POLITECNICO DI TORINO

Department of Mechanical and Aerospace Engineering (DIMEAS)

Politecnico di Torino

EMBRY-RIDDLE Aeronautical University
PRESCOTT, ARIZONA

Master's Degree in Aerospace Engineering

High-Precision Computation and Classification of Periodic Orbits in the Circular Restricted Three-Body Problem

Supervisors:
Prof. Lorenzo Casalino
Dr. Davide Conte

Candidate:
Jacopo Nazzaro

A.Y. 2024-2025

Acknoledgements

First of all, I would like to thank my parents and my sister. You are the ones who gave me the opportunity to study far from home, at a “prestigious” university, and allowed me to experience this journey as if all of this were normal, without ever making me feel its cost in any sense. If you are holding this thesis today and I am holding a degree, it is above all thanks to you. Each of you, in your own way, has made me feel supported over these years, always trying to offer me thoughtful advice, even when I was not particularly keen to hear it. Without your support, your guidance and your constant encouragement to expect something more from myself, I would very likely not be here and would not be the person I am today. This achievement, for better or for worse, is something we share.

I would also like to thank my friends, both those I have known forever and those I met during my university years. You have helped me keep a balance between study and real life, between commitments and lightness, and you have made these years less rigid and more genuinely lived. Knowing that I had people around me with whom I could laugh, vent, or simply sit in silence has mattered just as much as any academic result.

I also wish to express my sincere gratitude to Professor Lorenzo Casalino for having accepted and supervised this thesis work. Having the opportunity to work on a topic that is closely aligned with my interests, within a structured and rigorous framework, is something I do not take for granted.

A special thank you goes to Professor Davide Conte. I am grateful to him first of all for giving me the opportunity to spend a research period at Embry-Riddle Aeronautical University in the United States, an experience that has had a concrete impact both on the content of this thesis and on my personal development. During that time, he was a constant point of reference for me: his presence was both reassuring and inspiring. Working with him has been truly formative, from both a technical and a human point of view; he has been a mentor with whom it was natural to collaborate, able to make people feel at ease while at the same time raising the bar. His support, his willingness to engage in discussion and the way he stands behind his students’ work have turned the thesis experience into something I can genuinely be proud of.

To all the people who, in different ways, have contributed to this journey, I owe my gratitude, even if it is not always easy to fully express it on a single page.

Abstract

This thesis falls within the scope of the restricted three-body problem (CR3BP), a dynamic model capable of describing, with realism and manageable complexity, the motion of a body of negligible mass subject to the gravitational attraction of two primary bodies in circular orbit. The CR3BP is now a fundamental reference for the preliminary analysis of missions in multi-body environments, widely used in scientific and exploration missions. Despite the existence of the NASA Jet Propulsion Laboratory (JPL) public catalog of periodic orbits, this resource is limited to a finite set of precomputed orbital systems and families, and does not allow for the autonomous generation of new solutions or the easy extension of the database to other three-body systems.

The objective of the thesis is to develop a numerical tool, implemented in Python and equipped with a graphical user interface (GUI), which acts as an intermediate link between the NASA JPL database and the more complex proprietary mission design environments. The tool allows for high-precision calculation, classification, and analysis of periodic orbits in the CR3BP, replicating and expanding the functionality of the JPL dynamic catalog, and making it possible to define new three-body systems, generate initial conditions for new orbits, and correct them towards periodicity.

After reviewing the equations of motion in the synodic system, the Lagrangian points, their stability framework, and the role of the effective potential and Jacobi constant, the thesis analyzes the main orbital families considered in the tool: Lyapunov orbits, Distant Retrograde Orbits (DRO), Short and Long Period Orbits, Halo, Vertical and Axial orbits. For each family, the geometric aspects and global stability indices are studied as a function of the dynamic parameters.

The methodological core of the work is represented by dynamic systems theory and differential correction algorithms. Starting from variational equations, the State Transition Matrix (STM) is derived, which is used to formulate and solve two-point boundary value problems using shooting methods. A single shooting algorithm, in which the correction acts on a single initial state vector, and a multiple shooting algorithm, which divides the orbit into several arcs and introduces intermediate state vectors as correction variables, are implemented and compared. A numerical campaign in the Earth–Moon system shows that single shooting is particularly efficient for planar or highly symmetric orbits, while multiple shooting is more robust for complex three-dimensional orbits.

To derive the initial conditions of Halo orbits, Richardson's analytical approximation is first implemented, which provides third-order solutions for the initial conditions starting from a prescribed vertical amplitude; its accuracy is sufficient in some cases, but not always adequate to guarantee the convergence of the correction methods. To

overcome these limitations, the thesis introduces a Genetic Algorithm (GA) specifically designed to search for quasi-periodic initial conditions, using as a cost function the deviation from periodicity measured on the velocity components at the plane section. Initially validated on a Northern Halo of the Earth-Moon system, the GA is then applied to systems not covered by the JPL catalog, such as Pluto-Charon and the binary star system α Centauri AB, demonstrating the ability to identify initial conditions with a high probability of periodicity even in the absence of reference solutions.

A further original contribution is the development of an algorithm for estimating the surface coverage of a celestial body by a probe in periodic orbit. The surface is discretized using a spherical Fibonacci lattice, which distributes an arbitrary number of points almost uniformly on the sphere, while the field of view of the onboard instrument is translated into an imprint angle and a visibility criterion formulated in matrix terms. The procedure returns the fraction of the surface observed and an estimate of the numerical error, which decreases with the square root of the number of discretization points.

All the methodologies developed are integrated into a single software tool with a GUI, divided into modules dedicated to the orbit database, the definition of new systems, differential correction, the derivation of initial conditions, and coverage analysis. In conclusion, the thesis creates a flexible environment for high-precision calculation and classification of periodic orbits in the CR3BP, capable of extending and enhancing the NASA JPL database and supporting the preliminary phases of mission design in different scenarios.

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

2BP Two-Body Problem

3BP Three-Body Problem

ACE Advanced Composition Explorer

AGI Analytical Graphic Inc.

API Application Programming Interface

ARM Asteroid Redirect Mission

AU Astronomical Unit

CAPSTONE Cislunar Autonomous Positioning System Technology Operations and Navigation Experiment

CR3BP Circular Restricted Three-Body Problem

DART Double Asteroid Redirection Test

DROs Distant Retrograde Orbits

DST Dynamical Systems Theory

EoMs Equations of Motion

FOV Field Of View

FDAB Flight Dynamics Analysis Branch

GA Genetic Algorithm

GMAS Goddard Mission Analysis System

GSFC Goddard Space Flight Center

ISEE International Sun-Earth Explorer

JWST	James Webb Space Telescope
JPL	Jet Propulsion Laboratory
LY	Light Year
LPOs	Long Period Orbits
MAP	Microwave Anisotropy Probe
NRHOs	Near Rectilinear Halo Orbits
ODEs	Ordinary Differential Equations
SOHO	Solar Heliospheric Observatory
SPOs	Short Period Orbits
SLS	Space Launch System
STK	System Tool Kit
STM	State Transition Matrix
TPBVP	Two-Point Boundary Value Problem

Introduction

Modern space design increasingly operates in gravitational environments where more than two bodies are considered, and the hypothesis of the two-body problem is no longer sufficient. The addition of just a third body breaks down the Keplerian integrability and introduces several non-linearities. In this context, the *Circular Restricted Three-Body Problem* (CR3BP) hypothesis is important because it is the simplest model capable of preserving the essential dynamical structures of a three-body system, while mantaining a manageable complexity. In fact, this hypothesis eliminates the need of complete ephemerides, reduces the number of parameters and enables the analysis in non-dimensional units as discussed later. Thus, the CR3BP is the basis of many missions, including those operated in the lunar environment such, as the Cislunar Autonomous Positioning System Technology Operations and Navigation Experiment (CAPSTONE) [1]; and missions like the Double Asteroid Redirection Test (DART) [2], OSIRIS-REx [3] and Hayabusa [4], which are designed for the observation and sample collection from the surfaces of asteroids in the vicinity of the Earth.

The CR3BP has been the subject of numerous studies over the past 250 years, with contributions from illustrious figures like Euler, Lagrange and Jacobi. Due to the technological limitations at the time, first attempts focused mainly on finding an analytical solution to the problem. These attempts soon resulted in the identification of just one integral of motion, which was later demostrated to be the only integral of motion that could be derived for the CR3BP. In the following years, the advent of more reliable numerical methods made it possible to solve the equations of motion of the CR3BP, first computing the location of the equilibrium points of a three-body system and then deriving the initial conditions of the so-called periodic orbits. The identification of the equilibrium, or libration, points, has made it possible to carry out missions whose main objective has been scientific investigations into the physics of stars and galaxies.

One example of a mission of this kind, which made use of the three-body dynamics, is the *International Sun-Earth Explorer* (ISEE-3). This mission was the first to be designed using a combination of analytical and numerical methods to predetermine the reference libration orbit, along with proven operational numerical techniques for targeting and optimization [5, 6]. ISEE-3 was designed and supported by the *Flight Dynamics Analysis Branch* (FDAB) of the *Goddard Space Flight Center*, which used the reference software for all missions designed before 1990, namely the *Goddard Mission Analysis System* (GMAS). GMAS was the first software capable of propagating complex orbits with realistic perturbations models, incorporating also attitude control functions, evaluation of eclipse periods and ground coverage. Furthermore, it enabled user to connect proprietary modules to perform customized analyses [7]. GMAS was widely employed to design periodic orbits and advanced configurations in the CR3BP

before the development of new generation softwares. One of the main limitations of GMAS was its implementation on a mainframe computer, which at the time was the only resource capable of computing high accuracy and high fidelity trajectories for orbits around the libration points.

From the mid-1990s onwards, GMAS was replaced by Swingby [8, 9], a new operational PC program developed by GSFC. Thanks to this new trajectory and mission design software, it became possible to carry out new missions based on orbits built around the libration points, such as the *Solar Heliospheric Observatory* (SOHO), the *Advanced Composition Explorer* (ACE), the *Microwave Anisotropy Probe* (MAP), and *Genesis*. The main new feature was the interactive graphical interface, which provided instantaneous feedback on the trajectory design in multiple reference frame. Other innovations included the ability to change physical constants to describe different three-body systens and select the perturbations to be included into the simulation. Swingby was designed to act as a generic tool capable of supporting missions of various natures. Thanks to its numerical accuracy and the multiple mission design functions it was equipped with, Swingby became the new standard in mission design and formed the basis for the development of new tools like *Astrogator*.

Astrogator is, more precisely, a tool from Analytical Graphics Inc.'s (AGI's) Systems Tool Kit (STK). This tool has been used to design missions like *Arcus* [10], ARTEMIS [11] e *James Webb Space Telescope* (JWST) [12]. It is evident that STK and its tools have become the new reference point for the mission and trajectory design, giving a huge contribution to the planning of missions that will mark fundamental milestones in the scientific research and human exploration since the 2010s.

Besides these proprietary and commercial softwares, such as *STK/Astrogator*, few instruments deal with the fundamental topics related to the CR3BP, therefore enabling a preliminary phase of mission planning and trajectory design. The main resource is a database provided by NASA. Such database, as will be highlighted hereafter, guarantees free access to data related to several families of periodic orbits. However, rather than being a truly useful tool for mission and trajectory design, this database can be seen as a complement to software that performs these tasks.

1.1 State of the art

Due to the growing interest in the space sector and space exploration, new products designed to enhance preliminary mission design are continuously released. Most of these, like those previously mentioned, provide extremely accurate tools for reliably designing missions. However, the majority of these products require licences upwards of several hundreds of thousand of dollars.

The next section discusses and analyzes the only instrument provided to the public free of charge. However, this too has significantly limited functions for mission and

trajectory design.

1.1.1 Database of three-body periodic orbits by NASA JPL

NASA's *Jet Propulsion Laboratory* (JPL) provides free access to a web database[1] of periodic orbits, computed in the context of the CR3BP, whose theory will be discussed in more detail in Chapter 2. This database, also known as *dynamical orbit catalog* contains 727,774 precomputed orbits in total and is updated whenever new orbits are derived. Periodic orbits that are already available in the catalog are derived using a robust multiple shooting differential correction algorithm based on a generalized constraint and free variable method within a pseudo-arclength continuation scheme. All stored data can be accessed via the web interface provided for users, shown in Figure 1.1 and the *Application Programming Interface* (API) service, which allows to obtain datas with the highest available resolution.

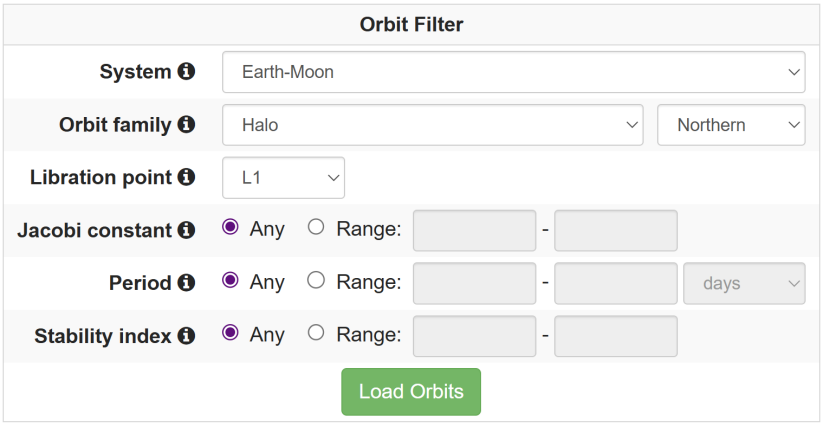


Figure 1.1: User interface to access the three-body periodic orbits database provided by NASA [13]

Using the graphic interface, the user can select a system amongst the seven available, which are:

- Sun-Earth system;
- Sun-Mars system;
- Earth-Moon system;
- Mars-Phobos system;
- Jupiter-Europa;
- Saturn-Enceladus;

[1] https://ssd.jpl.nasa.gov/tools/periodic_orbits.html#/periodic

- Saturn-Titan.

Additionally, the user can choose the family of periodic orbits of interest, and specify a few parameters to filter the database. These parameters include the Lagrangian point around which the orbit is built, the orbital period, the Jacobi constant, and the stability index. The output data are organized into two different sections. The first one containts two tables showing the physical properties of the system within the selected family of periodic orbits is evaluated. As shown in Figure 1.2 and Figure 1.3, one table reports the ratio of the masses of the two celestial bodies and the units used to non-dimensionalize the output data. The second table simply shows the location of the Lagrangian points of the system in non-dimensionalized units.

Libration Points			
	x (LU)	y (LU)	z (LU)
L1:	0.83691513	0.00000000	0.00000000
L2:	1.15568217	0.00000000	0.00000000
L3:	-1.00506265	0.00000000	0.00000000
L4:	0.48784941	0.86602540	0.00000000
L5:	0.48784941	-0.86602540	0.00000000

Figure 1.2: Lagrange points location provided by NASA JPL for the Earth-Moon system [13]

Earth-Moon System	
Mass ratio	1.215058560962404E-2
Length unit, LU (km)	389703
Time unit, TU (s)	382981
Moon radius (km)	1737.1

Figure 1.3: System physical properties provided by NASA JPL for the Earth-Moon system [13]

The second table gathers all data collected from the dynamical orbit catalog according to the user's selections. Each row of the table corresponds to a specific periodic orbit, characterised by the following parameters:

- an ID to identify the orbit;
- six elements that form the initial conditions vector;

- the Jacobi constant;
- the orbital period expressed in both non-dimensional unit and days;
- the stability index.

Figure 1.4a and Figure 1.4b show how these data are presented to the user. They break down the results table in order to offer a clearer view of the data it contains.

Id	x_0 (LU)	y_0 (LU)	z_0 (LU)	vx_0 (LU/TU)	vy_0 (LU/TU)	vz_0 (LU/TU)
1177	4.8784941344943100E-1	6.7212754533055685E-1	1.7393275630338904E-35	-2.4535031351852712E-1	1.3805858489246356E-1	-3.0357784458458681E-35
1176	4.8784941344943100E-1	6.7216923728902811E-1	-7.2825089275806241E-36	-2.4526519697961138E-1	1.3806761685537203E-1	-2.0648201120666586E-35
1175	4.8784941344943100E-1	6.7225250351835308E-1	3.6327183544224268E-35	-2.4509526852053465E-1	1.3808574702093177E-1	-8.0294529355816456E-36
1174	4.8784941344943100E-1	6.7233561532084207E-1	2.5949037794401997E-36	-2.4492574251405616E-1	1.3810396270628361E-1	-5.3815784419575166E-35
1173	4.8784941344943100E-1	6.7241857567449559E-1	3.285271382089387E-35	-2.4475661452281836E-1	1.3812226057131133E-1	5.5489578172550714E-35
1172	4.8784941344943100E-1	6.7250138754494959E-1	4.5840296317423360E-35	-2.4458788012162963E-1	1.3814063729918719E-1	6.7822717964584302E-35
1171	4.8784941344943100E-1	6.7258405388648068E-1	6.4910155203933610E-35	-2.4441953489616769E-1	1.3815908595556714E-1	1.3350677089464130E-34
1170	4.8784941344943100E-1	6.7266657764124527E-1	8.9776544334129049E-35	-2.4425157444421650E-1	1.3817761418895516E-1	1.4798224914104851E-34
1169	4.8784941344943100E-1	6.7274896173959375E-1	-8.4139681579951548E-36	-2.4408399437515671E-1	1.3819620783060588E-1	-2.3649247381865502E-36
1168	4.8784941344943100E-1	6.7283120909965788E-1	1.5448016150720334E-35	-2.4391679031072444E-1	1.3821486729460083E-1	8.3094409962681292E-35
1167	4.8784941344943100E-1	6.7291332262792491E-1	2.4540648625323044E-35	-2.4374995788432260E-1	1.3823358937738295E-1	7.0734454205769571E-35
1166	4.8784941344943100E-1	6.7299530521843409E-1	6.7007402429693656E-35	-2.4358349274205024E-1	1.3825237089837944E-1	1.6579245792394469E-35
1165	4.8784941344943100E-1	6.7307715975325422E-1	-6.1177134120607425E-36	-2.4341739054230577E-1	1.3827120869946330E-1	-3.3997868019672978E-35
1164	4.8784941344943100E-1	6.7315888910235666E-1	-3.4823239541392924E-35	-2.4325164695592086E-1	1.3829009964503541E-1	-2.3789617937742991E-35
1163	4.8784941344943100E-1	6.7324049612368675E-1	-3.4627194433434666E-35	-2.4308625766624406E-1	1.3830904062184401E-1	-4.8751056806461951E-35
1162	4.8784941344943100E-1	6.7089965235028037E-1	-1.3053042838190889E-34	-2.4786546846693094E-1	1.3780826788225226E-1	-1.9771272233655465E-34
1161	4.8784941344943100E-1	6.7081350092759440E-1	1.0213686743302540E-34	-2.4804244829934605E-1	1.3779197892824685E-1	1.1269243936142491E-34
1160	4.8784941344943100E-1	6.7072714271307576E-1	5.5750968899546872E-35	-2.4821990760151771E-1	1.3777583577148897E-1	8.2118565980506477E-35
1159	4.8784941344943100E-1	6.7064057452822623E-1	2.8516896020223668E-35	-2.4839785098692305E-1	1.3775984216284307E-1	4.5219798308194873E-36
1158	4.8784941344943100E-1	6.7055379318420183E-1	2.3640752598470888E-35	-2.4857628307703569E-1	1.3774400187573130E-1	9.2804360642458019E-37
1157	4.8784941344943100E-1	6.7046679548200916E-1	-3.0765442225904004E-35	-2.4875520850094354E-1	1.3772831870608485E-1	-1.2975804594147210E-35
1156	4.8784941344943100E-1	6.7037957821252891E-1	1.6013539136061244E-35	-2.4893463189529932E-1	1.3771279647227502E-1	8.7139763568440805E-35
1155	4.8784941344943100E-1	6.7029213815650279E-1	1.9812038928148076E-35	-2.4911455790424530E-1	1.3769743901516335E-1	-4.3078837386263623E-35
1154	4.8784941344943100E-1	6.7401020078783935E-1	6.3135893418161220E-35	-2.4153191245668412E-1	1.3849089336569220E-1	3.6376591041157267E-35
1153	4.8784941344943100E-1	6.7409069282507805E-1	2.6682091683609122E-35	-2.4136998595472248E-1	1.3851017732227067E-1	4.0018235805806250E-35

(a)

Jacobi constant (LU^2/TU^2)	Period (TU)	Period (days)	Stability index
2.9979610288038199E+0	2.4071477560861904E+1	1.0670052670717605E+2	1.0000000775225699E+0
2.9979610078108498E+0	2.4068108193305424E+1	1.0668559146724169E+2	1.0000000097302200E+0
2.9979608920360601E+0	2.4061367081246001E+1	1.0665571045119214E+2	1.0000000057806600E+0
2.9979606782880102E+0	2.4054622843454499E+1	1.0662581557985439E+2	1.0000000160384499E+0
2.9979603670592900E+0	2.4047875532062960E+1	1.0659590708431172E+2	1.0000000000000000E+0
2.9979599588407302E+0	2.4041125199071516E+1	1.0656598519506271E+2	1.0000000000000000E+0
2.9979594541212502E+0	2.4034371896355985E+1	1.0653605014205492E+2	1.0000000002671900E+0
2.9979588533879400E+0	2.4027615675655774E+1	1.0650610215463122E+2	1.0000000019503099E+0
2.9979581571260101E+0	2.4020856588575164E+1	1.0647614146153555E+2	1.0000000011667101E+0
2.9979573658188099E+0	2.4014094686575707E+1	1.0644616829087910E+2	1.0000000057976599E+0
2.9979564799477498E+0	2.4007330020978436E+1	1.0641618287015027E+2	1.0000000021924300E+0
2.9979554999923899E+0	2.4000562642952289E+1	1.0638618542616327E+2	1.0000000007838401E+0
2.9979544264303399E+0	2.3993792603518184E+1	1.0635617618507615E+2	1.0000000015462500E+0
2.9979532597373200E+0	2.3987019953541594E+1	1.0632615537235792E+2	1.0000000036025500E+0
2.9979520003870599E+0	2.3980244743733468E+1	1.0629612321279268E+2	1.0000000000000000E+0
2.9979506471768800E+0	2.4168814000001895E+1	1.0713198544491988E+2	1.3919892931674700E+0
2.9979491269315099E+0	2.4175497923815488E+1	1.0716161296527372E+2	1.4204190416330600E+0
2.9979475000693698E+0	2.4182177820450089E+1	1.0719122263453721E+2	1.4489839962851401E+0
2.9979457660669602E+0	2.4188853636237951E+1	1.0722081421481991E+2	1.4776834255749900E+0
2.9979439243991699E+0	2.4195525317470096E+1	1.0725038746804844E+2	1.5065165906251901E+0
2.9979419745393501E+0	2.4202192810402419E+1	1.0727994215599379E+2	1.5354827382268901E+0
2.9979399159592699E+0	2.4208856061259951E+1	1.0730947804029020E+2	1.5645811095406399E+0
2.9979377481291500E+0	2.4215515016242335E+1	1.0733899488245923E+2	1.5938109374938600E+0
2.9979355046877298E+0	2.3915762174926414E+1	1.0601029430853252E+2	1.0000000000000000E+0
2.9979333005456601E+0	2.3908963106316445E+1	1.0598015639115840E+2	1.0000000007798000E+0

(b)

Figure 1.4: Output table provided by NASA JPL for the Long Period orbit family around L_4 in the Earth-Moon system [13]

These data can be used to perform several tasks, such as plotting one or more orbits,

generating a "*.csv*" showing the initial conditions of the selected orbits or the state vector of each point that composes the orbit, or observing the trends in the Jacobi constant and stability index.

Although this website represents the only way to freely access data on periodic orbits under the hypothesis of the restricted three-body problem, it has limitations. The only systems available are the those provided to the user, as well as the only orbits that can be obtained are the precomputed ones. Consequently, users cannot obtain informations on any other system or derive the initial conditions for new orbits independently.

1.2 Objectives

The objective of this thesis is to develop a tool that put itself, due to its characteristics, between the NASA JPL database and the more complex mission design softwares mentioned previously. The tool's primary function must be to retrieve periodic solutions in the CR3BP and extend this function to different dynamical systems. This will enable initial conditions for new periodic orbits in new sistems to be derived with sufficient accuracy and then provided to differential correction algorithms as an initial guess.

Here follows the list of the particular objectives that the thesis set out to achieve:

- to comprehend the theoretical basis and the fundamental characteristics of the CR3BP;
- to replicate the structure and the essential functions of the dynamical orbit catalog produced by NASA JPL;
- to analyze the sensitivity of the differential correction method to variations in the initial conditions, in order to identify its operational limits and critical failure points;
- to design and develop an algorithm that generates the initial conditions for new Halo orbits and use differential correction algorithms to ensure these initial conditions are periodic;
- to validate the results obtained through comparisons with analytical solutions or theoretical approximations derived from the mathematical framework of the CR3BP.
- to design and develop an algorithm that can evaluate the percentage of surface of a celestial body that is visible as the spacecraft moves along a specific periodic orbit;
- to obtain results that can be extrapolated to realistic scenarios with the potential to be applied in the early phases of space mission design;

- to create a graphic user interface (GUI) that enables the user to easily and intuitively perform all the functions at his disposal;

1.3 Methodology

Two programming languages have been considered to code the tool: MATLAB and Python.

The first one is an environment created by Cleve Molner in 1984. MATLAB distinguishes for the presence of numerous toolboxes, an integrated IDE and Simulink. Its computational speed and the vector/matrix syntax make MATLAB intuitive and well-suited to designing control algorithms, signal processing and numerical algorithms based on the use of matrices.

Python is a general-purpose open-source programming language created in 1991 by Guido von Rossum. Thanks to its extensive range of libraries, Python represents a valid alternative to MATLAB. It is also an excellent choice for scientific computing, machine learning and deep learning applications, as well as for creating *Graphical User Interfaces* (GUI). Finally, Python's interoperability with other programming languages such as C/C++, makes it ideal for coding of "end-to-end" software.

In order to create a freely usable and distributable tool with native web integration and no licence limitations for the general public, Python is the obvious choice for this thesis work. However, it is worth noting that MATLAB is now the standard for technical calculations in academic and corporate environments.

1.4 Thesis overview

This thesis is structured in 6 chapters.

Chapter 1 introduced the scenarios in which the CR3BP is applied, as well as its importance. It provided a historical overview of mission design software from the 1990s to the present day. The NASA JPL database was analysed due to its public access, and its functions, pros and cons were described as state of the art. Finally, the objectives of this thesis were outlined, as well as the programming language used to code the tool;

Chapter 2 introduces the theoretical framework on which the thesis is based. Firstly, it presents the fundamental studies that, during the years, contributed to the definition of the *Circular Restricted Three-Body Problem*. It then introduces the two reference frames used and the units used to non-dimensionalize the equations of motion, which are derived right after in the rotational reference frame. Finally, key concepts are presented and analyzed. These include the Lagrangian points and their stability, as well as

the Jacobi constant and how the value of this last parameter influences the extension of the zero-velocity curves. This thorough analysis of fundamental theoretical concepts is essential for the comprehension of the characteristics and the behaviour of a three-body system.

Chapter 3 explore the topic of periodic orbits in the CR3BP, presenting the characteristics necessary for them to be considered as such. Furthermore, it shows the procedure to integrate the equations of motion, in order to produce a visual representation of the periodic orbits. An analisys of the periodic orbits that are available in the tool follows, investigating the geometric and stability characteristics of each family. As well as providing a preliminary graphical representation, this chapter enables the reader to understand the nature of the different families of periodic orbits and, as a consequence, their possible applications in various operational scenarios.

Chapter 4 provides a brief introduction to the *Dynamical System Theory*, before focusing its analysis on the differential correction algorithms. First, the *State Transition Matrix* is derived, with particular attention given to its main characteristics and its importance for the differential correction algorithms. Next, the two implemented algorithms are thouroughly explained, presenting the mathematical discussion, the structure of the algoritm and an application case to demonstrate how it works. Finally, the results obtained when testing both algorithms with the same initial conditions are presented. Thus, after explaining the theoretical basis of the algorithms, the reader is offered a comparison to help understand the scenarios in which the two algorithms ensure the best performance in correcting the initial conditions.

Chapter 5 presents methods to derive the initial conditions for Halo orbits. The first method is Richardson's approximation, which yields third-order analytical solutions for the initial conditions of Halo orbits. After proving the utility and limitations of this approach, the theoretical functioning of a genetic algorithm and its fundamental steps are described. The genetic algorithm is then tested using reference initial conditions to demonstrate its reliability. Lastly, the results of the application of the genetic algorithm to two new three-body systems are presented. These results highlight the capacity of the genetic algorithm to find initial conditions highly likely to be periodic using the analytical approximation as a starting point.

Chapter 6 describes the final architecture of the tool, analyzing each of its modules and explaining how they work. Next, several future improvements are mentioned that would make the tool more complete and suitable to further aid in designing missions. Finally, the conclusions summarize the work done and explain how the thesis objectives have been satisfied.

The Circular Restricted Three-Body Problem

The *Three-Body Problem* (3BP) is a theory of the motion of an isolated system of three arbitrary point masses m_i. According to Newton's Law of Universal Gravitation and Second Law of Motion [14], the motion of every mass is a result of the mutual gravitational attraction, leading to the following equations of motion

$$m_i \frac{d^2\mathbf{r}_i}{dt^2} = G \sum_{j \neq 1} \frac{m_i m_j}{r_{ji}^3} \mathbf{r}_{ji} \tag{2.1}$$

where G is the Gravitational Constant equal to $6.6743 \times 10^{-11}\,\mathrm{N\,m^2\,kg^{-2}}$, m_i denotes the mass of *i-th* body and, $\mathbf{r}_{ji}$ is the relative position of P_j with respect to P_i, defined as

$$\mathbf{r}_{ji} = \mathbf{r}_j - \mathbf{r}_i \tag{2.2}$$

Equation (2.1) informs us that the trajectory of any body is defined by three second-order differential equations. Hence, a standard three-body system has nine second-order differential equations defining it, or alternatively eighteen first-order equations. Thus, eighteen initial conditions would be required in order to fully define the system's evolution analitically. However, when there are three or more bodies, only ten integrals of motion are available, suggesting that the system is not entirely integrable. Specifically, the first integral of motion was obtained by Jacobi in 1836 [15] and is presently called the *Jacobi integral*, which will be examined in Section 2.3. Later, in 1899, Poincarè showed that this is the only precise constant of motion for the general problem of three bodies [16]. Nevertheless, under these assumptions, analytical solutions, named *homographic solutions*, may be found, in which the system's geometric configuration is constant in time. These solutions have been obtained imposing zero angular momentum and equal masses, i.e. $m_1 = m_2 = m_3$. Among the ensuing orbits was one initially postulated by Moore in 1993 [17] and later confirmed by Chenciner and Montgomery in 2000 [18]. In this orbit, the three masses form a figure-eight curve, trace a rigid equilateral triangle that rotates at constant angular velocity inside it circumscribing circle, as depicted in Figure 2.1.

Of the numerous configurations shown to be numerically stable, this is merely an example. Despite this, the presence of such solutions in physical space is considered highly unlikely.

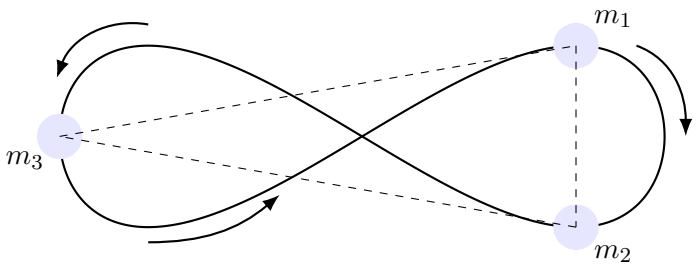


Figure 2.1: Figure-eight loop with equal masses

One simplifying assumption which is best suited to the majority of planetary systems that have been observed is that the mass of one of the bodies is negligible in comparison to the other two, which are referred to as primaries. With this in hand, the negligible mass does not influence the primaries' motion. This leads to a reformulation of the problem as so-called *Restricted Three-Body Problem* (R3BP). Therefore, the analysis of the system dynamics is the analysis of relative motion of the third body with respect to the primaries, which orbit about their common center of mass. A classic example is the motion of a spacecraft in the Earth-Moon system.

A second approximation is the primaries being in circular orbits around each other, assumption valid for most planet-moon and Sun-planet pairs in the Solar System. This is the formulation of the problem first suggested by Euler in 1767 [19], starting the break up from the R3BP towards the *Circular Restricted Three-Body Problem* (CR3BP).

A further variation of the problem has the primaries orbiting in elliptical curves, resulting in the so-called *Elliptic Restricted Three-Body Problem* (ER3BP). Nevertheless, in the rest of this research, only the CR3BP will be used, since it is an adequate model for the dynamical description of the previously cited primary body pairs, and for many others that will be used as examples throughout this thesis.

2.1 Equations of Motion

To get the equations of motion (EoMs) of the CR3BP, the configuration shown in Figure 2.2 is used. Figure 2.2 shows two frames: an inertial frame, denoted by **IJK**; and a rotating frame, or synonymously synodic frame, denoted by **ijk**. The latter is the barycenter of the two primaries with uniform angular velocity ω, which by the circular motion assumption coincides with the mean angular motion of the primaries

$$\boldsymbol{\omega} = \sqrt{\frac{GM}{R^3}}\mathbf{k} \tag{2.3}$$

where M is the total mass of the two primaries, and R is the distance between them.

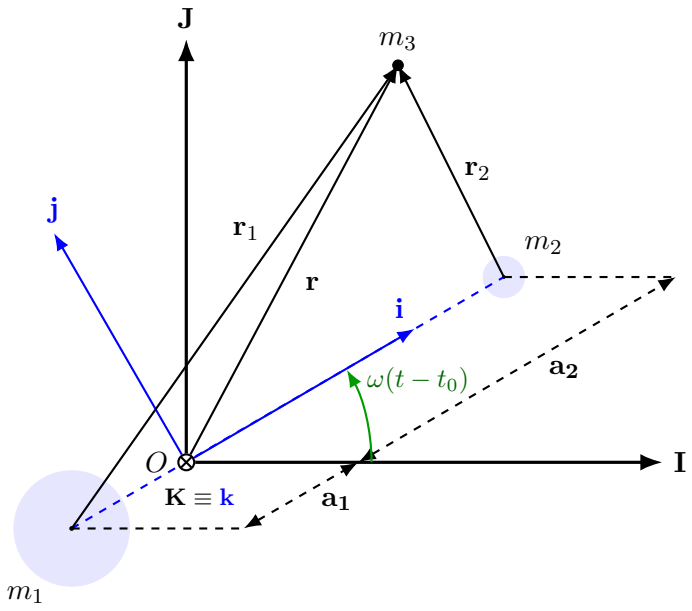


Figure 2.2: Geometric representation of the CR3BP

The relationship that defines the transformation from the synodic frame to the inertial frame is given by the following equation

$$\begin{Bmatrix} \mathbf{I} \\ \mathbf{J} \\ \mathbf{K} \end{Bmatrix} = \begin{bmatrix} \cos\omega(t-t_0) & -\sin\omega(t-t_0) & 0 \\ \sin\omega(t-t_0) & \cos\omega(t-t_0) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} \mathbf{i} \\ \mathbf{j} \\ \mathbf{k} \end{Bmatrix} = \mathbf{R} \begin{Bmatrix} \mathbf{i} \\ \mathbf{j} \\ \mathbf{k} \end{Bmatrix} \tag{2.4}$$

where $\mathbf{R}$ is the direct cosine matrix that transforms between the two reference frames.

The EoMs may be written as well in the synodic and in the inertial frame of reference. In the case at hand, the synodic frame is employed, as it enables the eradication of the time dependency from the equations of motion. In addition, the periodicity of the orbit in the CR3BP setting arises precisely due to the reference frame being chosen accordingly.

An additional simplification, widely used in the literature involves the nondimensionalization of the variables under consideration. This process eliminates the necessity of knowing the absolute values of the masses and the distance between the primaries, requiring instead only their relative relationships. The process of nondimensionalization is performed through division of the dimensional variables by suitable reference quantities, namely:

- *Length Unit*, corresponding to the sum of the distances of the two primaries from the barycenter

$$LU = a_1 + a_2$$

- *Mass Unit*, which is equal to the sum of the masses of the primaries

$$MU = m_1 + m_2$$

- *Time Unit*, derived from the dimensional part of the orbital period

$$TU = \sqrt{\frac{R^3}{GM}}$$

Because of the normalization used, the total mass of the system, the primary separation and the Gravitational Constant are all taken to be unity, while the orbital period T is simplified to 2π.

Moreover, since the total mass M of the system is nondimentionalized to unity, the smaller mass m_2 can be set to a fraction of M

$$m_2 = \mu \tag{2.5}$$

Consequently, the mass of the larger primary becomes

$$m_1 = M - m_2 = 1 - \mu \tag{2.6}$$

The μ parameter, conventionally referred to as the *mass ratio*, can thus range from a minimum of 0 for the classical *Two-Body Problem* (2BP) to a maximum of 0.5 for a system of two equal primaries, similar to many binary star systems.

Additionally, by definition of the barycenter, the distance of every primary from center of mass can be expressed as a function of the mass ratio

$$m_1 a_1 = m_2 a_2 \quad \Longrightarrow \quad \frac{a_1}{a_2} = \frac{\mu}{1-\mu} \tag{2.7}$$

From the Equation (2.7), it follows that the nondimensional distances of the primaries from the center of mass are

$$\begin{aligned} a_1 &= \mu \\ a_2 &= 1 - \mu \end{aligned} \tag{2.8}$$

that is, each is equal to the nondimensional mass of the opposite body, such that their distances are complementary with respect to unity.

As per the above assumption, the EoMs can now be written with respect to the synodic reference frame, as depicted in Figure 2.3. The third body state vector m_3 whose motion is to be examined is defined in the rotating frame and included both position and velocity terms

$$\mathbf{X} = \left(x, y, z, \frac{dx}{dt}, \frac{dy}{dt}, \frac{dz}{dt}\right) \tag{2.9}$$

The position vector of m_3 in the synodic frame is expressed as

$$\mathbf{r} = x\mathbf{i} + y\mathbf{j} + z\mathbf{k} \tag{2.10}$$

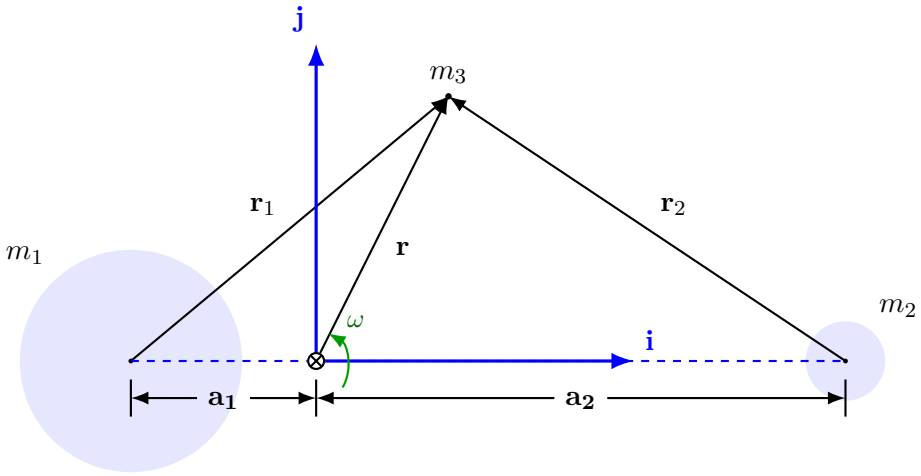


Figure 2.3: Synodic reference frame

Differentiating Equation (2.10) with respect to time provides the velocity of the body within the rotating frame

$$\frac{d\mathbf{r}^{(R)}}{dt} = \frac{dx}{dt}\mathbf{i} + \frac{dy}{dt}\mathbf{j} + \frac{dz}{dt}\mathbf{k} \tag{2.11}$$

where the superscript *(R)* represents the rotational reference frame.

A second differentiation of Equation (2.11) leads the acceleration vector in the synodic frame

$$\frac{d^2\mathbf{r}^{(R)}}{dt^2} = \frac{d^2x}{dt^2}\mathbf{i} + \frac{d^2y}{dt^2}\mathbf{j} + \frac{d^2z}{dt^2}\mathbf{k} \tag{2.12}$$

Both the velocity and acceleration vectors can also be expressed in the inertial reference frame.

The transformation of the velocity is given by

$$\frac{d\mathbf{r}^{(I)}}{dt} = \frac{d\mathbf{r}^{(R)}}{dt} + \boldsymbol{\omega}^{(R/I)} \times \mathbf{r} \tag{2.13}$$

where the superscript *(I)* stands for intertial reference frame.

The corresponding acceleration expression is

$$\frac{d^2\mathbf{r}^{(I)}}{dt^2} = \frac{d^2\mathbf{r}^{(R)}}{dt^2} + 2\boldsymbol{\omega}^{(R/I)} \times \frac{d\mathbf{r}^{(R)}}{dt} + \frac{d\boldsymbol{\omega}^{(R/I)}}{dt} \times \mathbf{r} + \boldsymbol{\omega}^{(R/I)} \times \left(\boldsymbol{\omega}^{(R/I)} \times \mathbf{r}\right) \tag{2.14}$$

It can be observed in Equation (2.14) that the inertial frame acceleration, and hence the one experienced in Newton's second law, consists of four separate terms:

- $\frac{d^2\mathbf{r}^{(R)}}{dt^2}$ is the acceleration of the third body in the rotating frame;
- $2\boldsymbol{\omega}^{(R/I)} \times \frac{d\mathbf{r}^{(R)}}{dt}$ is the Coriolis acceleration;

- $\frac{d\boldsymbol{\omega}^{(R/I)}}{dt} \times \mathbf{r}$ is the acceleration due to a non-constant motion of the synodic frame with respect to the inertial frame;
- $\boldsymbol{\omega}^{(R/I)} \times \left(\boldsymbol{\omega}^{(R/I)} \times \mathbf{r}\right)$ is the centripetal acceleration.

Since the two primaries' motion around their barycenter is circular and in the xy plane, the angular velocity vector ω is constat and along the z-axis

$$\boldsymbol{\omega} = \begin{pmatrix} 0 \\ 0 \\ \omega \end{pmatrix} \tag{2.15}$$

Hence, the third term on the right hand side of Equation (2.14) is dropped, and the Coriolis and centripetal accelerations reduce to

$$2\boldsymbol{\omega}^{(R/I)} \times \frac{d\mathbf{r}^{(R)}}{dt} = 2 \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ 0 & 0 & \omega \\ \frac{dx}{dt} & \frac{dy}{dt} & \frac{dz}{dt} \end{vmatrix} = 2\omega \frac{dx}{dt}\mathbf{i} - 2\omega \frac{dy}{dt}\mathbf{j} \tag{2.16}$$

$$\boldsymbol{\omega}^{(R/I)} \times \left(\boldsymbol{\omega}^{(R/I)} \times \mathbf{r}\right) = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ 0 & 0 & \omega \\ -\omega y & \omega x & 0 \end{vmatrix} = -\omega^2(x\mathbf{i} + y\mathbf{j}) \tag{2.17}$$

By putting Equations (2.16) and (2.17) in Equation (2.14), one get the expression of the acceleration of the third body with respect to the inertial frame, expressed as function of the synodic frame coordinates

$$\frac{d^2\mathbf{r}^{(I)}}{dt^2} = \left(\frac{d^2x}{dt^2} - 2\frac{dy}{dt} - x\right)\mathbf{i} + \left(\frac{d^2y}{dt^2} + 2\frac{dx}{dt} - y\right)\mathbf{j} + \frac{d^2z}{dt^2}\mathbf{k} \tag{2.18}$$

By applying the Universal Gravitational Law as given in Equation (2.1) the gravitational acceleration on the third body can be calculated. In order to calculate the contributions from the two primaries, one needs to find the position vectors from the primaries towards the third mass m_3. Referring to Equation (2.2) and the configuration described in Fig. (2.3), the relative position vectors are

$$\mathbf{r}_1 = \mathbf{r} - a_1\mathbf{i} = (x + \mu)\mathbf{i} + y\mathbf{j} + z\mathbf{k} \tag{2.19}$$

$$\mathbf{r}_2 = \mathbf{r} - a_2\mathbf{i} = (x - 1 + \mu)\mathbf{i} + y\mathbf{j} + z\mathbf{k} \tag{2.20}$$

with their magnitudes given by

$$\begin{aligned} r_1 &= \sqrt{(x + \mu)^2 + y^2 + z^2} \\ r_2 &= \sqrt{(x - 1 + \mu)^2 + y^2 + z^2} \end{aligned} \tag{2.21}$$

According to Law of Universal Gravitation, the accelerations exerted by the two pri-

maries on m_3 are

$$\begin{aligned} \mathbf{a}_1 &= -\frac{(1-\mu)\mathbf{r}_1}{r_1^3} \\ \mathbf{a}_2 &= -\frac{\mu\mathbf{r}_2}{r_2^3} \end{aligned} \tag{2.22}$$

Summing these contributions results in the total gravitational acceleration acting on the third body

$$\mathbf{a}_g = -\frac{(1-\mu)\mathbf{r}_1}{r_1^3} - \frac{\mu\mathbf{r}_2}{r_2^3} \tag{2.23}$$

Care should be taken to note that, due to previous nondimensionalization, both the whole mass M and the gravitational constant G are assigned unity values and therefore omitted from Equation (2.23). Had this normalization not been carried out, they would have featured in the expression explicitly.

It should be noted further that the acceleration in Equation (2.23) is equivalent to the previously obtained one in Equation (2.14).

Hence, setting them equal and solving for the **ijk** components provides the final form of the EoMs: a set of three second-order, non-linear, and coupled differential equations

$$\begin{cases} \dfrac{d^2x}{dt^2} - 2\dfrac{dy}{dt} - x = -\dfrac{(1-\mu)(x+\mu)}{r_1^3} - \dfrac{\mu(x-1+\mu)}{r_2^3} \\ \dfrac{d^2y}{dt^2} + 2\dfrac{dx}{dt} - y = -\dfrac{(1-\mu)y}{r_1^3} - \dfrac{\mu y}{r_2^3} \\ \dfrac{d^2z}{dt^2} = -\dfrac{(1-\mu)z}{r_1^3} - \dfrac{\mu z}{r_2^3} \end{cases} \tag{2.24}$$

The EoMs provided in Equation (2.24) can be re-expressed alternatively by introducing the pseudo-potential function U

$$U = \frac{1}{2}(x^2+y^2) + \frac{1-\mu}{r_1} + \frac{\mu}{r_2} \tag{2.25}$$

Based on the formulation provided in Equation (2.25), the EoMs can be expressed more briefly as

$$\begin{cases} \dfrac{d^2x}{dt^2} - 2\dfrac{dy}{dt} = \dfrac{\partial U}{\partial x} \\ \dfrac{d^2y}{dt^2} + 2\dfrac{dx}{dt} = \dfrac{\partial U}{\partial y} \\ \dfrac{d^2z}{dt^2} = \dfrac{\partial U}{\partial z} \end{cases} \tag{2.26}$$

The pseudo-potential U consists of the two primaries' gravitational potential together with an additional term, known as centrifugal potential, associated with the first term in Equation (2.25), and is caused by taking rotational reference frame. The pseudo-potential thus describes the effective potential experienced by the third body in the synodic reference frame. Figure 2.4 shows this function for the mass ratio $\mu = 0.15$. As can be seen, the pseudo-potential tends to infinity at the locations of the two main

bodies. This is due to the fact that the distance between the third body and one of the primaries approaches zero in the region. Moreover, the maximum corresponding to the more massive primary is of larger radial size. This is a direct consequence of the mass-dependent nature of the gravitational term in the pseudo-potential: since the potential contribution of each primary is a function of its mass, the stronger attaction of the heavier body produces a deeper and wider potential well around it.

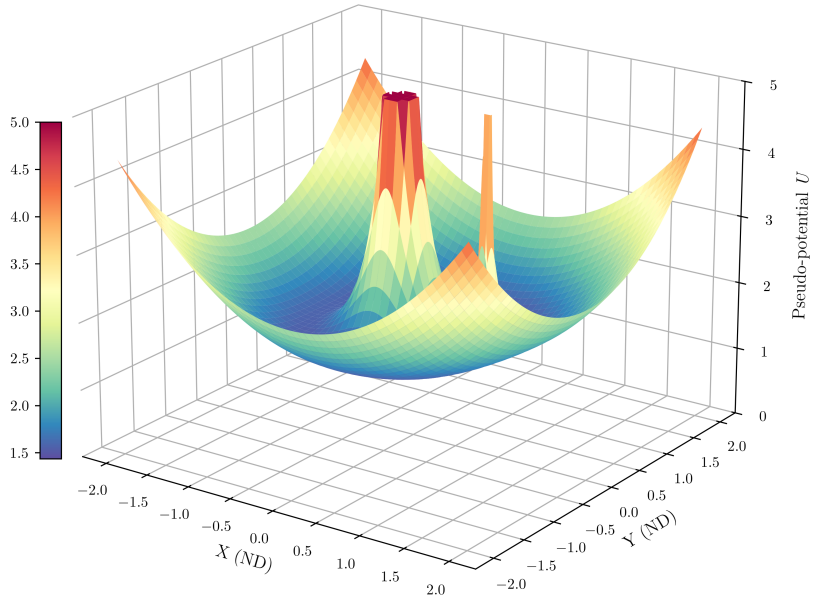


Figure 2.4: Pseudo-potential function

Equation (2.26) will be utilized in the subsequent sections to find the Lagrange points and to obtain the Jacobi integral.

2.2 Lagrange Points

Once the EoMs have been obtained, the question that arises is whether or not there exist equilibrium points within the synodic frame. These are equivalent to configurations where the third body is at rest in reference to the primaries. Within the synodic frame, this implies that the third body is stationary with respect to position in time. The location of such points is of inherent importance in the CR3BP, as they are the starting point of families of periodic and quasi-periodic orbits.

In order to find points with such characteristics, one must enforce that both velocity and acceleration be zero at these points. This is equivalent to insisting that the spatial derivatives of the pseudo-potential function U be zero.

In view of these parameters, the EoMs provided in Equation (2.26) reduce to the

following algebraic system

$$\begin{cases} x - \dfrac{(1-\mu)(x+\mu)}{r_1^3} - \dfrac{\mu(x-1+\mu)}{r_2^3} = 0 \\ \left[1 - \dfrac{1-\mu}{r_1^3} - \dfrac{\mu}{r_2^3}\right] y = 0 \\ \left[-\dfrac{1-\mu}{r_1^3} - \dfrac{\mu}{r_2^3}\right] z = 0 \end{cases} \tag{2.27}$$

It is immediately evident that the equation along the z-axis holds only if $z = 0$. Therefore, all the equilibrium points, or *libration points*, or *Lagrange points*, named after Lagrange who first calculated them in 1772 [20], lie in the **i-j** plane, which coincides with the plane of motion of the primaries.

In order to find the coordinates of the Lagrange points in the synodic reference frame, one must solve the x- and y- equations of Equation (2.27). The y-equation has two different solutions, $y = 0$ and $y \neq 0$, which inherently distinguish the libration points into two distinct groups: the equilateral points ($y \neq 0$) and the collinear points ($y = 0$).

2.2.1 Equilateral Points

The first group of Lagrange points is identified imposing that $y \neq 0$ within the equation along the y-direction. Recalling that all Lagrange points lie in the **i-j** plane, the resulting system to be solved becomes

$$\begin{cases} x - \dfrac{(1-\mu)(x+\mu)}{r_1^3} - \dfrac{\mu(x-1+\mu)}{r_2^3} = 0 \\ 1 - \dfrac{1-\mu}{r_1^3} - \dfrac{\mu}{r_2^3} = 0 \end{cases} \tag{2.28}$$

By manipulating the relations in Equation (2.28) the conditions required to determine the equilateral points can be derived.

More precisely, multiplying the second equation by $x - 1 + \mu$ and subtracting it from the first one gives

$$\cancel{x} - \frac{(1-\mu)(x+\mu)}{r_1^3} - \cancel{\frac{\mu(x-1+\mu)}{r_2^3}} - \cancel{x} + 1 - \mu + \frac{(1-\mu)(x-1+\mu)}{r_1^3} + \cancel{\frac{\mu(x-1+\mu)}{r_2^3}} = 0$$

which simplifies to

$$\frac{1-\mu}{r_1^3} = 1 - \mu$$

implying that $r_1 = 1$.

Likewise, multiplying the second equation by $x + \mu$ and once again subtracting it

from the first yields

$$\cancel{x} - \cancel{\frac{(1-\mu)(x+\mu)}{r_1^3}} - \frac{\mu(x-1+\mu)}{r_2^3} - \cancel{x} + 1 - \mu + \cancel{\frac{(1-\mu)(x-1+\mu)}{r_1^3}} + \frac{\mu(x-1+\mu)}{r_2^3} = 0$$

which results in

$$\frac{\mu}{r_2^3} = \mu$$

hence obtaining $r_2 = 1$.

Referencing the general equations for the distance of the third body from each primary given by Equation (2.21) and observing that these distances are equal to each other, the relation is obtained

$$\sqrt{(x+\mu)^2 + y^2} = \sqrt{(x-1+\mu)^2 + y^2} \tag{2.29}$$

Solving for x, Equation (2.29) becomes

$$x = \frac{1}{2} - \mu \tag{2.30}$$

From this outcome, and the unitary nature of both of these distances, the y-coordinates of the two equilateral points are

$$y = \pm\frac{\sqrt{3}}{2} \tag{2.31}$$

Therefore, the equilateral points lie on the vertices of two unit length equilateral triangles, hence their designation. The equilateral points are usually denoted as L_4 and L_5, and their coordinates are as follows

$$\begin{aligned} L_4: &\quad \left(\frac{1}{2} - \mu, \frac{\sqrt{3}}{2}, 0\right) \\ L_5: &\quad \left(\frac{1}{2} - \mu, -\frac{\sqrt{3}}{2}, 0\right) \end{aligned} \tag{2.32}$$

2.2.2 Collinear Points

The coordinates of the second group of Lagrange points can be obtained by using the second solution of the equation in the y-direction of Equation (2.28), i.e., $y = 0$. This type of condition imposes that all the second set of Lagrange points be located on the x-axis of the synodic frame. By thus substituting $y = 0$ and $z = 0$, the given relation simplifies to

$$x - \frac{(1-\mu)(x+\mu)}{|x+\mu|^3} - \frac{\mu(x-1+\mu)}{|x-1+\mu|^3} = 0 \tag{2.33}$$

where the absolute values in the denominators are used to enforce that the terms are positive because they are physical distances. There are three real roots to Equation (2.33), which are the positions of points that are collinear along the x-axis. The points are often represented as L_1, L_2, and L_3.

However, unlike for the equilateral points, the computation of collinear points' positions involves the solution of an equation that lacks direct analytical solutions. As such, two different approaches can be used based on mass ratio μ value. If μ is small enough, Equation (2.33) can be approximated and approximate analytical solutions for the coordinates of collinear points can be derived

$$\begin{aligned} L_1 : &\quad \left(-\left[\frac{\mu}{3}\right]^{\frac{1}{3}} + 1, 0, 0 \right) \\ L_2 : &\quad \left(\left[\frac{\mu}{3}\right]^{\frac{1}{3}} + 1, 0, 0 \right) \\ L_3 : &\quad \left(-1 - \left[\frac{5\mu}{12}\right], 0, 0 \right) \end{aligned} \tag{2.34}$$

If instead the mass ratio μ is not very small, then the above approximate coordinates may serve as a good initial guess for a numerical solution of Equation (2.33). For instance, the Newton-Raphson method [21] may be used to solve Equation (2.33) and find the numerical solutions corresponding to the x-coordinates of L_1, L_2 and L_3. Approximate solutions tend to be near true roots and also improve convergence of the numerical method. Figure 2.5 illustrates how the position of the collinear points vary with the mass ratio μ. The horizontal axis has been truncated to the interval $[0, 0.2]$ since it includes all the values relevant to the Solar System systems.

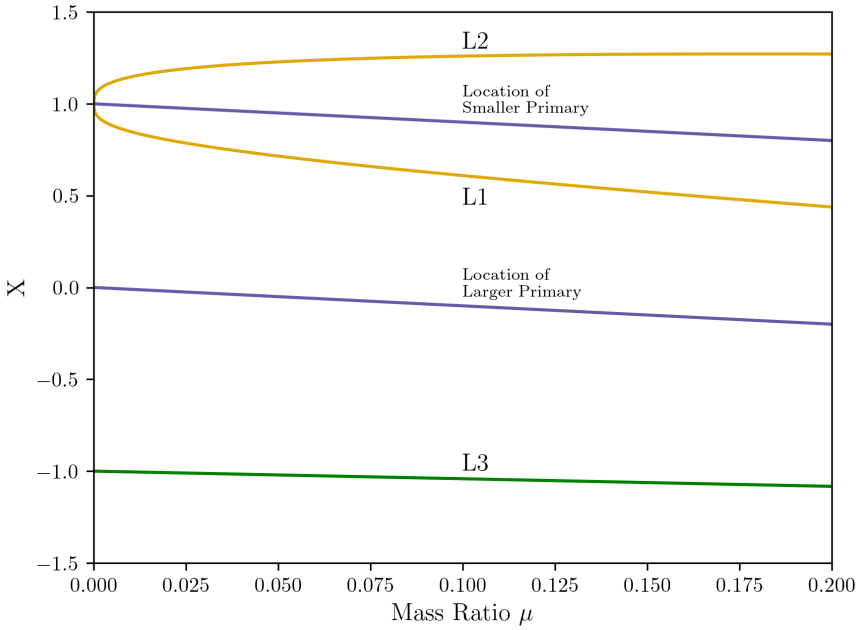


Figure 2.5: Variation of the collinear points as a function of μ

Figure 2.6 present a few representative examples, with the horizontal axis converted to a logarithmic scale to better emphasize the difference in μ values.

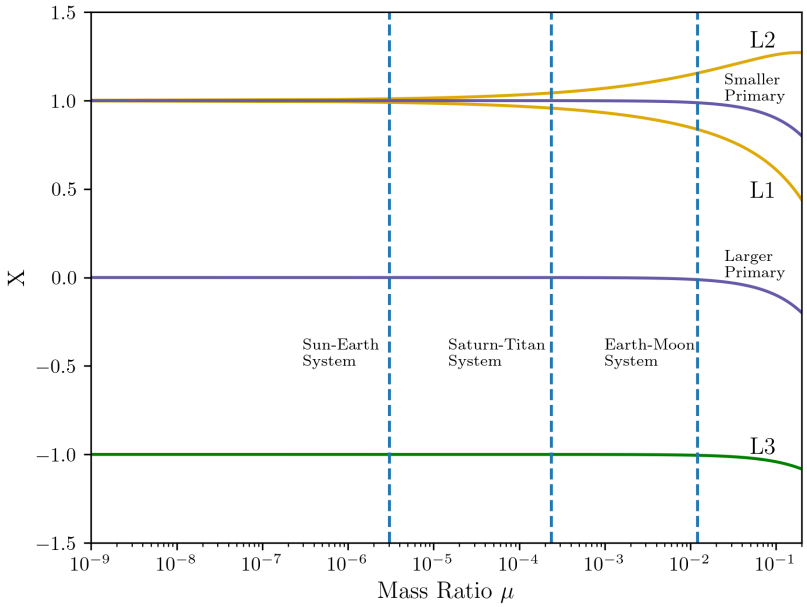


Figure 2.6: Zoomed-in plot of the collinear point locations with μ on a logarithmic scale

Figure 2.6 present a few representative examples of three-body systems, with the horizontal axis converted to a logarithmic scale to better emphasize the difference in μ values. It should be noted that:

- L_1 is always confined to the interval $[-\mu, 1-\mu]$ and tends to zero as μ approaches 0.5;
- L_2 lies within the interval $[1-\mu, \infty[$ and exhibits a behavior similar to that of L_1;
- L_3 in located in $]-\infty, -1]$ and tends to $-\infty$ as μ approaches to 0.5.

2.2.3 Stability of the Lagrange Points

Having determined the position of the Lagrange points, one must determine the conditions under which these points are stable. A Lagrange point is stable if, when a body is put at its position and a small perturbation is given to its initial state, the body does not deviate from the point over time. To formulate this condition mathematically, the initial coordinates of the third body are expressed as small deviations from the equilibrium position

$$\begin{aligned} x &= x_e + \delta x \\ y &= y_e + \delta y \\ z &= z_e + \delta z \end{aligned} \tag{2.35}$$

where x_e, y_e and z_e denote the coordinates of a given Lagrangian pointy, while δx, δy and δz represents small perturbations applied along each axis. In practise, since all

Lagrangian points lie in the xy-plane, it follows that $z_e = 0$

With these perturbed conditions, the expressions for the inverse cube of the distances of the third body to each primary, originally established by Equation (2.21), are now

$$\begin{aligned} r_1^{-3} &= \left[(x_e + \delta x + \mu)^2 + (y_e + \delta y)^2 + \delta z^2\right]^{-\frac{3}{2}} \\ r_2^{-3} &= \left[(x_e + \delta x - 1 + \mu)^2 + (y_e + \delta y)^2 + \delta z^2\right]^{-\frac{3}{2}} \end{aligned} \tag{2.36}$$

Multiplying everything out, disregarding the higher-order terms, and using the binomial theorem

$$\begin{aligned} r_1^{-3} &\approx r_{1e}^{-3} - 3r_{1e}^{-5}\left[(x_e + \mu)\delta x + y_e \delta y\right] \\ r_2^{-3} &\approx r_{2e}^{-3} - 3r_{2e}^{-5}\left[(x_e - 1 + \mu)\delta x + y_e \delta y\right] \end{aligned} \tag{2.37}$$

where r_{ie} represent the distance of the *i-th* primary from the location of the Lagrange points, hence it can be written as

$$r_{ie} = \begin{cases} x_e & \text{for } L_1,\ L_2,\ L_3 \\ 1 & \text{for } L_4,\ L_5 \end{cases} \tag{2.38}$$

By getting the first and second derivative of Equation (2.35) and substituting them, along with Equation (2.37), into Equation (2.24), yields the linearized EoMs around the Lagrange points

$$\begin{cases} \delta\ddot{x} - 2\delta\dot{y} - (1 - A)\delta x - B\delta y = 0 \\ \delta\ddot{y} + 2\delta\dot{x} - B\delta x - (1 - C) = 0 \\ \delta\ddot{z} + D\delta z = 0 \end{cases} \tag{2.39}$$

where $\delta\dot{x}$ and $\delta\ddot{x}$ are respectively the first and second derivative with respect to time. The coefficients A, B, C and D are constants defined as follows

$$\begin{aligned} A &= (1 - \mu)\left[\frac{1}{r_{1e}^3} - 3\frac{(x_e + \mu)^2}{r_{1e}^5}\right] + \mu\left[\frac{1}{r_{2e}^3} - 3\frac{(x_e - 1 + \mu)^2}{r_{2e}^5}\right] \\ B &= 3(1 - \mu)\frac{(x_e + \mu)y_e}{r_{1e}^5} + 3(1 - \mu)\frac{(x_e - 1 + \mu)y_e}{r_{2e}^5} \\ C &= (1 - \mu)\left[\frac{1}{r_{1e}^3} - 3\frac{y_e^2}{r_{1e}^5}\right] + \mu\left[\frac{1}{r_{2e}^3} - 3\frac{y_e^2}{r_{2e}^5}\right] \\ D &= \frac{1 - \mu}{r_{1e}^3} + \frac{\mu}{r_{2e}^3} \end{aligned} \tag{2.40}$$

Because the motion along the z-direction is a simple harmonic motion and therefore it is always stable, the attention can be focused on the first two relations of Equation (2.39), leading to the 2D linearized EoMs

$$\begin{cases} \delta\ddot{x} - 2\delta\dot{y} - (1 - A)\delta x - B\delta y = 0 \\ \delta\ddot{y} + 2\delta\dot{x} - B\delta x - (1 - C) = 0 \end{cases} \tag{2.41}$$

It is assumed that Equation (2.41) admits a solution of the following kind

$$\begin{aligned} \delta x &= c_1 e^{\lambda t} \\ \delta y &= c_2 e^{\lambda t} \end{aligned} \tag{2.42}$$

The system can be expressed in matrix form by inserting the first and second derivative of Equation (2.42) into Equation (2.41)

$$\begin{bmatrix} \lambda^2 - (1-A) & -B - 2\lambda \\ -B + 2\lambda & \lambda^2 - (1-C) \end{bmatrix} \begin{Bmatrix} \delta x \\ \delta y \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \tag{2.43}$$

which represents a typical eigenvalue problem. The trivial solution $\delta x = \delta y = 0$, one of the two allowed, does not provide any additional information regarding the stability of the system. Such solution corresponds to the third body being located at one of the Lagrangian points exactly. The non-trivial solution is of greater interest and can be derived when the matrix in Equation (2.43) is invertible, namely non-singular. Computing the determinant and setting it equal to zero yeilds the characteristic equation. This is a quadratic equation in λ^2

$$\lambda^4 + \lambda^2 (2 + C + A) + \left(1 - C - A + AC - B^2\right) = 0 \tag{2.44}$$

whose solutions are given by

$$\lambda = \pm\sqrt{-2 - C - A \pm \sqrt{2C + 2A + \frac{1}{4}C^2 - \frac{1}{2}AC + \frac{1}{4}A^2 + B^2}} \tag{2.45}$$

In order for the system to be stable, all roots must be purely immaginary or have a negative real part.

For istance, when the equilateral points are considered, it is known that both distances from the primaries r_{1e} and r_{2e} are unitary. Therefore, Equation (2.44) becomes

$$\lambda^4 + \lambda^2 - \left(\frac{27}{4}\right)\mu(\mu - 1) = 0 \tag{2.46}$$

whose roots are given by

$$\lambda^2 = \frac{1}{2}\left(-1 \pm \sqrt{1 - 27\mu(1-\mu)}\right) \tag{2.47}$$

To ensure the solutions are purely imaginary, the argument of the root must be negative

$$1 - 27\mu(1 - \mu) < 0 \tag{2.48}$$

Knowing that $\mu \in [0, 0.5]$, Equation (2.48) is only satisfied if

$$\mu < \frac{1}{18}\left(9 - \sqrt{69}\right) \approx 0.0385209 \tag{2.49}$$

This value corresponds to the critical mass ratio μ^*.

Hence, L_4 and L_5 are marginally stable if and only if $\mu < \mu^*$, a condition that applies to each planetary system in the Solar System. The only exception is the Pluto-Charon system, whose mass ratio is $\mu = 0.108511$.

A similar analysis can be conducted for the collinear points L_1, L_2 and L_3 as well. Unlike the previous case, the roots of the characteristic equation have always positive real parts, meaning that these points are always unstable.

Figure 2.7 finally shows a representation of the five Lagrange points in the **i-j** plane for a system with an arbitrary μ set to 0.1.

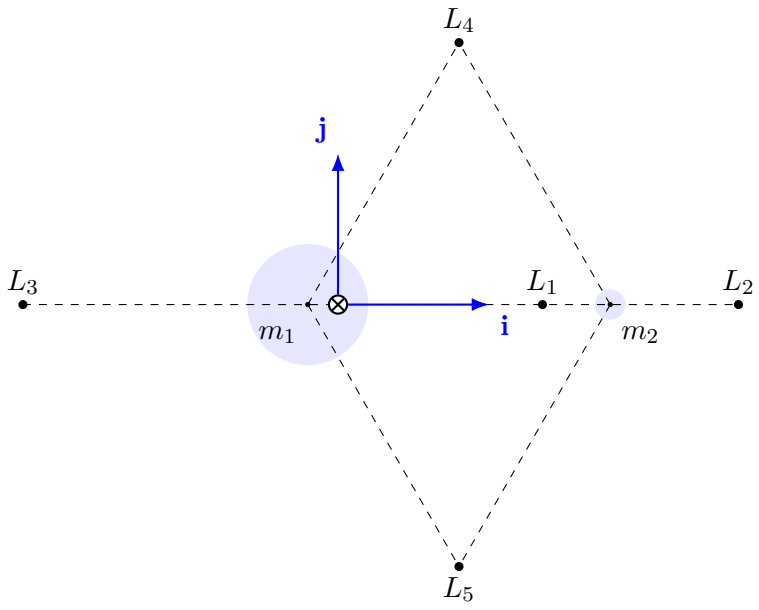


Figure 2.7: Lagrange points in the CR3BP

2.3 Jacobi Integral

As anticipated in the introduction to this chapter, the Jacobi integral (or Jacobi constant) is the only conservation law that is valid in the context of the CR3BP [15]. In particular, this integral of motion is the only one that provides informations on the dynamics of the third body with respect to the primaries.

To derive the Jacobi constant, the EoMs presented in Equation (2.24) are used as a starting point. The first step is to multiply each equation by the first derivative of the corresponding spatial variable, and then the three resultant expressions are summed

up

$$\begin{aligned}&\left[\frac{d^2x}{dt^2} - 2\frac{dy}{dt} - x + \frac{(1-\mu)(x+\mu)}{r_1^3} + \frac{\mu(x-1+\mu)}{r_2^3}\right]\frac{dx}{dt}\\ &+ \left[\frac{d^2y}{dt^2} + 2\frac{dx}{dt} - y + \frac{(1-\mu)y}{r_1^3} + \frac{\mu y}{r_2^3}\right]\frac{dy}{dt}\\ &+ \left[\frac{d^2z}{dt^2} + \frac{(1-\mu)z}{r_1^3} + \frac{\mu z}{r_2^3}\right]\frac{dz}{dt} = 0\end{aligned} \tag{2.50}$$

Making the products explicit and grouping the terms yields

$$\begin{aligned}\frac{dx}{dt}\frac{d^2x}{dt^2} + \frac{dy}{dt}\frac{d^2y}{dt^2} + \frac{dz}{dt}\frac{d^2z}{dt^2} - x\frac{dx}{dt} - y\frac{dy}{dt} = &-\frac{(1-\mu)}{r_1^3}\left[(x+\mu)\frac{dx}{dt} + y\frac{dy}{dt} + z\frac{dz}{dt}\right]\\ &-\frac{\mu}{r_2^3}\left[(x-1+\mu)\frac{dx}{dt} + y\frac{dy}{dt} + z\frac{dz}{dt}\right]\end{aligned} \tag{2.51}$$

The first three terms on the left-hand side form the time derivative of the kinetic energy in the synodic reference frame

$$\frac{dx}{dt}\frac{d^2x}{dt^2} + \frac{dy}{dt}\frac{d^2y}{dt^2} + \frac{dz}{dt}\frac{d^2z}{dt^2} = \frac{1}{2}\frac{d}{dt}\left(V^2\right) \tag{2.52}$$

where V is the magnitude of the velocity vector, while the remaining two terms on the left-hand side represent the opposite of the derivative of the centrifugal potential

$$-x\frac{dx}{dt} - y\frac{dy}{dt} = \frac{1}{2}\frac{d}{dt}\left(x^2+y^2\right) \tag{2.53}$$

Acknowledging that the expressions bewtween square brackets on the right-hand side represent the squared distances from the primaries and substituting Equations (2.52) and (2.53) into Equation (2.51), yields the following

$$\frac{d}{dt}\left[\frac{1}{2}V^2 - \frac{1}{2}\left(x^2+y^2\right) - \frac{1-\mu}{r_1} - \frac{\mu}{r_2}\right] = 0 \tag{2.54}$$

Integrating Equation (2.54) with respect to time, the expression of the Jacobi constant is obtained

$$J' = \frac{1}{2}\left[\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2 + \left(\frac{dz}{dt}\right)^2\right] - \frac{1}{2}\left(x^2+y^2\right) - \frac{1-\mu}{r_1} - \frac{\mu}{r_2} \tag{2.55}$$

In the continuation of the thesis an alternative formulation will be employed, which will allow the Jacobi constant to be conceived as a positive number

$$J = \frac{2(1-\mu)}{r_1} + \frac{2\mu}{r_2} + \left(x^2+y^2\right) - \left[\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2 + \left(\frac{dz}{dt}\right)^2\right] \tag{2.56}$$

Equivalently, the Jacobi constant can be express as a function of the pseudo-potential U

$$J = 2U - V^2 \tag{2.57}$$

Equation (2.57) shows that the Jacobi integral represents the total energy of the third body, which depends on the initial conditions. Due to the dependence of the kinetic energy on velocity, a decrease of the latter corresponds to an increase in the pseudo-potential. When the velocity reaches zero, the third body will be located on a zero-velocity curve, which represents an invalicable limit for a given value of J. These curves define the borders of the *Hill's regions*, namely the portions of the synodic plane within which the third body can move freely. Taking the Earth-Moon system as an example, Figure 2.8 shows the accessible regions when the Jacobi constant is set to $J = 3.3$. Specifically, the colored region corresponds to the region inaccessible to the third body. Therefore, under these conditions a body orbiting the Earth would be unable to reach the Moon, and vice versa.

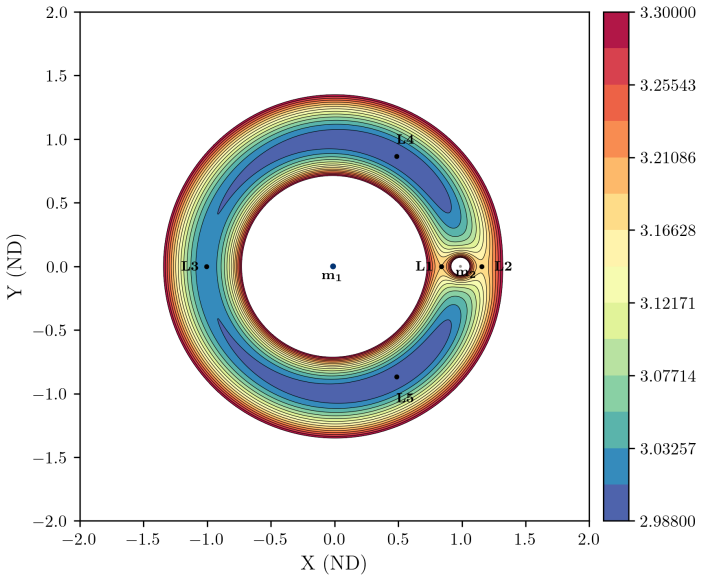


Figure 2.8: Zero-velocity curves for Earth-Moon system for $J = 3.3$

For the Lagrange points to be accessible, the value of the Jacobi constant must vary. Using Equation (2.56) it is possible to compute the constant's value at each Lagrange point location, assuming a condition of zero velocity. Considering L_1, a value of $J = 3.18834$ is derived. Assuming such value for J, the two internal regions extend exactly to L_1, but they are not yet connected. In order for this to happen, it is sufficient to slightly reduce the value of the Jacobi constant.

Figure 2.9 presents the situation described above: the two regions do indeed connect when $J = 3.18$. Thanks to the bottleneck that appears around L_1, the second primary is accessible.

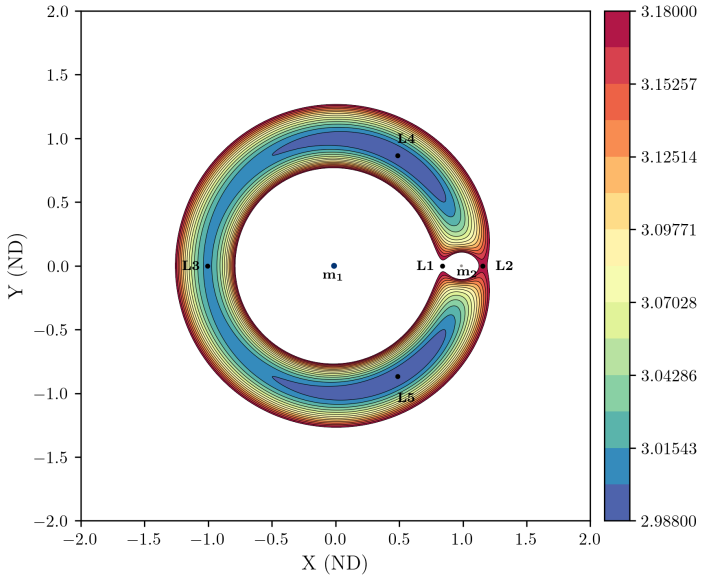


Figure 2.9: Zero-velocity curves for Earth-Moon system for $J = 3.18$

A further reduction of the Jacobi constant to $J = 3.17216$ lead to the region around the smaller primary to extent to L_2, as showed by Figure 2.10.

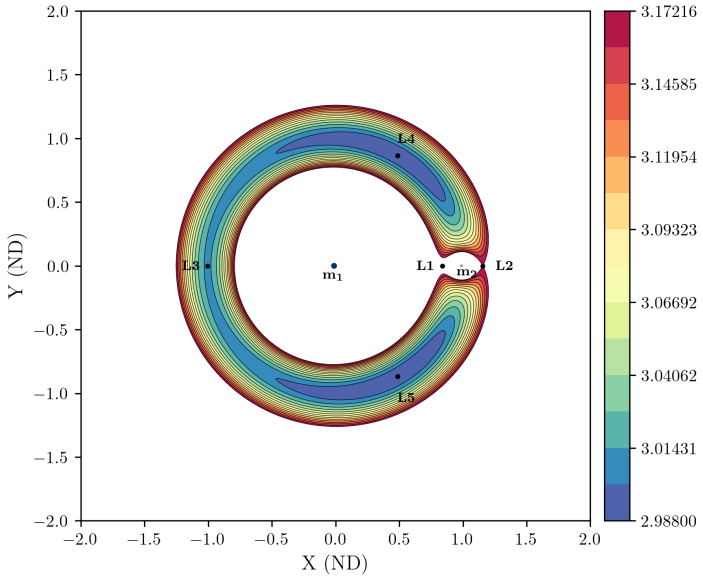


Figure 2.10: Zero-velocity curves for Earth-Moon system for $J = 3.17216$

In order for the third body to reach L_3, which is located on the opposite side of the bigger primary, the energy of the body must increase significantly. With $J = 3.012147$, Figure 2.11 shows that the unaccessible regions becoming narrower, which benefits an enlargment of the Hill's regions. Additionally, it can be deduced that, once the third collinear point is reached, the equilateral points become accessible following a slight increase in the constant. Indeed, employing the coordinates of these points, a value of

the Jacobi constant of $J = 2.987997$ is computed, which opens the entire synodic plane to the third body. Due to the symmetry of L_4 and L_5 with respect to the x-axis, the same value applies to both points.

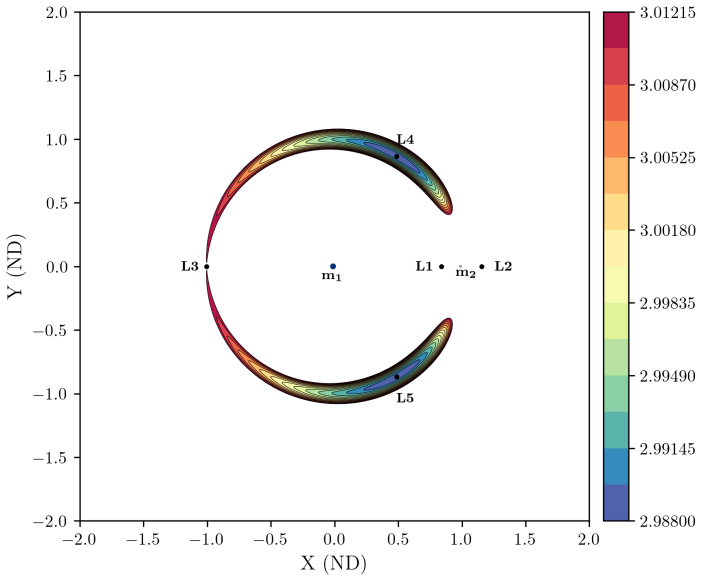


Figure 2.11: Zero-velocity curves for Earth-Moon system for $J = 3.012147$

Table 2.1 summarizes the values of the Jacobi constant for each of the Lagrange points for the Earth-Moon system. It becomes clear that the enumeration of the Lagrange points follows the sequence of values of the constant. L_1 is associated with the highest value of J, which corresponds to the least amout of energy, making it the easiest to reach. Proceding towards the equilateral points, a greater amount of energy is required, meaning they are the most difficult to achieve.

Table 2.1: Values of Jacobi constant at Earth-Moon Lagrangian points

Lagrange point	L_1	L_2	L_3	L_4	L_5
Jacobi constant	3.18834	3.17216	3.012147	2.987997	2.987997

Periodic Orbits of the CR3BP

The analysis of the periodic solutions for the CR3BP is a topic that has been broadly debated since the 1960s, particularly following the launch of the *International Sun-Earth Explorer* (ISEE) mission in 1974, which had the objective to study the interaction between the Earth's magnetic field and the solar wind. Starting from the concept of solution, it can be concluded that it is almost impossible to obtain full information on any orbit of a non-integrable dynamical systems, unless the orbit is asymptotic, quasi-periodic or periodic. More precisely, there is no numerical method that can handle the case of $t \longrightarrow \infty$.

If the dynamical system under analysis is integrable or can be linearized, the previous conclusion does not apply. In fact, asymptotic or periodic solutions can be derived by analysing the motion around the Lagrange points. For simple periodic orbits, object of the thesis, a relation of the kind

$$\dot{\mathbf{X}} = f(\mathbf{X}_0) \tag{3.1}$$

can be established and, consequently, the group of initial conditions corresponds to the set of points belonging to the curve $f(\mathbf{X}_0)$.

As Szebehely reports in his work [22], the starting point is the definition of periodic motion in a dynamical system. Motion is periodic if the same configuration repeats at regular time intervals. For example, in a fixed coordinate frame of the 2BP, the solution repeats after one period, in accordance with

$$T = 2\pi/n \tag{3.2}$$

where n is the mean angular motion. Introducing the synodic reference frame, which rotates with a period $T_{syn} = 2\pi$, periodic motion exists in the rotational frame when n is a rational number.

Moving on to the three-body problem, the system of differential equations that describes the CR3BP can be expressed as

$$\frac{dx_i}{dt} = X_i(x_1, \ldots, x_n) \qquad i = 1, \ldots, n \tag{3.3}$$

The system has a solution that is equal to

$$x_i = \varphi_i(t) \tag{3.4}$$

which represents the vector of initial conditions of the orbit. This vector has six components and corresponds to

$$\mathbf{X} = [x, y, z, \dot{x}, \dot{y}, \dot{z}] \tag{3.5}$$

If the function φ_i shows the property

$$\varphi_i(0) = \varphi_i(T) \tag{3.6}$$

namely if the initial value of φ_i is equal to the value obtained after one period, hence $\varphi_i(t) = \varphi_i(t+T)$, then the function φ_i is periodic in t with an orbital period of T and $x_i = \varphi_i(t)$ is a periodic solution of the system of differential equations.

A function $x(t)$ is almost periodic if, for any given $\epsilon > 0$ there is an $E(\epsilon)$ such that for each real number α there is a T that satisfies

$$\alpha \leq T \leq \alpha + E(\epsilon) \tag{3.7}$$

and

$$|x(t+T) - x(t)| \leq \epsilon \quad \text{for all t} \tag{3.8}$$

If the conditions expressed by Equations (3.7) and (3.8) are satisfied, the function is almost periodic. Alternatively, it is possible to affirm that the function is periodic with an error equal to ϵ.

As previously mentioned, this thesis focuses exclusively on deriving and correcting periodic solutions, and does not cover non-periodic orbits in the CR3BP.

3.1 Numerical integration of initial conditions

To obtain a proper visual and numerical representation of periodic orbits in the CR3BP, it is necessary to integrate the EoMs using a set of initial conditions. In order for the orbit to be periodic, the initial conditions vector must also be periodic. If this condition is not met, the initial conditions can be corrected using differential correction algorithms, as detailed in Chapter 4. Assuming that the initial state vector can generate a periodic orbit, the EoMs can be numerically integrated using the `solve_ivp` integrator, that belongs to the `scipy.integrate` Python library. The selected integration method is the `DOP853`, i.e. the explicit Dormand-Price method. The reasons that led to this choice are as follows:

- `DOP853` is an eighth-order method, which is a higher order than the 4^{th} order Runge-Kutta methods already implemented in the `scipy.integrate` library. A higher order usually implies that fewer steps are required to meet the fixed tolerance;

- `DOP853` uses a variable step method, i.e., integration steps are distributed autonomously by the method, setting longer steps for a more regular path and shorter steps in corrispondence with events or discontinuities;
- `DOP853` is an explicit method, which is part of a class of methods more suited to non-stiff problems like the CR3BP compared to implicit methods.

The numerical integration of the EoMs takes place by setting both a relative tolerance `r_tol` and an absolute tolerances `a_tol` equal to 10^{-13}.

3.2 Analysis of orbit families

3.2.1 Lyapunov orbits

Lyapunov orbits are named after the Russian mathematician Aleksandr Lyapunov, whose research represents the basis for evaluating the stability of non-linear dynamical systems. The existence of such orbits is ensured by the *Lyapunov center theorem*, which in general guarantees the existence of periodic orbits near a central equilibrium point for bidimensional Hamiltonian systems [23]. In fact, small perturbations in the equilibrium point generate a family of periodic orbits for systems of this kind. In the context of the *Lyapunov* orbits, the equilibrium points are the collinear points. Planar orbits symmetrical with respect to the xz-plane extend around these points. This particular geometry results in an initial state vector with several infinitesimal components that can be approximated to zero without introducing significant errors, such that

$$\mathbf{X}_0 = [x_0, 0, 0, 0, \dot{y}_0, 0] \tag{3.9}$$

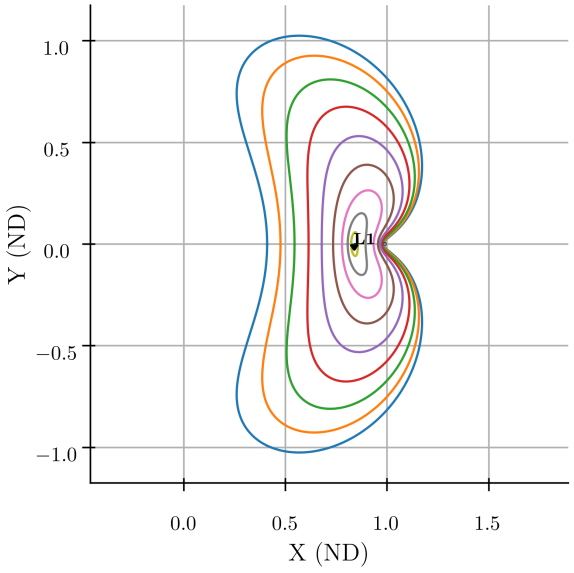


Figure 3.1: Overview of the Lyapunov orbits around L_1 in the Earth-Moon system

Figure 3.1 shows an overview of the orbits built around L_1 in the Earth-Moon system that belong to such family. More specifically, *Lyapunov* orbits built around L_1 ensure uninterrupted communication between the surfaces of both celestial bodies. *Lyapunov* orbits built around L_2 and L_3 allow a favorable access to the equatorial region of the closest body, respectively the secondary body and the primary body. Even if all these orbits can guarantee almost continuous coverage of the equatorial region, they are characterized by a significant high stability index and are therefore unstable. Figure 3.2 highlights the trend in the stability index of the orbits built around L_1 with the Jacobi constant value. This results in very high stationkeeping costs. If the perturbations were not counteracted, the trajectory would deviate from its nominal periodic orbit. This is the main reason why *Lyapunov* orbits have limited applicability. However, their significant instability makes them ideal candidates for low Δv staging orbits for both orbit insertion and departure from 2D orbits, e.g. *Distante Retrograde Orbits* as discussed in the next subsection. Therefore, *Lyapunov* orbits and *Lyapunov*-like orbits can be employed for the insertion along more stable orbits or for descent onto the surface of a celestial body.

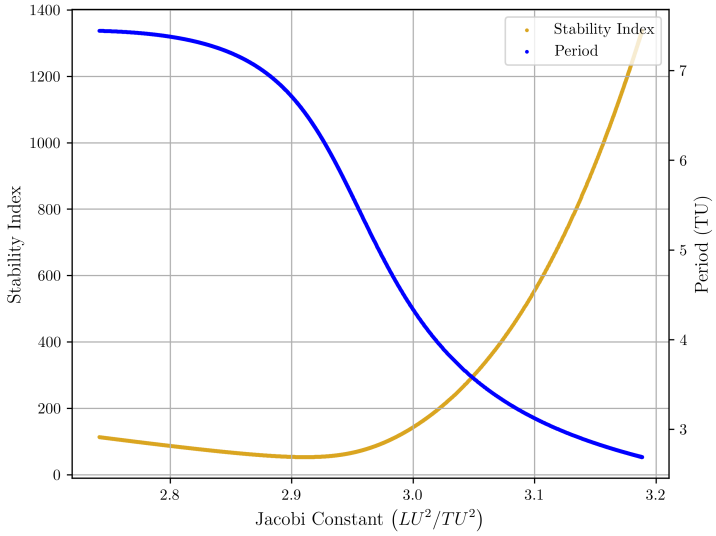


Figure 3.2: Stability plot of the Lyapunov orbit familiy around L_1 in the Earth-Moon system

3.2.2 Distant Retrograde Orbits

The *Distant Retrograde Orbits* (DROs) family is a second example of a group of planar orbits that exhibits symmetry with respect to the xz-plane. The focus of this thesis is on 2D DROs, although 3D DROs also exist. Due to this similarity with Lyapunov orbits, the initial condition vector for a generic DRO simplifies as showed by Equation (3.9). Analyzing the name of this family reveals two main features:

- *Distant* states that the DROs develop, on average, at a significant distance from the center of the secondary body, to the extent that they encompass both L_1 and

L_2. As can be seen in Figure 3.3, the amplitude of most of the orbits higher than the distance between the two Lagrange points;

- *Retrograde* provides information regarding the motion of the spacecraft. In fact, the spacecraft orbits around the smaller celestial body in the opposite direction to that of the revolution of the secondary body around the primary body. However, this is only true in the synodic reference frame, while in the inertial reference frame the motion of the spacecraft is prograde.

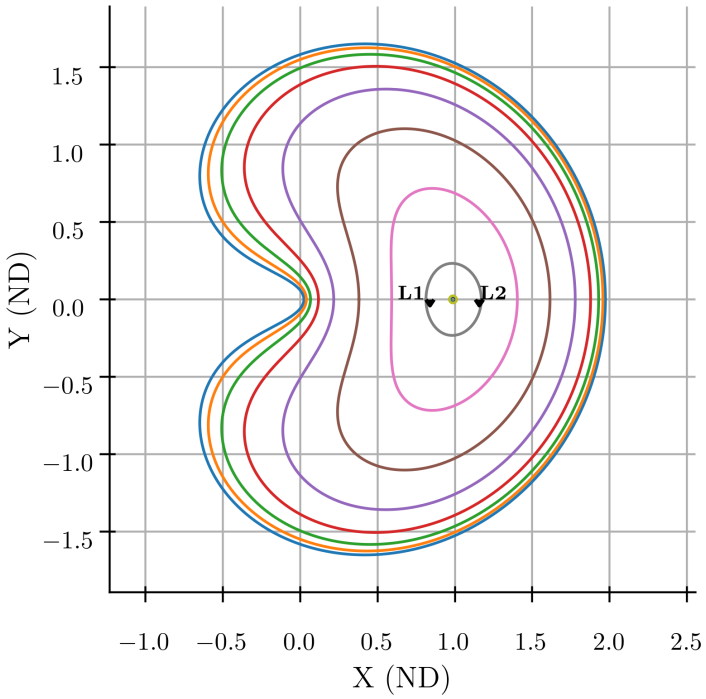


Figure 3.3: Overview of the DROs in the Earth-Moon system

Another characteristic of DROs is their stability, which is caused by the interaction with the two aforementioned Lagrange points. This feature can be observed in Figure 3.4, that shows the trend of the stability index, as a function of Jacobi constant. Because the value of the stability index is always very close to unity, DROs are marginally stable. Several studies confirm that a spacecraft moving along a DRO around the Moon requires few to no stationkeeping maneuvres over several decades [24,25]. Therefore, combining a very small Δv required for stationkeeping and the relatively high stability, DROs are highly recommended as longterm parking orbits. It is indeed possible to transition from a DRO to Halo orbits or Near Rectilinear Halo Orbits (NRHOs) with an equally low propulsive cost.

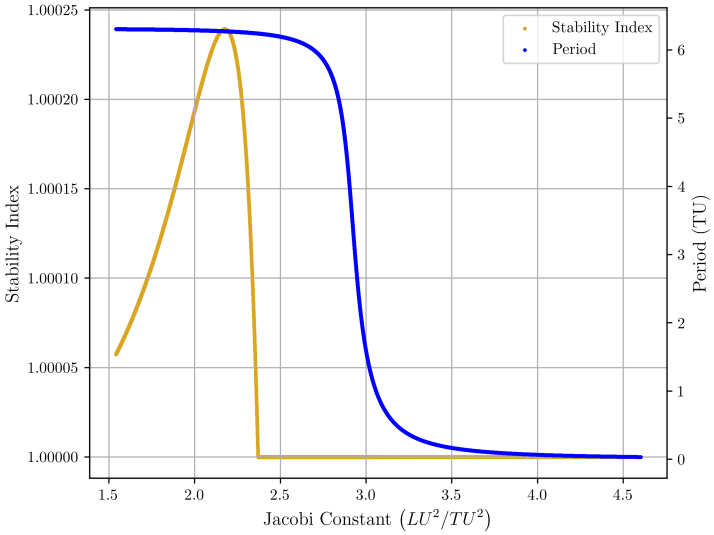


Figure 3.4: Stability plot of the DROs family in the Earth-Moon system

Although DROs offer more applications than Lyapunov orbits, only one mission has been carried out using a DRO to date: the NASA Artemis I mission. The two main elements of the mission were the Space Launch System (SLS) rocket and the unmanned Orion module. The objectives of the mission were to test both of the components and to study the permanence of Orion in a DRO around the Moon [26]. This test was performed in anticipation of the upcoming Artemis missions. Additionally, the previous *Asteroid Redirect Mission* (ARM) was scheduled by NASA to study the use of DROs. The plan was to intercept an asteroid in the vicinity of the Earth and redirect it into a lunar DRO. However, the mission was cancelled in 2017 [27].

3.2.3 Short Period and Long Period Orbits

The group of planar orbits ends with two orbit families built around the equilateral points: *Short Period Orbits* (SPOs) and *Long Period Orbits* (LPOs). Both families are characterized by a variable geometry. Figure 3.5 and Figure 3.6 show that orbits near the reference equilateral point have a quasi-elliptic shape. As the amplitude and period of the orbits increase, they begin to curve around the primary, incorporating L_3 too. This is particularly true for SPOs, because LPOs tend to assume more complex shapes and do not only to curve around the primary.

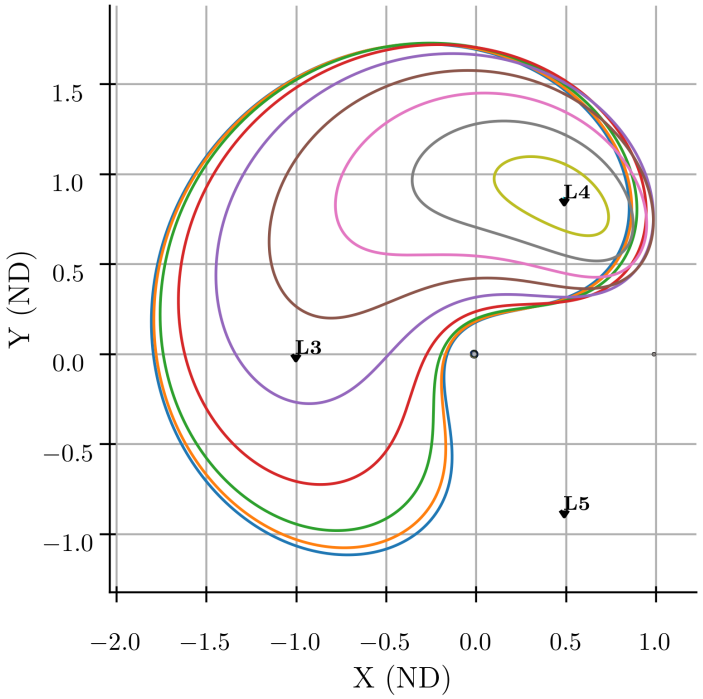


Figure 3.5: Overview of the SPOs family around L_4 in the Earth-Moon system

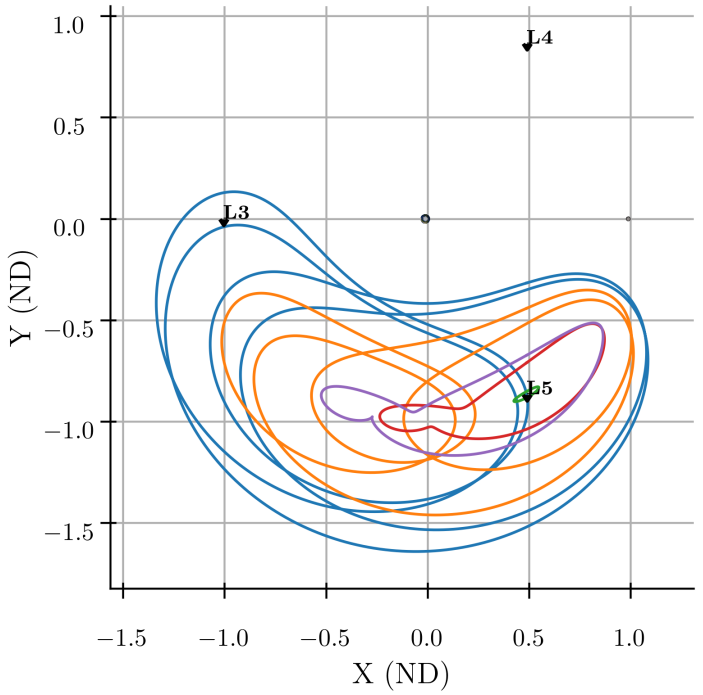


Figure 3.6: Overview of the LPOs family around L_5 in the Earth-Moon system

As their classification suggests, the main difference relative to SPOs and LPOs, is their orbital period. SPOs have a period that is comparable to the revolution period of both primaries, whereas LPOs are distinguished by a significantly longer period with respect to the primaries. Figure 3.7 shows how a generic LPO built around L_4 requires more

than one passage around the Lagrange point to come back to its starting point.

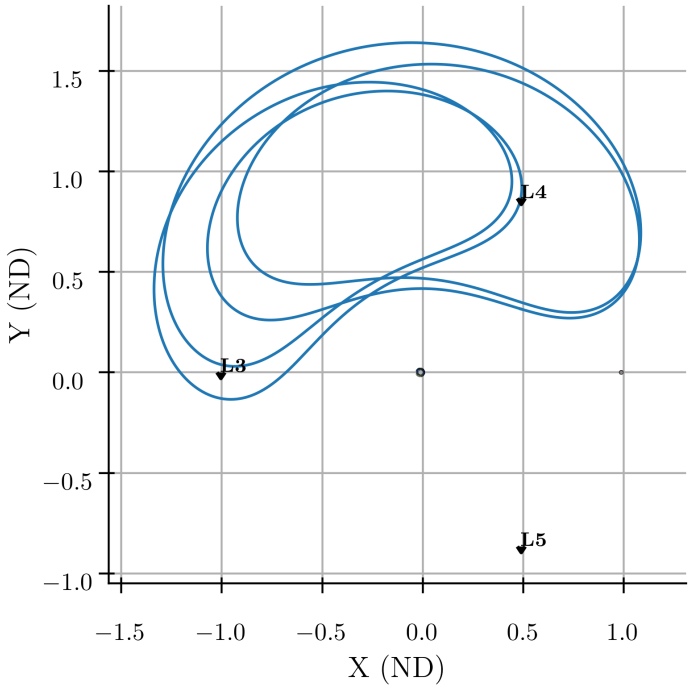


Figure 3.7: Generic LPO built around L_4 in the Earth-Moon system

Unlike Lyapunov orbits, SPOs and LPOs should be stable orbits, due to the stability of the equilateral points. However, Figure 3.8 shows that SPOs are stable orbits, with almost every orbit built around L_4 having a stability index practically equal to one. But, Figure 3.9 highlights that LPOs are indeed marginally stable, with only a restricted number of orbits being stable.

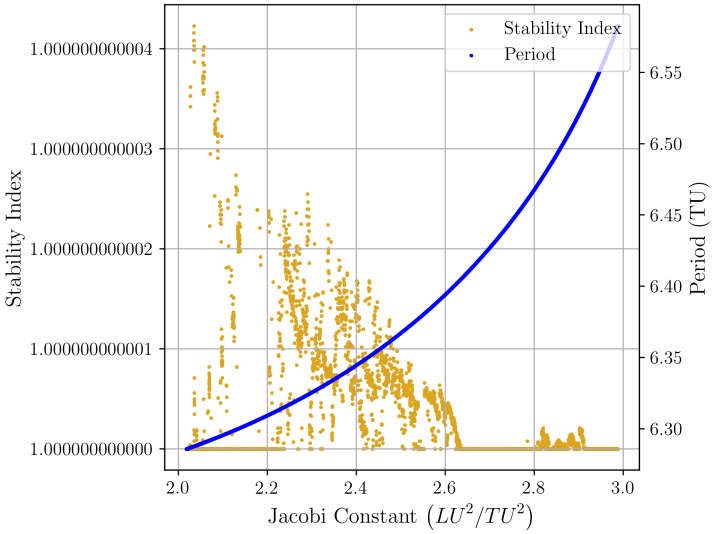


Figure 3.8: Stability plot of the SPOs orbit family around L_4 in the Earth-Moon system

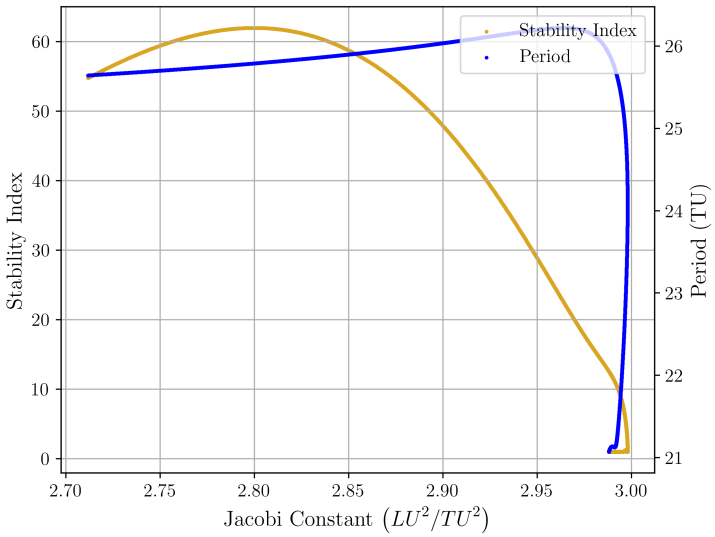


Figure 3.9: Stability plot of the LPOs orbit family around L_5 in the Earth-Moon system

Therefore, these orbits can be useful for observing missions of the outer layers of a the celestial body or studying the interplanetary material that accumulates in these regions. One concept of a mission that will use these kind of orbits is the *Helio-Magnetism Investigation from the Sun to Earth* (HeMISE) mission. This mission will consist of two twin satellites that will investigate the Sun's photosferic and magnetic fields, as well as the propagation of electromagnetic waves along the Sun-Earth line [28].

3.2.4 Halo orbits

This orbit family is named after the term "halo", which was first used by Farquhar [29] in 1973 to describe a key characteristic of this orbit family: when observed from one of the two primaries, a generic Halo orbit appears to create a halo around the other primary. This happens even if the orbit does not extends exactly around the celestial body, and it is not planar. Figure 3.10 shows a representation of a generic Halo orbit built around L_2 in the Earth-Moon system projected along the *yz-plane*. From this view, the orbit happens to look planar, even if it is not, and the behaviour described earlier can be observed. As mentioned before, Halo orbits are not planar, but three-dimensional. Indeed, they can be distinguished based on their most distant position along the z-axis, a quantity also known as *z-amplitude*, A_z. Other than A_z, Halo orbits can be classified as *Northern* Halo orbits or *Southern* Halo orbits, depending on their spatial orientation.

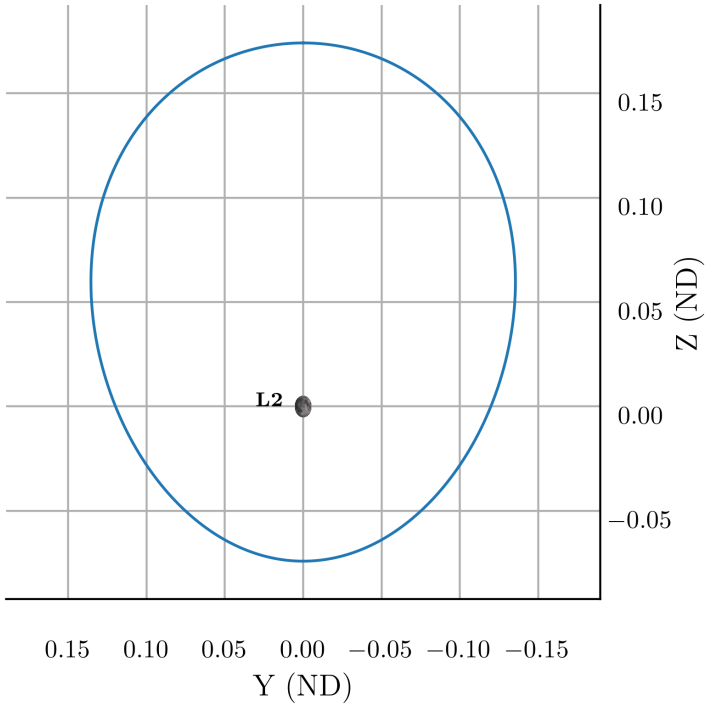


Figure 3.10: Generic Halo Northern orbit around L_2 in the Earth-Moon system, shown in the yz-plane

In particular, starting from a Northern Halo orbit it is always possible to identify the corresponding Southern Halo orbit by simply inverting the sign of the z component in the initial state vector, and viceversa. The state vector for a generic Halo orbit is

$$\mathbf{X}_0 = [x_0, 0, z_0, 0, \dot{y}_0, 0] \tag{3.10}$$

and it is perpendicular to the xz-plane. As a consequence, the resultant orbit will also be symmetric with respect to the xz-plane. Figure 3.11 highlights this characteristic.

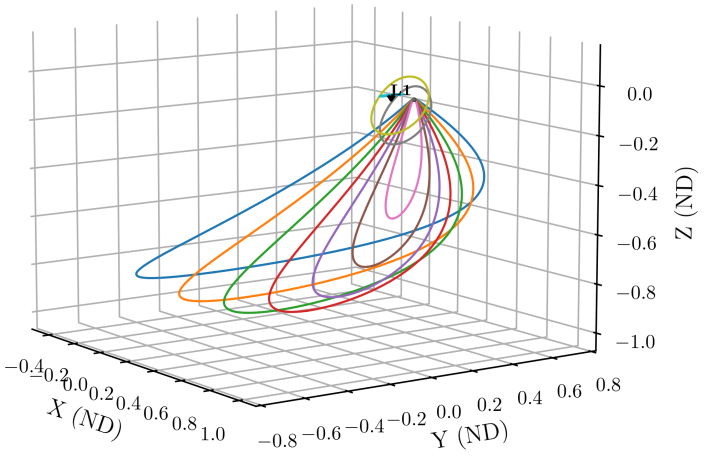


Figure 3.11: Overview of the Halo Southern orbits family around L_1 in the Earth-Moon system

Figure 3.11 shows an overview of Southern Halo orbits built around L_1 in the Earth-Moon system and highlights another peculiar characteristic of Halo orbits. As they approach the closer celestial body, they become more elongated and are known as *Near Rectilinear Halo Orbits* (NRHOs). Furthermore, a clear trend for the orbital period can be traced: as the orbits approach the closer celestial body, the orbital period decreases.

Finally, it is useful to analyze the stability of such orbits. The three groups of orbits, each built around one of the collinear points, exhibit different trends for the stability index. As shown in Figure 3.12, orbits built around L_1 are mainly unstable, except for a small group of orbits having a stability index very close to unity.

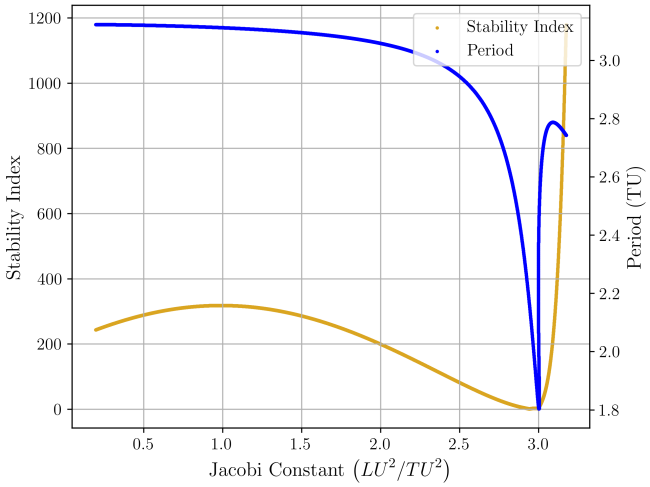


Figure 3.12: Stability plot of the Halo Northern orbit family around L_1 in the Earth-Moon system

Similarly, the majority of orbits built around L_2 are unstable, but the group of stable orbits in much wider that the previous one, as can be observed in Figure 3.13. It should also be noted that orbits with a stability index of one fall into the NRHO group.

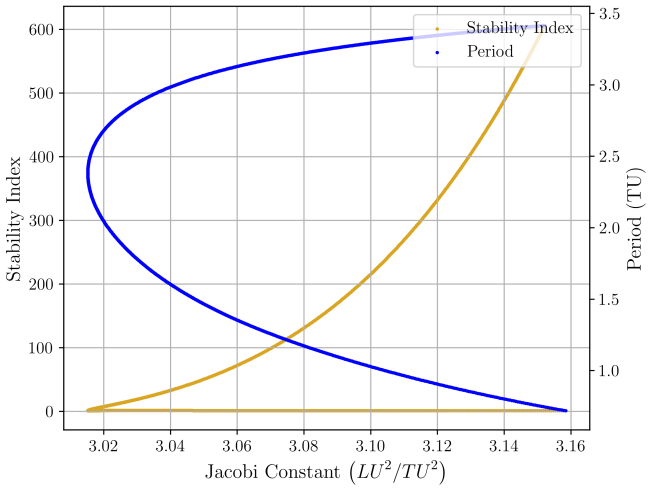


Figure 3.13: Stability plot of the Halo Northern orbit family around L_2 in the Earth-Moon system

Orbits built around L_3 are mostly marginally stable, with a group of stable orbits that share the same characteristics as those built around L_2, as can be observed in Figure 3.14.

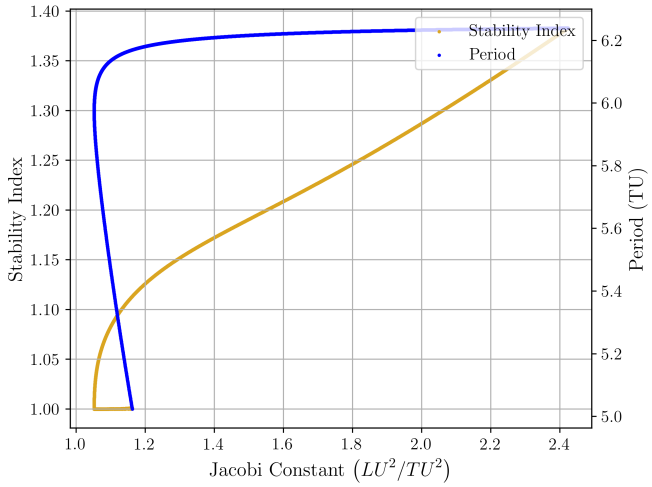


Figure 3.14: Stability plot of the Halo Northern orbit family around L_3 in the Earth-Moon system

Halo orbits have a wide range of applications. Thanks to their particular geometry, which allows for uninterrupted communications between the two celestial bodies, and their extended orbital periods. Table 3.1 lists some of the missions that have used Halo orbits to meet their objectives.

Table 3.1: Missions using Halo orbits

Mission (Agency, Year)	**Halo Orbit Type**	**Main Objective**
ISEE-3/ICE (NASA/ESA, 1978)	Sun–Earth L1	Solar wind/magnetotail science
WIND (NASA, 1994)	Sun–Earth L1	Solar wind and interplanetary plasma & fields
SOHO (ESA/NASA, 1995)	Sun–Earth L1	Solar physics and helioseismology, space weather monitoring.
JWST (NASA/ESA/CSA, 2021)	Sun–Earth L2	Infrared astronomy: first light, galaxy assembly, exoplanet atmospheres
ACE (NASA, 1997)	Sun–Earth L1	Energetic particles/solar wind; real-time space weather
WMAP (NASA, 2001)	Sun–Earth L2	Cosmic Microwave Background anisotropy and cosmology
Genesis (NASA, 2001)	Sun–Earth L1	Solar-wind *sample return* for solar composition studies
Queqiao-1 (CNSA, 2018)	Earth–Moon L2	Comms relay for Chang'e-4, far-side support and radio experiments
Spektr-RG (Roscosmos/DLR, 2019)	Sun–Earth L2	X-ray astronomy, all-sky surveys
Euclid (ESA, 2023)	Sun–Earth L2	Cosmology
Artemis III (NASA, 2027)	Earth-Moon L2 NRHO	Lunar environment & magnetotail dynamics

ISEE and WIND missions were the first to employ Halo orbits built in the Sun-Earth system to conduct in-depth research on solar activity. Among future missions are the

Artemis missions and the assembly of the Lunar Gateway. The Artemis missions aim to return humans to the Moon's surface, while the creation of the Lunar Gateway would lead to the first orbiting station beyond low Earth orbit.

3.2.5 Axial and Vertical orbits

Another example of three-dimensional orbit families are the *Axial* and *Vertical* orbit families. Both these families exist in proximity to the collinear and the equilateral points. Vertical orbits built around the collinear points originate from a bifurcation of the Lyapunov orbits. In fact, these orbits are also referred to as *Vertical Lyapunov* orbits, due to their link with the Lyapunov orbit family. The origin of these orbits can be recognised by analyzing their geometry. Figure 3.15 shows a group of generic Vertical orbits built around L_1 in the Earth-Moon system. It can be noted that initial orbits have a two-lobe geometry that develops primarily along the z-direction. As the orbital period increases, the two-lobe structure of the orbits is mantained, but the lobes tend to converge towards the xy-plane surrounding the reference celestial body.

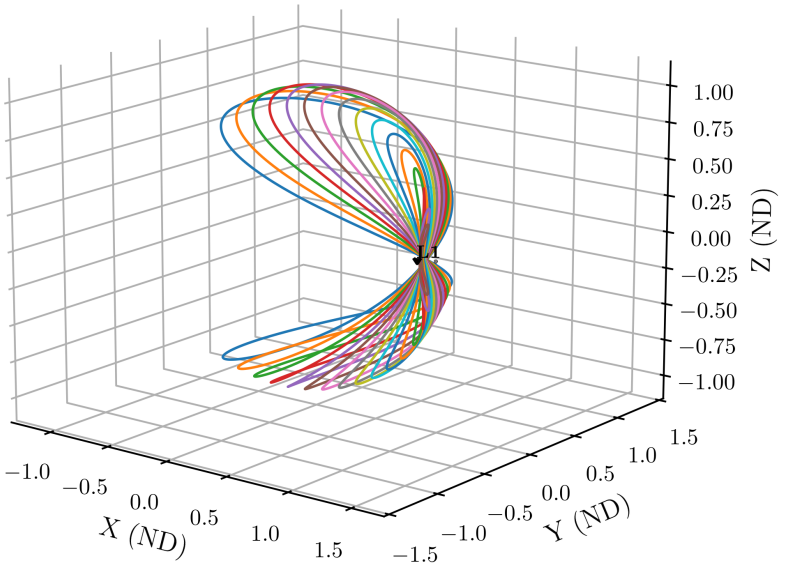


Figure 3.15: Overview of the Vertical orbits family around L_1 in the Earth-Moon system

Axial orbits are characterised by a geometry similar to that of Vertical orbits. For orbits built around L_1 and L_2 in particular, a structure consisting of two lobes extending outside the xy-plane can be observed. However, these orbits rotate around the y-axis. Consequently, there is a transition from orbits ensuring a better coverage of polar regions of the reference celestial body to quasi-planar orbits that guarantee primarily coverage of the equatorial region. This behaviour can be observed in Figure 3.16.

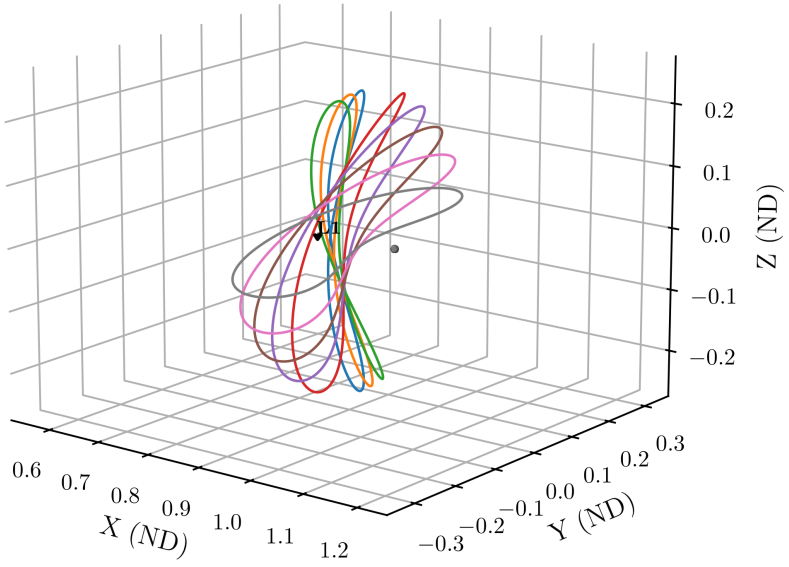


Figure 3.16: Overview of the Axial orbits family around L_1 in the Earth-Moon system

Axial orbits built around the equilateral points exhibit a less intuitive evolution of their geometry. Orbits with shorter orbital periods assume an elliptic shape in the vicinity of the primary body. As they approach the equilateral point around which the orbits are built, the orbits' structure evolves in the previously observed two-lobe geometry with greater extension outside the xy-plane. Figure 3.17 illustrates this characteristic.

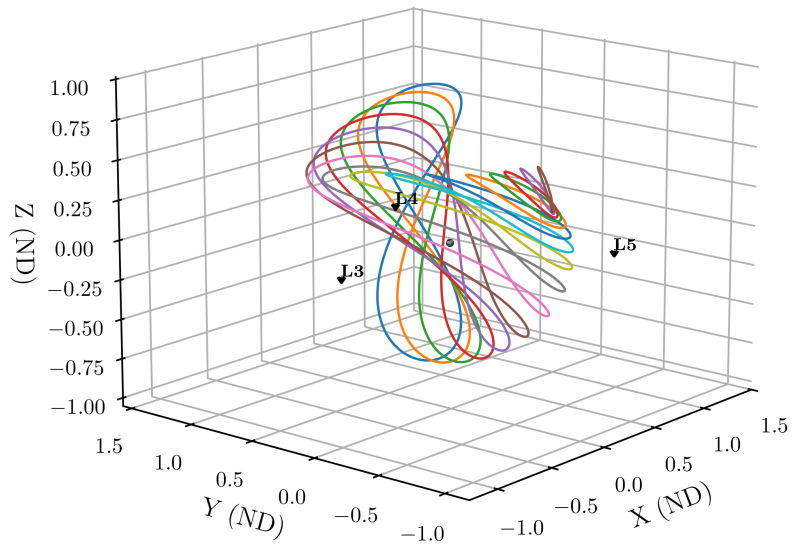


Figure 3.17: Overview of the Axial orbits family around L_4 in the Earth-Moon system

Both families are characterised by a predictable trend in the stability index: orbits formed around the collinear points tend to be unstable, whereas most of those in the vicinity of the equilateral points are marginally stable, with a variable number of stable orbits. Figure 3.18 and Figure 3.19 confirm that around the collinear points the orbits are broadly unstable, with stability index values in the hundreds.

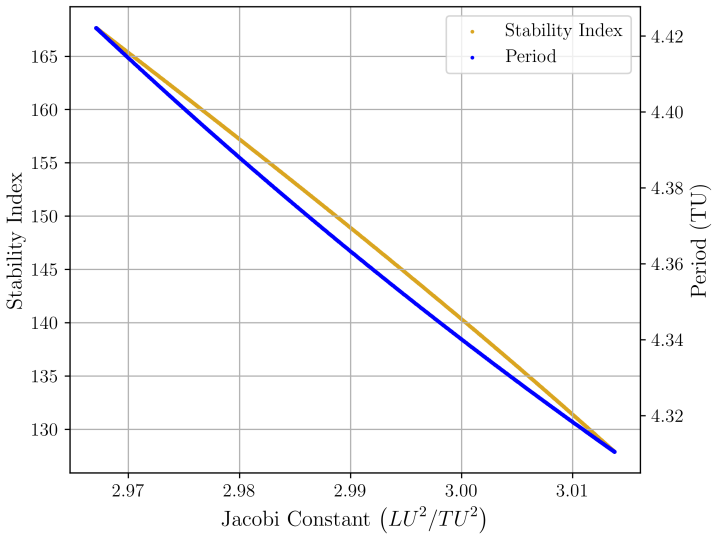


Figure 3.18: Stability plot of the Axial orbit family around L_2 in the Earth-Moon system

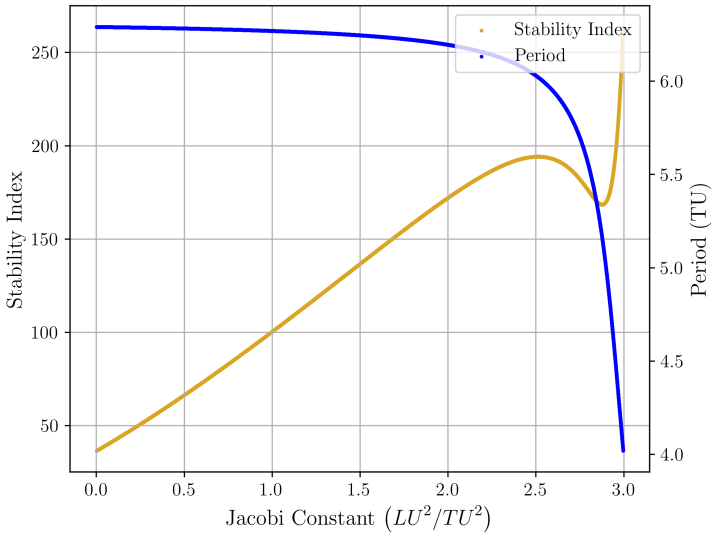


Figure 3.19: Stability plot of the Vertical orbit family around L_1 in the Earth-Moon system

Figure 3.20 and Figure 3.21 show a similar behaviour to these observed for the SPOs and LPOs in Subsection 3.2.3. In fact, Figure 3.20 shows that orbits Axial orbits built around L_4 are marginally stable, with a consistent number of orbits having a stability index equal to one, hence being stable. On the other hand, Figure 3.21 points out that orbits built around L_5 are also marginally stable, but the the value of the stability index varies less and stays very close to unit.

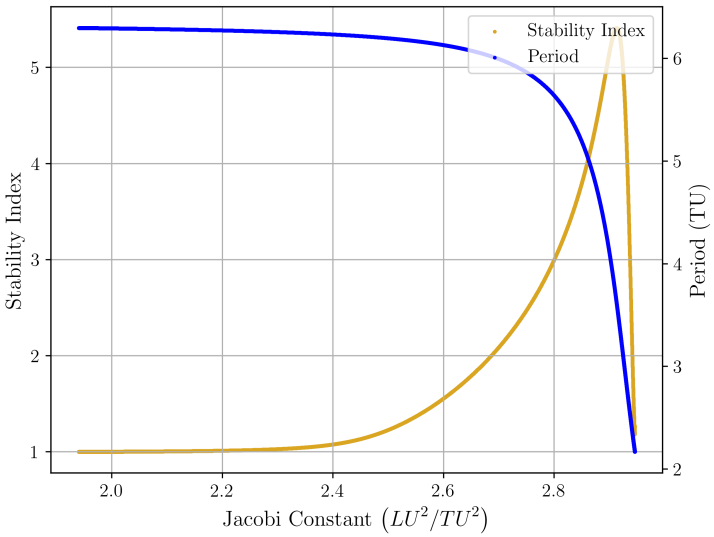


Figure 3.20: Stability plot of the Axial orbit family around L_4 in the Earth-Moon system

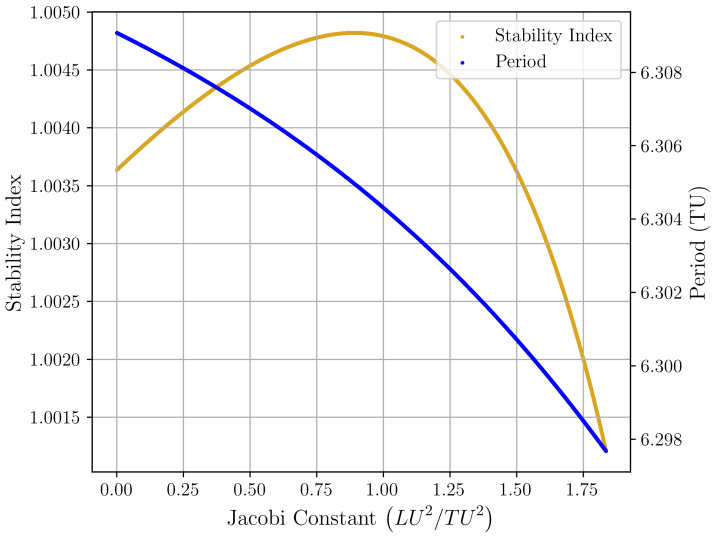


Figure 3.21: Stability plot of the Vertical orbit family around L_5 in the Earth-Moon system

Employment possibilities of the aforementioned orbits are strictly connected to their structure. Both Axial and, in particular, Vertical orbits around collinear points can be taken into consideration for observing the polar regions of the reference celestial body. Axial orbits extending around the equilateral points can be used as transfer orbits due to their proximity to bifurcation points with other orbit families.

3.2.6 Other orbits

To complete the analysis of the periodic solutions of the CR3BP, two families of three-dimensional orbits are considered: *Butterfly* orbits and *Dragonfly* orbits.

Similarly to *Vertical* orbits, Butterfly orbits also derive from a bifurcation that involves an already known orbit family: Halo orbits. This link with Halo orbits explains why even Butterfly orbits can be distinguished as either *Northern* or *Southern* orbits. Taking a look at Figure 3.22, that shows two generic Northern Butterfly orbits in the Earth-Moon system, it can be noted that the geometry of these orbits is more complex that that of Halo orbits. In particular, Butterfly orbits that are closer to the secondary body have a U-shape and are symmetric with respect to the xy-plane. As the dimensions of the orbit increase, the shape becomes more complex with two large lobes developing around both L_1 and L_2.

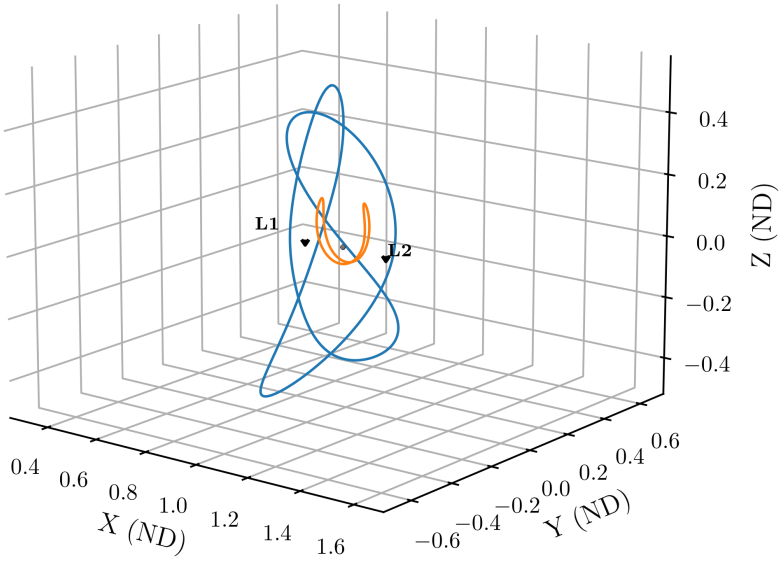


Figure 3.22: Northern Butterfly orbits in the Earth-Moon system

Dragonfly orbits also originate from a bifurcation point of different families, mainly Halo orbits built around L_2. This connection with Halo orbits can be observed in Figure 3.23, which shows two random Northtern Dragonfly orbits in the Earth-Moon system. The orbit closer to the Moon, the orange orbit in Figure 3.23, happens to be very close to the Northern Halo orbits seen before. Moving away from the Moon, geometry becomes more complex and the orbit assumes a two-lobe configuration, with the upper lobe being very extended.

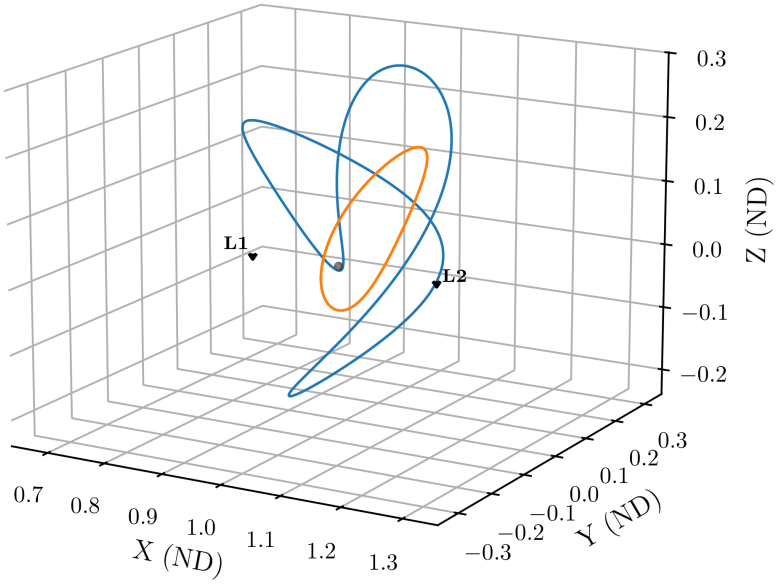


Figure 3.23: Northern Dragonfly orbits in the Earth-Moon system

Both of these orbit families are generally unstable like the Halo orbits from which they derive. This characteristic is confirmed by Figure 3.24 and Figure 3.25, where it can be seen that both families have stability index values that are high enough to consider them unstable. Figure 3.24 also shows cusp points, which can be attributed to the bifurcation points from which these orbits originate.

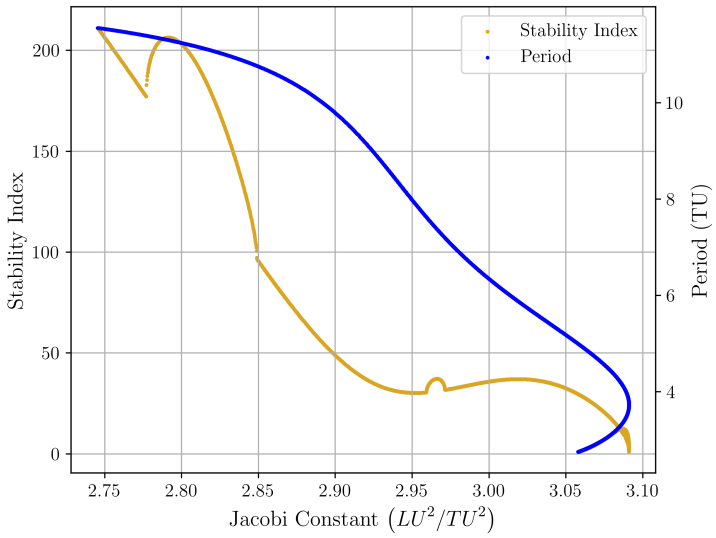


Figure 3.24: Stability plot of the Butterfly Northern orbit family in the Earth-Moon system

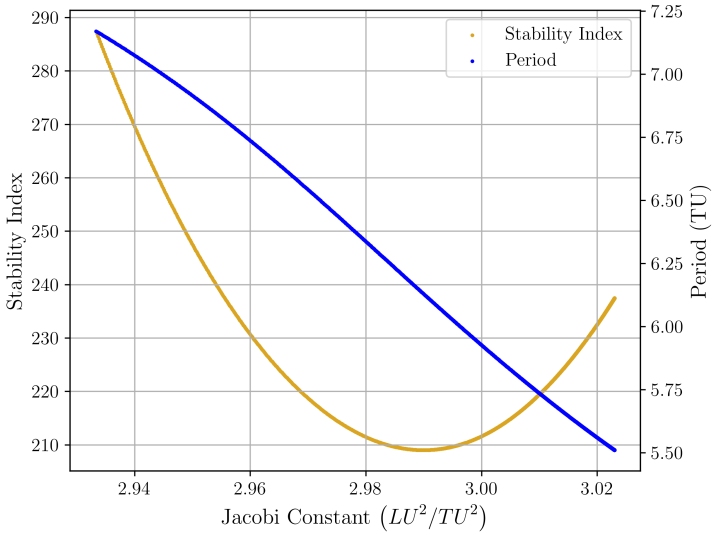


Figure 3.25: Stability plot of the Dragonfly Northern orbit family in the Earth-Moon system

However, this instability makes them attractive for use as transfer trajectories with low Δv. Moreover, Butterfly orbits have attracted a lot of interest due to their peculiar shape that provides excellent visibility of the lunar south pole. As well as providing a favorable observation window of the polar region, they also the guarantee of satisfactory communications with Earth.

Dynamical System Theory and Differential Correction

Dynamical Systems Theory (DST) was first introduced in the 19th century by Henri Poincaré and it is also known as *chaos theory* [30]. This theory can be used to study complex, time-evolving systems analyzing their response to the initial conditions. In mathematics, a dynamical system that is highly sensitive to the intial conditions is referred to as chaotic. An example of such this type of system is a generic N-body system. In fact, due to the presence of several celestial bodies that gravitationally interact between each other, a slight change in the initial conditions of a periodic orbit results in a completely different orbit that is likely not periodic. DST not only helps deduce the governing equations of the chaotic motion of a body in the N-body problem, but provides also a procedure to discover and validate orbit families that can be used for interplanetary transfer or as parking orbits, such as for *Genesis* [31], JWST [32] and CAPSTONE [33].

Section 4.1 shows how the *State Transition Matrix* (STM) is derived and its major charachteristics explained. This matrix plays a fundamental role due to the sensitivity of the chaotic systems to the initial conditions. The STM describes how a point evolves along the trajectory over time and it is necessary for the *differential correction* algorithms to work properly. These algorithms will be analized in detail in Section 4.2, with a particular focus on their use in obtaining periodic initial conditions in the CR3BP.

4.1 State Transition Matrix

It is assumed that there is a trajectory $\phi(t, t_0)$ going from a starting point $x(t_0)$ to a final point $x(t)$ that respects the dynamics expressed by

$$\dot{x} = f(x), \qquad x \in \mathbb{R}^6 \tag{4.1}$$

Such trajectory corresponds to the flow map of the dynamical system expressed by Equation (4.1) and allows to map points from their location at the initial instant t_0 to their position at time t. It is immediate to verify that the flow map satisfies the Equation (4.1)

$$\frac{d\phi(t; x_0)}{dt} = f(\phi(t; x_0)), \qquad \text{with} \quad \phi(t_0; x_0) = x_0 \tag{4.2}$$

A reference trajectory $\bar{x}(t)$ connecting the points $\bar{x}_0$ and $\bar{x}_1$ is assumed to exist. An alternative trajectory starting from a perturbed initial point $\bar{x}_0 + \delta\bar{x}_0$, will evolve with

the displacement

$$\delta\bar{x}(t) = \phi(t; \bar{x}_0 + \delta\bar{x}_0) - \phi(t; \bar{x}_0) \tag{4.3}$$

Performing a Taylor series expansion of Equation (4.3) yields

$$\delta\bar{x}(t) = \frac{\partial\phi(t; \bar{x}_0)}{\partial x_0}\delta\bar{x}_0 + \text{higher order terms} \tag{4.4}$$

where the matrix $\frac{\partial\phi(t;\bar{x}_0)}{\partial x_0}$ is denoted as *state transition matrix* (STM), or *sensitivity matrix*. The use of the STM for linear systems leads to an exact analytical solution, but introduces an approximation when it comes to non-linear systems. Hence, in the CR3BP the STM performs a linear mapping between small initial and final displacements. Therefore, it is possible to quantify the sensitivity of the final state to a variation in the initial state. Due to this property, the STM plays a key role in *differential correction* algorithms that provide initial conditions for periodic orbits. Equation (4.4) can then be written as

$$\delta\bar{x} = \mathbf{\Phi}(t, t_0)\delta\bar{x}_0 \tag{4.5}$$

with the STM designated by $\mathbf{\Phi}(t, t_0)$. This matrix can also be expressed using the partial derivatives of the state vector elements computed at a generic time t with respect to the same quantities evaluated at the initial instant t_0.

$$\mathbf{\Phi}(t, t_0) = \begin{bmatrix} \frac{\partial x}{\partial x_0} & \frac{\partial x}{\partial y_0} & \frac{\partial x}{\partial z_0} & \frac{\partial}{\partial \dot{x}_0} & \frac{\partial x}{\partial \dot{y}_0} & \frac{\partial x}{\partial \dot{z}_0} \\ \frac{\partial y}{\partial x_0} & \frac{\partial y}{\partial y_0} & \frac{\partial y}{\partial z_0} & \frac{\partial y}{\partial \dot{x}_0} & \frac{\partial y}{\partial \dot{y}_0} & \frac{\partial y}{\partial \dot{z}_0} \\ \frac{\partial z}{\partial x_0} & \frac{\partial z}{\partial y_0} & \frac{\partial z}{\partial z_0} & \frac{\partial z}{\partial \dot{x}_0} & \frac{\partial z}{\partial \dot{y}_0} & \frac{\partial z}{\partial \dot{z}_0} \\ \frac{\partial \dot{x}}{\partial x_0} & \frac{\partial \dot{x}}{\partial y_0} & \frac{\partial \dot{x}}{\partial z_0} & \frac{\partial \dot{x}}{\partial \dot{x}_0} & \frac{\partial \dot{x}}{\partial \dot{y}_0} & \frac{\partial \dot{x}}{\partial \dot{z}_0} \\ \frac{\partial \dot{y}}{\partial x_0} & \frac{\partial \dot{y}}{\partial y_0} & \frac{\partial \dot{y}}{\partial z_0} & \frac{\partial \dot{y}}{\partial \dot{x}_0} & \frac{\partial \dot{y}}{\partial \dot{y}_0} & \frac{\partial \dot{y}}{\partial \dot{z}_0} \\ \frac{\partial \dot{z}}{\partial x_0} & \frac{\partial \dot{z}}{\partial y_0} & \frac{\partial \dot{z}}{\partial z_0} & \frac{\partial \dot{z}}{\partial \dot{x}_0} & \frac{\partial \dot{z}}{\partial \dot{y}_0} & \frac{\partial \dot{z}}{\partial \dot{z}_0} \end{bmatrix} = \left[\begin{array}{c|c} \mathbf{\Phi}_{\rho\rho} & \mathbf{\Phi}_{\rho V} \\ \hline \mathbf{\Phi}_{\rho V} & \mathbf{\Phi}_{VV} \end{array}\right] \tag{4.6}$$

The STM is a 6×6 matrix, within which four 3×3 submatrices of partial derivatives can be recognised. More specifically, the subscripts ρ and V respectively represent position and velocity. Differentiating Equation (4.2) with respect to x_0 the relation that describes how the STM evolves along the trajectory is obtained

$$\frac{d}{dt}\frac{\partial\phi(t; \bar{x}_0)}{\partial x_0} = \mathbf{A}(t)\frac{\partial\phi(t; \bar{x}_0)}{\partial x_0} \tag{4.7}$$

Recalling the definition of the STM, Equation (4.7) can be reformulated as follows

$$\dot{\mathbf{\Phi}}(t, t_0) = \mathbf{A}(t)\mathbf{\Phi}(t, t_0) \tag{4.8}$$

where the matrix $\mathbf{A}(t)$ is the Jacobian matrix of the system.

This last matrix is obtained by recalling Equation (2.25), which defines the pseudo-potential U, and expanding it in a Taylor series. Taking into consideration an expansion

with respect to the variable x

$$\begin{aligned}\frac{\partial U}{\partial x} = &\frac{\partial}{\partial x}\left(\frac{\partial U}{\partial x}\right)x + \frac{\partial}{\partial y}\left(\frac{\partial U}{\partial x}\right)y + \frac{\partial}{\partial z}\left(\frac{\partial U}{\partial x}\right)z + \frac{\partial}{\partial \dot{x}}\left(\frac{\partial U}{\partial x}\right)\left(\frac{dx}{dt}\right) \\ &+ \frac{\partial}{\partial \dot{y}}\left(\frac{\partial U}{\partial x}\right)\left(\frac{dy}{dt}\right) + \frac{\partial}{\partial \dot{z}}\left(\frac{\partial U}{\partial x}\right)\left(\frac{dz}{dt}\right)\end{aligned} \tag{4.9}$$

However, since U is not a function of the velocities $(\dot{x}, \dot{y}, \dot{z})$, Equation (4.9) becomes

$$\begin{aligned}\frac{\partial U}{\partial x} &= \frac{\partial}{\partial x}\left(\frac{\partial U}{\partial x}\right)x + \frac{\partial}{\partial y}\left(\frac{\partial U}{\partial x}\right)y + \frac{\partial}{\partial z}\left(\frac{\partial U}{\partial x}\right)z \\ &= \left(\frac{\partial^2 U}{\partial x^2}\right)x + \left(\frac{\partial^2 U}{\partial x \partial y}\right)y + \left(\frac{\partial^2 U}{\partial x \partial z}\right)z\end{aligned} \tag{4.10}$$

Repeating the same process for the variables y and z yields

$$\frac{\partial U}{\partial y} = \left(\frac{\partial^2 U}{\partial x \partial y}\right)x + \left(\frac{\partial^2 U}{\partial^2 y}\right)y + \left(\frac{\partial^2 U}{\partial y \partial z}\right)z \tag{4.11}$$

$$\frac{\partial U}{\partial z} = \left(\frac{\partial^2 U}{\partial x \partial z}\right)x + \left(\frac{\partial^2 U}{\partial y \partial z}\right)y + \left(\frac{\partial^2 U}{\partial^2 z}\right)z \tag{4.12}$$

By substituting Equations (4.10), (4.11) and (4.12) into Equation (2.26), the EoMs can be expressed as follows

$$\begin{cases}\dfrac{d^2x}{dt^2} - 2\dfrac{dy}{dt} = \left(\dfrac{\partial^2 U}{\partial x^2}\right)x + \left(\dfrac{\partial^2 U}{\partial x \partial y}\right)y + \left(\dfrac{\partial^2 U}{\partial x \partial z}\right)z \\ \dfrac{d^2y}{dt^2} + 2\dfrac{dx}{dt} = \left(\dfrac{\partial^2 U}{\partial x \partial y}\right)x + \left(\dfrac{\partial^2 U}{\partial^2 y}\right)y + \left(\dfrac{\partial^2 U}{\partial y \partial z}\right)z \\ \dfrac{d^2z}{dt^2} = \left(\dfrac{\partial^2 U}{\partial x \partial z}\right)x + \left(\dfrac{\partial^2 U}{\partial y \partial z}\right)y + \left(\dfrac{\partial^2 U}{\partial^2 z}\right)z\end{cases} \tag{4.13}$$

whose partial derivatives are made explicit hereafter

$$\begin{aligned}\frac{\partial^2 U}{\partial x^2} &= 1 - \frac{(1-\mu)}{r_1^3} - \frac{\mu}{r_2^3} + \frac{3(1-\mu)(x+\mu)^2}{r_1^5} + \frac{3\mu(x-1+\mu)^2}{r_2^5} \\ \frac{\partial^2 U}{\partial x \partial y} &= \frac{3(1-\mu)(x+\mu)y}{r_1^5} + \frac{3\mu(x-1+\mu)y}{r_2^5} \\ \frac{\partial^2 U}{\partial x \partial z} &= \frac{3(1-\mu)(x+\mu)z}{r_1^5} + \frac{3\mu(x-1+\mu)z}{r_2^5} \\ \frac{\partial^2 U}{\partial y^2} &= 1 - \frac{(1-\mu)}{r_1^3} - \frac{\mu}{r_2^3} + \frac{3(1-\mu)y^2}{r_1^5} + \frac{3\mu y^2}{r_2^5} \\ \frac{\partial^2 U}{\partial y \partial z} &= \frac{3(1-\mu)yz}{r_1^5} + \frac{3\mu yz}{r_2^5} \\ \frac{\partial^2 U}{\partial z^2} &= -\frac{(1-\mu)}{r_1^3} - \frac{\mu}{r_2^3} + \frac{3(1-\mu)z^2}{r_1^5} + \frac{3\mu z^2}{r_2^5}\end{aligned} \tag{4.14}$$

Rewriting the system shown by Equation (4.13) in matrix form reveals the appearence of the system's Jacobian matrix $\mathbf{A}(t)$

$$\frac{d\mathbf{X}}{dt} = \frac{d}{dt}\begin{Bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \left(\frac{\partial^2 U}{\partial x^2}\right) & \left(\frac{\partial^2 U}{\partial x \partial y}\right) & \left(\frac{\partial^2 U}{\partial x \partial z}\right) & 0 & 2 & 0 \\ \left(\frac{\partial^2 U}{\partial x \partial y}\right) & \left(\frac{\partial^2 U}{\partial y^2}\right) & \left(\frac{\partial^2 U}{\partial y \partial z}\right) & -2 & 0 & 0 \\ \left(\frac{\partial^2 U}{\partial x \partial z}\right) & \left(\frac{\partial^2 U}{\partial y \partial z}\right) & \left(\frac{\partial^2 U}{\partial z^2}\right) & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{Bmatrix} = \mathbf{A}(t)\mathbf{X} \tag{4.15}$$

Similarly to the STM, $\mathbf{A}(t)$ is a 6×6 matrix composed by four 3×3 square submatrices

$$\mathbf{A}(t) = \left[\begin{array}{c|c} \mathbf{0} & \mathbf{I} \\ \hline \mathbf{U} & \mathbf{\Omega} \end{array}\right] \tag{4.16}$$

Equation (4.8) shows that the time evolution of the STM is described by a set of 36 ordinary differential equations (ODEs), via the matrix $\mathbf{A}(t)$. However, the Jacobian matrix is non-autonomous, but it depends on a specific solution or reference trajectory. Therefore, to evaluate the STM, it is necessary not only to numerically integrate Equation (4.8), but also the EoMs simultaneously in order to retrieve the required trajectory. The final set of equations comprises 42 ODEs and requires the initial conditions to be solved. The initial conditions for the state vector $\mathbf{X}$ are problem-specific, while the initial condition for the STM is by definition $\mathbf{\Phi}(t_0, t_0) = \mathbf{I}$, with $\mathbf{I}$ equal to the 6×6 identity matrix.

The STM computed after an entire period is also known as *monodromy matrix*, when a periodi solution, i.e. a periodic orbit, is considered. This matrix can be used to assess the stability of the orbit, i.e., whether an initial perturbation increases or decreases over time.

$$\mathbf{M} \equiv \mathbf{\Phi}(T) = \frac{\partial \phi(T; \bar{x}_0)}{\partial x_0} \tag{4.17}$$

The characteristics of such matrix are derived from the results of Floquet theory [34]:

- the monodromy matrix of a periodic orbit in the CR3BP must possess at least two eigenvalues equal to one;
- the monodromy matrix of an autonomous Hamiltonian system is symplectic;
- if λ is an eigenvalue of the matrix $\mathbf{M}$, then λ^{-1}, $\bar{\lambda}$ (complex conjugate) e $\bar{\lambda}^{-1}$ are eigenvalues of the monodromy matrix and have the same multiplicity.

After analyzing the mathematical derivation of STM and listing its properties, the next step is to discuss the differential correction algorithms, which are based precisely on this matrix.

4.2 Differential Correction

As mentioned in the previous section, there are several families of periodic orbits in the context of the CR3BP. Each family is a periodic solution of the EoMs and exhibits distinctive characteristics. One of the most interesting properties of these orbits is the periodicity. Given the initial conditions of an orbit, expressed in terms of the state vector $\mathbf{X}$, and the period T, the orbit is defined periodic if the following condition applies

$$\mathbf{X}_f = \mathbf{X}(T) = \mathbf{X}_0 \tag{4.18}$$

where the subscript *"f"* indicates the state vector after a period, while the subscript *"0"* denotes the state vector at the initial time. According to Equation (4.18), the state vector evaluated after a period must be equal to the initial state vector for an orbit to be considered periodic. However, it is usually difficult to identify initial conditions that lead to a periodic CR3BP orbit, unless the correct numerical methodology is employed. The following sections will discuss two algorithms belonging to the *differential correction* family. The principle at the base of these algorithms can be described as a *targeting* process. The idea for both algorithms is to iteratively apply small variations to an end of the trajectory in order to target a desired point at the other end.

The main difference between the two algorithms is the number of sections used to split the trajectory, hence the number of points distributed along the path. The first version of the algorithm implemented is the simplest, known as *single shooting*; while the evolution of this method is called *multiple shooting*.

The conclusive section of this chapter will show that both algorithms converge to the desired periodic solution if the estimate of the initial conditions is sufficiently accurate. Despite that, due to their characteristics, their convergence speeds and fields of application are quite different, even though they belong to the same family of algorithms.

4.2.1 Single shooting method

The *single shooting* algorithm for differential correction was developed in the 1960s. The algorithm was created in response to the necessity for numerical correction of sets of initial conditions to obtain periodic orbits in the CR3BP.

The first documented work was carried out by Broucke [35], who produced a database of periodic orbits for the Earth-Moon system. The idea at the base of this work was the future foundation of the method: iteratively correcting the initial conditions using the STM. In 1973 Hénon [36] introduced the algorithm into the context of DST, describing it as a standard method for identifying periodic orbits correcting errors in Poincaré sections. Subsequently, Farquhar [37] used the algorithm during his study on three-dimensional orbits, which are now knows as halo orbits. Based on Farquhar's work, Howell [38] investigated halo orbits and proposed a formulation of the

single shooting differential correction, which is still widely adopted in literature today. Howell exploited the symmetry characteristics of this orbit family to reduce the number of variables to be corrected at each iteration.

Moving on to the mathematical formulation of the method, the starting point is the problem setting as a *Two-Point Boundary Value Problem* (TPBVP). The goal is to correct the initial state vector $\mathbf{X}_0$, so that when integrated for a specified time, it yields to a final state $\mathbf{X}_f$ that satisfies Equation (4.18). The fundamental equation of the *single shooting* method derives from a Taylor series expansion arrested to the first order of the EoMs around ($\mathbf{X}_0$,T)

$$\delta\mathbf{X}_f = \mathbf{\Phi}(T)\delta\mathbf{X}_0 + \frac{\partial\mathbf{X}_f}{\partial t}\delta T \tag{4.19}$$

where:

- $\mathbf{\Phi}$(T) is the STM evaluated after a period;
- $\delta\mathbf{X}_0$ is the vector containing the corrections to be computed and applied to the initial state vector to obtain periodic initial conditions;
- $\delta\mathbf{X}_f$ represents the variation of the final state due to the application of $\delta\mathbf{X}_0$ to the initial state;
- δT is the period change that has to be computed at each iteration.

As mentioned in Section 4.1 the STM must be derived integrating a system of 42 ODEs, which includes the EoMs. However, the period of a periodic orbit is generally unknown a priori. Therefore, the number of unknowns which have to be computed each time increases to 43. The high computational cost can be reduced using the symmetry properties of certain orbits families. Doing so, the initial conditions vector simplifies and the number of unknowns to be calculated decreases.

To demostrate this, the formulation proposed by Howell for halo orbits is recalled [38]. As discussed in Section periodic orbits, such orbits are perpendicular to the **x-z** plane and are then denoted by an initial condition vector in the form of

$$\mathbf{X}_0 = \begin{bmatrix} x_0 & 0 & z_0 & 0 & \dot{y}_0 & 0 \end{bmatrix} \tag{4.20}$$

It is not only the initial state vector that can be simplified, but even the state vector at the end of the integration process. Indeed, to respect the symmetry of halo orbits with respect to the *x-z* plane, the following conditions must hold

$$y\left(t_f\right) = 0, \quad \dot{x}\left(t_f\right) = 0, \quad \dot{z}\left(t_f\right) = 0 \tag{4.21}$$

where $t_f = T/2$, because the **x-z** plane crossing takes place after half period. Since the period is an unknown, it needs to be derived explicitly. In this case, it is possible

to use the constraint $y(t_f) = \delta y = 0$ to express δT as funcion of the initial state variables. Extracting the second row from the system expressed by Equation (4.19), which corresponds to the constraint

$$\delta y = 0 = \mathbf{\Phi}_{21}\delta x_0 + \mathbf{\Phi}_{23}\delta z_0 + \mathbf{\Phi}_{25}\delta \dot{y}_0 + \dot{y}_f \delta T \tag{4.22}$$

which solved for δT yields

$$\delta T = -\frac{1}{\dot{y}_f}\left(\mathbf{\Phi}_{21}\delta x_0 + \mathbf{\Phi}_{23}\delta z_0 + \mathbf{\Phi}_{25}\delta \dot{y}_0\right) \tag{4.23}$$

Based on these considerations, the remaining constraints are $\dot{x}_f = 0$ and $\dot{z}_f = 0$. The initial conditions that can be corrected are the three non-zero components of the state vector. Therefore, Equation 4.19 simplifies as follows

$$\begin{Bmatrix} \cancel{\delta x_f} \\ \cancel{\delta y_f} \\ \cancel{\delta z_f} \\ \delta \dot{x}_f \\ \cancel{\delta \dot{y}_f} \\ \delta \dot{z}_f \end{Bmatrix} = \begin{bmatrix} \cancel{\Phi_{11}} & \cancel{\Phi_{12}} & \Phi_{13} & \cancel{\Phi_{14}} & \Phi_{15} & \cancel{\Phi_{16}} \\ \cancel{\Phi_{21}} & \cancel{\Phi_{22}} & \Phi_{23} & \cancel{\Phi_{24}} & \Phi_{25} & \cancel{\Phi_{26}} \\ \cancel{\Phi_{31}} & \cancel{\Phi_{32}} & \Phi_{33} & \cancel{\Phi_{34}} & \Phi_{35} & \cancel{\Phi_{36}} \\ \Phi_{41} & \Phi_{42} & \Phi_{43} & \Phi_{44} & \Phi_{45} & \Phi_{46} \\ \cancel{\Phi_{51}} & \cancel{\Phi_{52}} & \Phi_{53} & \cancel{\Phi_{54}} & \Phi_{55} & \cancel{\Phi_{56}} \\ \Phi_{61} & \Phi_{62} & \Phi_{63} & \Phi_{64} & \Phi_{65} & \Phi_{66} \end{bmatrix} \begin{Bmatrix} \delta x_0 \\ \cancel{\delta y_0} \\ \delta z_0 \\ \cancel{\delta \dot{x}_0} \\ \delta \dot{y}_0 \\ \cancel{\delta \dot{z}_0} \end{Bmatrix} + \begin{Bmatrix} \cancel{\delta \ddot{x}_f} \\ \cancel{\delta \ddot{y}_f} \\ \cancel{\delta \ddot{z}_f} \\ \delta \ddot{x}_f \\ \cancel{\delta \ddot{y}_f} \\ \delta \ddot{z}_f \end{Bmatrix} \delta T \tag{4.24}$$

Clearly, this form of the system is undetermined, due to presence of just two equations from which to derive three unknowns. Consequently, one variable must be fixed to solve the system and compute the two free variables. Usually, z_0 is chosen as the fixed variable because halo orbits can be classified with the maximum value along the z axis, known as *z-amplitude*, A_z. Accordingly, the variation that apply to x_0 and $\dot{y}_0$ are computed as

$$\begin{Bmatrix} \delta x_0 \\ \delta \dot{y}_0 \end{Bmatrix} = \left[\begin{bmatrix} \Phi_{41} & \Phi_{45} \\ \Phi_{61} & \Phi_{65} \end{bmatrix} - \frac{1}{\ddot{y}_f} \begin{Bmatrix} \ddot{x}_f \\ \ddot{z}_f \end{Bmatrix} \begin{bmatrix} \Phi_{21} & \Phi_{25} \end{bmatrix} \right]^{-1} \begin{Bmatrix} \delta \dot{x}_f \\ \delta \dot{z}_f \end{Bmatrix} \tag{4.25}$$

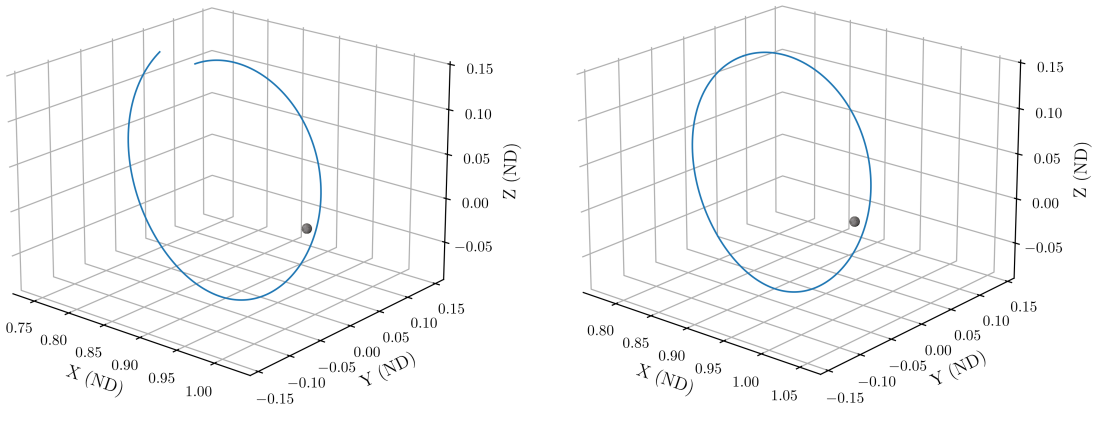
Alternatively, one the three variables can be arbitrarily fixed and the other two can be corrected. This correction process continues until the variations in the final state exceed the specified tolerance threshold, e.g. 10^{-10}. Algorithm 1 outlines the key steps of the single shooting method.

It is useful to observe that the period is not calculated using Equation (4.23), but it is obtained by interrupting the time integration of the initial conditions when the x-z plane is crossed.

Algorithm 1 Single shooting differential correction for Halo orbits

Define initial conditions: $\mathbf{X}_0 = [\, x_0,\, 0,\, z_0,\, 0,\, \dot{y}_0,\, 0\,]^\top$
Set the tolerance: tol $\leftarrow 10^{-14}$ (for example)
Set the maximum number of iterations: $k_{\max} \in \mathbb{N}^+$
while $(|\dot{x}_f| > \text{tol}$ **or** $|\dot{z}_f| > \text{tol})$ **and** $k < k_{\max}$ **do**
    Integrate EoMs and STM using Eqs. (4.8)
    Extract from the augmented state: $\mathbf{X}_f = \mathbf{X}(t_f)$ and $\Phi_f = \Phi(t_f)$
    Differentiate the final state: $\dfrac{\partial \mathbf{X}_f}{\partial t} = [\, \dot{x}_f,\, \dot{y}_f,\, \dot{z}_f,\, \ddot{x}_f,\, \ddot{y}_f,\, \ddot{z}_f\,]^\top$
    Set the final variations: $\delta\dot{x}_f = -\dot{x}_f, \quad \delta\dot{z}_f = -\dot{z}_f$
    if $|\dot{y}_f| \leq \text{tol}$ **then**
        Periodic orbit found
    else
        Fix one initial component (x_0 *or* z_0 *or* $\dot{y}_0$) and free the other two.
        Compute $\delta\mathbf{X}_0$ using Eq. (4.24).
        Update the free initial components: $\mathbf{X}_0 \leftarrow \mathbf{X}_0 + \delta\mathbf{X}_0$.
    $k \leftarrow k + 1$

An example of the Algorithm 1 is shown hereafter, where the initial conditions of an existing Northern Halo orbit built around L_1 in the Earth-Moon system have been selected. The state vector provided as an initial guess is: $\mathbf{X}_0$=[0.836, 0, 0.1478446561518, 0, 0.256, 0] In this case, the exact value of the z component has been given, with the other two being approximated. Figures 4.1a and 4.1b show a comparison between the integrated initial state vector, Figure 4.1a, and the integrated initial conditions resulting from the *single shooting* algorithm, Figure 4.1b. Since the desired orbit is built around L_1, both figures show a sphere corresponding to the Moon for the sake of completeness.



(a) **(b)**

Figure 4.1: Comparison between a set of non-periodic initial conditions and a set of periodic initial conditions in the Earth-Moon system

Figure 4.1a allows to observe the non-periodicity of the initial conditions provided, while Figure 4.1b highlights the effectiveness of the algorithm. Indeed,the algorithm produces initial conditions that lead to a periodic orbit.

Figure 4.2 shows instead the evolution of the initial conditions with the number of iterations needed to reach convergence. Firstly, it can be affirmed that the initial guess is sufficiently accurate, because of the small number of iterations required to satisfy the convergence criterion. Additionally, the corrections made have a magnitude less than 10^{-3}, which indicates that the original intial conditions are close to those of a periodic orbit.

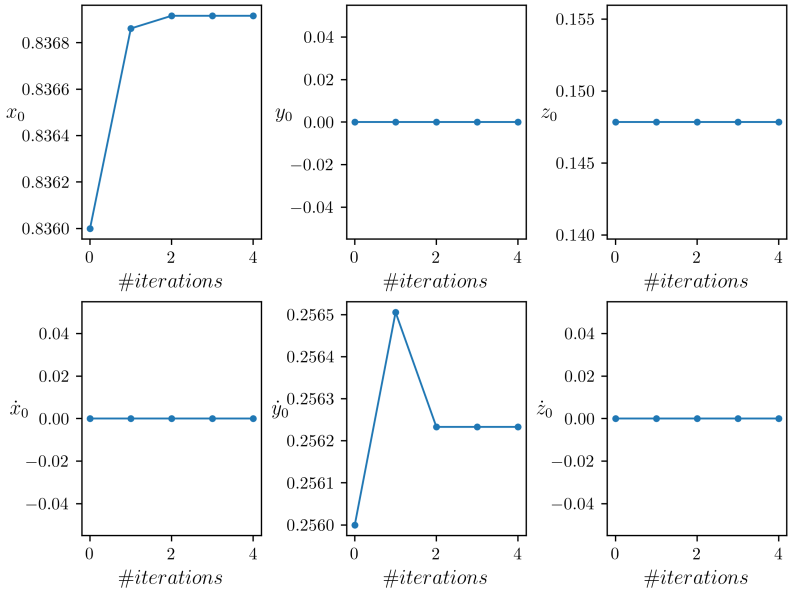


Figure 4.2: Initial conditions evolution during the single shooting differential correction

Finally, Figure 4.3 reports the values of $|\dot{x}_f|$ and $|\dot{z}_f|$ for each iteration. These values are used to verify the periodicity of the orbit. This final figure confirms the computational speed and the effectiveness of the algorithm when applied to this family of orbits. Indeed, after a few iterations the values of velocities along x- and z-axes at the final time can be considered null, because they drop below the set tolerance. Consequently, the resultant orbit crosses the x-z plane perpendicularly and can be considered a periodic halo orbit.

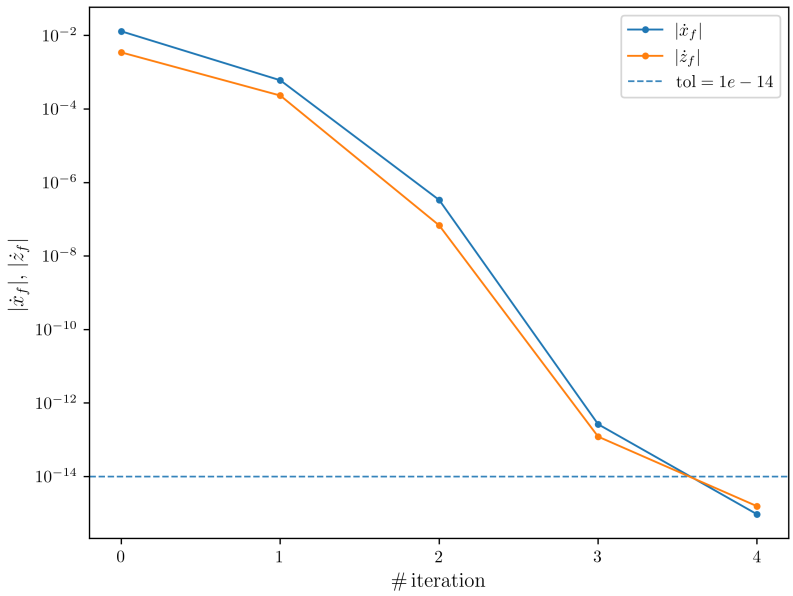


Figure 4.3: Final state vector variations at each iteration

4.2.2 Multiple shooting method

The *multiple shooting* method was developed as a numerical analysis technique that can overcome the limitations of the *single shooting* variant, such as convergence and conditioning problems.

Osborne [39] first proposed an approach based on the trajectory's fragmentation into multiple segments. According to this approach, the dynamical system is integrated on multiple arcs, which are connected together thanks to continuity conditions. Over the following years Bulircsh [40] highlighted the effectiveness of the method when applied to non-linear systems and to systems that are strongly sensitive to initial conditions. Deuflhard [41] demostrated that the *multiple shooting* algorithm can be interpreted as an extension of the Newton's method for considerably bad-conditioned systems. Between 1980s and 1990s, periodic orbits and interplanetary trajectories were formed by the union of multiple dynamical arcs, so the *multiple shooting* was widely employed due to its potential. The algorithm established itself as a robust method that can effectively handle long temporal arcs and an high numerical sensitivity.

The following mathematical description recalls the formulation proposed by Van Anderlecht [42]. The base principle of the method is the same as that of the *single shooting* algorithm: the initial conditions are updated until the convergence criterion is met. Similarly to the *single shooting*, the targeting problem is based on an n-free variables vector $\mathbf{X}$, subject to m constraint equations grouped in the vector $\mathbf{F}(\mathbf{X})$. The objective of the targeting problem is to identify a vector $\mathbf{X}^*$ that satisfies the constraints, i.e. ensures $\mathbf{F}(\mathbf{X}^i) = \mathbf{0}$. Given an estimate of the initial conditions $\mathbf{X}^i$, the equation for the initial conditions of the next iteration can be obtained by expanding

the constraints vector in a Taylor series.

$$\mathbf{F}(\mathbf{X}^i) + \mathbf{A}(\mathbf{X}^i) \cdot (\mathbf{X}^{i+1} - \mathbf{X}^i) = \mathbf{0} \tag{4.26}$$

where $\mathbf{A}(\mathbf{X}^i)$ is the $m \times n$ Jacobian matrix

$$\mathbf{A}(\mathbf{X}^i) = \begin{bmatrix} \frac{\partial F_1}{\partial X_1} & \frac{\partial F_1}{\partial X_2} & \cdots & \frac{\partial F_1}{\partial X_n} \\ \frac{\partial F_2}{\partial X_1} & \frac{\partial F_2}{\partial X_2} & \cdots & \frac{\partial F_2}{\partial X_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_m}{\partial X_1} & \frac{\partial F_m}{\partial X_2} & \cdots & \frac{\partial F_m}{\partial X_n} \end{bmatrix} \tag{4.27}$$

A solution can only be computed if the number of free variables is equal to or greater than the number of constraint equations. If this condition is met, two different scenarios can be observed

- when the number of the free variables is equal to the number of constraints ($n = m$), the only solution to Equation (4.26) can be calculated using Newton's method
$$\mathbf{X}^{i+1} = \mathbf{X}^i - \mathbf{A}(\mathbf{X}^i)^{-1}\mathbf{F}(\mathbf{X}^i) \tag{4.28}$$

- if the number of free variables is greater than the number of constraints ($n > m$), Equation (4.26) shows infinite solutions. Therefore, it must be selected the solution $\mathbf{X}^{i+1}$ closer to $\mathbf{X}^i$, so the first inherits most of the features of the solution computed at the previous iteration. Such solution is known as *minimun-norm solution* and is calculated as follows
$$\mathbf{X}^{i+1} = \mathbf{X}^i - \mathbf{A}(\mathbf{X}^i)^T \left[\mathbf{A}(\mathbf{X}^i)\mathbf{A}(\mathbf{X}^i)^T\right]^{-1} \mathbf{F}(\mathbf{X}^i) \tag{4.29}$$

In both cases, the initial conditions are updated until the order of magnitude of the norm of the constraints vector is greater than the specified tolerance, e.g. 10^{-10}. As mentioned, the *multiple shooting* is applied to a fragmented trajectory. Hence, $\mathbf{X}$ is a vector comprising n vectors that represent the initial conditions of each point fixed along the trajectory

$$\mathbf{X} = [\bar{x}_1, \bar{x}_2, \cdots, \bar{x}_{n-1}, \bar{x}_n]^T \tag{4.30}$$

where the notation $\bar{x}_i$ represents the vector of initial conditions of the *i-th* point. In the general case when the total integration period, that is the period T of the desired orbit, is not known a priori, it is added to the vector of free variables. Consequently, the number of unknowns is equal to $6n+1$. Providing an initial guess of the orbital period, the state of each internal point $\bar{x}_i$ can be integrated for a time $t_i = \frac{T}{n-1}$, reaching the point denoted $\bar{x}^t_{i+1}$. This process does not apply to the last point $\bar{x}_n$. Figure 4.4 illustrates how the *multiple shooting* operates. After that, the vector $\mathbf{F}(\mathbf{X})$, that contains all the constraint equations that ensure the final trajectory to be closed, is analyzed.

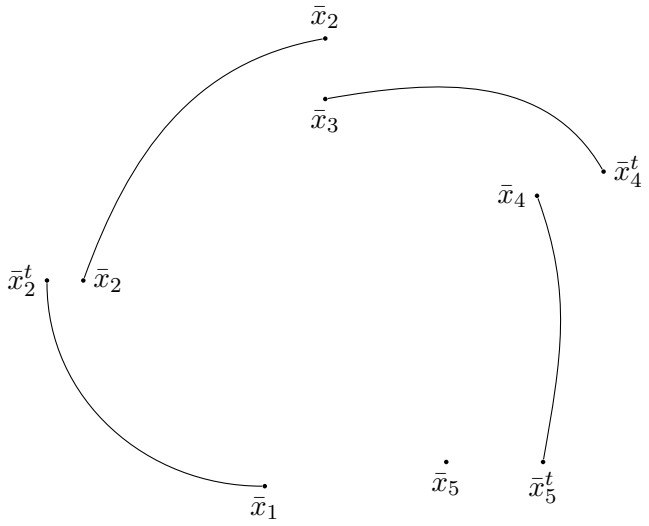


Figure 4.4: Multiple Shooting Targeting Scheme

Inside the vector, three subvectors can be identified, each of them corresponding to a specified constraint:

- the first is $\mathbf{F}_C$, a $6(n-1)$ variables vector that comprises the continuity conditions between the segments. This ensures that the condition $\bar{x}_{i+1}^t = \bar{x}_{i+1}$ is satisfied. The structure of the vector is presented below

$$\mathbf{F}_C = \begin{bmatrix} \bar{x}_2^t - \bar{x}_2 \\ \bar{x}_3^t - \bar{x}_3 \\ \vdots \\ \bar{x}_n^t - \bar{x}_n \end{bmatrix} \tag{4.31}$$

- $\mathbf{F}_P$ is a five elements vector that imposes the periodicity condition, so it is responsible for the finale state to match the initial state

$$\mathbf{F}_P = \begin{bmatrix} x_n^t - x_1 \\ y_n^t - y_1 \\ z_n^t - z_1 \\ \dot{x}_n^t - \dot{x}_1 \\ \dot{z}_n^t - \dot{z}_1 \end{bmatrix} \tag{4.32}$$

- $\mathbf{F}_L$ is known as normalization or system closure constraint and ensures that the motion happens in the proximity of the selected Lagrangian point

$$\mathbf{F}_L = y_1 - y_{L_i} \tag{4.33}$$

Hence, the residuals vector is presented as a vector containing $6n$ elements, assembled as follows

$$\mathbf{F}(\mathbf{X}) = \begin{bmatrix} \mathbf{F}_C \\ \mathbf{F}_P \\ \mathbf{F}_L \end{bmatrix} \tag{4.34}$$

Finally, the Jacobian matrix $A(X)$, which has dimensions $6n \times (6n+1)$, is shown

$$\mathbf{A}(\mathbf{X}) = \begin{bmatrix} \mathbf{\Phi}_{1(6\times6)} & -\mathbf{I}_{6\times6} & \mathbf{0}_{6\times6} & \cdots & \cdots & \mathbf{0}_{6\times6} & \frac{1}{n-1}\dot{\bar{x}}_2^t \\ \mathbf{0}_{6\times6} & \ddots & \ddots & \ddots & & \vdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \mathbf{0}_{6\times6} & \vdots \\ \mathbf{0}_{6\times6} & \cdots & \cdots & \mathbf{0}_{6\times6} & \mathbf{\Phi}_{n-1(6\times6)} & -\mathbf{I}_{6\times6} & \frac{1}{n-1}\dot{\bar{x}}_n^t \\ \frac{\partial F_P}{\partial \bar{x}_1} & \mathbf{0}_{5\times6} & \cdots & \mathbf{0}_{5\times6} & \frac{\partial F_P}{\partial \bar{x}_{n-1}} & \mathbf{0}_{5\times6} & \frac{\partial F_P}{\partial T} \\ \frac{\partial F_L}{\partial \bar{x}_1} & \mathbf{0}_{1\times6} & \cdots & \cdots & \cdots & \mathbf{0}_{1\times6} & 0 \end{bmatrix} \tag{4.35}$$

The Jacobian matrix employed for the *multiple shooting* is made up of submatrices of different nature. The dimensions of each submatrix are specified by the subscript reported, while they are explained hereafter:

- $\mathbf{\Phi}_{i(6\times6)}$ represents the STM relative to each segment of the trajectory;
- $-\mathbf{I}_{6\times6}$ is the identity matrix with the sign changed;
- $\mathbf{0}_{6\times6}$ indicates a zero square matrix, however within the Jacobian matrix the zero rectangular matrix $\mathbf{0}_{5\times6}$ and the zero row vector $\mathbf{0}_{1\times6}$ can be observed as well;
- $\frac{\partial F_P}{\partial \bar{x}_1}$ is a 5×6 matrix and, due to the structure of $\mathbf{F}_P$, corresponds to a negative rectangular identity matrix

$$\frac{\partial F_P}{\partial \bar{x}_1} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \tag{4.36}$$

- $\frac{\partial F_P}{\partial \bar{x}_{n-1}}$ is a 5×6 matrix too and contains the elements of STM that relates the integrated state $\bar{x}_n^t$ to the state $\bar{x}_{n-1}$

$$\frac{\partial F_P}{\partial \bar{x}_{n-1}} = \begin{bmatrix} \frac{\partial x_n^t}{\partial x_{n-1}} & \frac{\partial x_n^t}{\partial y_{n-1}} & \frac{\partial x_n^t}{\partial z_{n-1}} & \frac{\partial x_n^t}{\partial \dot{x}_{n-1}} & \frac{\partial x_n^t}{\partial \dot{y}_{n-1}} & \frac{\partial x_n^t}{\partial \dot{z}_{n-1}} \\ \frac{\partial y_n^t}{\partial x_{n-1}} & \frac{\partial y_n^t}{\partial y_{n-1}} & \frac{\partial y_n^t}{\partial z_{n-1}} & \frac{\partial y_n^t}{\partial \dot{x}_{n-1}} & \frac{\partial y_n^t}{\partial \dot{y}_{n-1}} & \frac{\partial y_n^t}{\partial \dot{z}_{n-1}} \\ \frac{\partial z_n^t}{\partial x_{n-1}} & \frac{\partial z_n^t}{\partial y_{n-1}} & \frac{\partial z_n^t}{\partial z_{n-1}} & \frac{\partial z_n^t}{\partial \dot{x}_{n-1}} & \frac{\partial z_n^t}{\partial \dot{y}_{n-1}} & \frac{\partial z_n^t}{\partial \dot{z}_{n-1}} \\ \frac{\partial \dot{x}_n^t}{\partial x_{n-1}} & \frac{\partial \dot{x}_n^t}{\partial y_{n-1}} & \frac{\partial \dot{x}_n^t}{\partial z_{n-1}} & \frac{\partial \dot{x}_n^t}{\partial \dot{x}_{n-1}} & \frac{\partial \dot{x}_n^t}{\partial \dot{y}_{n-1}} & \frac{\partial \dot{x}_n^t}{\partial \dot{z}_{n-1}} \\ \frac{\partial \dot{z}_n^t}{\partial x_{n-1}} & \frac{\partial \dot{z}_n^t}{\partial y_{n-1}} & \frac{\partial \dot{z}_n^t}{\partial z_{n-1}} & \frac{\partial \dot{z}_n^t}{\partial \dot{x}_{n-1}} & \frac{\partial \dot{z}_n^t}{\partial \dot{y}_{n-1}} & \frac{\partial \dot{z}_n^t}{\partial \dot{z}_{n-1}} \end{bmatrix} \tag{4.37}$$

- $\frac{\partial F_P}{\partial T}$ is a column vector made up of five elements corresponding to the derivatives of the $\mathbf{F}_P$ elements with respect to the period

$$\frac{\partial F_P}{\partial T} = \frac{1}{n-1} \begin{bmatrix} \dot{x}_n^t \\ \dot{y}_n^t \\ \dot{z}_n^t \\ \ddot{x}_n^t \\ \ddot{z}_n^t \end{bmatrix} \tag{4.38}$$

- $\frac{\partial F_P}{\partial \bar{x}_1}$ is a row vector of dimension 6 formed by the derivatives of $\mathbf{F}_P$ with respect to the initial state

$$\frac{\partial F_P}{\partial \bar{x}_1} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.39}$$

The need to compute the state of all points distributed along the trajectory at each iteration makes the *multiple shooting* method more computationally expensive than the *single shooting* method. Similar to the latter, depending on the family of orbits considered, the problem can be simplified by exploiting its symmetry characteristics. Furthermore, the problem can be also simplified by fixing the period of the orbit. In fact, this alone causes the Jacobian to "lose" a column and have a dimension of $6n \times 6n$. As in the case of *single shooting*, the Algorithm 2 below shows the basic steps of the *multiple shooting* method.

Algorithm 2 Multiple shooting differential correction for periodic orbits

1: Define initial conditions: $\bar{x}_0$
2: Set the number of sections: N and define $n \leftarrow N + 1$
3: Set the initial period guess: T and the step $h \leftarrow T/N$
4: Set the tolerance: tol $\leftarrow 10^{-12}$ (for example)
5: Set the maximum number of iterations: $k_{\max} \in \mathbb{N}^+$
6: Initialize the nodes by propagation:
7: **for** $i = 0$ **to** $N - 1$ **do** ▷ Derive n initial points
8: $\quad \bar{x}_{i+1} \leftarrow$ Propagate $\bar{x}_i$ via EoMs for time h
9: Define the unknowns vector: $\mathbf{X} \leftarrow \left[\text{vec}(\bar{x}_0, \ldots, \bar{x}_N),\, T \right]^\top$
10: $k \leftarrow 0$
11: **while** $\|\mathbf{F}(\mathbf{X})\| >$ tol **and** $k < k_{\max}$ **do**
12: $\quad$ Compute the constraints vector $\mathbf{F}(\mathbf{X})$ using Eq. (4.34)
13: $\quad$ Compute the Jacobian $\mathbf{A}(\mathbf{X})$ using Eq. (4.35)
14: $\quad$ Compute the correction $\delta\mathbf{X}$ using the minimum-norm solution, Eq. (4.29)
15: $\quad$ Update the unknowns: $\mathbf{X} \leftarrow \mathbf{X} + \delta\mathbf{X}$
16: $\quad k \leftarrow k + 1$

In order to reduce the computational cost of the algorithm, corrections are calculated and then applied only to the components that are indicated based on the desired orbit.

In the case of planar orbits, such as lyapunov orbits, the state of the first point is

$$\bar{x}_1 = [x, 0, 0, 0, \dot{y}, 0] \tag{4.40}$$

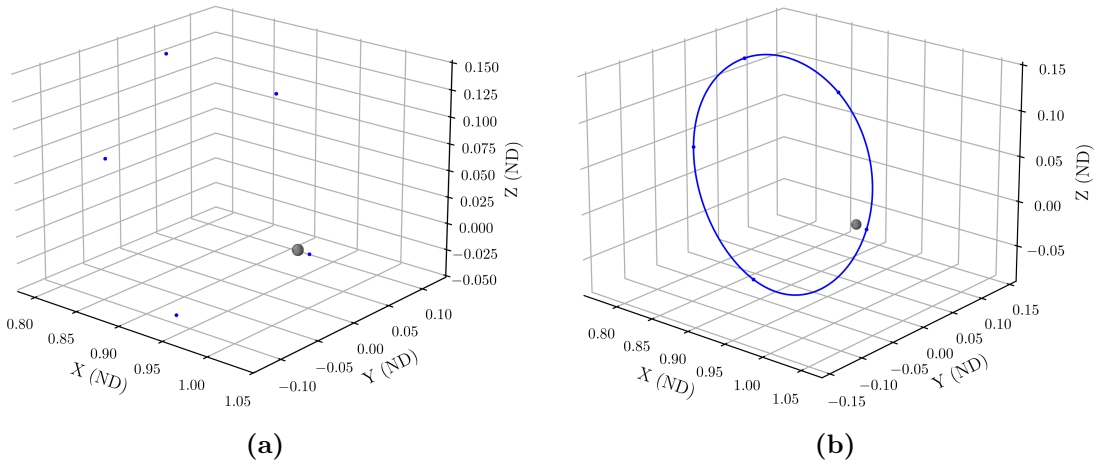
so the corrections only affect the two non-zero components. To comply with the orbital planarity constraint, the corrections applied to the other points will affect the components $x, y, \dot{x}$ and $\dot{y}$. In the case of three-dimensional orbits, such as halo orbits, the initial state is corrected on the basis of the simplifications that can be made. The other points, on the other hand, will have corrections applied to all six components.

As in the previous case, it is here reported the application of the described algorithm to a Northern Halo orbit built around L_1 in the Earth-Moon system. The state vector provided as initial guess is: $\mathbf{X}_0 = [8.36E-1, 0, 1.47E-1, 0, 2.56E-1, 0]$. Unlike in the single shooting case, all the three non-zero components of the state vector are corrected simultaneously. This is possibile because the exact period of the orbit is known and fixed at $T = 2.7450787982481035$. According to the outlined procedure, the first step is to generate a succession of points along the trajectory on which to apply the algorithm. Table 4.1 reports the state vector of each point when the trajectory is broken down into N=5 segments.

Table 4.1: State vectors of shooting nodes for the multiple-shooting differential correction algorithm

# Point	1	2	3	4	5	6
State Vector	$\begin{bmatrix} 0.836 \\ 0 \\ 0.147 \\ 0 \\ 0.256 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0.8675271216 \\ 0.1157426143 \\ 0.0877227778 \\ 0.1034881572 \\ 0.1231293310 \\ 0.2034091588 \end{bmatrix}$	$\begin{bmatrix} -0.2335785765 \\ -0.2303298607 \\ 0.0740720024 \\ -0.0471620415 \\ 0.0949455145 \\ 0.9249685338 \end{bmatrix}$	$\begin{bmatrix} 0.9224273300 \\ -0.0858432924 \\ -0.0564606959 \\ -0.0851718919 \\ -0.2526981507 \\ 0.2107888683 \end{bmatrix}$	$\begin{bmatrix} 0.8533441582 \\ -0.1160295676 \\ 0.0756763728 \\ -0.1379383445 \\ 0.1179140501 \\ 0.2123482462 \end{bmatrix}$	$\begin{bmatrix} 0.7915650923 \\ 0.0099900162 \\ 0.1455954923 \\ -0.0813554382 \\ 0.3060879524 \\ 0.0288544313 \end{bmatrix}$

Figure 4.5a shows how the shooting points are distributed at the initial time around L_1 and the Moon, that is the celestial body displayed. Applying the *multiple shooting* algorithm to this initial scheme and by iteratively fixing the state vector of each internal point, results the trajectory reported in Figure 4.5b. Thanks to the corrections applied, integrating the state of a point for a fraction of the period leads directly to the following point, ensuring continuity between points and therefore the periodicity of the orbit.



(a) **(b)**

Figure 4.5: Application of the multiple shooting differential correction algorithm to a set of initial nodes leading to a periodic connected orbit

Figure 4.6 illustrates the evolution of the state vector of the first point across the iterative procedure. At first, it can be noted that all the non-zero components vary, in line with the hypothesis made. Secondly, it is evident that the number of iterations required to meet the convergence criterion is in the hundreds. Such increase in the number of iterations is related to the convergence criterion used. In fact, it is not sufficient that the corrections applied to the state vector of the first point fall below the fixed tolerance. This condition must also be met by the residuals vector norm, so the continuity between the points is ensured. Clearly, if the number of the shooting points increases, the number of conditions to be satisfied increases as well and so does the number of iterations needed, hence the convergence speed decreases.

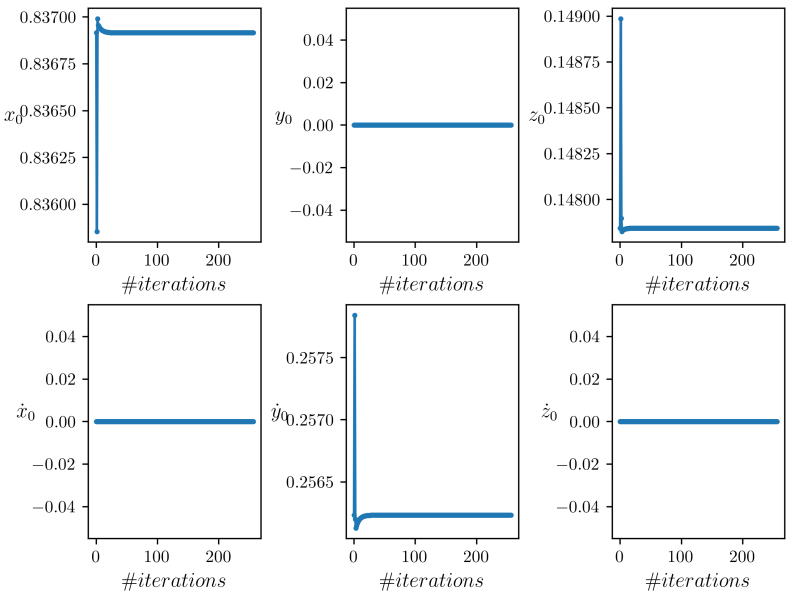
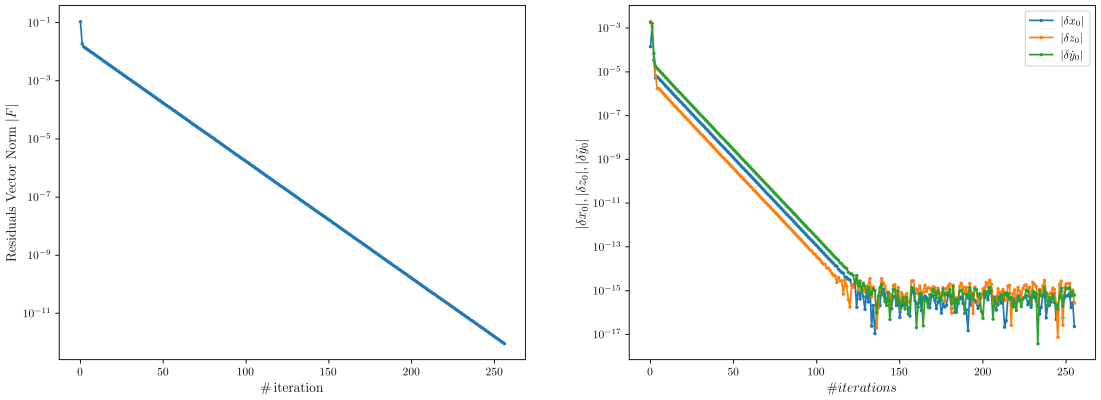


Figure 4.6: Initial conditions evolution during the multiple shooting differential correction

Figure 4.7a shows how the residuals vector evolves with each iteration, while Figure 4.7b reports the absolute values of the corrections applied to the components of the initial guess state vector. As can be seen from the plots, the norm of the individual corrections decreases faster than the residuals vector. In conclusion, even if the initial point is corrected to lead to periodic initial conditions, the *multiple shooting* algorithm stops only when all points are fixed in order to ensure the continuity between the segments that comprise the final trajectory.



(a) **(b)**

Figure 4.7: Residuals vector norm and absolute value of corrections applied to the initial state vector

4.3 Perfomance analysis and numerical comparison

This section will analyze the performance of the two algorithms when applied to different orbit families in the Earth-Moon system. Both algorithms were tested with a maximum number of iterations `max_iter` $= 1000$ and a relative tolerance `tol` $= 10^{-14}$.

The performance of the *single shooting* algorithm was evaluated according to the following procedure: given the periodic initial conditions for an orbit, the state vector was approximated to determine the number of iterations required to correct such initial conditions. For each orbit, the initial guess was provided with a different number of decimal digits at each attempt to observe the correlation between the accuracy of the initial guess and the resulting number of iterations. The following parameters are employed to evaluate the performance of the algorithm :

- Number of iterations required to meet the convergence criterion;
- Residuals norm, i.e. the norm of the variations of the final state $\delta\mathbf{X}_f$;

- Percent error on the initial conditions, computed as follows:

$$IC_{\%} = \frac{\|\mathbf{IC} - \mathbf{IC}_{\text{periodic}}\|}{\|\mathbf{IC}_{\text{periodic}}\|} \times 100\% \,. \tag{4.41}$$

  where **IC** is the initial conditions vector that is iteratively corrected and $\mathbf{IC}_{\text{periodic}}$ is the vector of periodic initial conditions that is known a priori;

- Percent error on the period, computed similarly to $IC_{\%}$

$$T_{\%} = \frac{|T - T_{real}|}{|T_{real}|} \tag{4.42}$$

  where T is the estimated period of the orbit, obtained by interrupting the integration process of the system of 42 ODEs, and T_{real} is the actual period of the periodic orbit.

Taking into consideration a Lyapunov orbit built around L_1 with periodic initial conditions equal to

$$\mathbf{X}_0 = [0.8026705755589522, 0, 0, 0, 0.338409540598485, 0] \tag{4.43}$$

Table 4.2 shows an example of the tests that were carried out. In accordance with the application principles of the algorithm, in turn one of the components was fixed while the algoritm corrected the free variable. Each test was perfomed approximing the free variable with an increasing number of decimal digits with each attempt to observe the algorithm's response.

Table 4.2: Results of the single shooting algorithm applied to a Lyapunov L_1 orbit in the Earth-Moon system

ICs	Fixed Variable	Convergence	Iterations	$\|\delta\mathbf{X}_f\|$	$IC_{\%}$	$T_{\%}$
$[0.8, 0, 0, 0, 0.338409540598485, 0]$	$\dot{y}$	yes	5	3.88×10^{-14}	3.44×10^{-13}	2.85×10^{-12}
$[0.802, 0, 0, 0, 0.338409540598485, 0]$	$\dot{y}$	yes	4	1.09×10^{-13}	3.57×10^{-13}	6.79×10^{-12}
$[0.8026, 0, 0, 0, 0.338409540598485, 0]$	$\dot{y}$	yes	4	1.56×10^{-14}	3.44×10^{-13}	1.69×10^{-12}
$[0.80267, 0, 0, 0, 0.338409540598485, 0]$	$\dot{y}$	yes	3	1.56×10^{-14}	3.44×10^{-13}	1.69×10^{-12}
$[0.8026705755589522, 0, 0, 0, 0.33, 0]$	x	yes	5	6.23×10^{-16}	2.63×10^{-12}	2.31×10^{-12}
$[0.8026705755589522, 0, 0, 0, 0.338, 0]$	x	yes	4	2.01×10^{-14}	2.71×10^{-12}	3.34×10^{-12}
$[0.8026705755589522, 0, 0, 0, 0.3384, 0]$	x	yes	3	1.16×10^{-14}	2.55×10^{-12}	2.13×10^{-12}

The *multiple shooting* algorithm was used providing always the exact period of a known periodic orbit and approximating all components of the state vector. As with the previous algorithm, the initial guess was provided with an increasing number of decimal digits in each component with each attempt. In addition, each provided state vector was tested with a different number of segments. To evaluate the performance of this algorithm, the same parameters listed for the *single shooting* algorithm were used, with one exception. Since the period of the orbit is always equal to the actual period of the periodic orbit, $T_{\%}$ was not computed. Instead, the number of segments used in each

attempt was tracked. Table 4.3 reports an example of the tests perfomed using the *multiple shooting* algorithm, taking into consideration the orbit whose periodic initial condition are expressed by Equation (4.43).

Table 4.3: Results of the multiple shooting algorithm applied to a Lyapunov L_1 orbit in the Earth-Moon system

ICs	Convergence	Segments	Iterations	$\|\delta \mathbf{X}_f\|$	$IC_{\%}$
$[0.8026, 0, 0, 0, 0.3384, 0]$	yes	3	109	8.77×10^{-13}	1.16×10^{-12}
$[0.8026, 0, 0, 0, 0.3384, 0]$	yes	5	169	9.55×10^{-13}	8.75×10^{-13}
$[0.8026, 0, 0, 0, 0.3384, 0]$	yes	7	261	9.54×10^{-13}	1.03×10^{-12}
$[0.8026, 0, 0, 0, 0.3384, 0]$	yes	9	335	9.49×10^{-13}	1.98×10^{-12}
$[0.8026, 0, 0, 0, 0.3384, 0]$	yes	11	413	9.54×10^{-13}	1.06×10^{-12}
$[0.80267, 0, 0, 0, 0.3384, 0]$	yes	3	94	9.76×10^{-13}	1.06×10^{-12}
$[0.80267, 0, 0, 0, 0.3384, 0]$	yes	5	161	9.62×10^{-13}	6.96×10^{-13}
$[0.80267, 0, 0, 0, 0.3384, 0]$	yes	7	214	9.44×10^{-13}	8.33×10^{-13}
$[0.80267, 0, 0, 0, 0.3384, 0]$	yes	9	262	9.59×10^{-13}	7.38×10^{-13}
$[0.80267, 0, 0, 0, 0.3384, 0]$	yes	11	308	9.77×10^{-13}	1.04×10^{-12}
$[0.80267, 0, 0, 0, 0.338409, 0]$	yes	3	90	8.43×10^{-13}	1.20×10^{-12}
$[0.80267, 0, 0, 0, 0.338409, 0]$	yes	5	153	9.64×10^{-13}	1.20×10^{-12}
$[0.80267, 0, 0, 0, 0.338409, 0]$	yes	7	204	9.29×10^{-13}	1.44×10^{-12}
$[0.80267, 0, 0, 0, 0.338409, 0]$	yes	9	250	9.78×10^{-13}	1.27×10^{-12}
$[0.80267, 0, 0, 0, 0.338409, 0]$	yes	11	296	9.59×10^{-13}	7.38×10^{-13}

Figure 4.8 shows a visual representation of the results relating to the proposed example.

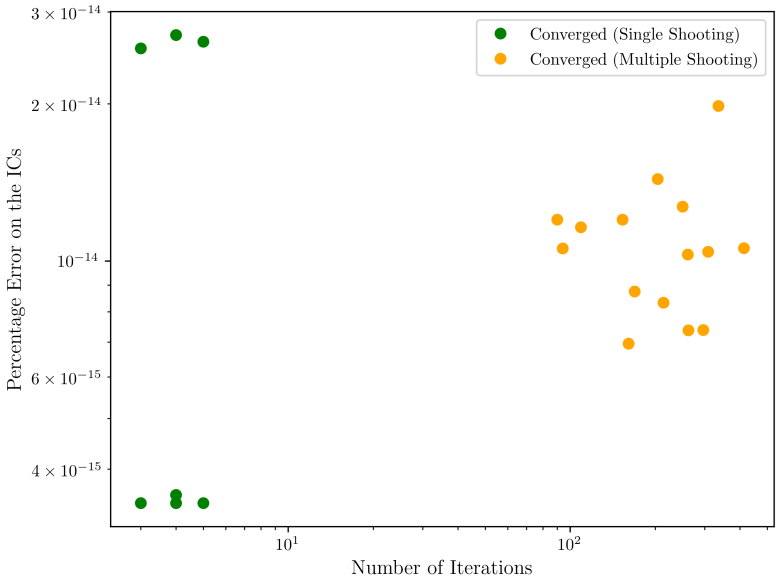


Figure 4.8: Convergence outcome of single shooting and multiple shooting algorithms applied to Lyapunov L_1 orbit in the Earth-Moon system

It is noticible that the *single shooting* algorithm exhibits a greater speed compared to the *multiple shooting* version. The number of iterations required to meet the conver-

gence criterion ranges from a few for the *single shooting* algorithm to several hundred for the *multiple shooting* algorithm. Furthermore, the example shown can be considered well-behaved because each attempt converged, correcting effectively the initial conditions. This occurs because it is easier to correct the initial conditions of a planar orbit than those of a 3D orbit.

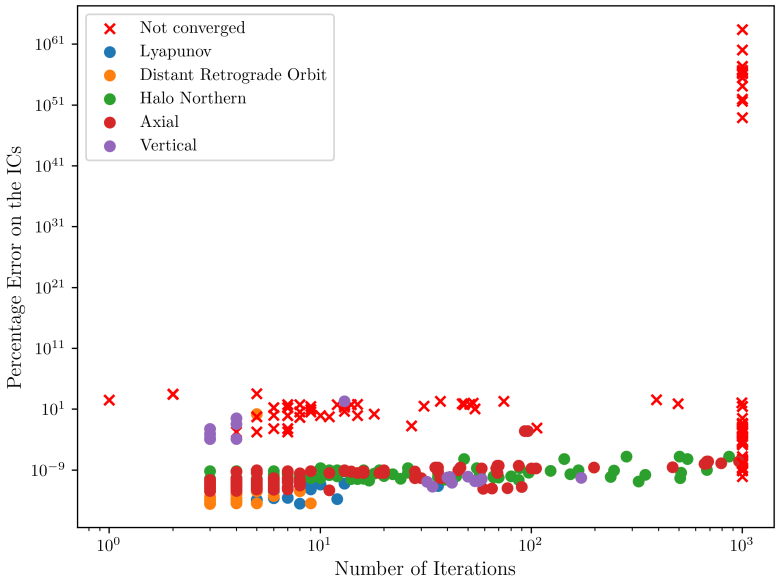
However, when all the tests performed are taken into consideration, the convergence rate of almost all analyzed orbit families is lower than 100%. Table 4.4 compares the percentage of the converged tests for each orbit family. Correcting the initial condition vectors of DROs with the *single shooting* algorithm results in a 100% accuracy mainly because these orbits are two-dimensional and marginally stable, unlike all other orbits which are unstable.

Table 4.4: Convergence rates of single and multiple shooting for different orbit families in the Earth-Moon system

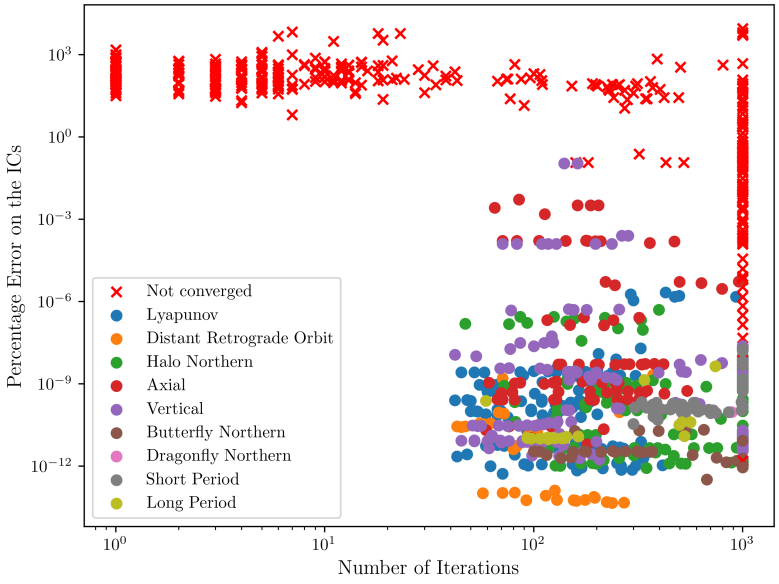
Orbit Family	**Single Shooting**	**Multiple Shooting**
Lyapunov	69.39%	56.94%
DRO	100.00%	53.33%
Halo Northern	90.62%	72.67%
Axial	80.00%	62.22%
Vertical	25.00%	91.85%
Butterfly Northern	-	71.11%
Short Period	-	61.33%

The results reveal a crucial detail: the *single shooting* algorithm outperforms the *multiple shooting* when it is employed to correct the initial conditions of planar or symmetric orbits, with a state vector of less than three non-null components and a relatively short period. These are the ideal operating conditions for the *single shooting*, enabling it to work on well-conditioned systems and guarantee convergence even when the initial guess is significantly less accurate. In these circumstances the use of the *multiple shooting* algorithm is not recommended due to its higher computational cost. However, *multiple shooting* algorithm is useful for correcting an initial conditions vector with almost all non-null components that belongs to a 3D orbit family with no symmetry or difficult to determine. Providing an accurate initial guess, *multiple shooting* is able to correct the initial conditions in most cases and to ensure the periodicity of the orbit. However, alongside the high computational cost, a low convergence speed is generally expected. In fact, it is common for the number of iterations required to be in the hundreds. Figures 4.9a and 4.9b show the outcomes of all the tests performed, identifying the orbit families with different color markers for the *single shooting* and the *multiple shooting* algorithm, respectively. In Figure 4.9b it is evident that the majority of positive tests required more than 100 iterations. Figure 4.9a shows, instead, that the number of iterations needed is relatively low, but increases with the complexity of

the initial conditions vector.



(a)



(b)

Figure 4.9: Convergence results of single shooting (a) and multiple shooting (b)

A second piece of data highlighted by both figures is the number of unsuccessful attempts. The number of such attempts is significantly higher for *multiple shooting* than for *single shooting* for two main reasons:

- Table 4.3 shows that the initial guess was provided with a minimum of three or four decimal digits. Having such a small number of significant figures can lead to a bad-conditioned system, especially if the initial state vector has more non-null components, hence decreasing the probability of convergence;

- the *multiple shooting* method corrects not only the initial conditions vector, but also the state vector of each internal shooting point. Consequently, if the trajectory is broken down into an increasing number of segments, there's a good chance that the method will not converge to a solution within the maximum number of iterations fixed.

In summary, the *single shooting* algorithm is the optimal choice for correcting the initial condition vector of a planar or symmetric orbit with a short orbital period. In most cases, even an approximate initial guess is sufficient to obtain a periodic initial condition vector after a few iterations. The *multiple shooting* algorithm is more appropriate when the initial state vector belongs to orbit families with complex geometry and long orbital periods. Furthermore, the state vector has to be provided with sufficient accuracy in order to converge to a proper solution. Figures 4.10 and 4.11 show a comparison of the performance of the two algorithms used with Vertical orbits built around the collinear points. It should be noted that these tests were conducted only on orbits built around the collinear points because the *single shooting* algorithm does not apply to Vertical orbits built around the equilateral points. Moreover, it is evident that the number of tests carried out with the *multiple shooting* algorithm is higher than that with the *single shooting* algorithm, which was affected by several bad-conditioned problems. Figure 4.10 shows that when the *single shooting* algotithm takes fewer iterations when it is able to converge to a solution. However, Figure 4.11 highlights that the *multiple shooting* algorithm is more reliable with this orbit family due to the higher number of successful attempts.

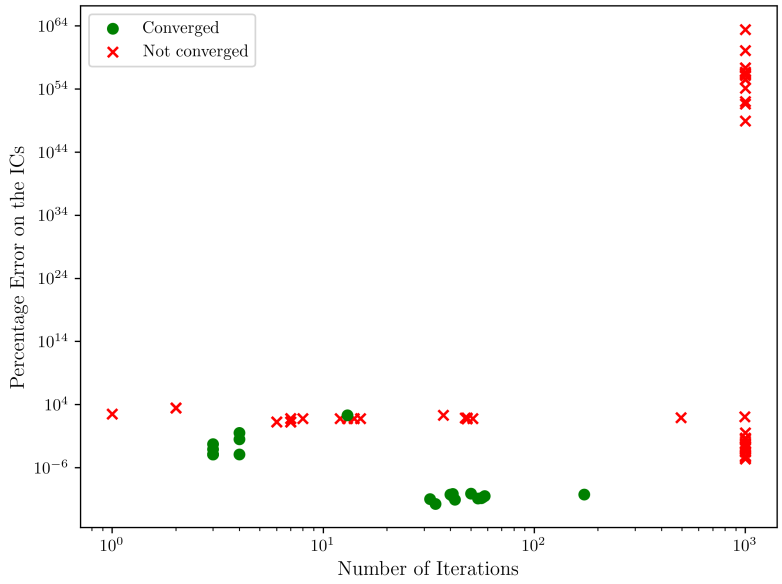


Figure 4.10: Convergence outcome of single shooting algorithm applied to Vertical orbits around collinear Lagrangian points in the Earth-Moon system

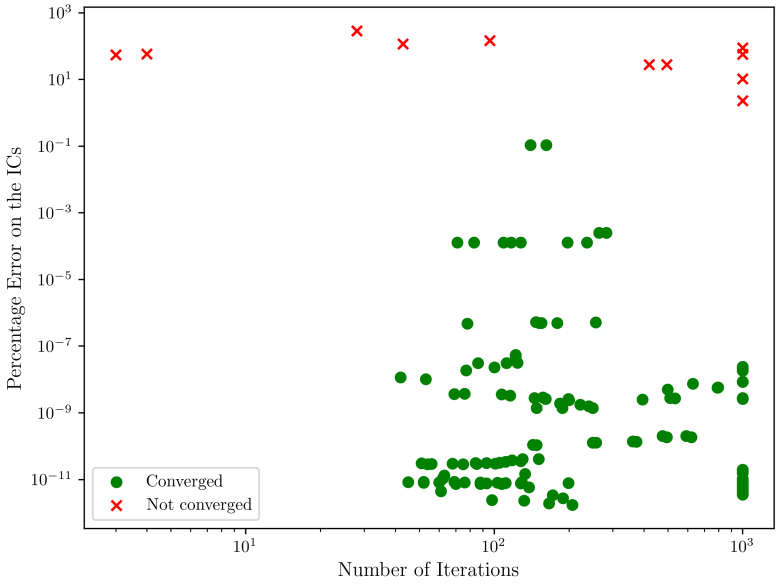


Figure 4.11: Convergence outcome of multiple shooting algorithm applied to Vertical orbits around collinear Lagrangian points in the Earth-Moon system

Initial conditions derivation for any CR3BP system

As discussed in Chapter 4, in most cases the vector of initial conditions must be corrected to ensure the periodicity of the orbit. Furthermore, in order for the differential correction algorithms to perform correctly and obtain a periodic solution, it is essential to provide an accurate estimate of the initial state vector. Thus far, the vector of initial conditions has been taken from the NASA JPL database and approximated to verify the performance of the differential correction algorithms. This chapter aims to develop a method to compute the initial conditions of Halo orbits for systems that are not listed in the NASA JPL database. This CR3BP orbit family has been chosen due to its broad use in numerous real missions, as previously mentioned.

First, a method to analytically approximate the initial conditions of Halo orbits is described. These initial conditions are necessary for the next step, where a genetic algorithm is used to generate new Halo orbits in systems that are not mentioned in the NASA JPL dynamical orbit catalog. This algorithm is the key to derive the initial conditions of the new desired periodic halo orbits. The following section focuses on the results obtained applying the whole procedure to two new systems: Pluto-Charon and α Centauri. The last section provides a detailed description of the algorithm that enables the user of the tool to perform coverage analysis using a derived periodic orbit.

5.1 Approximating initial conditions in the CR3BP

This section describes the method based on the Richardson's work [43] that allows to derive the initial conditions for Halo orbits. More precisely, the method aims to identify third-order analytical solutions for periodic halo-type motions around the collinear points in the CR3BP.

First, the EoMs are the linearized as expressed in Equation (2.26). However, Richardson uses a reference frame centered on one of the collinear points. Consequently, in order for Equation (2.26) to be formally equivalent to the EoMs employed by Richardson, it is necessary to transition from the synodic frame to the frame centered on the Lagrangian point and to non-dimentionalize the variables employed. The

following relations are used to achieve this

$$\begin{aligned} x &= \frac{x_{\text{syn}} - x_{L_i}}{\gamma_L} \\ y &= \frac{y_{L_i}}{\gamma_L} \\ z &= \frac{z_{L_i}}{\gamma_L} \end{aligned} \tag{5.1}$$

where x_{L_i}, y_{L_i}, z_{L_i} are the coordinates of the chosen collinear point, and γ_L is a non-dimensional quantity calculated as follows

$$\begin{aligned} \gamma_L &= \frac{r_{2_L}}{LU} \quad \text{for motion about } L_1 \text{ or } L_2 \\ \gamma_L &= \frac{r_{1_L}}{LU} \quad \text{for motion about } L_3 \end{aligned} \tag{5.2}$$

with r_{1_L} and r_{2_L} representing the distances of the primaries from the reference Lagrangian point. However, since the distances have been non-dimentionalized, these two variables are equal to one in their respective applications. The linearized EoMs admit solutions of the kind

$$\begin{aligned} x &= -A_x \cos \lambda t + \phi \\ y &= k A_x \sin \lambda t + \phi \\ z &= A_z \sin \lambda t + \psi \end{aligned} \tag{5.3}$$

which describe a periodic halo-type motion. A_x and A_z are connected by an algebraic relation, as are the phase angles ϕ and ψ. Both relations will be explained shortly. Linearized solutions are just the first step in obtaining the third-order solutions for a periodic motion around the collinear points.

In order to accomplish this, the EoMs must be rewritten around one collinear point using the Lagrangian L, expressed as

$$L = \frac{1}{2}(\dot{\boldsymbol{\rho}} \cdot \dot{\boldsymbol{\rho}}) + Gm_1 \left[\frac{1}{|\mathbf{r_1} - \boldsymbol{\rho}|} - \frac{\mathbf{r_1} \cdot \boldsymbol{\rho}}{|\mathbf{r_1}|^3} \right] + Gm_2 \left[\frac{1}{|\mathbf{r_2} - \boldsymbol{\rho}|} - \frac{\mathbf{r_2} \cdot \boldsymbol{\rho}}{|\mathbf{r_2}|^3} \right] \tag{5.4}$$

where $\boldsymbol{\rho}$ is the position vector of the mass m_3 with respect to the Lagrangian point. Expanding the two gravitational contributions $\frac{Gm_1}{\mathbf{r_1}}$ and $\frac{Gm_2}{\mathbf{r_2}}$ in series, Equation (5.4) is rewritten as

$$L = \frac{1}{2}(\boldsymbol{\rho}^* \cdot \boldsymbol{\rho}^*) + \sum_{n=2}^{\infty} c_n \rho^n P_n(x/\rho) \tag{5.5}$$

with the $(^*)$ operator indicating the derivation with respect to the variable $s = n_1 t$, where n_1 corresponds to the mean angular motion of the system. $P_n(x/\rho)$ represents the n-th Legendre polynomial, whose constants c_n are retrieved according to the following

equation

$$\begin{aligned} c_n &= \frac{1}{\gamma_L^3}\left[(\pm 1)^n \mu + (-1)^n \frac{(1-\mu)\gamma_L^{n+1}}{(1\mp\gamma_L)^{n+1}}\right], \quad \text{for } L_1 \text{ or } L_1 \\ c_n &= \frac{1}{\gamma_L^3}\left[1-\mu+\frac{\mu\gamma_L^{n+1}}{(1+\gamma_L)^{n+1}}\right], \quad \text{for } L_3 \end{aligned} \tag{5.6}$$

Using Equations (5.5) and (5.6), the EoMs can be written as

$$\begin{aligned} x^{**} - 2y^* - (1+2c_2)x &= \sum_{n=2}^{\infty}(n+1)c_{n+1}\rho^n P_n(x/\rho) \\ y^{**} + 2x^* + (c_2-1)y &= \sum_{n=3}^{\infty} c_n y \rho^{n-2}\tilde{P}_n(x/\rho) \\ z^{**} + c_2 z &= \sum_{n=3}^{\infty} c_n z \rho^{n-2}\tilde{P}_n(x/\rho) \end{aligned} \tag{5.7}$$

with

$$\tilde{P}_n(x/\rho) = \sum_{k=0}^{[(n-2/2]} (3+4k-2n)P_{n-2k-2}(x/\rho) \tag{5.8}$$

To derive periodic halo-type solutions from Equation (5.7), the amplitudes of the linearized solutions, A_x and A_z, must be large enough so that the nonlinear contributions to the system produce equal eigenfrequencies. Based on this hypothesis, Richardson's EoMs are

$$\begin{aligned} x^{**} - 2y^* - (1+2c_2)x &= 0 \\ y^{**} + 2x^* + (c_2-1)y &= 0 \\ z^{**} + \lambda^2 z &= 0 \end{aligned} \tag{5.9}$$

where the coefficient c_2 in Equation (5.7) is replaced by λ^2. These two terms are necessary to compute the correction Δ

$$\Delta = \lambda^2 - c_2 \tag{5.10}$$

which must be applied to the right-hand side of the z-equation of Equation (5.7). Thus, the z-equation becomes

$$z^{**} + c_2 z = \sum_{n=3}^{\infty} c_n z \rho^{n-2}\tilde{P}_n(x/\rho) + \Delta z \tag{5.11}$$

As a result of the successive approximations procedure, secular terms that are desired to be eliminated, appear. To eliminate them, the Linsted-Poincaré method is applied, introducing the variable τ to substitute the temporal variable t. This variable is of the kind

$$\tau = \omega s \tag{5.12}$$

where ω is the correction frequency equal to

$$\omega = 1 + \sum_{n\geq 1} \omega_n \quad \text{with} \quad \omega_n \leq 1 \tag{5.13}$$

Equation (5.7) is then rewritten as a function of τ

$$\begin{aligned}
&\omega^2 x'' - 2\omega y' - (1+2c_2)x = \frac{3}{2}c_3(2x^2 - y^2 - z^2) + 2c_4 x(2x^2 - 3y^2 - 3z^2) \\
&\omega^2 y'' + 2\omega x' + (c_2 - 1)y = -3c_2 xy - \frac{3}{2}c_4 y(4x^2 - y^2 - z^2) \\
&\omega^2 z'' + \lambda^2 z = -3c_3 xz - \frac{3}{2}c_4 z(4x^2 - y^2 - z^2)
\end{aligned} \tag{5.14}$$

Third-order solutions can be derived, without the secular terms, by specifying properly the values of ω_n. In particular, from the second-order equations is known that $\omega_1 = 0$, while the following relation holds for ω_2

$$\omega_2 = s_1 A_x^2 + s_2 A_z^2 \tag{5.15}$$

with constants s_1 and s_2 reported in the Appendix A. To ignore the secular terms in the z-equation as well, the two relations between the amplitudes and the phase angles must be used. The relationship between the amplitudes expresses as follows

$$l_1 A_x^2 + l_2 A_z^2 + \Delta = 0 \tag{5.16}$$

where the expressions of l_1 and l_2 are listed in Appendix A. Finally, the aforementioned constraint-relationship between the phase angles is

$$\psi = \phi + \frac{n\pi}{2} \quad \text{with} \quad n = 1, 3 \tag{5.17}$$

Based on these considerations, the third-order solutions for a periodic motion around each collinear point are derived

$$\begin{aligned}
x &= a_{21}A_x^2 + a_{22}A_z^2 - A_x \cos\tau_1 + (a_{23}A_x^2 - a_{24}A_z^2)\cos 2\tau_1 + (a_{31}A_x^3 - a_{32}A_x A_z^2)\cos 3\tau_1 \\
y &= kA_x \sin\tau_1 + (b_{21}A_x^2 - b_{22}A_z^2)\sin 2\tau_1 + (b_{31}A_x^3 - b_{32}A_x A_z^2)\sin 3\tau_1 \\
z &= \delta_n A_z \cos\tau_1 + \delta_n d_{21} A_x A_z(\cos 2\tau_1 - 3) + \delta_n(d_{32}A_z A_x^2 - d_{31}A_z^3)\cos 3\tau_1 \\
\dot{x} &= \lambda\omega n_1 \left[A_x \sin\tau_1 - 2(a_{23}A_x^2 - a_{24}A_z^2)\sin 2\tau_1 - 3(a_{31}A_x^3 - a_{32}A_x A_z^2)\sin 3\tau_1\right] \\
\dot{y} &= \lambda\omega n_1 \left[kA_x \cos\tau_1 + 2(b_{21}A_x^2 - b_{22}A_z^2)\cos 2\tau_1 + 3(b_{31}A_x^3 - b_{32}A_x A_z^2)\cos 3\tau_1\right] \\
\dot{z} &= \lambda\omega n_1 \left[-\delta_n A_z \sin\tau_1 - 2\delta_n d_{21} A_x A_z \sin 2\tau_1 - 3\delta_n(d_{32}A_z A_x^2 - d_{31}A_z^3)\sin 3\tau_1\right]
\end{aligned} \tag{5.18}$$

where the switch function δ_n is expressed as

$$\delta_n = 2 - n \quad \text{with} \quad n = 1, 3 \tag{5.19}$$

and, according to the value of n, it distinguishes between Halo Northern orbits and Halo Southern orbits, respectively. Finally, τ_1 is equal to

$$\tau_1 = \lambda\tau + \phi \tag{5.20}$$

with constants a_{ij}, b_{ij} e d_{ij} listed in Appendix A.

However, the initial conditions computed according to this procedure belong to the reference frame centered on the Lagrangian point. Therefore, to express them with respect to the synodic reference frame, the inverse of Equation (5.1) must be used as the final step.

Algorithm 3 shows the sequence of operations necessary to obtain the initial conditions of an Halo orbit using the Richardson method.

Algorithm 3 Construction of initial conditions for a periodic Halo orbit

1: Choose the system and the collinear Lagrangian point
2: Set the A_z amplitude and the phase angle ϕ
3: Compute the local non-dimensional distance: γ_L according to Eq. (5.2)
4: Compute Lagrangian constants c_n using Eq. 5.6
5: Compute the linearized frequency λ and constant k
6: Use Appendix A to compute the second- and third-order coefficients
7: Solve Eq. (5.16) for the in-plane amplitude: A_x
8: Compute the Lindstedt–Poincaré frequency correction with Eq. (5.15)
9: Compute ψ with Eq. (5.17) choosing $n = 1$ for Northern or n_3 for Southern orbit
10: Evaluate the third-order solution at $\tau = 0$ according to Eq. (5.18)

The initial conditions produced as output by Algorithm 3 must be corrected using one of the differential correction algorithms to ensure that they lead to a periodic orbit. As mentioned in Section 4.3, using the single shooting method is sufficient for this orbit family.

Table 5.1 lists the values of the mass ratio μ and the non-dimensional position along the x-axis of the collinear points in the four new systems to which the Richardson procedure has been applied.

Table 5.1: Mass ratios and non-dimentionalized positions collinear points of new systems

System	**Mass ratio**	L_1	L_2	L_3
Sun-Mercury	1.66×10^{-7}	0.99619396701	1.00381538196	-1.00000006917
Jupiter-Ganymede	7.81×10^{-5}	0.97058373638	1.02984548118	-1.00003252728
Pluto-Charon	0.1087	0.59262439711	1.26258391589	-1.04523234461
α Centauri A-α Centauri B	0.4519	0.06792829736	1.21460314181	-1.18130951202

In addition to some three-body systems belonging to the solar system, Table 5.1 includes also the stellar system α Centauri. This is actually a triple stellar system, and in fact Proxima Centauri has been demonstrated to be gravitationally connected to the system. However, it is neglected due to its large distance from the barycenter of the binary system α Centauri AB. Such hypothesis allows to consider a system with a considerably high mass ratio and analyze a limit case to which the CR3BP hypothesis can be applied.

Figures 5.1 and 5.2 are an example of the results that can be obtained applying Algorithm 3 and Algorithm 1 to derive initial conditions and correct them in the Pluto-Charon system.

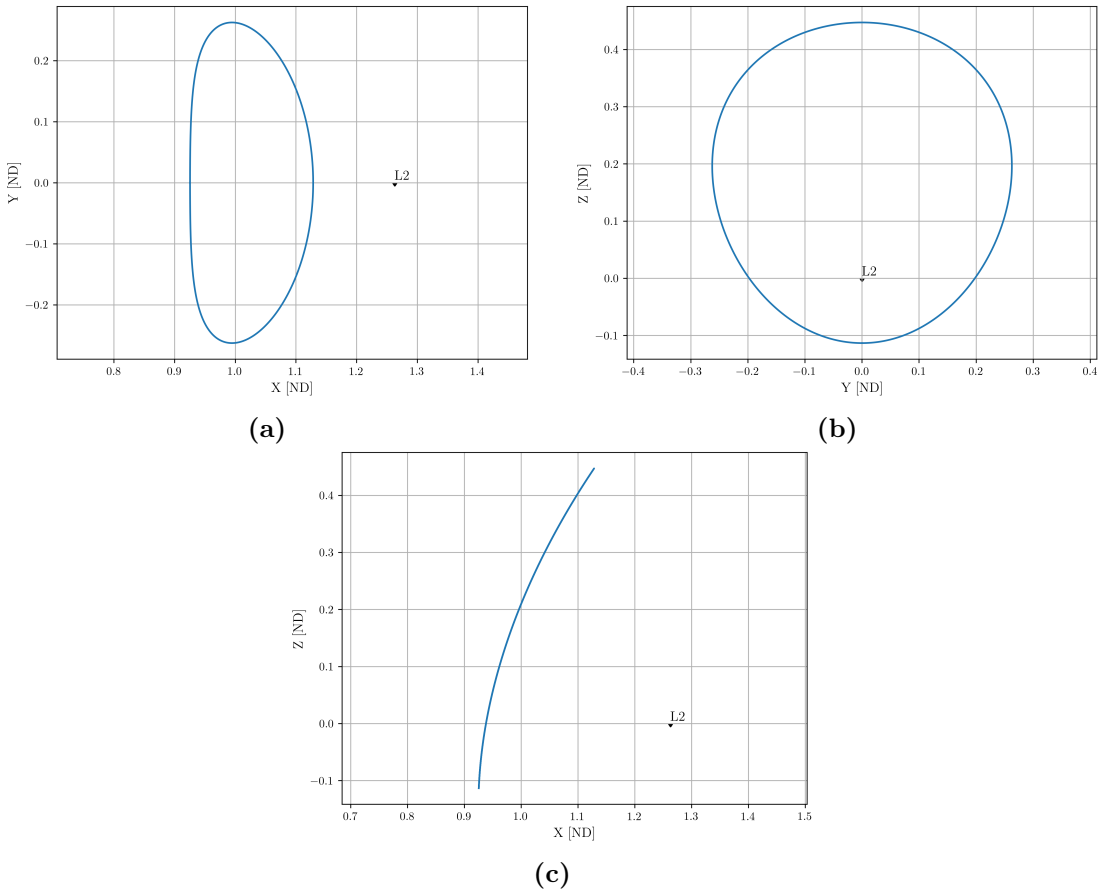


Figure 5.1: 2D view of the computed Northern Halo orbit around L_2 in the Pluto-Charon system with $A_z = 2800$ km: **(a)** x-y plane, **(b)** y-z plane, **(c)** x-z plane

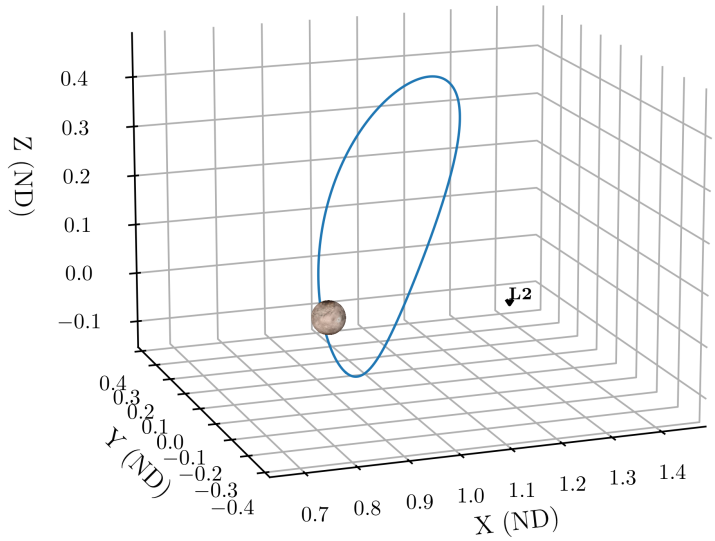


Figure 5.2: 3D view of the computed Northern Halo orbit around L_2 in the Pluto-Charon system with $A_z = 2800$ km

More examples, using other systems cited in Table 5.1, are shown in Appendix B. Looking at the results, it can be seen that, apart from the Pluto-Charon system, it was not possible to identify at least one periodic orbit around each of the collinear points for any other system. Consequently, Richardson's detailed procedure does not always produce satisfactory results, i.e. sets of initial conditions that are sufficiently accurate to be corrected effectively. In fact, several attempts carried out in different systems have produced negative results or have shown that the method is strongly influenced by the value of A_z provided. To overcome this problem, genetic algorithms can be used, which search within an interval for the presence of a possible solution.

5.2 Genetic Algorithm

Genetic Algorithms (GAs) are a class of optimization algorithms belonging to the research field of the evolutionary computation. GAs are based on random processes, just like evolutionary biological processes, and shares with these some fundamental terms, such as *population*, *generation* and *chromosomes*. Unlike natural biological processes, the creator of a GA can set the level of randomization and the level of control. GAs have advantages over other search algorithm because their randomic nature enables them to explore wider search spaces without requiring additional information about the problem. This feature places GAs ahead of other optimization methods, which instead require information about linearity, continuity or derivatives to find a solution.

The mechanism underlying GAs is straightforward. Starting from an initial population, the suitability of each individual is evaluated using a specific function, and the best individuals are chosen to produce the new generation. Through this process, the algorithm tends to the solution of the problem by producing individuals, i.e. possible

solutions, which are more suitable with each generation. It should be noted that the objective of the GA is not to identify all possible solution within a certain search interval, but rather to obtain solutions that are good enough to be provided to a differential correction algorithm as initial guesses for the initial conditions of a periodic orbit.

5.2.1 Individuals of the population

In order to proceed with the creation of the GA, it is first necessary to determine the characteristics of the individuals forming the population. Since the goal is to find possible initial conditions for periodic Halo orbits, each individual is represented by a two-component vector

$$\mathbf{v} = [x_0, \dot{y}_0] \tag{5.21}$$

It is already known that the state vector of a periodic Halo orbit has just three non-zero components, one of which is kept fixed during the differential correction process. Consequently, chosing z as the invariant variable, each individual has two chromosomes, corresponding to x_0 and $\dot{y}_0$, which the GA will manipulate.

Another fundamental step in initializing the population is defining the search interval. To verify the GA's performance, a periodic solution is drawn from the NASA JPL database and the two intervals are built simmetrically around the nominal values fot the x and $\dot{y}$ components as follows

$$\begin{aligned} \mathbf{v}_{\text{min}} &= \mathbf{v}_{\text{p}} - \delta \cdot \mathbf{v}_{\text{p}} \\ \mathbf{v}_{\text{max}} &= \mathbf{v}_{\text{p}} + \delta \cdot \mathbf{v}_{\text{p}} \end{aligned} \tag{5.22}$$

dove where $\mathbf{v}_{\text{p}}$ represents the periodic solution known a priori. The parameter δ must be chosen carefully to avoid two possible scenarios:

- the extremes of the search interval are too close to $\mathbf{v}_{\text{p}}$ and cause the GA to converge too quickly;
- the search intervals are too wide, which negatively affect the GA's efficiency.

5.2.2 Fitness function

After the characteristics of the individuals have been stated, their suitability must be evaluated using the *fitness function*, which is usually indicated as the cost of the individual. The definition of this function is crucial for correctly selecting the individuals. In this case, the symmetry characteristics of Halo orbits, discussed in Section 3.2.4, can be used. Halo orbits are known to cross the xz-plane perpendicularly to it, hence in two distinct points $\dot{y}$ has a non-zero value, while $\dot{x}$ and $\dot{z}$ are null. Using this feature, the EoMs are integrated using the $\mathbf{v}$-components and the fixed z component as initial conditions. The integration process stops when an event function detects that the xz-plane has been crossed. After the integration, the cost of the individual is computed

using the following fitness function

$$f_{\text{fit}} = |\dot{x}| + |\dot{z}| \tag{5.23}$$

The goal of the GA is to minimize the value of the fitness function so that the solution is coherent with the definition of a Halo orbit. The absolute values of the two velocities $\dot{x}$ and $\dot{z}$ are used to compute the sum, preventing the fitness function from approaching zero without ensuring symmetry. If the event function is not triggered during the integration, the cost of the individual is set to be $f_{\text{fit}} = 10^4$. This passage helps to strongly penalize less suitable individuals. These individuals are more likely to be excluded from creating the new generation and be progressively eliminated.

5.2.3 Natural selection

Natural selection plays a crucial role in creating a new generation of individuals. Indeed, this process leads to the selection of the best individuals to be the future parents. Specifically, individuals are sorted according to their fitness value, from highest to lowest. Only a fraction of the whole population is selected to advance to the next steps, typically setting the survival rate to 50%. Therefore, at each iteration of the GA, half of the population consisting of the lowest-cost individuals survives, while the other half is replaced by the new generation.

A different approach to natural selection is the *thresholding*. With this technique, individuals with a fitness function value that is less than the threshold survive. Initially, just a few individuals survive, but this number increases in subsequent generations.

Even though thresholding allows to select the best individuals without previously sorting the population, classic natural selection is chosen for the GA.

5.2.4 Selection

The selection process consists in pairing the individuals who survived natural selection so that new individuals can be generated in the next mating phase. The pairing process aims to create a number of couples equal to $N_{\text{pairs}} = N_{\text{pop}} - N_{\text{par}}$, with $N_{\text{pop}}$is the total population size and N_{par} is the number of parents. The most common selection methods are as follows [44]:

- Pairing from top to bottom, selecting two individuals at a time until the necessary number of couples is reached. This approach is very easy to implement, but it doesn't accurately model nature
- Random pairing is performed by simply randomically extracting the index of the individuals which form the couple;
- Weighted random pairing associates a selection probability to each individual which is inversely proportional to its cost. Then, the extraction of a random

number determines the selection of a certain individual. This type of selection is also known as *roulette wheel weighting* and it can be implemented using the method of the *rank weighting*. In practice, the probability associated with an individual is computed as a function of its position n in the sorting

$$P_n = \frac{N_{\text{par}} - n + 1}{\sum_{n=1}^{N_{\text{par}}} n} \tag{5.24}$$

According to the calculated probabilities and the random numbers extracted, an individual may be paired with itself or more couples may be formed from the same individuals. To avoid this, a control mechanism can be implemented to force the extraction to be repeated so until no two couples are the same.;

- Tournament selection randomly selects a small group of individuals, usually two or three, and identifies the least expensive among them. This individual is one of the parents of the couple. The tournament repeats until all couples are formed. This selection method is usually preceded by thresholding, so that the population is never sorted.

Since roulette wheel and tournament selection are both standard GAs selection mechanisms, the roulette wheel rank weighted selection has been chosen for implementation.

5.2.5 Mating

After selecting and pairing the individuals, the mating or crossover phase follows. This is the process that leads to the next generation of individuals. Mating can occur according to a fixed probability, but in this case it is imposed to happen for each couple. Crossover creates new individuals which inherit their parents' characteristics. To accomplish this, chromosomes of the new individuals are computed as a weighted average of their parents'. The implemented equations are as follows

$$\begin{aligned} \mathbf{v}_1^{\text{new}} &= \alpha \cdot \mathbf{v_1} + (1-\alpha) \cdot \mathbf{v_2} \\ \mathbf{v}_2^{\text{new}} &= \alpha \cdot \mathbf{v_2} + (1-\alpha) \cdot \mathbf{v_1} \end{aligned} \tag{5.25}$$

where α is a random number belonging to [0,1] that is indipendently extracted for each mating operation. Because of the random nature of α, it is unlikely, though not impossible, that it will be equal to one, resulting in the new individuals being copies of the parents.

5.2.6 Mutation

Mutations are applied to the new individuals to encourage a higher genetic diversity and prevent the GA from stalling. Without mutations, the GA can converge to a local optimum rather than a global optimum of the fitness function. To avoid this behavior,

mutations are set to occur with a 20% probability. When a mutation occurs, a small perturbation is applied to both chromosomes of the individual. This perturbation is modeled as random noise that follows a normal distribution. Therefore, the random perturbation has a mean value of zero and a standard deviation of one. Altought it is unlikely that the perturbation will have a high value, it is anyway scaled by a factor β before being applied, according to the following

$$\mathbf{v}_{\text{mut}} = \mathbf{v} + \beta \cdot \boldsymbol{\eta} \quad \text{with} \quad \boldsymbol{\eta} \sim N(0, \mathbf{I}) \tag{5.26}$$

where $\boldsymbol{\eta}$ si the vector that contains the two perturbations to be applied, which are randomly extracted independently .

The application of perturbations can cause the chromosomes to exceed the search interval limitations. It is sufficient to verify that the limits are still respected after the mutation. If not, the mutated chromosome can be rounded to the nearest interval extreme.

5.2.7 GA performances

This section will test the GA implemented, which is more schematically describe in Algorithm 4.

Algorithm 4 Genetic algorithm for periodic Halo orbit initial conditions

Choose NASA JPL nominal values z_{nom} and $\mathbf{v}_{\text{nom}} = [x_0, \dot{y_0}]$
Choose GA parameters: perturbation factor δ, population size N_{pop}, number of generations N_{gen}, number of runs N_{run}, mass ratio μ, threshold ε
for $j = 1, 2$ **do** ▷ Building search intervals
    $[\underline{b}_j, \bar{b}_j] \leftarrow IC_j(1 \pm \delta)$
for $r = 1, \ldots, N_{\text{run}}$ **do** ▷ Monte Carlo over different GA runs
    Initialize population $\mathcal{P}^{(0)} = \{\mathbf{v}_k^{(0)}\}_{k=1}^{N_{\text{pop}}}$:
    while $i < N_{\text{gen}}$ **and** flag $= 0$ **do** ▷ Main GA loop
        for each individual $\mathbf{y}_k^{(i)} \in \mathcal{P}^{(i)}$ **do**
            Integrate EoMs on $t \in [0, 2\pi]$ with plane-crossing event $y = 0$
            if event is detected **then**
                $f_{\text{fit}_k} \leftarrow |\dot{x}_k| + |\dot{z}_k|$
            else
                $f_{\text{fit}_k} \leftarrow 10^4$
        Sort $\mathcal{P}^{(i)}$ in ascending order of f_k
        Extract best fitness $f_{\text{best}}^{(i)}$ and best individual $\mathbf{y}_{\text{best}}^{(i)}$
        if $f_{\text{best}}^{(i)} < \varepsilon$ **then**
            flag $\leftarrow 1$
        Keep the best $N_{\text{par}} = N_{\text{pop}}/2$ individuals from $S^{(i)}$

Assign rank-based selection probabilities p_j and cumulative sums P_j
Use roulette-wheel sampling on P_j to form N_{par} couples of distinct parents
for each couple of parents $(\mathbf{v}_1, \mathbf{v}_2)$ **do**
    Draw $\beta_1, \beta_2 \sim \mathcal{U}(0,1)$
    $\mathbf{v}_1^{\text{new}} \leftarrow \beta_1 \mathbf{v}_1 + (1-\beta_1)\mathbf{v}_2; \quad \mathbf{v}_2^{\text{new}} \leftarrow \beta_2 \mathbf{v}_2 + (1-\beta_2)\mathbf{v}_1$
    if mutation is enabled **then**
        With $p_m = 0.2$, perturb $\mathbf{v}_1^{\text{new}}, \mathbf{v}_2^{\text{new}}$ with Gaussian noise
        Project mutated components back into their bounds $[\underline{b}_j, \overline{b}_j]$
    Add $\mathbf{v}_1^{\text{new}}$ and $\mathbf{v}_2^{\text{new}}$ to $\mathcal{P}^{(i+1)}$
$\mathcal{P}^{(i)} \leftarrow \mathcal{P}^{(i+1)}, \quad i \leftarrow i+1$

As previously mentioned, the reference initial conditions are selected from those available in the NASA JPL database. Specifically, the initial conditions vector describes a Halo Northern orbit built around L_2 in the Earth-Moon system, which has a stability index equal to $index = 100.3033$. In such way, the influence of the orbit stability on convergence capacity of the GA should be mitigated.

Table 5.2 presents the values of the parameters used to set the GA. The threshold value, as well as the parameter δ used to create the search interval bounds, have been chosen to prevent premature convergence.

Table 5.2: Setting parameters for the convergence test using a Halo Northern orbit built around L_2 in the Earth-Moon system

Parameter	Symbol	Value
Number of runs	N	10
Population size	P	100
Max generations	G_{max}	100
Threshold	ϵ	10^{-4}
Mutation probability	p_m	0.2
Mutation scale factor	α	0.05
Relative search radius	δ	0.05

The solution found at the end of a run is strongly related to the starting seed, namely the initial population randomly generated. This mainly affects the number of generations required to reach the threshold and the quality of the computed solution. Repeated use of the GA lead to different solutions, therefore it allows to select the solution with the lowest cost to be used as an initial guess for the differential correction algorithm. Considering the following nominal initial conditions

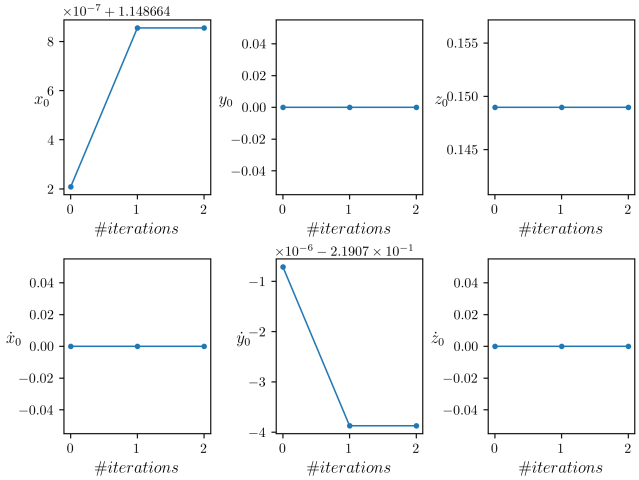
$$\mathbf{X}_{\text{JPL}} = [1.1486648559117889, 0, 0.14897322998167725, 0, -0.21907387052814919, 0] \tag{5.27}$$

the results obtained at the end of the ten runs set are summarized in Table 5.3.

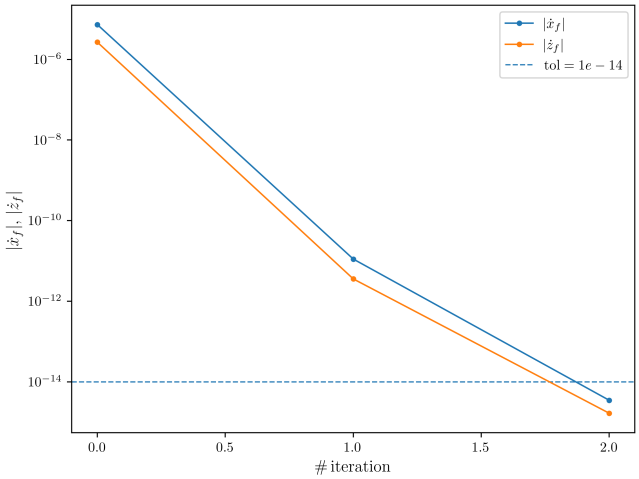
Table 5.3: GA results for the Halo Northern reference orbit built around L_2 in the Earth-Moon system

Convergence [%]	**Mean Gen.**	**Best cost**	**Best solution**
50%	8.8	9.98×10^{-6}	$[1.14866420854413, -0.219070715292363]$

Even with the threshold set to $\epsilon = 10^{-4}$, the best solution found across the ten runs has a cost that is two orders of magnitude lower. However, the chosen threshold proved to be very strict, indeed half of the runs failed to converge and produced solutions with a mean cost of 10^{-3}. The average number of the generations required can be interpreted as a positive result because, when the GA converges, it does so quickly. To assess the validity of the solution obtained, it is sufficient to analyze the results of the single shooting differential correction algorithm showed in Figures 5.3a and 4.3.



(a)



(b)

Figure 5.3: Results of the single shooting differential correction algorithm applied to the set of initial conditions provided by the GA

These graphs prove that the initial guess coming given by the GA is very accurate. Indeed differential correction algorithm converges in only two iterations, with a tolerance of 10^{-14}.

Repeating the test with a modified threshold of $\epsilon = 10^{-2}$, yields different results, particularly with regard to the convergence rate and the average number of iterations needed to satisfy the imposed tolerance are concerned, as shown by Table 5.4.

Table 5.4: GA results for the Halo Northern reference orbit built around L_2 in the Earth-Moon system with modified threshold

Convergence [%]	**Mean Gen.**	**Best cost**	**Best solution**
100%	1.6	1.03×10^{-3}	$[1.14859986681057, -0.218339208434893]$

Using a less restrictive threshold, the GA can still find a solution, but the solution has a higher cost. However, the single shooting differential correction effectively corrects the derived initial conditions in just one additional iteration compared to the previous test. Therefore, it is reasonable to assume that solutions obtained by setting a threshold of $\epsilon = 10^{-3}$ are precise enough to serve as initial guess to the differential correction and the computational cost is not too high.

5.3 Derivation of initial conditions using GA

Once the GA structure is detailed, it can be integrated into the initial conditions derivation process for new three-body systems. Unlike the tests conducted so far, the initial conditions are derived by imposing few constraints and without knowing any solutions in advance. Specifically, the characteristics of the desired orbit are defined by choosing the three-body system in which deriving the initial conditions, the Lagrangian point around which to build the orbit and its amplitude A_z. Finally, since the process regards Halo orbits, the desired orbits must be specified as either Northern or Southern. This information is needed to set Richardson's procedure, depicted in Section 5.1, to obtain an estimation of the initial conditions. This first result, is essential for generating the search intervals of the GA.

Built the search intervals or radius δ around the first estimate of the initial conditions, the GA parameters are set to begin finding a solution. Due to the different circumstances for the application of the GA compared to the tests performed in Section 5.2.7, Table 5.5 reports the values of the parameters used from now on.

Table 5.5: Genetic algorithm parameters used in the derivation of initial conditions for Pluto-Charon and α Centauri AB

Parameter	Symbol	Value
Number of runs	N	100
Population size	P	100
Max generations	G_{max}	100
Threshold	ϵ	10^{-2}
Mutation probability	p_m	0.2
Mutation scale factor	α	0.05
Relative search radius	δ	0.3

The number of runs is increased to 100 to improve the likelihood of finding low-cost solutions, hence best initial guesses fot the differential correction. Furthermore, since the solution is unknown, a threshold of 10^{-2} is considered sufficient. Finally, the search intervals are built using $\delta = 0.3$. This means that the GA will investigate a space spanning 30% above and below Richardson's estimation. This decision is justified by the fact that the solution may not be located in the immediate vicinity of the estimation. Therefore, wider intervals give more the GA room for maneuver and increase the chance of finding a solution.

Finally, once the lowest-cost solutions is selected, the initial conditions vector consisting of the two chromosomes $[x_0, \dot{y}_0]$ and the z coordinate obtained by fixing the value of A_z is corrected. The initial conditions vector returned by the single shooting differential correction represents the initial state that leads to a periodic orbits that responds to the initial constraints.

5.3.1 Pluto-Charon system

The first system chosen to demonstrate the functioning of the GA that does not belong to the NASA JPL database is Pluto-Charon. Pluto, was once classified as the ninth planet of the solare system but was reclassified as dwarf planet in 2006, after the discovery of Eris, a dwarf planet with a mass that is 27% higher than Pluto's. Although Pluto has five known moons, Charos is selected due to its unique characteristics. Indeed, Charon's diameter is half that of Pluto, and its mass is one-eighth that of Pluto. Unlike many other planetary systems in the solar system, where moon's dimensionts are remarkably smaller than the mother body, Pluto-Charon is a binary system. Because of Charon's dimensions the barycenter of this system lies outside both bodies. This characteristic is confirmed by the value of the mass ratio, that is $\mu = 0.10873$. For comparison, this mass ratio is approximately ten times higher than that of the Earth-Moon system, which is already considered to be quite high. This high mass ratio not only makes the dynamics more complex and chaotic, but according to Equation (2.48), makes the equilateral points L_4 and L_5 unstable. These characteristics make the

Pluto-Charon system a good benchmark for the GA capabilities, where Richardson's analytical approximation begins to struggle.

A total of seven new Halo orbits are derived, each of them with a different amplitude, A_z, and orientation. Table 5.6 summarizes the characteristics of these orbits, which are used as input parameters for the analytical approximation of the initial conditions.

Table 5.6: Features of each Halo orbit in the Pluto-Charon system used as inputs for Richardson's approximation

Orbit	$\mathbf{A_z}$ **[ND]**	$\mathbf{A_z}$ **[km]**	**Orientation**	$\mathbf{L}_i$	**Mass ratio**
Orbit 1	0.25	1470	Southern		
Orbit 2	0.50	2950	Northern		
Orbit 3	0.75	4400	Northern		
Orbit 4	1.0	5860	Southern	L_1	0.10873
Orbit 5	1.25	7340	Southern		
Orbit 6	1.51	8850	Northern		
Orbit 7	1.75	10270	Northern		

Table 5.7 shows instead the set of initial conditions that were analytically approximated using the procedure detailed by Richardson.

Table 5.7: Initial conditions derived using Richardson's analytical approximation for each Halo orbit in the Pluto-Charon system

Orbit	**Analytical ICs**
Orbit 1	[0.572, 0, -0.0782026630800778, 0, 0.29, 0]
Orbit 2	[0.574, 0, 0.1604394860979834, 0, 0.412, 0]
Orbit 3	[0.585, 0, 0.2454337465373808, 0, 0.518, 0]
Orbit 4	[0.607, 0, -0.3358252995570846, 0, 0.584, 0]
Orbit 5	[0.64, 0, -0.4326959572476554, 0, 0.59, 0]
Orbit 6	[0.685, 0, 0.5372827659114511, 0, 0.517, 0]
Orbit 7	[0.739, 0, 0.6412207406270349, 0, 0.36, 0]

These initial conditions cannot be effectively corrected by the single shooting differential correction algorithm, especially when the $A_z \geq 0.6$. Consequently, employing a genetic algorithm is crucial to find a candidate solution within search intervals wide enough and centered on the x_0 and $\dot{y}_0$ components of the aforementioned initial conditions, which are approximated analytically.

Applying the GA using the parameters introduced in Table 5.5 and considering each individual with the characteristics described in Section 5.2.1, yields the results that are shown in Table 5.8.

Table 5.8: Results of the genetic algorithm for each Halo orbit in the Pluto-Charon system using $\epsilon = 10^{-2}$

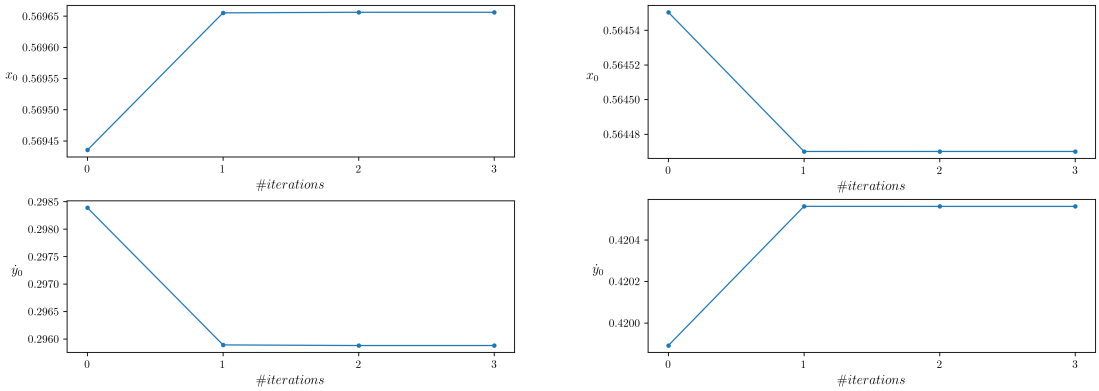
Orbit	Convergence [%]	Mean Gen.	Best cost	Best individual $[\mathbf{x_0}, \dot{\mathbf{y}}_\mathbf{0}]$
Orbit 1	94	10.75	6.705×10^{-4}	[0.56943553242901, 0.29839056639836]
Orbit 2	84	15.89	1.009×10^{-3}	[0.56455042560736, 0.4198921937556]
Orbit 3	97	9.24	4.085×10^{-4}	[0.556822250625615, 0.5337951187651]
Orbit 4	100	4.46	2.049×10^{-4}	[0.554195504589591, 0.6108414122852]
Orbit 5	100	15.91	4.674×10^{-4}	[0.61969235978080, 0.48115632001899]
Orbit 6	73	38.14	7.513×10^{-4}	[0.62710429350244, 0.38703794864584]
Orbit 7	35	28.51	1.859×10^{-3}	[0.57229330790619, 0.40353606152459]

A comparison of the solutions produced by the GA and the analytical estimation provided by Richadson's approximation, reveals that the approximations are very accurate for small values of A_z. In fact, a maximum percentage variation of 5% in the components $[x_0, \dot{y}_0]$ can be observed. However, this difference increases significantly as A_z increases, reaching a miximum percentage difference of almost 30% in the remaining four orbit components. This confirms that the analytical approximation is strongly limited. Table 5.9 shows that GA solutions are good initial guesses for the differential correction.

Table 5.9: Individuals obtained after the differential correction and absolute differences between the values derived by the GA and the differential correction for each Halo orbit in the Pluto-Charon system

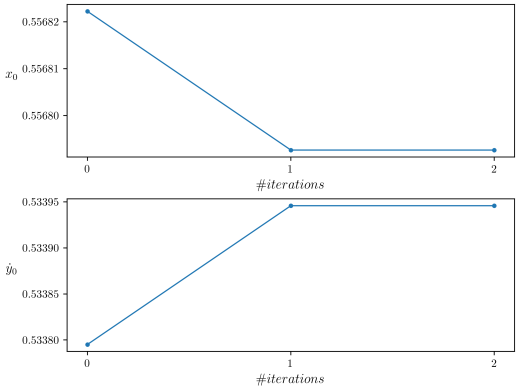
Orbit	Corrected $[\mathbf{x_0}, \dot{\mathbf{y}}_\mathbf{0}]$	$\lvert\boldsymbol{\Delta x_0}\rvert$	$\lvert\boldsymbol{\Delta \dot{y}_0}\rvert$
Orbit 1	[0.5696563323846876, 0.2958828502255298]	2.207×10^{-4}	2.507×10^{-3}
Orbit 2	[0.5644700840793219, 0.4205618476240396]	8.034×10^{-5}	6.696×10^{-4}
Orbit 3	[0.5567926282163066, 0.5339458056928368]	2.962×10^{-5}	1.506×10^{-4}
Orbit 4	[0.5542066273225952, 0.610763585831493]	1.112×10^{-5}	7.782×10^{-5}
Orbit 5	[0.6197836879799687, 0.4810328678007726]	9.133×10^{-5}	1.234×10^{-4}
Orbit 6	[0.6271700963729365, 0.3868877612009191]	6.581×10^{-5}	1.502×10^{-4}
Orbit 7	[0.5721940607310713, 0.4036804800195342]	9.925×10^{-5}	1.444×10^{-4}

The columns reporting the difference in absolute value of the corrected components witness that single shooting differential correction make always modifications starting from the fourth or fifth decimal digit. Thus, GA solutions are very close to an effective periodic solution and only require minor adjustments. As previously discussed, these minimal corrections are the actual discriminating factor between a non-periodic and a periodic, symmetrical solutions. Further confirmation of the quality of the GA solution is the speed of the correction process. Indeed, Figures 5.4 and 5.5 show that no more than three iterations are always needed to comply with a tolerance $\texttt{tol} = 10^{-13}$.

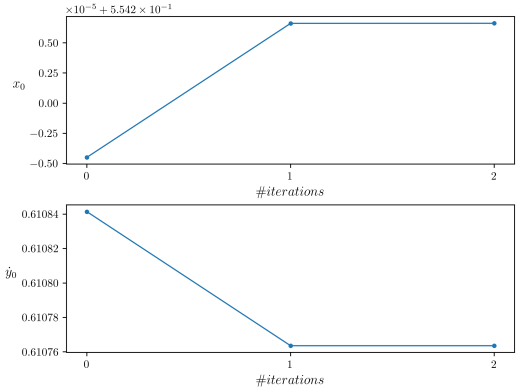


(a) Orbit 1

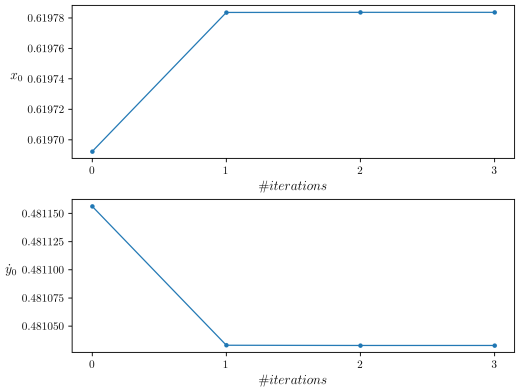
(b) Orbit 2



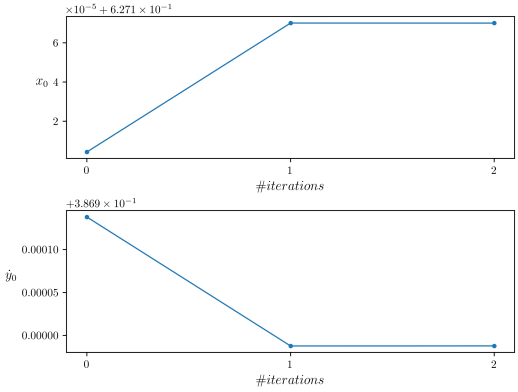
(c) Orbit 3



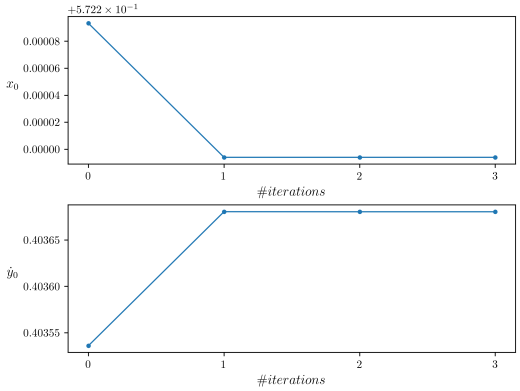
(d) Orbit 4



(e) Orbit 5

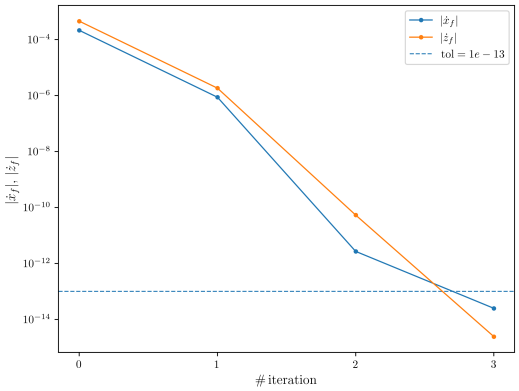


(f) Orbit 6

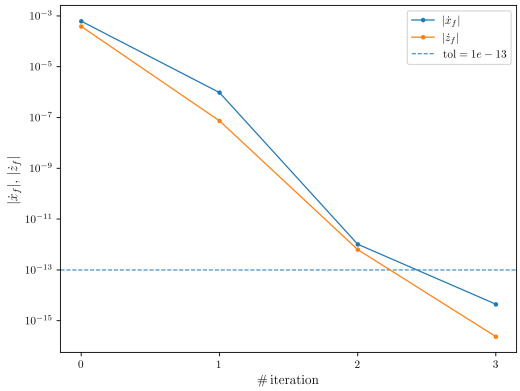


(g) Orbit 7

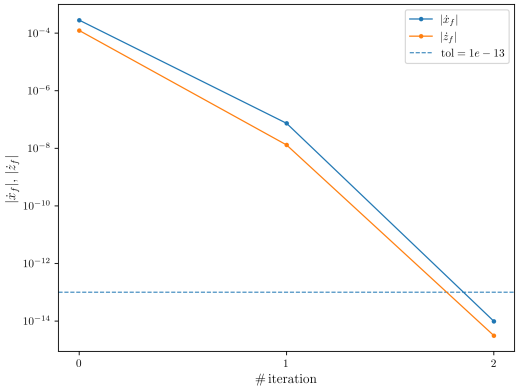
Figure 5.4: Initial conditions evolution throughout the differential correction process for each orbit in the Pluto-Charon system



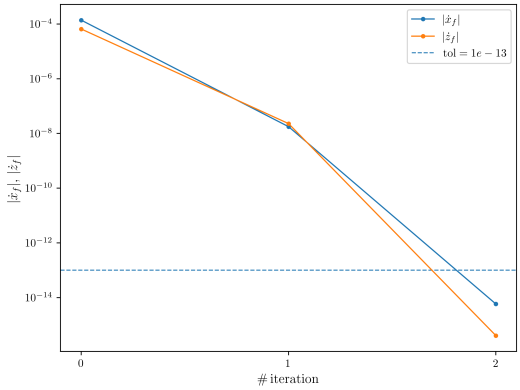
(a) Orbit 1



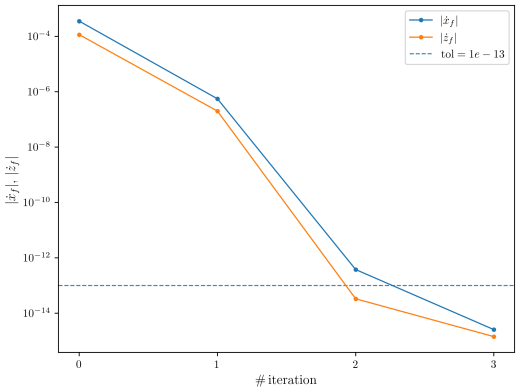
(b) Orbit 2



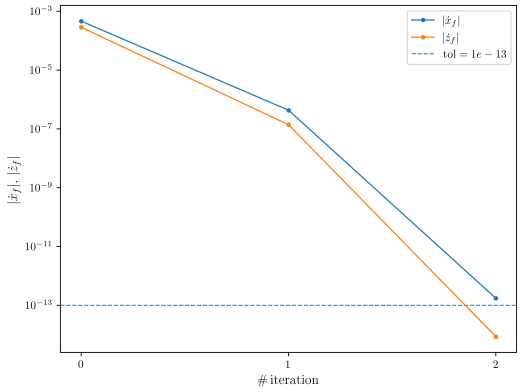
(c) Orbit 3



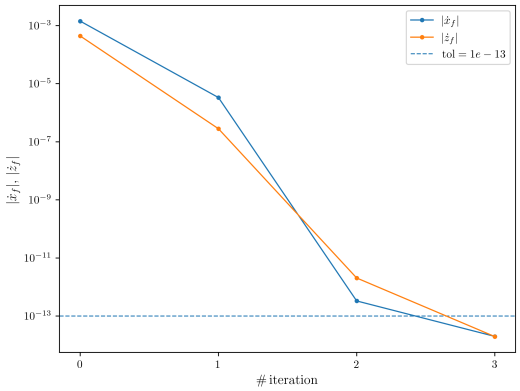
(d) Orbit 4



(e) Orbit 5



(f) Orbit 6



(g) Orbit 7

Figure 5.5: Final state velocities throughout the differential correction process for each orbit in the Pluto-Charon system

Figures 5.6 and 5.7 graphically show the orbits resulting from the integration of the initial conditions obtained as the output of the differential correction.

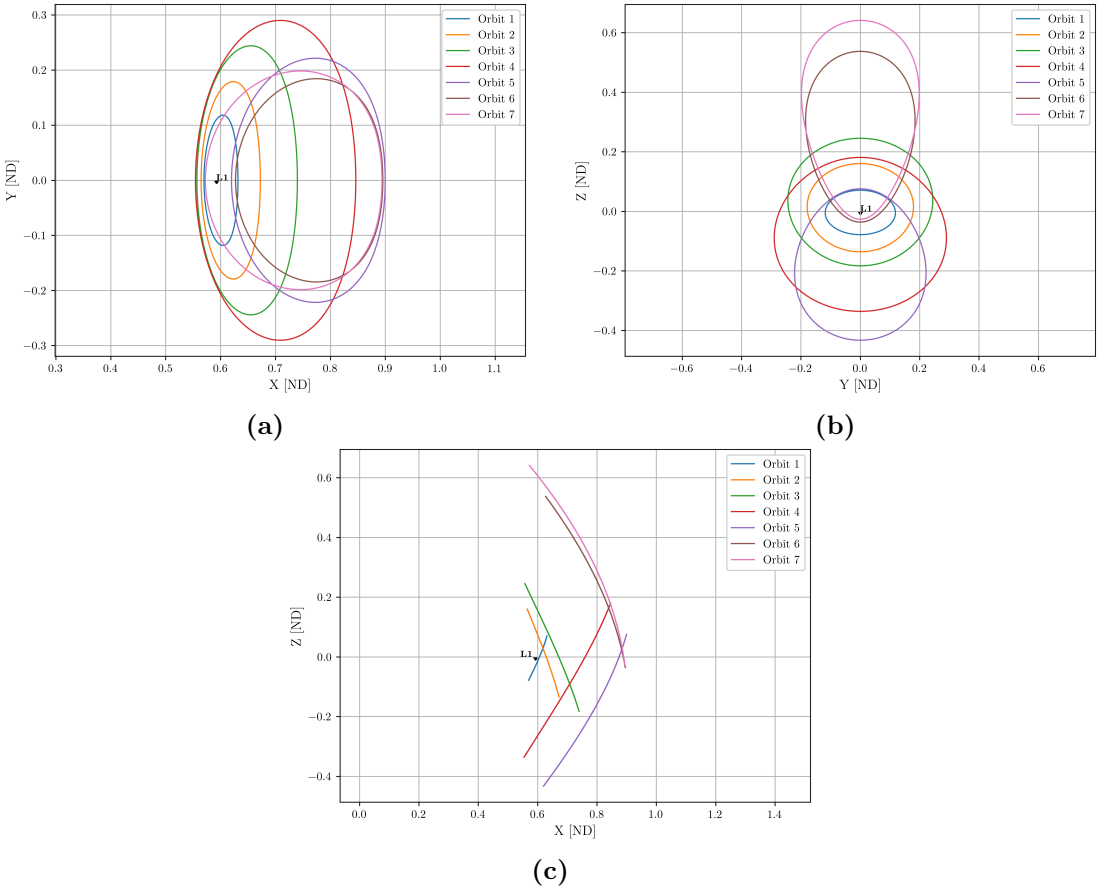


Figure 5.6: 2D view of the Halo orbits obtained using the GA around L1 in the Pluto-Charon system: (a) x-y plane, (b) y-z plane, (c) x-z plane

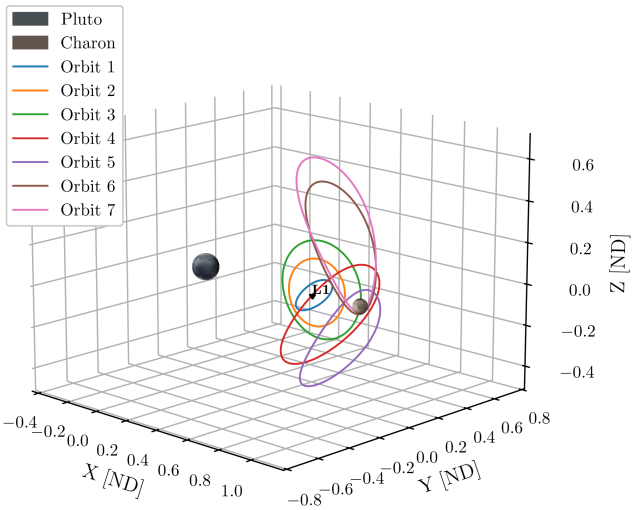


Figure 5.7: 3D view of the Halo orbits obtained using the GA around L1 in the Pluto-Charon system

Finally, Table 5.10 reports the periods and stability indexes of the resultant orbits. The value of the stability index marginally affects the GA's convergence percentage, which is instead more influenced by the z-amplitude. However, the correlation between the stability index and A_z, as described in Section 3.2.4 can be reasserted. As A_z increases, Halo orbits tend to become more elongated, falling into the NRHO family. As shown, these orbits have stability indexes close to or equal to one, and this trend is confirmed by the data presented in Table 5.10.

Table 5.10: Periods and stability indexes of the Halo orbits built around L_1 obtained using the GA in the Pluto-Charon system

Orbit	**A_z [ND]**	**A_z [km]**	**Period [ND]**	**Period [Days]**	**Stability index**
Orbit 1	0.25	1470	2.465	2.514	1124.36
Orbit 2	0.50	2950	2.559	2.609	464.26
Orbit 3	0.75	4400	2.662	2.715	114.98
Orbit 4	1.0	5860	2.606	2.658	14.69
Orbit 5	1.25	7340	2.166	2.209	1.36
Orbit 6	1.51	8850	2.229	2.273	1.11
Orbit 7	1.75	10270	2.445	2.494	4.41

5.3.2 α Centauri AB system

α Centauri is a triple star system, and also the closest stellar system to the solar system. Proxima Centauri is indeed located 4.365 light-years (ly) away, making it the closer star to Earth after the Sun. This system consists of the pair α Centauri A e α Centauri B, which are a yellow and an orange dwarf respectively, and are close to each other; and Proxima Centauri, a red dwarf much farther away. Table 5.11 reports the characteristics of the three bodies belonging to the system.

Table 5.11: α Centauri bodies characteristics

	α Centauri A	**α Centauri B**	**Proxima Centauri**
Distance from Earth	4.37 ly	4.37 ly	4.24 ly
Mass	1.1 $M_\odot$	0.9 $M_\odot$	0.12 $M_\odot$
Radius	1.22 $R_\odot$	0.86 $R_\odot$	0.14 $R_\odot$
Luminosity	151%$_\odot$	50%$_\odot$	0.15%$_\odot$
Surface Temp	5800 K	5300 K	3000 K
Periapsis	11.2 AU		4500 AU
Apoapsis	36 AU		13000 AU
Eccentricity	0.52		0.5
Orbital period	80 years		550,000 years

The subscript $\odot$ indicates that the mass, radius and luminosity are expressed as a

function of those of the Sun.

These data help explain why the analysis should focus on the system formed by α Centauri A ed α Centauri B, also known as α Centauri AB, altought α Centauri is a system composed by three bodies. These two stars have comparable masses and are much closer than Proxima Centauri, which is still linked to the system. Furthermore, eccentricity values highlight that the orbit along which the two main stars move, is highy eccentric, contrary to what is observed in planetay systems of the solar system. Consequently, applying the CR3BP hypothesis to this system seems unwarranted. However, analyzing α Centauri AB serves a different purpose, that is testing the potential of the algorithms described so far in unique conditions. Therefore, even if deriving Halo orbits in this system does not respond to a practical interest, it represents an excellent testing ground for GA in a system with a mass ratio close to its limit value. If the outcome is positive, GA could be more likely used to find periodic Halo orbits in an asteroid systems.

Table 5.12 reports the desired characteristics of the new orbits. Besides the mass ratio, the z amplitudes are notable, reaching values in the hundreds of astronomical units (AU). However, these orders of magnitude are not surprising given the physical characteristics of the system.

Table 5.12: Features of each Halo orbit in the α Centauri AB system used as inputs for Richardson's approximation

Orbit	**A_z [ND]**	**A_z [km]**	**Orientation**	**L_i**	**Mass ratio**
Orbit 1	0.1	237×10^6	Northern	L_2	0.451918
Orbit 2	0.3	708×10^6	Northern		
Orbit 3	0.6	1.02×10^9	Southern	L_1	
Orbit 4	0.9	1.53×10^9	Northern		

Using the procedure detailed in Section 5.1 yields the analytical initial conditions listed in Table 5.13.

Table 5.13: Initial conditions derived using Richardson's analytical approximation for each Halo orbit in the α Centauri AB system

Orbit	**Analytical ICs**
Orbit 1	[0.981, 0, 0.0507194238999922, 0, 0.777, 0]
Orbit 2	[0.928, 0, 0.1464499509721667, 0, 0.921, 0]
Orbit 3	[0.005, 0, -0.2929985669405203, 0, 0.693, 0]
Orbit 4	[-0.018, 0, 0.443029897974902, 0, 0.683, 0]

Consequently, the GA set with the same parameters shown in Table 5.5 is used. However, a small clarification is necessary. Due to the value of μ close to its theoretical limit, the point L_1 has an almost zero value. Therefore, constructing the search inter-

vals using the parameter $\delta = 0.3$ leads to an extremely narrow search space. Thus, the two orbits built around L_1 are derived using a search interval set at [-0.3, 0.3] for the x_0 component, and the interval used to find the correct value of the $\dot{y}_0$ component is built using $\delta = 1$. Following these modifications, the GA produced the results reported in Table 5.14.

Table 5.14: Results of the genetic algorithm for each Halo orbit in the α Centauri AB system using $\epsilon = 10^{-2}$

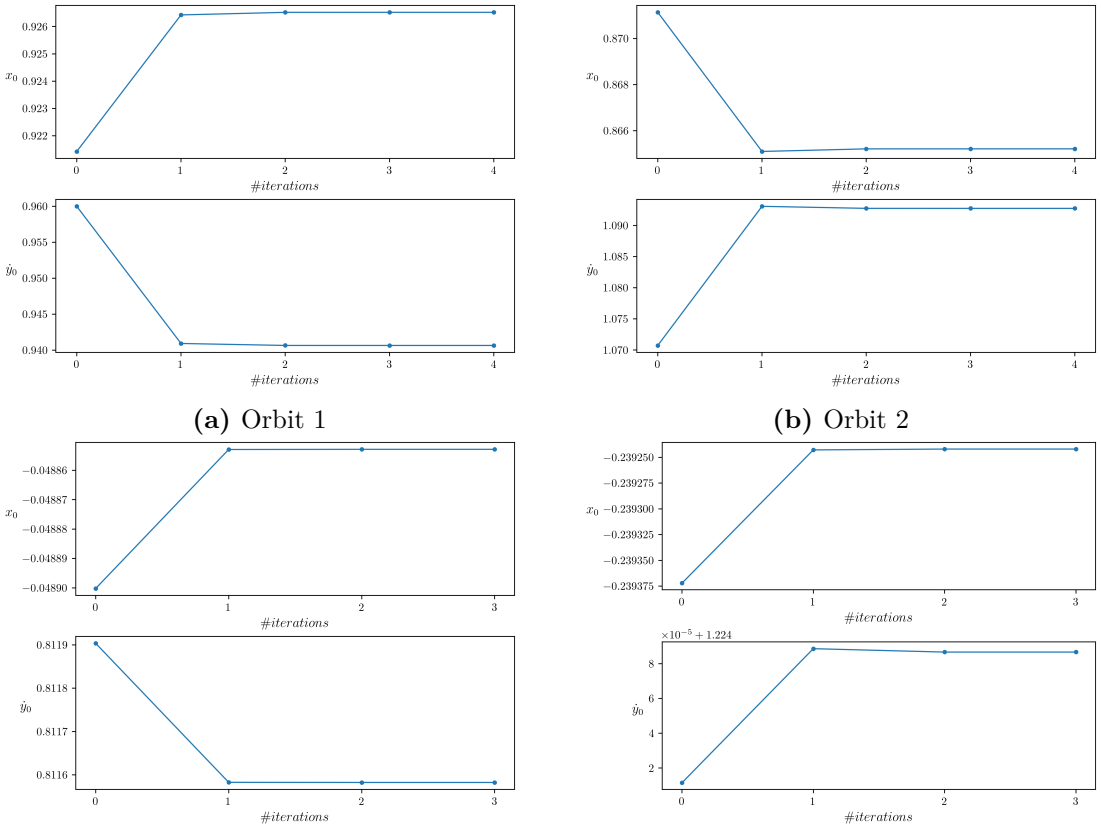
Orbit	**Convergence** [%]	**Mean Gen.**	**Best cost**	**Best individual** $[\mathbf{x_0}, \dot{\mathbf{y}}_\mathbf{0}]$
Orbit 1	100	10.89	1.092×10^{-3}	[0.92142307321223, 0.9599892620113]
Orbit 2	56	18.21	2.469×10^{-3}	[0.87114365964844, 1.07073686198129]
Orbit 3	96	15.43	6.216×10^{-4}	[-0.00489002006232, 0.81190347014181]
Orbit 4	90	4.76	9.008×10^{-4}	[-0.239372069454730, 1.22401154734194]

For the orbits built around the L_2, the GA exhibited performance similar to that resulted in Section 5.3.1. For orbits built around L_1 the GA benefited from the modifications made to the setting parameters. Otherwise, no good-fit solution would be found. The results obtained proved to be excellent initial guesses with which to begin the differential correction process. This can be seen by the results in Table 5.15.

Table 5.15: Individuals obtained after the differential correction and absolute differences between the values derived by the GA and the differential correction for each Halo orbit in the α Centauri AB system

Orbit	**Corrected** $[\mathbf{x_0}, \dot{\mathbf{y}}_\mathbf{0}]$	$\lvert\boldsymbol{\Delta x_0}\rvert$	$\lvert\boldsymbol{\Delta \dot{y}_0}\rvert$
Orbit 1	[0.9265211138800251, 0.9406654266665474]	5.098×10^{-3}	1.932×10^{-2}
Orbit 2	[0.8652085866621616, 1.0927332836174912]	5.935×10^{-3}	2.199×10^{-2}
Orbit 3	[-0.0488528635021841, 0.8115824891875094]	4.734×10^{-5}	3.209×10^{-4}
Orbit 4	[-0.239241981480691, 1.2240866726673463]	1.301×10^{-4}	7.512×10^{-5}

Figures 5.8 and 5.9 show that all sets of initial conditions are corrected quickly, requiring a maximum of four iterations.



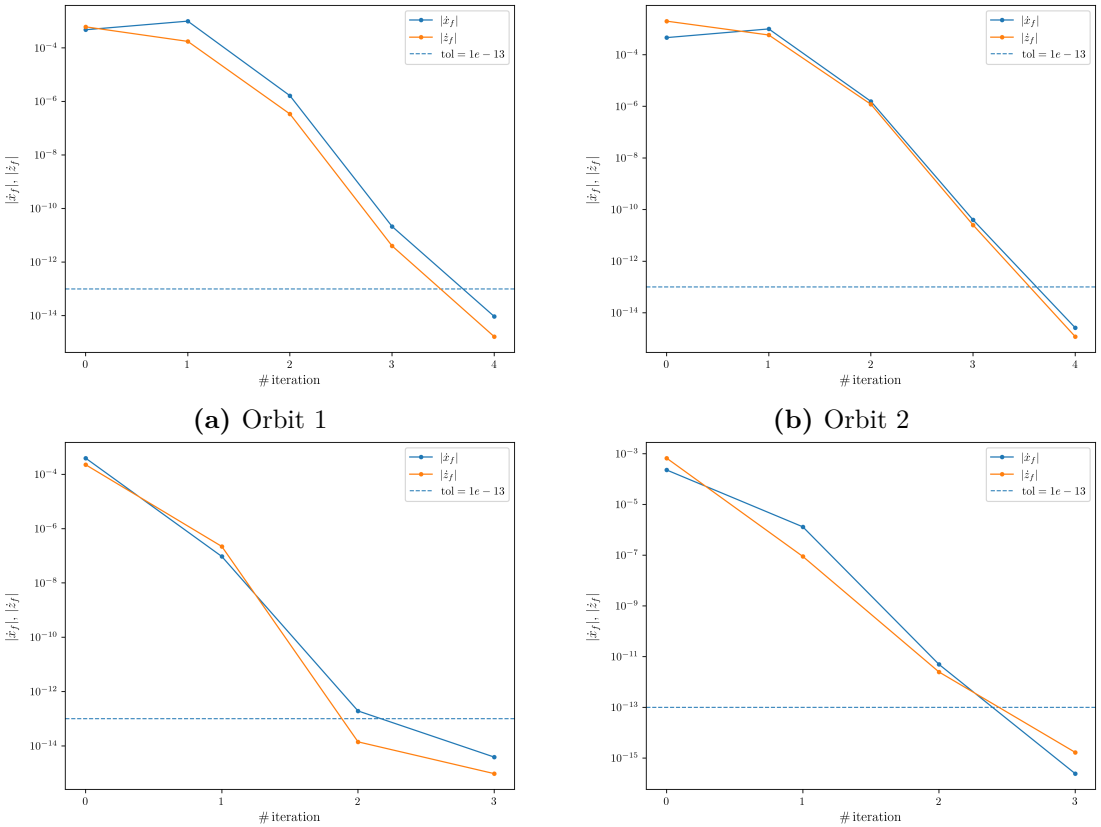
(a) Orbit 1

(b) Orbit 2

(c) Orbit 3

(d) Orbit 4

Figure 5.8: Initial conditions evolution throughout the differential correction process for each orbit in the α Centauri AB system



(a) Orbit 1

(b) Orbit 2

(c) Orbit 3

(d) Orbit 4

Figure 5.9: Final state velocities throughout the differential correction process for each orbit in the α Centauri AB system

Finally, the orbits whose periods and stability indexes are summarized in Table 5.16 are shown in Figures 5.10 and 5.11.

Table 5.16: Periods and stability indexes of the Halo orbits obtained using the GA in the α Centauri AB system

Orbit	**A_z [ND]**	**A_z [km]**	**Period [ND]**	**Period [years]**	**Stability index**
Orbit 1	0.1	237×10^6	6.676	60.28	97.69
Orbit 2	0.3	708×10^6	4.608	59.41	73.54
Orbit 3	0.6	1.02×10^9	2.613	33.68	137.99
Orbit 4	0.9	1.53×10^9	2.791	35.98	10.24

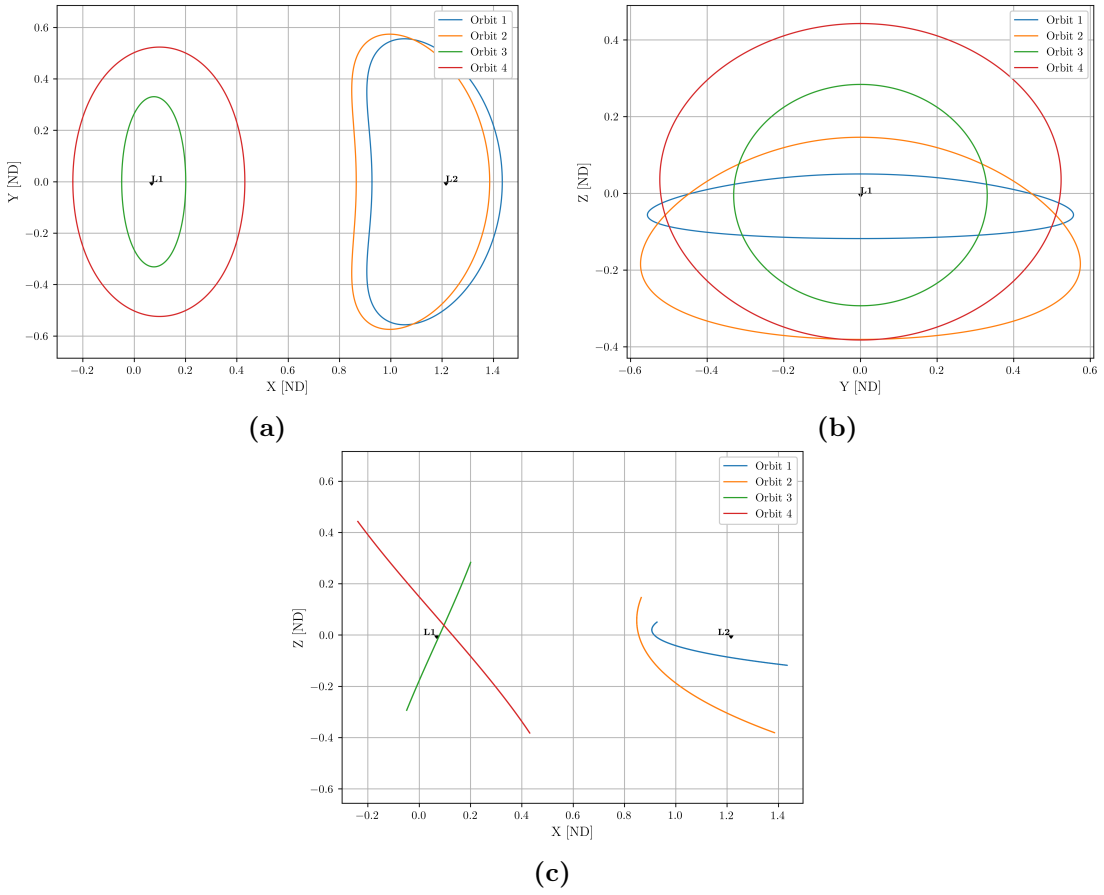


Figure 5.10: 2D view of the Halo orbits obtained using the GA in the α Centauri AB system: (a) x-y plane, (b) y-z plane, (c) x-z plane

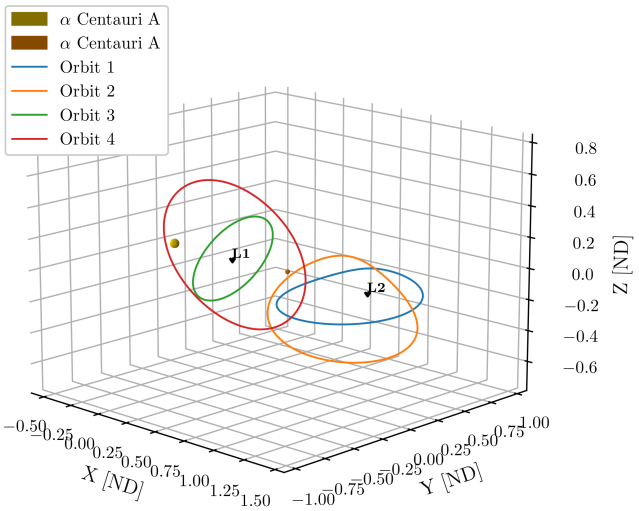


Figure 5.11: 3D view of the Halo orbits obtained using the GA in the α Centauri AB system

5.4 Celestial body coverage analysis

Once a periodic orbit has been chosen in a certain system, the tool enables the user to calculate the percentage of surface area covered by that orbit. This section will explain how the algorithm works and the mathematical criteria that ensure the achievement of the desired objectives.

The first step is to evaluate the position of the spacecraft, hence the position of the onboard observation instrument, with respect to the celestial body to be observed. This operation is straightforward because the position of the celestial body is known a-priori and the position of the spacecraft can be obtained by integrating the EoMs using the initial conditions of the chosen periodic orbit, both with respect to the synodic reference frame. Once these quantities are known, the spacecraft's position with respect to the center of the observed celestial body can be simply computed as follows

$$\mathbf{r}_s = \mathbf{X}_3 - \mathbf{R}_c \tag{5.28}$$

where:

- $\mathbf{X}_3$ indicate the position of the spacecraft in the synodic reference frame obtained considering the first three components of the state vector $\mathbf{X}$ at a certain time;
- $\mathbf{R}_c$ represents the position of the celestial body in the synodic reference frame;
- $\mathbf{r}_s$ is the relative position of the spacecraft with respect to the celestial body.

After that, the *field of view* (FOV) of the onboard instrument is retrieved to evaluate the coverage percentage. There are actually two ways to set the FOV value. First, the user can assign an arbitrary value, which will be used to perform the coverage analysis. Otherwise, if no value is provided, the FOV is automatically computed. Specifically, the theoretical maximum FOV value is computed for each position occupied by the spacecraft along the orbit, using the following equation

$$\alpha_{max} = \arcsin \frac{R}{r_s} \tag{5.29}$$

where R represents the value of the radius of the celestial body observed.

Figure 5.12 graphically represents the FOV and clarifies the origin of Equation (5.29).

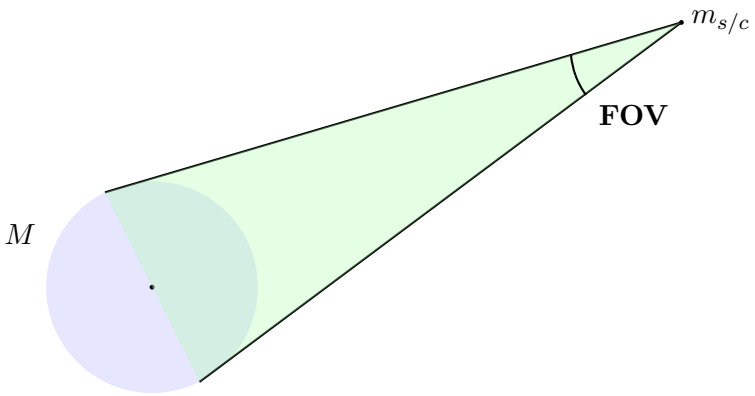


Figure 5.12: Field of view (FOV) of the sensor on board the spacecraft pointed towards the body M

To quantify the percentage of surface area observed, it is necessary to convert the instrument's FOV into a ground coverage angle, also known as footprint angle ψ. This angle determines the portion of surface that is effectively intercepted by the instrument's FOV as a function of the spacecraft's position and the radius of the celestial body, and it is calculated as

$$\psi = \arccos \frac{r_s(1 - \cos^2 \alpha) + \cos \alpha \sqrt{R^2 - r_s^2 \sin^2 \alpha}}{R} \tag{5.30}$$

By exploiting the footprint angle, the ground area observed is calculated as follows

$$A_{obs} = 2\pi R^2 (1 - \cos \psi) \tag{5.31}$$

Consequently, the percentage of the celestial body's surface area observed at a certain time is

$$A_{\text{cov}_i} = \frac{A_{obs}}{A_{tot}} = \frac{1 - \cos \psi}{2} \tag{5.32}$$

However, Equations (5.31) and (5.32) are not sufficient to evaluate the observed surface area fraction after moving along an orbit for one period. It is indeed necessary to discretize the surface of the celestial body and determine the number of samples that are intercepted at least once by the set FOV during one period.

5.4.1 Spherical Fibonacci lattice

To achieve the most uniform discretization of the spherical surface under observation, the *spherical Fibonacci lattice* is employed. This method distributes a sequence of points along the sphere in the form of a spiral trajectory using the golden angle. This angle, as will be shown later, is computed using the golden step, which is a defining feature of the Fibonacci lattice.
Firstly, the points are distributed uniformly along the z-direction. The z-coordinates

are derived as follows

$$z_k = 1 - 2\frac{k + 0.5}{M} \quad \text{with} \quad k = 0, \ldots, M - 1 \tag{5.33}$$

where M is the number of total points that has to be distributed along the sphere. The 0.5 offset introduced in Equation (5.33) avoids the points being placed at the poles of the sphere.

As the points are distributed along a unit radius sphere, the following equation holds

$$x^2 + y^2 + z^2 = 1 \tag{5.34}$$

which can be also written as

$$x^2 + y^2 = 1 - z^2 \tag{5.35}$$

Therefore, by substituing z_k in Equation (5.35) the distance r_k between the center of the sphere and the points lying along the sphere at its intersection with the plane z_k is obtained.

To complete the sphere sampling, the x- and y-coordinates of the points must also be computed. In order to place almost uniformly the points along the z_k plane, the aforementioned golden angle is computed. This angle is computed using the golden ratio φ, which is the limit of the ratio between two consecutive Fibonacci lattice numbers, n_i and n_{i+1}, when $i \to \infty$, namely

$$\varphi = \frac{1 + \sqrt{5}}{2} \tag{5.36}$$

The golden angle is then computed as

$$\theta_k = 2\pi\frac{k}{\varphi} \quad \text{with} \quad k = 0, \ldots, M - 1 \tag{5.37}$$

Once the golden angles have been computed for each value of k, the vectors containing the coordinates of each point, scaled using the distance r_k, can be derived

$$\mathbf{u}_k = [r_k \cdot \cos\theta_k, r_k \cdot \sin\theta_k, z_k] \quad \text{with} \quad k = 0, \ldots, M - 1 \tag{5.38}$$

Figure 5.13 shows how the points are distributed along the surface of a generic celestial body, for example the Earth, in accordance with the depicted procedure.

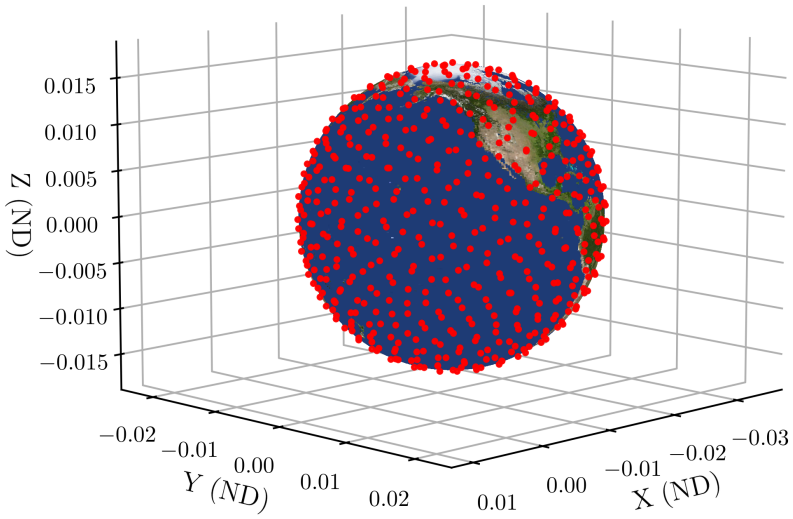


Figure 5.13: Earth's surface discretization using the spherical Fibonacci lattice with M=500 points

5.4.2 Visibility criterion

Once the surface of the celestial body under observation has been discretized, the number of points falling within the FOV must be evaluated. The criterion used takes into account the angle between the observation direction and the perpendicular direction to the surface at each point of the spherical Fibonacci lattice. The observation direction is assumed to be always nadir-pointed, i.e. the bisector of the FOV is perpendicular to the surface of the celestial body at every time step. The visibility criterion is based on the following equation

$$\mathbf{u}_k \cdot \mathbf{r}_i \geq \cos \psi_i \tag{5.39}$$

where

- $\mathbf{u}_k$ indicates the direction at the k-th point of the spherical Fibonacci lattice that is perpendicular to the spherical surface;
- $\mathbf{r}_i$ represents the nadir-pointed direction of the observation instrument at a certain time;
- ψ_i is the footprint angle computed at a certain time.

For a point on the spherical surface to be within the FOV, it is necessary that its angular distance from the observation direction to be less than the footprint angle. Consequently, the cosine of this angle must be greater than or equal to the cosine of the footprint angle, as defined by Equation (5.39).

To perform the coverage analysis considering the entire periodic orbit, this must be discretised as well. The visibility criterion is then applied to each position along the orbital trajectory, taking into account all the points distributed along the spherical

surface. Therefore, the equation implemented is in a matrix form as follows

$$\mathbf{D} = \mathbf{U}\mathbf{R}^T \tag{5.40}$$

Terms presented by Equation (5.40) are explained below

- $\mathbf{U} \in \mathbb{R}^{M\times 3}$ is the matrix that contains, along each row, the coordinates of the points belonging to the spherical Fibonacci lattice;
- $\mathbf{R} \in \mathbb{R}^{N\times 3}$ containts, for each row, the coordinates of the N positions sampled along the orbit;
- $\mathbf{D} \in \mathbb{R}^{M\times N}$ is the resultant matrix containing the scalar products between all observation directions and all the directions at the points on the surface of the celestial body.

If one of the values of U satisfies Equation (5.39) at least once, then it falls within the FOV. Therefore, the percentage of the observed surface using a certain periodic orbit is equal to

$$A_{\text{cov}} = \frac{\#\text{observed points}}{M} \tag{5.41}$$

According the following relation, the error committed during this operation decreases as the number of points used to discretise the surface increases

$$\text{err} = \frac{1}{2\sqrt{M}} \tag{5.42}$$

Algorithm 5 outlines the key steps necessary to perform correctly the coverage analysis.

Algorithm 5 Surface coverage analysis with spherical Fibonacci sampling

1: Set number of orbit samples: N
2: Set number of sphere samples: M
3: Integrate the EoMs using the initial state vector $\mathbf{X}_0$
4: Derive the spacecraft positions $\{\mathbf{X}_3\}_{i=1}^{N}$ sampling the resultant trajectory
5: **for** $i = 1$ to N **do**
6: Compute spacecraft relative position $\mathbf{r}_{s_i}$ and line-of-sight (LOS) direction using Eq. (5.28)
7: Compute maximum theoretical FOV: α_{max_i} using Eq. (5.29)
8: Compute footprint angle: ψ_i via Eq. (5.30)
9: **for** $k = 1$ to M **do**
10: Derive points coordinates to discretise the body surface using Eq. (5.38)
11: Apply visibility criterion described by Eq. (5.40)
12: Count the number of points observed
13: Compute the coverage percent: A_{cov} using Eq. (5.41)
14: Calculate the evaluation error: err via Eq. (5.42)

Algorithm 5 is applied to an example orbit, i.e. a Halo Northern orbit, built around L_1 in the Earth-Moon system discretized with $N = 2000$ points. The initial state vector of this orbit corresponds to

$$\mathbf{X}_0 = [0.83225881783611, 0, 0.127216985561728, 0, 0.241121072266256, 0] \tag{5.43}$$

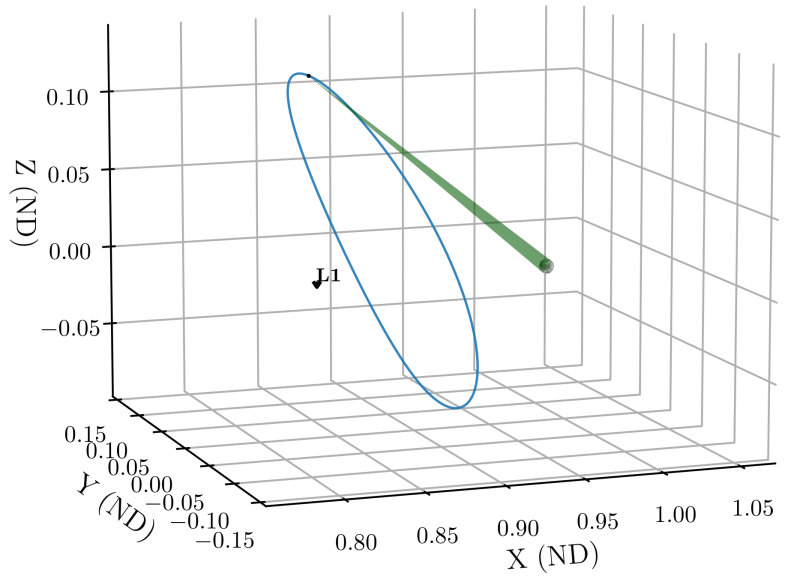


Figure 5.14: Observation geometry for a Halo Northern orbit built around L_1 in the Earth-Moon system

Figure 5.14 shows the FOV calculated at the position corresponding to $\mathbf{X}_0$. Figure 5.15 shows a zoomed-in view of the Moon. It displays the portion of surface observed with the FOV computed at the initial position, with the surface discretised using $M = 250,000$ points.

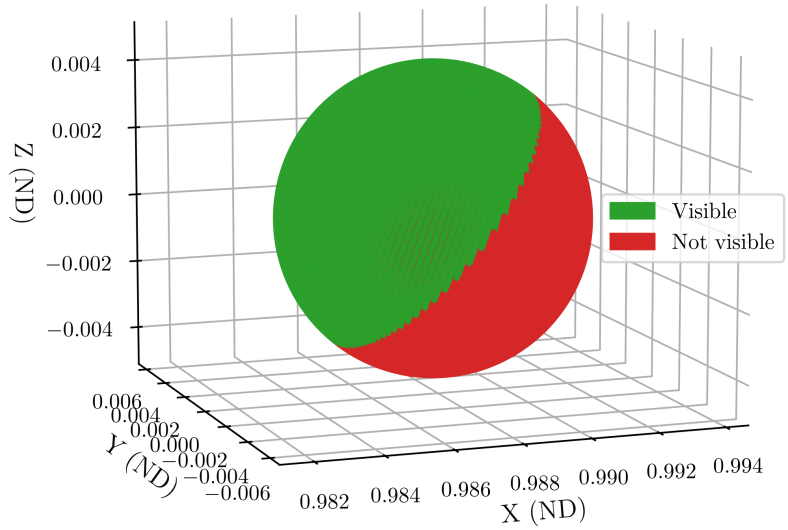


Figure 5.15: Instantaneous coverage of the Moon's surface evaluated at $\mathbf{X}_0$

Figure 5.16 consists of a cumulative coverage map of the Moon. The maximum number of observations per point corresponds to the number of points used to discretize the orbital trajectory. As it can be seen from the colorbar, regions of the lunar surface highlighted in warm colors underwent the highest number of observations. Conversely, regions denoted by cool colors are those that were less or not observed using the chosen orbit.

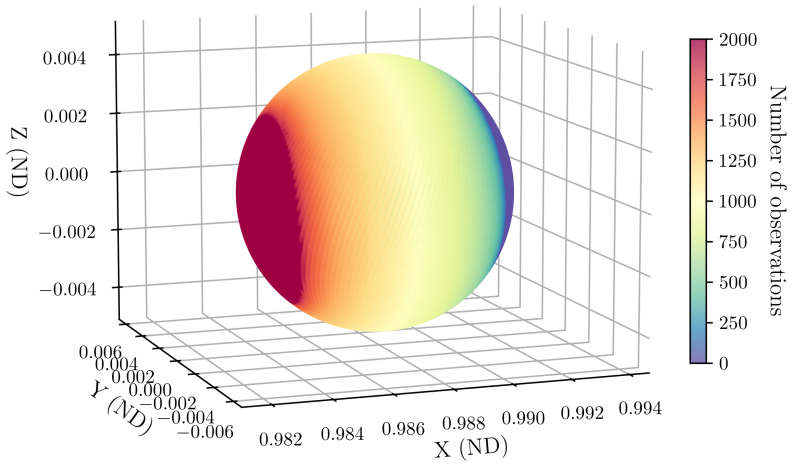


Figure 5.16: Moon's surface coverage map

The result emerging from Figure 5.16 is confirmed by the computed coverage percentage, which is $A_{\text{cov}} = 86.034\%$. Furthermore, the result is consistent with the geometry of Halo orbits. While these orbits do not ensure the total coverage of the surface to be observed, they guarantee good coverage of one primary body and almost uninterrupted communication with the other.

Finally, Figure 5.17 shows how the coverage percentage and the error committed in evaluating this latter evolve as function of M. In this case, the orbit used as example is the same as that used previously, $M \in [100, 1 \times 10^6]$ and $N = 2000$. In addition, the following parameters were calculated during the coverage analysis:

- The average of all the FOVs calculated in each position $FOV_{\text{avg}} = 1.7635$ř;
- The average of all the footprint angles obtained for each position along the orbit $\varphi_{\text{avg}} = 88.2365$ř
- The mean percentage of the surface observed $A_{\text{avg}} = 48.4614\%$, calculated by taking the average of the percentages observed at each position.

As can be seen, the coverage percentage approaches a constant value when M is at least equal to $100,000$ points. Contemporarly, the error tends to zero as M increseases, as mentioned before.

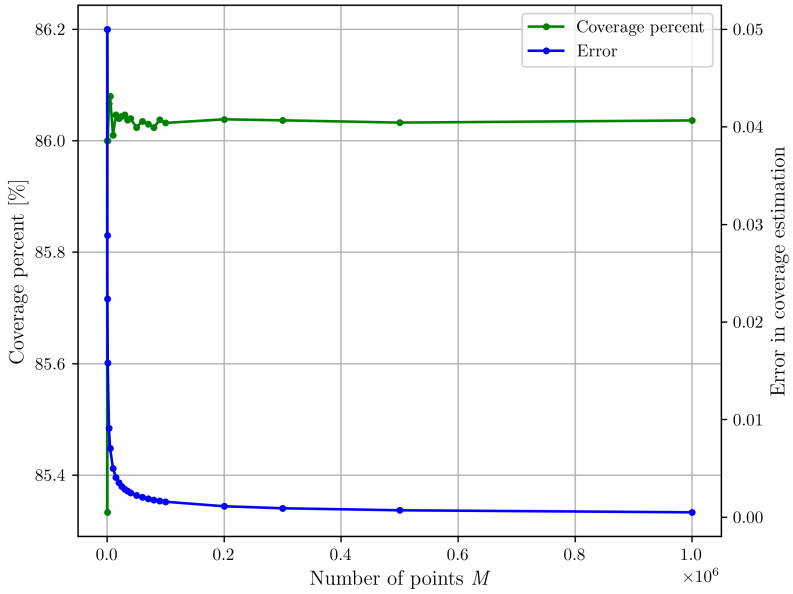


Figure 5.17: Coverage percentage and error in coverage estimation as functions of number of points

Orbit design tool, Future developments and Conclusions

6.1 GUI and Tool architecture

This section describes the GUI created in *Python* for the tool. The main objective is to realize a simple and intuitive interface that enables users to access all the algorithms presented thus far. For this reason, the GUI has been divided into more individual modules, that are analyzed separately.

6.1.1 Database

The first page of the GUI allows users to search the orbit database at its disposal. Most of these orbits are retrieved from the NASA JPL database via the appropriate API. The panel through which users can choose the parameters to filter the orbit database is represented in Figure 6.1.

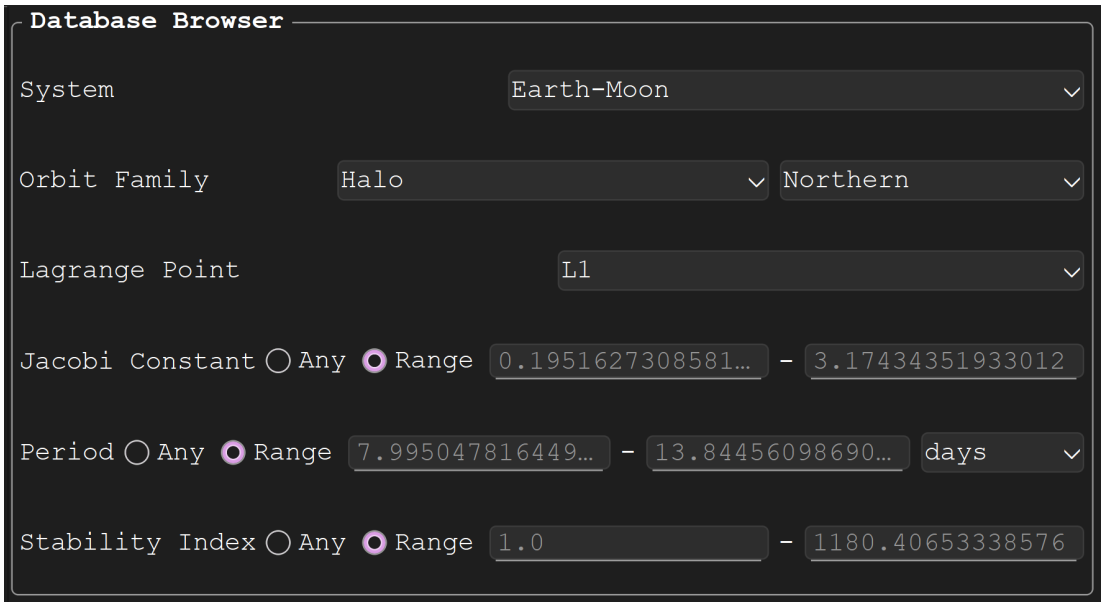


Figure 6.1: "Browse Database" menu of the GUI developed in Python for consulting the orbit database

The structure of the selection menu follows that of the same menu that is in the NASA JPL database. The menu was recreated with the same structure due to its excellent usability. Consequently, users can choose to receive as output data relative to one of the available orbit families in one of the systems of the database around the chosen

Lagrangian point. Moreover, users can further filter the database by entering arbitrary values for the Jacobi constant, orbital period, and stabiliy index, according to the limits suggested within each input line. These limits are determined by identifying the maximum and minimum values of these three parameters for the selected orbit family. This gives users an immediate idea of the ranges within which the three parameters lie. This is the only improvement made to the NASA JPL menu to access the database, which was already well-optimized.

After selecting the parameters, the database can be filtered to return the desired output data. The output data includes the physical characteristics of the chosen system, displayed in the appropriate section shown in Figure 6.2.

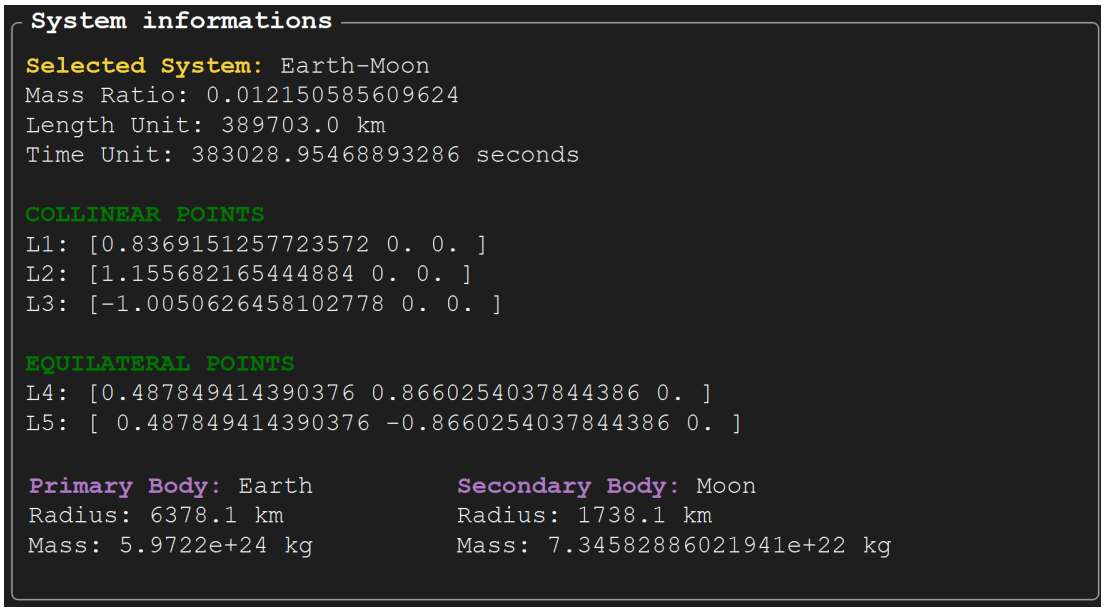
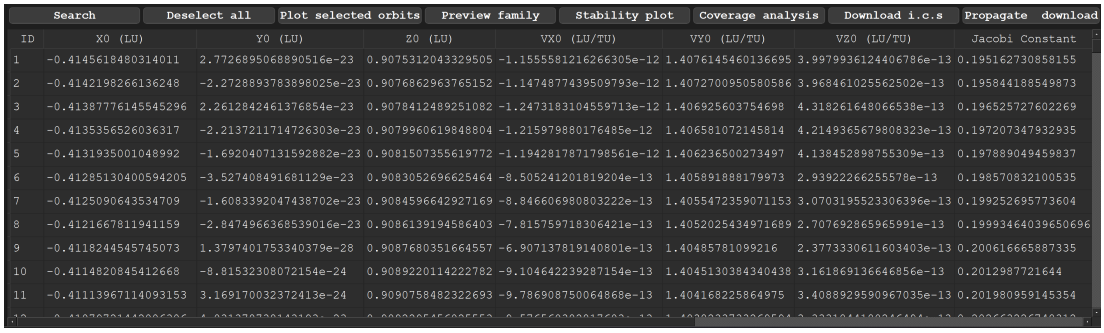


Figure 6.2: System information panel of the GUI for the selected CR3BP system, displaying its main physical and dynamical parameters

In order, the currently selected system is displayed, along with its fundamental parameters, such as the mass ratio, the length unit and the time unit. The latter two parameters are essential to convert all the data expressed in non-dimensional units into dimensional units, such as the coordinates of the Lagrangian points mentioned right below. Finally, radii and masses of the bodies forming the system are displayed. This information makes it easier to understand the output data presented in the following table. Due to the quantity of information contained in the table, it is divided into Figures 6.3a and 6.3b.



(a)

Period (TU)	Period (s)	Stability Index
3.123314392276159	1196170.0692683156	243.405726813375
3.123311261055463	1196168.8700702824	243.528729407559
3.123308118162728	1196167.66640208	243.65166172846097
3.123304963602633	1196166.4582655	243.774523672871
3.1233017973798534	1196165.2456623337	243.897315138911
3.123298619499044	1196164.0285943632	244.020036023816
3.1232954299648115	1196162.8070633537	244.142686225834
3.1232922287817795	1196161.5810710746	244.265265643009
3.1232890159545184	1196160.3506192777	244.387774172353
3.1232857914876204	1196159.1157097204	244.51021171269696
3.123282555385592	1196157.8763441294	244.632578162263

(b)

Figure 6.3: Output data table of the GUI listing the initial conditions and main dynamical parameters of all orbits in the selected family

In this case too, the output data are proposed in the same way as in the NASA JPL database. A table consisting of ten columns reports the following data:

- the first three columns present the x, y and z coordinates of the initial state in non-dimensional units;
- the successive three columns show the $\dot{x}$, $\dot{y}$ and $\dot{z}$ velocity components of the initial conditions vector, always expressed in non-dimensional units;
- the seventh column shows the Jacobi constant values, which indicate the energetic cost to access an orbit;
- two more columns show the orbital period expressed in non-dimensional units and in seconds;

- last column reports the stability index values for each orbit.

Figure 6.3a also illustrate all the action buttons of the first tool page. Each of these buttons allows users to perform a specific action. The function of each button is explained below, starting from the left:

- the `Search` button allows users to filter the database according to the options chosen in the previous menu shown in Figure 6.1;
- the `Deselect all` button simply deselects all the rows that users have selected in the output table;
- `Plot selected orbits` create a three-dimensional graphic using one or more selected orbits in the output table;
- `Preview family` automatically selects a small group of orbits belonging to the chosen family and creates a three-dimensional graphic that illustrates the trend of these orbits;
- the `Stability plot` button does not require any user selection in the output table and plots all stability indices and orbital periods in non-dimensional units as a function of the Jacobi constant for the chosen orbit family;
- `Coverage analysis` enables users to apply the algorithm depicted in Section 5.4 to obtain the previously discussed results for the selected orbit;
- `Download i.c.s` produces a *.csv* file containing all the selected rows in the output table or all the rows of the table if no rows are individually selected by the user;
- the `Propagante & download` button requires the selection of one table row, proceeds with the integration of the initial conditions of the selected orbit and create a *.csv* containing all the integration steps.

All of these elements make up the first page of the tool, which can be seen in its entirety in Figures 6.4.

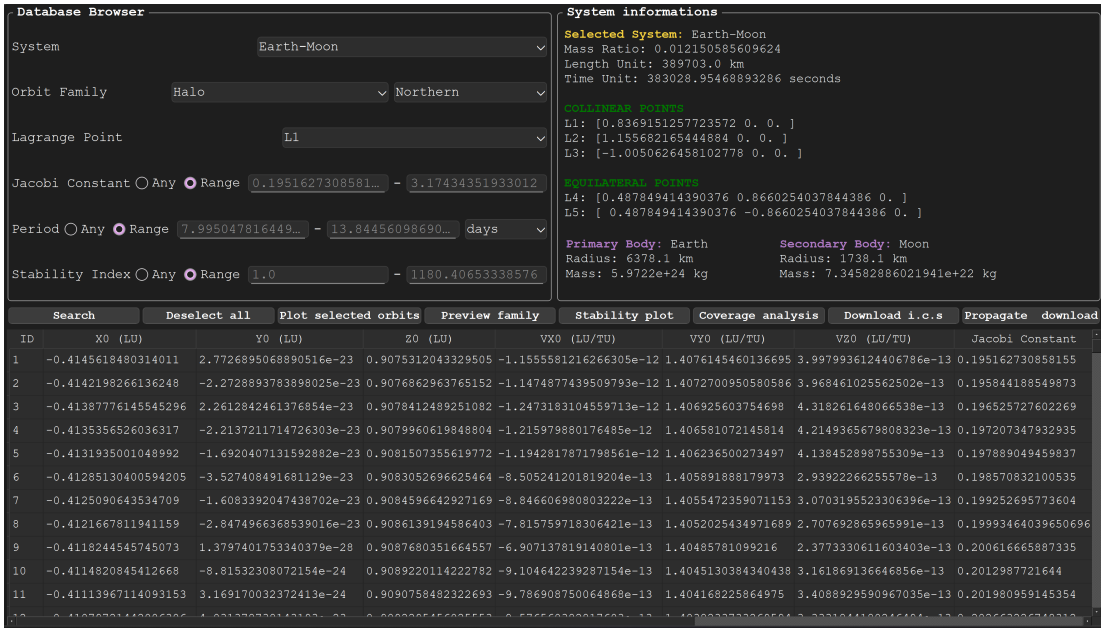


Figure 6.4: Main page of the GUI displaying system information, orbit selection controls, and the corresponding output data table for the selected orbit family

6.1.2 New system definition

One of the tool's innovations compared to the NASA JPL database is the panel that allows users to modify the database itself. For example, users can add a new three-body system, such as a planetary system not present in the NASA JPL database, a binary star system, or an asteroid system. The section of the GUI through which the user can proceed to the definition of a new system is depicted in Figure 6.5.

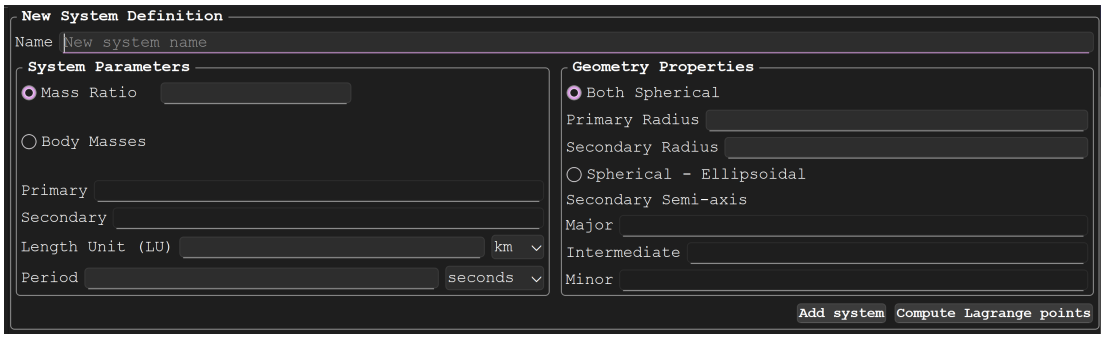


Figure 6.5: New system definition panel of the GUI used to insert custom three-body systems into the database

This interface is divided into two main sections. One section is used to specify the system's physical parameters. One of the most important parameters to define a new system is the mass ratio. Users have two options to do so: entering a mass ratio value directly or specifying the two body masses, in which case the mass ratio is computed. Next, users must define the length unit and the orbital period of the system. These

parameters can be provided using different units of measurement or using the non-dimensional units, by selecting them from the appropriate drop-down menu.

The second part of this interface is used to define the geometry of the two main bodies. For most of the systems to be analyzed using the spherical bodies hypothesis is sufficient. Thus, users can simply enter the values of the radii. However, some systems consists of a primary spherical body and a secondary body with a more irregular shape, such as Mars and its moons. Therefore, selecting `Spherical-Ellipsoidal` applies the ellipsoidal body hypothesis to the smaller body, allowing users to enter the three defining dimensions.

At the end of the panel, there are two action buttons. The `Add system` button, as it suggests, adds the system to the database, saving all the inserted parameters and computing the location of the Lagrangian points. If users do not want the system to be saved, using the `Compute Lagrange points` button computes the location of the Lagrangian points and provides the results in the form of a three-dimensional plot for visualization in the defined system.

6.1.3 Differential correction

The second new section of the tool allows users to provide arbitrary initial conditions and correct them using the differential correction algorithms. Thus, the tool is not limited to being a database, but becomes a computational tool that can be used to refine known non-periodic initial conditions. Figure 6.6 shows the section with which the users can exploit the differential correction algorithms.

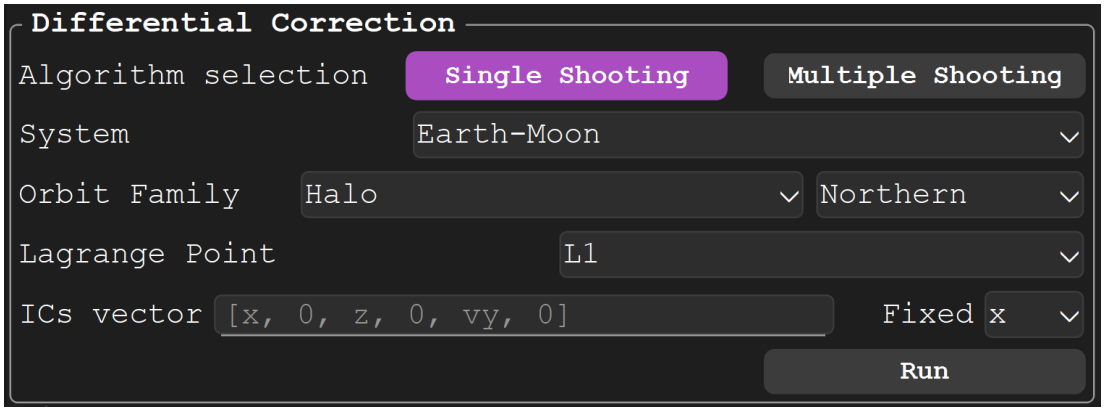


Figure 6.6: Differential correction setup panel of the GUI for selecting the algorithm and defining the target periodic orbit

Users can select either the single shooting or the multiple shooting algorithm to correct the provided set of initial conditions. In order for the algorithm to give a satisfying result, the characteristics of the desired periodic orbit must be declared using the appropriate drop-down menus. Finally, it is necessary to enter the initial conditions vector and select the variable to be fixed during the correction process. The graphic

representation of the periodic orbit and the graphs showing the corrections applied to the non-fixed components are then generated.

6.1.4 Initial conditions derivation

The last section of the GUI enables users to autonomously derive new initial conditions. This function is useful when the users define a new three-body system and want to derive new orbits, or when users want to obtain new orbits in a known system, but with different characteristics. Figure 6.7 shows the panel through which the users can access this function.

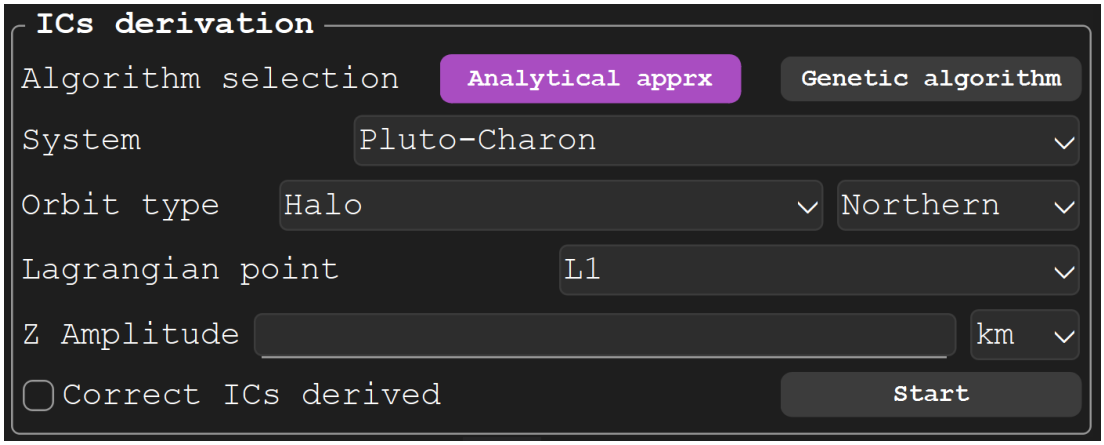


Figure 6.7: Initial conditions derivation panel of the GUI for generating Halo orbit seeds in the selected CR3BP system

Similarly to the section used to choose the differential correction algorithm to be used to correct new initial conditions, in this panel allows users to select either Richardson's analytical approximations or the genetic algorithm to compute new sets of initial conditions. These algorithms have already been discussed in detail in Chapter 5. Since the derivation of the initial conditions is limited to the Halo orbits, users must select the system, the orbit orientation and the Lagrangian point around which the orbit will be built. Finally, users must specify the z amplitude value in kilometers or in AU. At the end of the interface, there is a checkbox that, when selected, authorizes the automatic correction of the resultant initial state vector to produce a periodic vector of initial conditions. Otherwise, if this option is deselected, the result produced is the initial state vector computed by the selected algorithm without correction.

6.2 Conclusions

This thesis focuses on the development of a tool for deriving and analyzing the periodic orbits of the CR3BP, paying special attention to Halo orbits. One of the main objectives was to create a unique environment that integrates defining three-body systems,

deriving initial conditions and implementing differential corrections algorithms to render initial conditions periodic.

After clarifying the theoretical framework of the CR3BP, the main types of periodic orbits were adequately discussed. Among these, DROs and Halo orbits were found to be the most commonly used in real missions. For this reason, the successive initial conditions derivation focuses on the latter.

Next, the theoretical fundamentals and the characteristics of the two implemented differential correction algorithms were deeply analyzed. Both were tested with different families of periodic orbits and the numerical results were presented to clarify the suggested field of application. As demostraded, the single shooting algorithm is extremely effective, quick and reliable with simple vectors of initial conditions. In contrast, the multiple shooting algorithm requires more iterations and has a higher computational cost. However, it is better suited for correcting complex three-dimensional orbits with multiple non-null components in the initial state vector.

To generate new Halo orbits, it was demonstrated that just the Richardson's analytical approximations generally provide good initial estimations, but are not always sufficient to guarantee the convergence of the correction methods. Integrating a genetic algorithm overcomes these limitation by producing solutions with a high convergence percentage in the different analyzed systems, as well an accuracy of 10^{-4} for the non-null components, even in absence of a reference solution. This result, confirmed by the outcome of the subsequent differential correction, demonstrates the robustness and versatility of the proposed approach.

Finally, the coverage algorithm based on the spherical Fibonacci lattice enabled an effective and numerically stable estimate of the surface fraction observed along a periodic orbit. Simulations demonstrated that it is possible to reach coverages above 80% of the surface of the observed body for new Halo orbits considered, with rapid convergence of the estimation as the number of discretization points increases. These results, integrated within the GUI, confirm the tool's ability to support preliminary mission scenario analysis and lay the groundwork for future extensions that will provide more complete mission design functionalities.

6.3 Future work

Future implementations include the use of more advanced numerical methods than the Newton-Raphson method, to derive even more accurately the Lagrangian points, especially for systems with a mass ratio close to the limit value of 0.5.

Another important implementation is to increase the number of systems in the database, to include all the planetary systems in the solar system. Then, the genetic algorithm could be used to derive new Halo orbits in each system. Moreover, the derivation of initial conditions should not be limited to only few specimens. Instead,

hundreds of Halo orbits should be generated for each Lagrangian point, in order to decisively expand the orbit database.

Following this path, it would be of uselful for users to have algorithms that can derive initial conditions for not only Halo orbits, but also for the other orbit families described in Section 3. Thus, the database could be expanded to provide users with more options and become more complete.

Currently, the tool mainly focuses on the storage and the derivation of initial conditions for periodic orbits. In ordet to make a step toward creating a tool that can perform a preliminary mission anlysis, functions to compute transfer trajectories between different orbits and descent trajectories to the surface of a celestial body must be implemented. With these functions, it would be possible to estimate the necessary ΔV to perform a certain mission.

As far as the GUI is concerned, it is possible to introduce a page where users can adjust the parameters of the algorithms used or choose which results and graphs to receive as output. Finally, the fluidity of the three-dimensional plots could be significantly improved by using a better optimized library than `matplotlib`.

Bibliography

[1] NASA, "What is CAPSTONE?." https://www.nasa.gov/smallspacecraft/capstone/.

[2] NASA, "NASA's DART Mission Hits Asteroid in First-Ever Planetary Defense Test." https://www.nasa.gov/news-release/nasas-dart-mission-hits-asteroid-in-first-ever-planetary-defense-test/, Sept. 2022.

[3] NASA, "OSIRIS-REx: Asteroid sample return mission — press kit." https://www.nasa.gov/wp-content/uploads/2016/06/osiris-rex_press_kit_0.pdf, Aug. 2016.

[4] JAXA/ISAS, "HAYABUSA: Research in the asteroid 'sample & return' explorer." https://www.isas.jaxa.jp/e/special/2003/kawaguchi/index.shtml, May 2003.

[5] R. W. Farquhar, D. Muhonen, and D. Richardson, "Mission design for a halo orbiter of the earth," *Journal of Spacecraft and Rockets*, vol. 14, no. 3, pp. 170–177, 1977.

[6] R. W. Farquhar, "The Flight of ISEE-3/ICE: Origins, Mission History, and a Legacy," *The Journal of the Astronautical Sciences*, vol. 49, no. 1, pp. 23–73, 2001.

[7] F. E. Mcgarry, "Goddard mission analysis system (GMAS)," *NASA Tech Briefs*, vol. 14, no. 12, 1990.

[8] J. Carrico, C. Schiff, L. Roszman, H. Hooper, D. Folta, and K. Richon, "An interactive tool for design and support of lunar, gravity assist, and libration point trajectories," in *AIAA/AHS/ASEE Aerospace Design Conference*, no. AIAA-93-1126 in AIAA Paper, (Irvine, CA), 1993.

[9] J. Carrico, H. L. Hooper, L. Roszman, and C. Gramling, "Rapid design of gravity assist trajectories," in *Proceedings of the ESA Symposium on Spacecraft Flight Dynamics*, ESA SP-326, (Darmstadt, Germany), 1991.

[10] L. Plice, A. Dono Perez, L. Policastri, J. Carrico, and M. Loucks, "ARCUS mission design: Stable lunar-resonant high earth orbit for x-ray astronomy," in *AAS/AIAA Astrodynamics Specialist Conference*, no. AAS 18-271 in Advances in the Astronautical Sciences, (Snowbird, UT, USA), Aug. 2018.

[11] M. A. Woodard, D. C. Folta, and D. W. Woodfork, "ARTEMIS: The first mission to the lunar libration orbits," in *Proceedings of the 21st International Symposium on Space Flight Dynamics (ISSFD)*, (Toulouse, France), Sept. 2009.

[12] D. J. Dichmann, C. M. Alberding, and W. H. Yu, "Stationkeeping monte carlo simulation for the james webb space telescope," in *Proceedings of the 24th International Symposium on Space Flight Dynamics (ISSFD)*, (Laurel, Maryland, USA), May 2014.

[13] NASA Jet Propulsion Laboratory (JPL), "Three-body periodic orbits." https://ssd.jpl.nasa.gov/tools/periodic_orbits.html.

[14] I. Newton, *Philosophiæ Naturalis Principia Mathematica*. Royal Society, 1687.

[15] C. G. J. Jacobi, "Sur le movement d'un point et sur un cas particulier du problème des trois corps," *Comptes Rendus de l'Académie des Sciences de Paris*, vol. 3, pp. 59–61, 1836.

[16] H. Poincarè, *Les Méthodes Nouvelles de la Méchanique Céleste, Tome 3*. Gauthier-Villars, 1899.

[17] C. Moore, "Braids in classical dynamics," *Physical Review Letters*, vol. 70, no. 24, pp. 3675–3679, 1993.

[18] A. Chenciner and R. Montgomery, "A remarkable periodic solution of the three-body problem in the case of equal masses," *Annals of Mathematics*, vol. 152, no. 3, pp. 881–901, 2000.

[19] L. Euler, "De moto rectilineo trium corporum se mutuo attrahentium," *Novi Commentarii academiae scientiarum Petropolitanae*, vol. 11, p. 144–151, 1767.

[20] J.-L. Lagrange, "Essai sur le problème des trois corps," *Euvre*, vol. 6, pp. 229–331, 1772.

[21] J. Raphson, *Analysis aequationum universalis*. Th. Braddyll, 1690.

[22] V. Szebehely, *Theory of Orbits: The Restricted Problem of Three Bodies*, vol. 41 of *Mathematics in Science and Engineering*. Academic Press, 1967.

[23] A. M. Lyapunov, "Problème général de la stabilité du mouvement," *Annales de la Faculté des sciences de Toulouse: Mathématiques*, vol. 9, pp. 203–474, 1907.

[24] R. Whitley and R. Martinez, "Options for staging orbits in cislunar space," in *2016 IEEE Aerospace Conference*, pp. 1–9, IEEE, 2016.

[25] G. L. Condon and J. Williams, "Asteroid redirect crewed mission nominal design and performance," in *SpaceOps 2014 Conference*, American Institute of Aeronautics and Astronautics (AIAA), 2014.

[26] K. Mars, "Orion will go the distance in retrograde orbit during artemis i."

[27] D. D. Mazanek, R. G. Merrill, J. R. Brophy, and R. P. Mueller, "Asteroid redirect mission concept: A bold approach for utilizing space resources," *Acta Astronautica*, vol. 117, pp. 163–171, 2015.

[28] A. Bemporad, S. Fineschi, M. Focardi, F. Landini, M. Romoli, and M. Pancrazzi, "Hemise (helio-magnetism investigation from the sun to earth): a twin spacecraft mission at the sun–earth lagrangian points l4 and l5," in *40th COSPAR Scientific Assembly*, 2014.

[29] R. W. Farquhar and A. A. Kamel, "Quasi-periodic orbits about the translunar libration point," *Celestial Mechanics*, vol. 7, no. 4, pp. 458–473, 1973.

[30] H. Poincaré, "Sur le problème des trois corps et les équations de la dynamique," *Acta Mathematica*, vol. 13, pp. 1–270, 1890.

[31] W.-S. Koon, M. W. Lo, J. E. Marsden, and S. D. Ross, "The genesis trajectory and heteroclinic connections," in *AAS/AIAA Astrodynamics Specialist Conference*, (Girdwood, Alaska), 1999. AAS 99-451.

[32] J. Petersen, "L2 station keeping maneuver strategy for the james webb space telescope," in *AAS/AIAA Astrodynamics Specialist Conference*, 2019. AAS 19-8062; Available via NASA NTRS.

[33] N. L. Parrish, E. Kayser, S. Udupa, J. S. Parker, B. W. Cheetham, and D. C. Davis, "Ballistic lunar transfers to near rectilinear halo orbit: Operational considerations," in *AIAA SciTech 2020 Forum*, 2020.

[34] W. Koon, M. Lo, J. Marsden, and S. Ross, *Dynamical Systems, the Three-Body Problem and Space Mission Design*. Interdisciplinary Applied Mathematics, Springer New York, 2017.

[35] R. Broucke, *Periodic Orbits in the Elliptic Restricted Three-body Problem*. Jet Propulsion Laboratory, California Institute of Technology, 1969.

[36] M. Hénon, "Numerical exploration of the restricted problem. v – hill's case: periodic orbits and their stability," *Astronomy and Astrophysics*, vol. 28, pp. 415–426, 1973.

[37] R. Farquhar, *The Utilization of Halo Orbits in Advanced Lunar Operations*. National Aeronautics and Space Administration, 1971.

[38] K. Howell, "Three-dimensional, periodic, 'halo' orbits," *Celestial Mechanics*, vol. 32, pp. 53–71, 1984.

[39] M. Osborne, "Shooting methods for boundary value problems," *Journal of Mathematical Analysis and Applications*, vol. 27, no. 2, pp. 417–433, 1969.

[40] R. Bulirsch, "Die mehrzielmethode zur numerischen lösung von nichtlinearen randwertproblemen und aufgaben der optimalen steuerung," tech. rep., Carl-Cranz-Gesellschaft e.V., DLR Oberpfaffenhofen, 1971.

[41] P. Deuflhard, "A modified newton method for the solution of ill-conditioned systems of nonlinear equations with application to multiple shooting .," *Numerische Mathematik*, vol. 22, pp. 289–316, 1974.

[42] A. G. Van Anderlecht, "Tadpole orbits in the l4/l5 region: Construction and links to other families of periodic orbits," m.s. thesis, Purdue University, West Lafayette, IN, USA, 2016.

[43] D. Richardson, "Analytical construction of periodic orbits about the collinear points," *Celestial Mechanics*, vol. 22, pp. 241–253, 1980.

[44] R. Haupt and S. Haupt, *Practical Genetic Algorithms*. Wiley InterScience electronic collection, Wiley, 2004.

Appendix A

Richardson constants and coefficients

The linearized frequency λ is derived solving the following equation

$$\lambda^4 + (c_2 - 2)\lambda^2 - (c_2 - 1)(1 + c_2) = 0 \tag{A.1}$$

Constant k

$$k = \frac{1}{2\lambda}(\lambda^2 + 1 + 2c_2) = \frac{2\lambda}{\lambda^2 + 1 - c_2} \tag{A.2}$$

and coefficients

$$\begin{aligned} d_1 &= \frac{3\lambda^2}{k}\left[k(6\lambda^2 - 1) - 2\lambda\right] \\ d_2 &= \frac{8\lambda^2}{k}\left[k(11\lambda^2 - 1) - 2\lambda\right] \end{aligned} \tag{A.3}$$

are used to calculate the following coefficients

$$a_{21} = \frac{3c_3(k^2 - 2)}{4(1 + 2c_2)}$$

$$a_{22} = \frac{3c_3}{4(1 + 2c_2)}$$

$$a_{23} = -\frac{3c_3\lambda}{4kd_1}\left[3k^3\lambda - 6k(k - \lambda) + 4\right]$$

$$a_{24} = -\frac{3c_3\lambda}{4kd_1}(2 + 3k\lambda)$$

$$a_{31} = -\frac{9\lambda}{4d_2}\left[4c_3(ka_{23} - b_{21} + kc_4(4 + k^2)\right] + \left(\frac{9\lambda^2 + 1 - c_2}{2d_2}\right)\left[3c_3(2a_{23} - kb_{21}) + c_4(2 + 3k^2)\right]$$

$$a_{32} = -\frac{1}{d_2}\left\{\frac{9\lambda}{4}\left[4c_3(ka_{24} - b_{22}) + kc_4\right] + \frac{3}{2}(9\lambda^2 + 1 - c_2)\left[c_3(kb_{22} + d_{21} - 2a_{23}) - c_4\right]\right\}$$

$$b_{21} = -\frac{3c_3\lambda}{2d_1}(3k\lambda - 4)$$

$$b_{22} = \frac{3c_3\lambda}{d_1}$$

$$b_{31} = \frac{3}{8d_2}\left\{8\lambda\left[3c_3(kb_{21} - 2a_{23}) - c_4(2 + 3k^2)\right] + (9\lambda^2 + 1 + 2c_2)\left[4c_3(ka_{24} - b_{21}) + kc_4(4 + k^2)\right]\right\}$$

$$b_{32} = \frac{1}{d_2}\left\{9\lambda\left[c_3(kb_{22} + d_{21} - 2a_{24}) - c_4\right] + \frac{3}{8}(9\lambda^2 + 1 + 2c_2)\left[4c_3(ka_{24} - b_{22}) + kc_4\right]\right\}$$

$$d_{21} = -\frac{c_3}{2\lambda^2}$$

$$d_{31} = \frac{3}{64\lambda^2}(4c_3a_{24} + c_4)$$

$$d_{32} = \frac{3}{64\lambda^2}\left[4c_3(a_{23} - d_{21}) + c_4(4 + k^2)\right] \tag{A.4}$$

These coefficients are in turn used to calculate the constants used for frequency correction

$$\begin{aligned} s_1 &= \frac{1}{2\lambda\left[\lambda(1+k^2) - 2k\right]}\left\{\frac{3}{2}c_3\left[2a_{21}(k^2-2) - a_{23}(k^2+2) - 2kb_{21}\right] - \frac{3}{8}c_4(3k^4 - 8k^2 + 8)\right\} \\ s_2 &= \frac{1}{2\lambda\left[\lambda(1+k^2) - 2k\right]}\left\{\frac{3}{2}c_3\left[2a_{22}(k^2-2) + a_{24}(k^2+2) + 2kb_{22} + 5d_{21}\right] + \frac{3}{8}c_4(12 - k^2)\right\} \end{aligned} \tag{A.5}$$

These constants are also used in calculating the coefficients present in the amplitude-constraint relationship

$$\begin{aligned} l_1 &= a_1 + 2\lambda^2 s_1 \\ l_2 &= a_2 + 2\lambda^2 s_2 \end{aligned} \tag{A.6}$$

where

$$\begin{aligned} a_1 &= -\frac{3}{2}c_3(2a_{21} + a_{23} + 5d_{21}) - \frac{3}{8}(12 - k^2) \\ a_2 &= \frac{3}{2}(a_{24} - 2a_{22}) + \frac{9}{8}c_4 \end{aligned} \tag{A.7}$$

Appendix B

Halo orbit examples

Table B.1 reports, for each chosen system that is not part of the NASA JPL CR3BP database, the Lagrangian point around which the new orbit is built and the amplitude A_z of the desired orbit. The non-null components obtained after the differential correction and used to form the initial condition vector of each orbit are listed on the right side of the table.

System	**Lagrangian Point**	A_z	x_0	z_0	$\dot{y}_0$
Sun-Mercury	L_1	$100,000$	0.9958949694342745	0.001993350982731309	0.004765946126279004
Sun-Mercury	L_2	$70,000$	1.0028385518212544	0.0010732563263877	0.0048434215813002
Jupiter-Ganymede	L_1	$20,000$	0.9698633282185479	0.0221096801693863	0.0407853354070788
Jupiter-Ganymede	L_3	$600,000$	-1.448144096834162	1.0298940514131996	1.0939720719393502
Pluto-Charon	L_1	4000	0.559309869585235	0.2215250384692508	0.5047035682835929
Pluto-Charon	L_3	9000	-1.4722733017437595	0.7546912076888178	1.0667617850439732
α Centauri A-α Centauri B	L_1	5×10^7	0.0441641493924296	0.0142339944067844	0.2960231260380268
α Centauri A-α Centauri B	L_3	1×10^9	-1.4063340137055116	0.4026732826451843	0.7972232950301624

Table B.1: Initial conditions for Halo orbit generation in the CR3BP. Each row lists the system, target Lagrange point, prescribed vertical amplitude A_z, and the non-null components of the initial state vector

The results listed in Table B.1 are shown graphically below. Each orbit is depicted both in the three bidimensional planes of the synodic reference frame and three-dimensionale space.

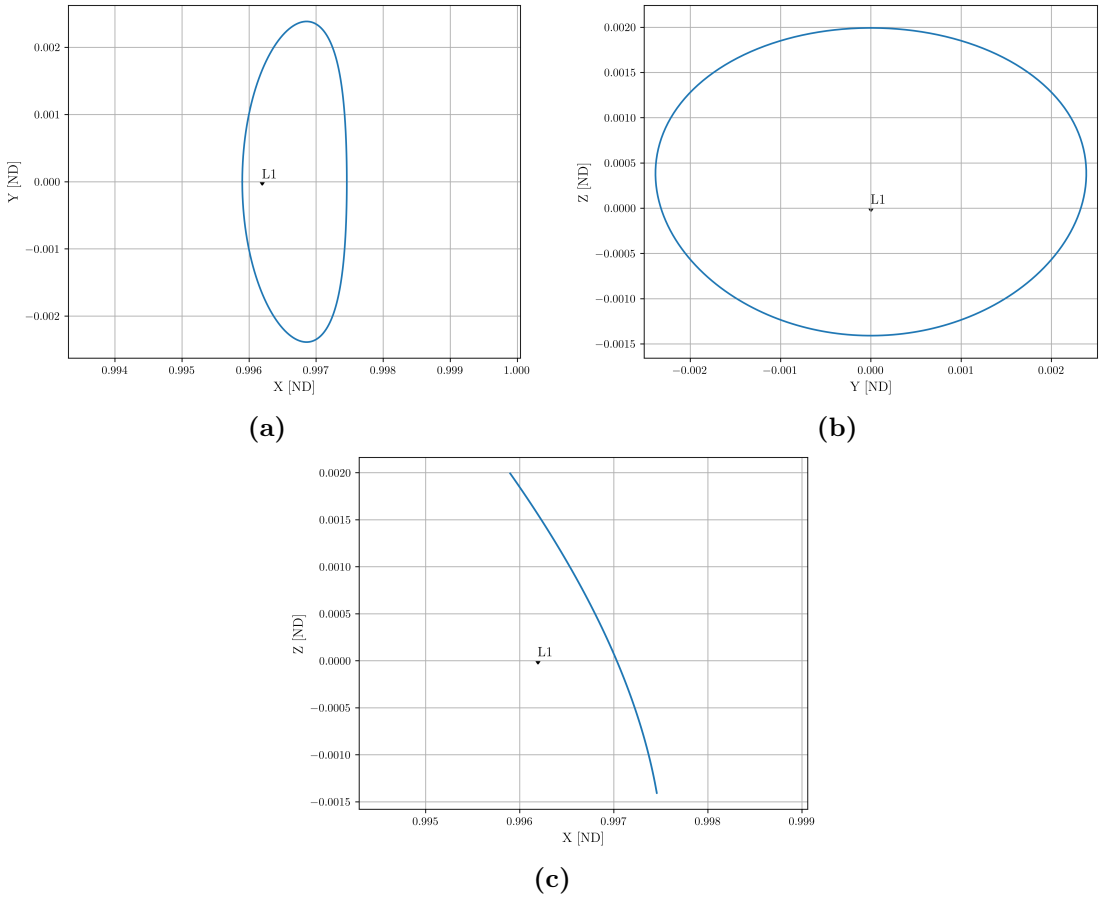


Figure B.1: 2D view of the computed Northern Halo orbit around L_1 in the Sun-Mercury system with $A_z = 100,000$ km: **(a)** x-y plane, **(b)** y-z plane, **(c)** x-z plane

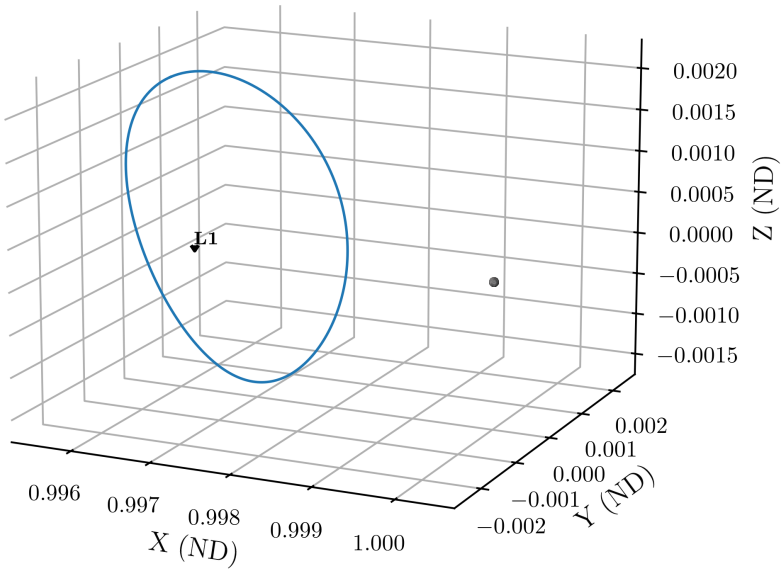


Figure B.2: 3D view of the computed Northern Halo orbit around L_1 in the Sun-Mercury system with $A_z = 100,000$ km

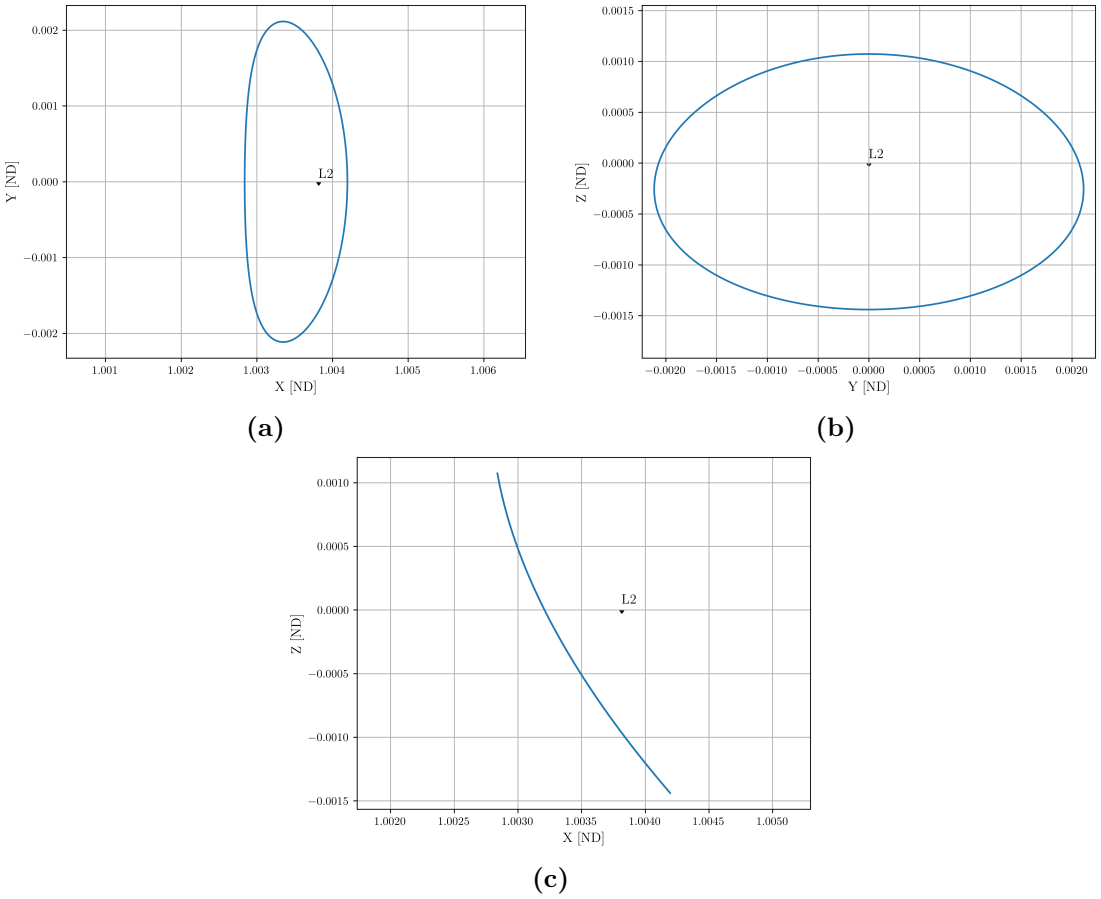


Figure B.3: 2D view of the computed Northern Halo orbit around L_2 in the Sun-Mercury system with $A_z = 70,000$ km: **(a)** x-y plane, **(b)** y-z plane, **(c)** x-z plane

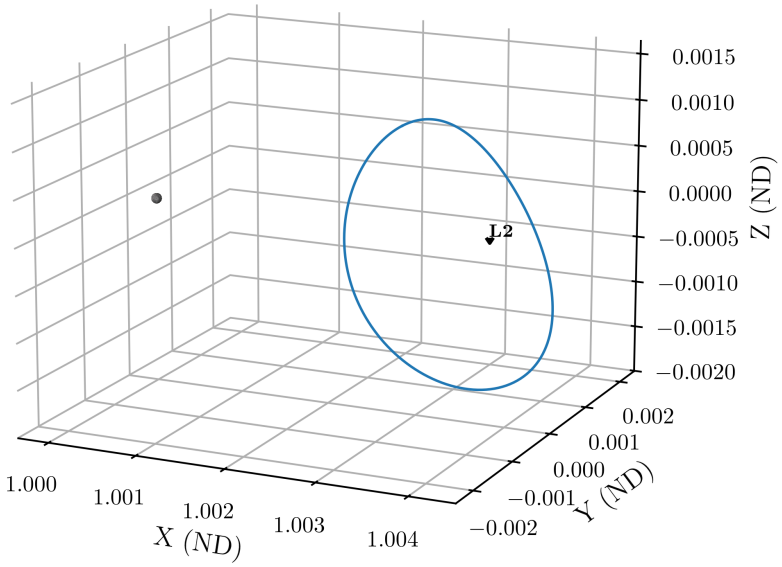


Figure B.4: 3D view of the computed Northern Halo orbit around L_2 in the Sun-Mercury system with $A_z = 70,000$ km

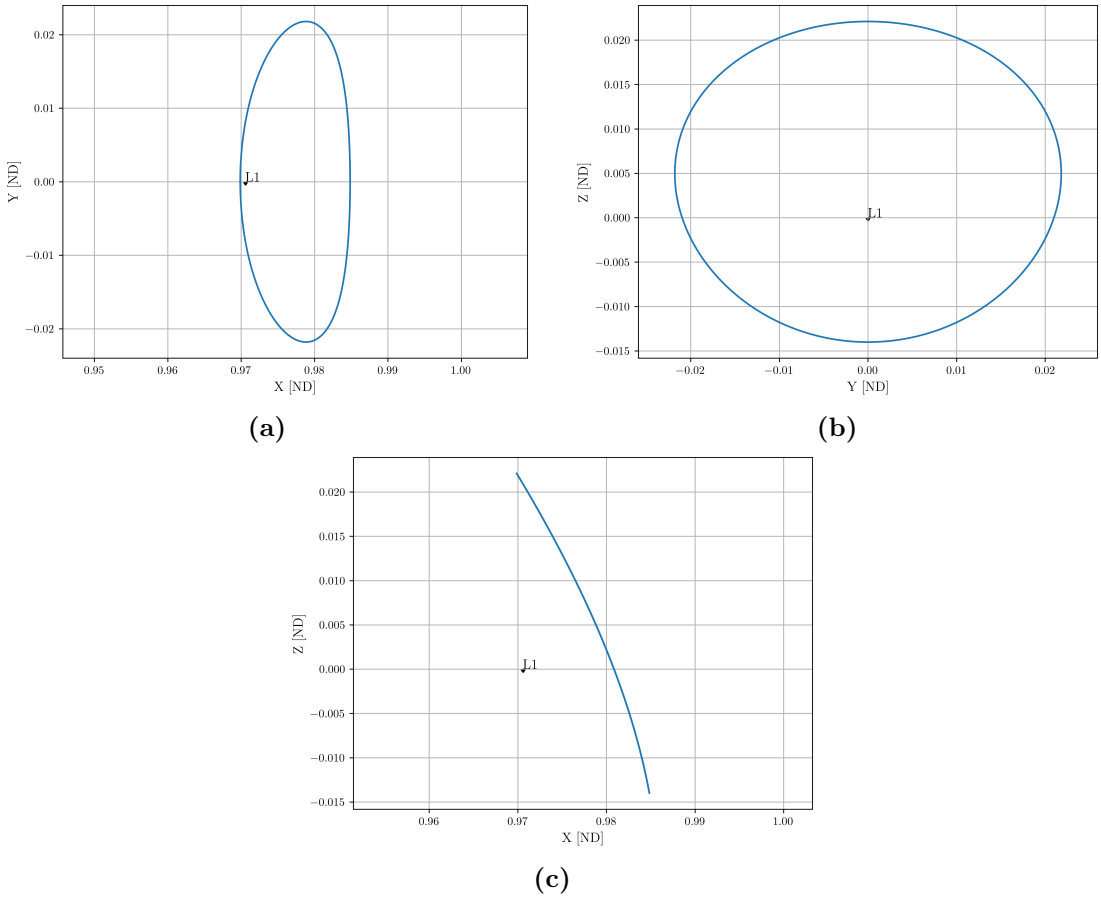


Figure B.5: 2D view of the computed Northern Halo orbit around L_1 in the Jupiter-Ganymede system with $A_z = 20,000$ km: **(a)** x-y plane, **(b)** y-z plane, **(c)** x-z plane

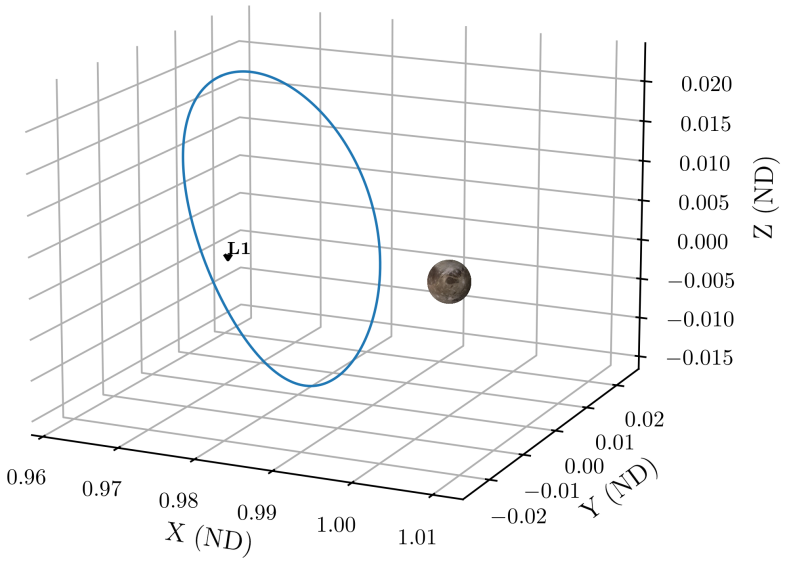


Figure B.6: 3D view of the computed Northern Halo orbit around L_1 in the Jupiter-Ganymede system with $A_z = 20,000$ km

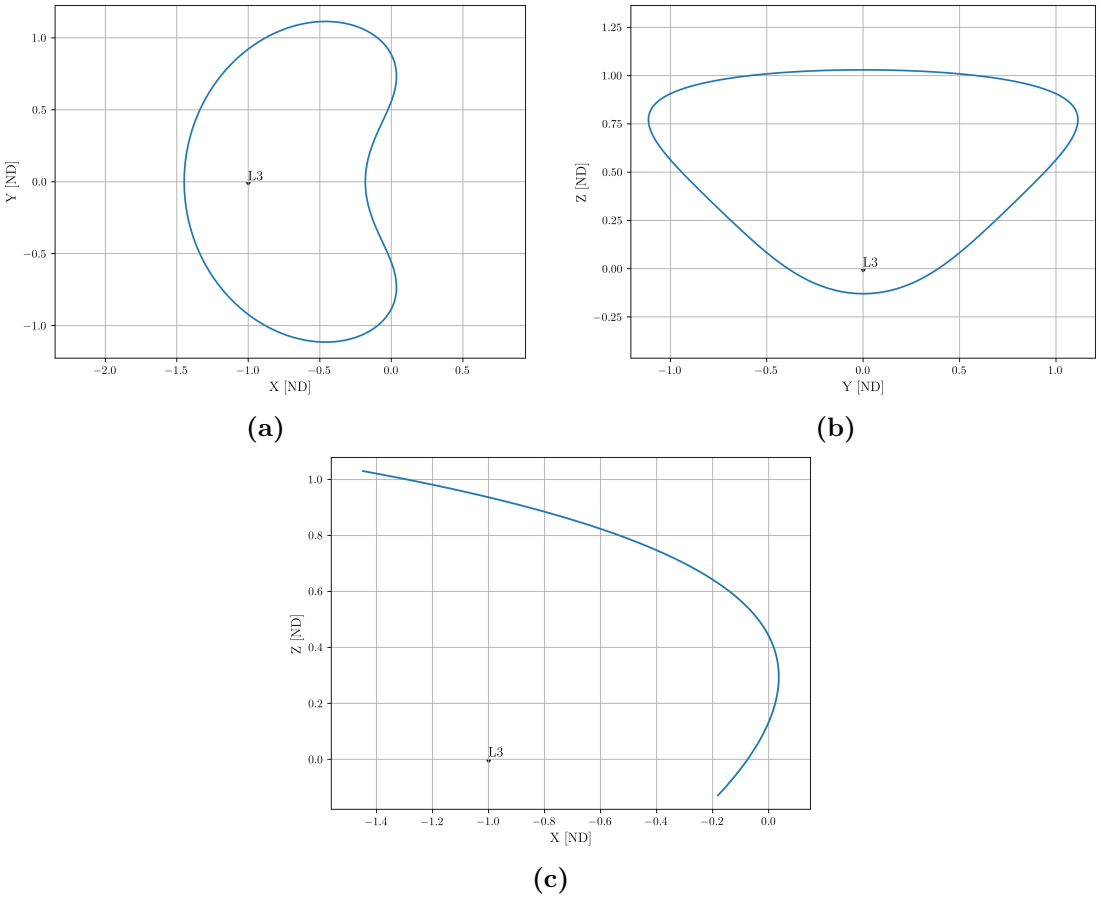


Figure B.7: 2D view of the computed Northern Halo orbit around L_3 in the Jupiter-Ganymede system with $A_z = 600,000$ km: **(a)** x-y plane, **(b)** y-z plane, **(c)** x-z plane

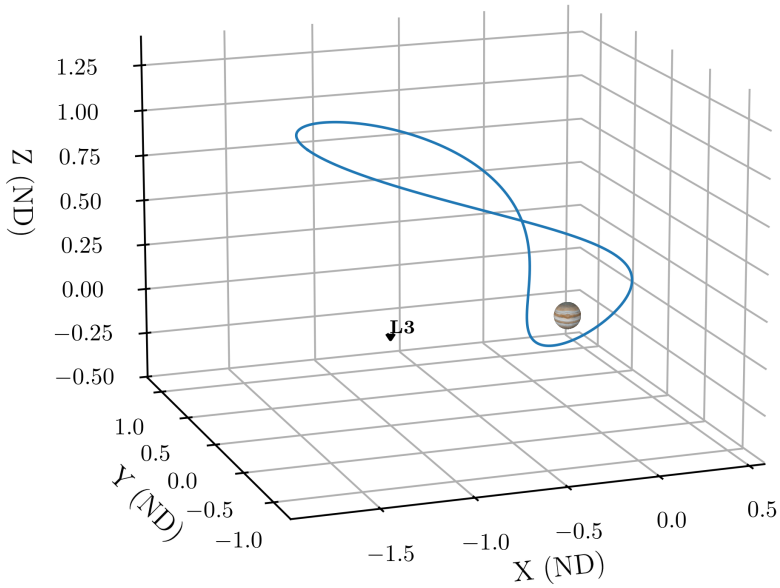


Figure B.8: 3D view of the computed Northern Halo orbit around L_3 in the Jupiter-Ganymede system with $A_z = 600,000$ km

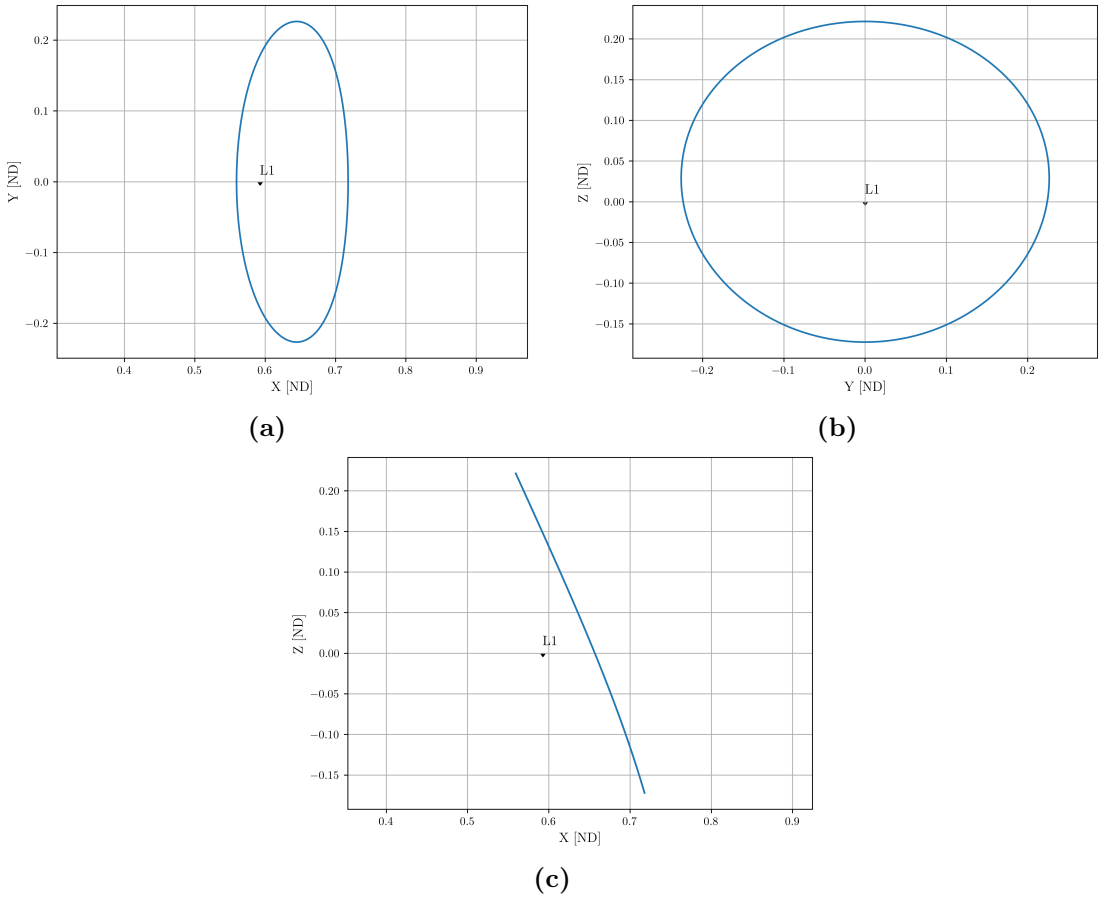


Figure B.9: 2D view of the computed Northern Halo orbit around L_1 in the Pluto-Charon system with $A_z = 4000$ km: **(a)** x-y plane, **(b)** y-z plane, **(c)** x-z plane

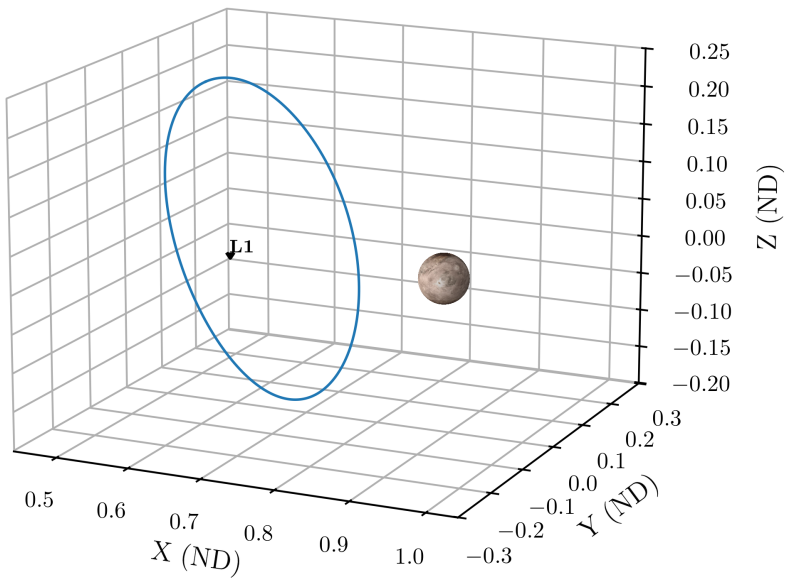


Figure B.10: 3D view of the computed Northern Halo orbit around L_1 in the Pluto-Charon system with $A_z = 4000$ km

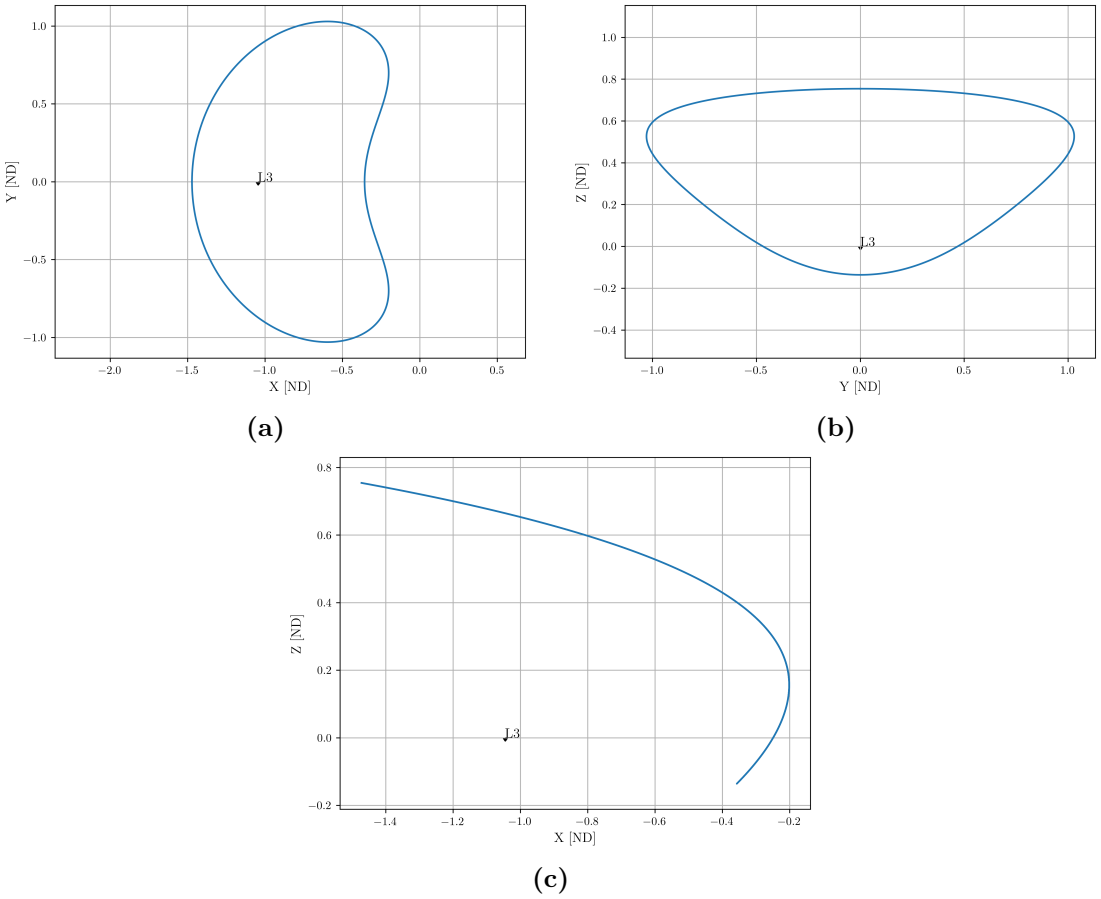


Figure B.11: 2D view of the computed Northern Halo orbit around L_3 in the Pluto-Charon system with $A_z = 9000$ km: **(a)** x-y plane, **(b)** y-z plane, **(c)** x-z plane

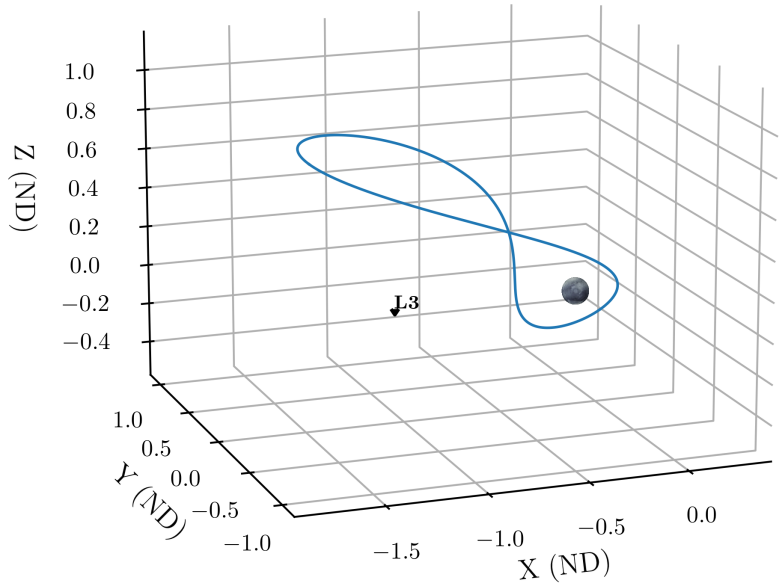


Figure B.12: 3D view of the computed Northern Halo orbit around L_3 in the Pluto-Charon system with $A_z = 9000$ km

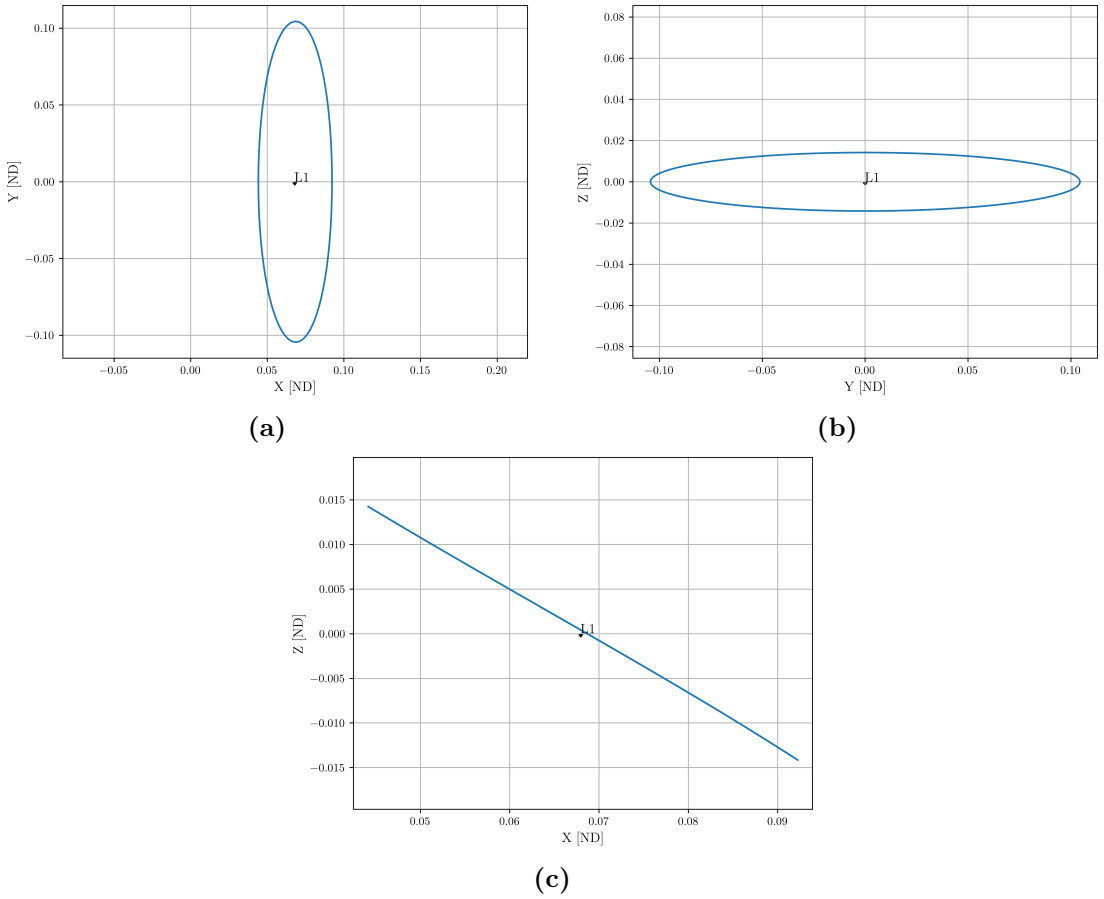


Figure B.13: 2D view of the computed Northern Halo orbit around L_1 in the α Centauri AB system with $A_z = 5 \times 10^7$ km: **(a)** x-y plane, **(b)** y-z plane, **(c)** x-z plane

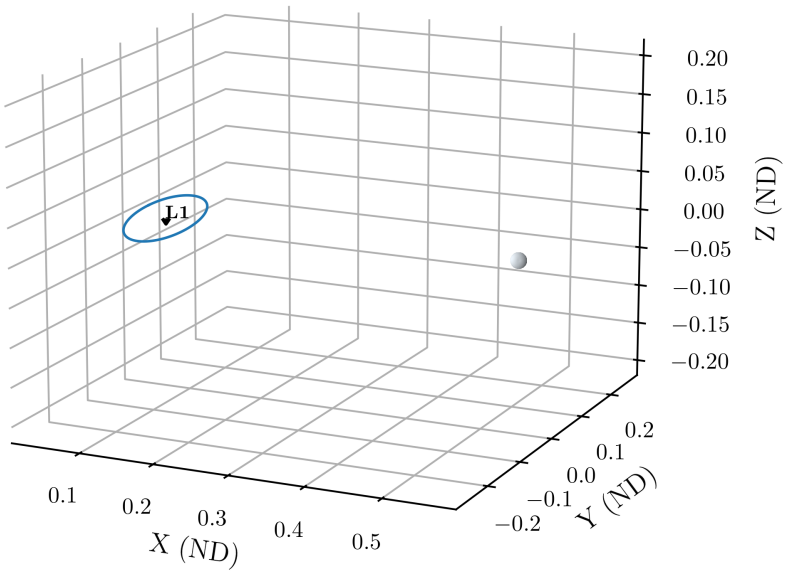


Figure B.14: 3D view of the computed Northern Halo orbit around L_1 in the α Centauri AB system with $A_z = 5 \times 10^7$ km and α Centauri B enlarged by a factor of 50x

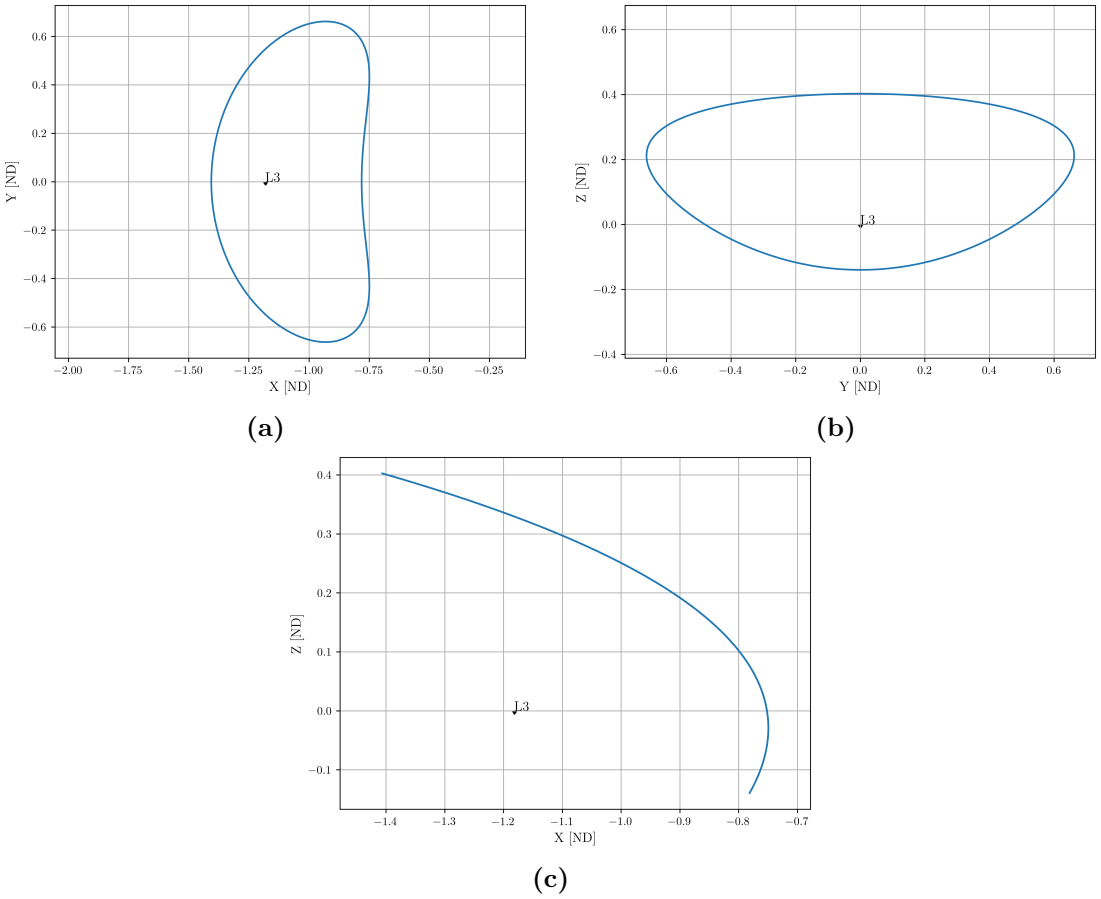


Figure B.15: 2D view of the computed Northern Halo orbit around L_3 in the α Centauri AB system with $A_z = 1 \times 10^9$ km: **(a)** x-y plane, **(b)** y-z plane, **(c)** x-z plane

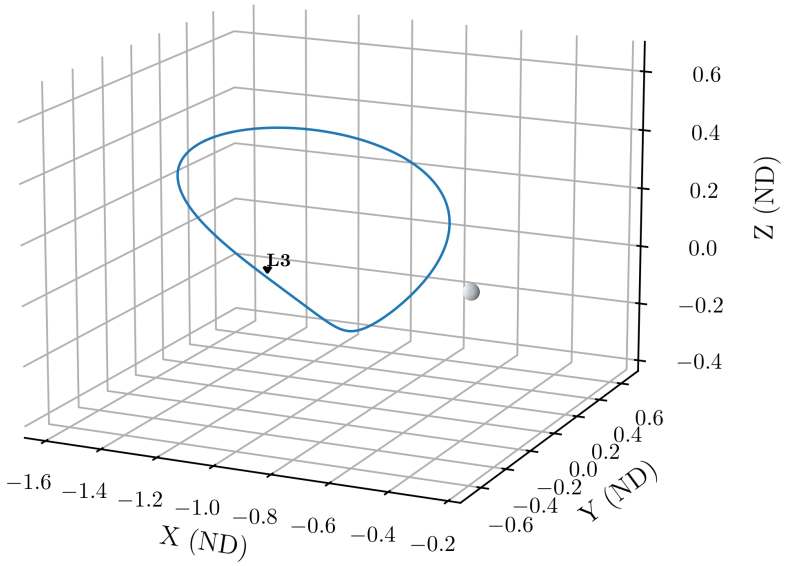


Figure B.16: 3D view of the computed Northern Halo orbit around L_3 in the α Centauri AB system with $A_z = 1 \times 10^9$ km and α Centauri A enlarged by a factor of 50x