ideas from Golub and Welsch, and Gautschi.  For a
%  condensed presentation see H.R. Schwarz, "Numerical Analysis: A
%  Comprehensive Introduction," 1989, Wiley.  Original MATLAB
%  implementation by H.W. Wilson and L.H. Turcotte, "Advanced Mathematics
%  and Mechanics Applications Using MATLAB," 2nd ed., 1998, CRC Press
```

```
beta = (1:n-1)./sqrt(4*(1:n-1).^2 - 1);
J = diag(beta,-1) + diag(beta,1);   % eig(J) needs J in full storage
[V,D]  = eig(J);
[x,ix] = sort(diag(D));             %nodes are eigenvalues, which are on diagonal of D
w = 2*V(1,ix)'.^2;                  %V(1,ix)? is column vector of first row of sorted V
end
```

```
function U = CollRTBP (t, x, Joption, mu)

Xmu = (x(1)+1-mu);
xmu = (x(1)-mu);
MU  = (1-mu);

r13 = (sqrt(xmu^2 + x(2)^2 + x(3)^2))^3;
r23 = (sqrt(Xmu^2 + x(2)^2 + x(3)^2))^3;

r15 = (sqrt(xmu^2 + x(2)^2 + x(3)^2))^5;
r25 = (sqrt(Xmu^2 + x(2)^2 + x(3)^2))^5;

Ux = x(1) - ((MU*xmu) /(r13))  - ((mu*Xmu) /(r23));
Uy = x(2) - ((x(2)*MU)/(r13))  - ((mu*x(2))/(r23));
Uz =      - ((x(3)*MU)/(r13))  - ((mu*x(3))/(r23));


% VECTOR FIELD

xdot = zeros(6,1);

xdot(1) = x(4);
xdot(2) = x(5);
xdot(3) = x(6);
xdot(4) = Ux + (2*x(5));
xdot(5) = Uy - (2*x(4));
xdot(6) = Uz;

U = xdot;


if Joption==1 %f(x)/dx jacobian


   Uxx = ((3*MU*(xmu^2))  /(r15))  + ((3*mu*(Xmu^2))  /(r25))   - (MU/(r13)) - (mu/(r23))
+ 1;
   Uxy = ((3*x(2)*MU*xmu) /(r15))  + ((3*mu*x(2)*Xmu) /(r25));
   Uxz = ((3*x(3)*MU*xmu) /(r15))  + ((3*mu*x(3)*Xmu) /(r25));

   Uyx = ((3*x(2)*MU*xmu) /(r15))  + ((3*mu*x(2)*Xmu) /(r25));
```

```
   Uyy = ((3*(x(2)^2)*MU) /(r15))  + ((3*mu*(x(2)^2)) /(r25))   - (MU/(r13)) - (mu/(r23)) +
1;
   Uyz = ((3*x(2)*x(3)*MU)/(r15))  + ((3*mu*x(2)*x(3))/(r25));

   Uzx = ((3*x(3)*MU*xmu) /(r15))  + ((3*mu*x(3)*Xmu) /(r25));
   Uzy = ((3*x(2)*x(3)*MU)/(r15))  + ((3*mu*x(2)*x(3))/(r25));
   Uzz = ((3*(x(3)^2)*MU) /(r15))  + ((3*mu*(x(3)^2)) /(r25))   - (MU/(r13)) - (mu/(r23));


xdot = zeros(6,6);
xdot(1,4) = 1;
xdot(2,5) = 1;
xdot(3,6) = 1;
xdot(4,1) = Uxx;
xdot(4,2) = Uxy;
xdot(4,3) = Uxz;
xdot(4,5) = 2;
xdot(5,1) = Uyx;
xdot(5,2) = Uyy;
xdot(5,3) = Uyz;
xdot(5,4) = -2;
xdot(6,1) = Uzx;
xdot(6,2) = Uzy;
xdot(6,3) = Uzz;

U = xdot;

%      x       y       z     Vx   Vy   Vz
% U = [ 0     0       0      1    0    0;
%         0     0       0      0    1    0;
%         0     0       0      0    0    1;
%       Uxx   Uxy   Uxz     0    2    0;
%       Uyx   Uyy   Uyz    -2    0    0;
%       Uzx   Uzy   Uzz     0    0    0];

end
end
```

This Page Intentionally Left Blank

Appendix C

In this section additional scripts are provided.

Collocation.m

```
clear all; close all; clc;

name = input('Enter the name of the file to read: ');

name2 = input('Enter the name of the file to save: ');

[T, X, P0, mu, N, QUAD, JC] = loaddata(name);

sd = 1e-1;

IC = X(1,:);
conv = 1e-9;
ANCH = 1;
[x, t, P, JC] = GLcollocation(IC, P0, N, QUAD, mu, ANCH, conv);

%Number of points to plot
m = 250;
%Animation
A=0;
[ti, xi] = plotter(t, x, P, m, mu, A);

savedata(t, x, QUAD, mu, JC, name2)
```

```
function [t, x, P, mu, N, QUAD, JC] = loaddata(name)

tx = load(name);

t = tx(:,2);
x = tx(:,3:8);
mu = tx(1,1);
N = (length(t)-1);
P = t(length(t));
QUAD = tx(2,1);
JC = tx(3,1);
end
```

```
function savedata(t, x, QUAD, mu, JC, name)

Mu = zeros(length(t),1);
```

```
Mu(1,1) = mu;
Mu(2,1) = QUAD;
Mu(3,1) = JC;
%save orbit data
tx = [Mu, t, x];
save(name,'tx','-ascii');

end
```

```
function [ti, xi]=plotter(t, x, P, m, mu, A)

ti=[];xi=[];

[ti, xi] = intersolution(t, x, P, m);

figure
subplot(3,1,1); plot(ti,xi(:,1));grid; title('Coordinates over time'); xlabel('Time'); ylabel('X');
subplot(3,1,2); plot(ti,xi(:,2));grid; xlabel('Time'); ylabel('Y');
subplot(3,1,3); plot(ti,xi(:,3));grid; xlabel('Time'); ylabel('Z');


figure
subplot(3,1,1); plot(ti,xi(:,4));grid;title('Velocities over time'); xlabel('Time'); ylabel('Vx');
subplot(3,1,2); plot(ti,xi(:,5));grid; xlabel('Time'); ylabel('Vy');
subplot(3,1,3); plot(ti,xi(:,6));grid; xlabel('Time'); ylabel('Vz');


figure
plot3(xi(:,1),xi(:,2),xi(:,3),'r');grid; title('Periodic orbit'); xlabel('X'); ylabel('Y');zlabel('Z')

figure
subplot(2,2,1); plot3 (xi(:,1),xi(:,2),xi(:,3),'r'); axis vis3d; grid; title(sprintf('Isometric view')); xlabel('X'); ylabel('Y');zlabel('Z')
subplot(2,2,2); plot (xi(:,1),xi(:,2),'r'); axis vis3d; grid;title(sprintf('X-Y view')); xlabel('X'); ylabel('Y');
subplot(2,2,3); plot (xi(:,1),xi(:,3),'r'); axis vis3d; grid;title(sprintf('X-Z view')); xlabel('X'); ylabel('Z');
subplot(2,2,4); plot (xi(:,2),xi(:,3),'r'); axis vis3d; grid;title(sprintf('Y-Z view')); xlabel('Y'); ylabel('Z');

figure
plot3(xi(:,1),xi(:,2),xi(:,3),'r');grid; title('Periodic orbit'); xlabel('X'); ylabel('Y');zlabel('Z')
hold on
plot3(mu-1,0,0,'marker','o','MarkerFaceColor','b')

% figure
% quiver3(x(:,1),x(:,2),x(:,3),x(:,4),x(:,5),x(:,6));title('Periodic orbit with velocities direction'); xlabel('X'); ylabel('Y');zlabel('Z')
```

```
if A==1
Animations(xi, length(xi));
ends
end
```

```
function [xsts, NXs, xuns, NXu, Jacobi_Constant, Cjst, Cjun] = Homoclinic_function(R,
V, P, m)

format long

%-------- Variables ------------

NXs = [];
NXss = [];
xstlength = [];
xsts = [];
NXu = [];
xuns = [];
xunlength = [];
Cjst =[];
Cjun =[];
Cjstin =[];

%-------- Constants --------------------------

pi = 3.1415926535897932384;
e  = 2.7182818284590452353;
AU = 149597870.7;
RAD= pi/180;
mu = 3.040423398444176e-6;
Mu = mu - 1;                                % Position of the Earth
OPTIONS = odeset('RelTol',1.e-12,'AbsTol',1.e-12); % Options for propagation

%------- Initial conditions ----------------------------

I = eye(6); STM0 = matrixtovector(I); % IC for STM

IC = [R V STM0];


%------- Propagation -----------------------------

[t,x] = ode45(@RTBPVF,[0 P],IC,OPTIONS,mu);


figure()
plot3(x(:,1),x(:,2),x(:,3),'r');grid;      title('Propagation      of      orbit');      xlabel('X');
ylabel('Y');zlabel('Z');
```

```
%------------ Forming Monodromy matrix ----------------

X = x(:,[1:6]);      % State vector
STM = x(:,[7:42]);   % Set of STMs in vector form

JC = computeJacobi (X(1,:), mu);
M = statetransitionmatrix(STM(length(STM),:));  % Momodromy matrix


%------------ Eigenvalues and eigenvector --------------

lambda = eig(M);    % Eigenvalues L1
[eV, eD] = eig(M);  % Eigenvector and eigenvalues as diagonal matrix


[lambdamin, pos] = min(lambda);          % The smallest eigenvalue and its position
Vst = eV(:,pos);                         % Eigenvector corresponding to the smallest
eigenvalu

[lambdamax, pos] = max(lambda);          % The largets eigenvalue and its position
Vun = eV(:,pos);                         % Eigenvector corresponding to the largest
eigenvalue

%------- Choose solutions for needed amount of points ----------------------

[ti, xi]=stepsolution(t, x, m);


XI = xi(:,[1:6]);      % State vector
STMI = xi(:,[7:42]);   % Set of STMs in vector form


figure()
plot3(XI(:,1),XI(:,2),XI(:,3),'r');grid;  title('Propagation of orbit');  xlabel('X');
ylabel('Y');zlabel('Z');

%---------- Forming STM matrixs for points  -------------
S = cell(1,length(ti));


for ii=1:length(ti)

   S{ii}=statetransitionmatrix(STMI(ii,:));

end

%---------- Normaliztion ---------------------

for ii = 1:length(ti)
```

```
Si=cell2mat(S(1,ii));    % Taking STM(t) (6 x 6)

Wst = Si*Vst;                    % Shift the position for each STM (6 x 1) stable
Wun = Si*Vun;                    % Shift the position for each STM (6 x 1) unstable

%----------- Stable L1
Lst = norm(Wst);
Vsti = Wst/Lst;

%----------- Unstable L2
Lun = norm(Wun);
Vuni = Wun/Lun;

%----------- New Initial conditions -------

eps = 200/AU;
if ii < (length(ti)-8)
   NXst = XI(ii,:) - eps*Vsti.';
else
   NXst = XI(ii,:) + eps*Vsti.';
end
NXs = [NXs; NXst];

NXun = XI(ii,:) - eps*Vuni.';          % '+' or '-' depending on the orbit

NXu = [NXu; NXun];

Cj = computeJacobi (NXun, mu);

Cjstin =[Cjstin; Cj];

end

%----------- Propagation of New initital conditions ----------------------

figure()
plot3(x(:,1),x(:,2),x(:,3),'b','LineWidth',4);grid;  title('Stable  and  Unstable  Manifolds');
xlabel('X'); ylabel('Y');zlabel('Z')
hold on
plot3(Mu,0,0,'marker','o','MarkerFaceColor','b')

for ii = 1:length(ti)
% create an options variable
flag = 0;

%---------------------------------- Stable manifold (propagates backward)

options = odeset('RelTol',1e-12,'AbsTol',1e-12,'Events',@event_function_stable);
```

```
[tst,xst, TE, VE] = ode45(@RTBP,[0 -3*P],NXs(ii,:),options,mu);

TE; % this is the time value where the event occurred
VE; % this is the value of X where the event occurred

plot3(xst(:,1),xst(:,2),xst(:,3),'g');

iteration = ii

xstlength = [xstlength; length(xst)];

xsts = [xsts; xst(length(xst), :)];

Cj = computeJacobi (xsts, mu);

Cjst =[Cjst; Cj];

%------------------------- Unstable manifold (propagates forward)

options = odeset('RelTol',1e-12,'AbsTol',1e-12,'Events',@event_function_unstable);

[tun,xun,TE,VE] = ode45(@RTBP,[0 10*P],NXu(ii,:),options,mu);

TE; % this is the time value where the event occurred
VE; % this is the value of X where the event occurred

NSV{ii} = xun; % Save each iteration

plot3(xun(:,1),xun(:,2),xun(:,3),'r');


xunlength = [xunlength; length(xun)];

xuns = [xuns; xun(length(xun), :)];

Cjs = computeJacobi (xuns, mu);

Cjun =[Cjun; Cjs];

end

figure()
plot(xsts (:,2),xsts(:,5),'-gs',...
   'LineWidth',1,...
   'MarkerSize',2,...
   'MarkerEdgeColor','g',...
   'MarkerFaceColor',[0.3,0.2,0.5]); grid; title('Y and Ydot'); xlabel('Y'); ylabel('Ydot');
hold on
plot(xuns(:,2),xuns(:,5),'-rs',...
   'LineWidth',1,...
```

```
    'MarkerSize',2,...
    'MarkerEdgeColor','r',...
    'MarkerFaceColor',[0.5,0.5,0.5]);
end
```