# PERSISTENT SPACE SURVEILLANCE
# USING THE THEORY OF FUNCTIONAL CONNECTIONS

A Thesis

by

JIWON BAE

Submitted to the Graduate and Professional School of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

| | |
|---|---|
| Chair of Committee, | Daniele Mortari |
| Committee Members, | John Valasek |
| | Daniel Selva |
| | J.Maurice Rojas |
| Head of Department, | Ivett Leyva |

August 2023

Major Subject: Aerospace Engineering

# ABSTRACT

The Persistent Space Surveillance (PSS) strategy is designed to continuously detect targets and overcome the previously studied Periodic Close Encounter (PCE) problem; i.e. the periodic surveillance of targets. This study addresses the traditional need to monitor military-threatening targets and repair or check the status of commercially expensive satellites, but also the modern need for space traffic management and the removal of space debris. In this study, persistent surveillance is defined as the need to keep observation distance below a certain threshold (i.e., distance from the target) while the target orbits the Earth. In addition, the optimal solution needs to maintain a proper monitoring distance despite the effects of perturbation, and eliminate the risk of Earth-impacting or hyperbolic trajectories. The proposed technique satisfies the following three constraints needed to be sufficiently close and compatible with the target's orbit: orbit shape, orbit orientation, and observation distance. The Genetic Algorithm (GA) is used to optimize orbital elements that satisfy the three constraints. To apply the proposed model to the numerical examples, propagation is performed using the Theory of Functional Connection (TFC), which has been verified in other studies. All of the above processes are verified by simulating the existing Low Earth Orbit (LEO), Geostationary Orbit (GEO), and Highly Elliptical Orbit (HEO) satellites as targets.

# DEDICATION

To my Family, father Joonghwan Bae, mother Misuk Lee, brother Jinho Bae, and girlfriend Caitlin Downing for teaching me the most dedication.

To Republic of Korea Army for giving me the chance to study abroad and widen horizons.

# CONTRIBUTORS AND FUNDING SOURCES

## Contributors

This work was supported by my thesis advisor, Dr. Mortari, from the Department of Aerospace Engineering. Additionally, my girlfriend Caitlin significantly contributed to the improvement of this work by providing valuable corrections to the English context.

## Funding Sources

There are no outside funding contributions to acknowledge related to the research and compilation of this thesis. The student gets funding for studying and staying in the United States from Republic of Korea Army (ROKA).

# NOMENCLATURE

| | |
|---|---|
| PSS | Persistent Space Surveillance |
| PCE | Periodic Close Encounter |
| GA | Genetic Algorithm |
| TFC | Theory of Functional Connection |
| LEO | Low Earth Orbit |
| GEO | Geostationary Orbit |
| HEO | Highly Elliptical Orbit |
| IVP | Initial Value Problem |
| BVP | Boundary Value Problem |
| TLE | Two Line Elements |
| $\boldsymbol{a}$ | Semi-major axis |
| $\boldsymbol{b}$ | Semi-minor axis |
| $\boldsymbol{i}$ | Inclination |
| $\boldsymbol{\Omega}$ | Right ascension of ascending node |
| $\boldsymbol{\omega}$ | Argument of periapsis |
| $\varphi$ | True anomaly |

## TABLE OF CONTENTS

## LIST OF FIGURES

FIGURE Page

# LIST OF TABLES

TABLE Page

# 1. INTRODUCTION

Nations all around the world are starting to fight for control in the new space area. Because of this, the count of satellites in orbit and the amount of space debris is growing fast [2]. In simpler terms, the skill to find and follow targets in space is getting more and more important. Watching targets in space is needed for rendezvous, removal of space debris [3], and space traffic management [4]. As a result, finding or watching targets has become a very important field to study. The latest studies in pursuing target finding and watching have paid attention to many areas, including advanced tracking techniques [5], on-board autonomous systems [6], and collaborative monitoring approaches [7].

Advanced tracking techniques : Researchers have been developing sophisticated tracking algorithms that can accurately predict and determine the position and motion of space targets. These algorithms employ various approaches such as machine learning, data fusion, and statistical methods to enhance the accuracy and reliability of target tracking, even in congested space environments.

On-board autonomous systems: Another area of research concentrates on equipping chaser satellites with on-board autonomous capabilities. These systems enable the chaser to make real-time decisions regarding orbital adjustments, target identification, and collision avoidance without relying heavily on ground-based support. Such autonomous capabilities not only enhance the resilience and efficiency of target monitoring but also reduce the operational costs associated with ground station support.

Collaborative monitoring approaches: Some recent studies have explored the potential of utilizing multiple satellites to collaboratively monitor targets in space. This approach involves coordinating the actions of several chaser satellites, allowing them to share information and tasks, and thereby improving the overall target tracking performance. Collaborative monitoring can also help distribute the operational workload among multiple satellites, reducing the risks and costs associated with individual satellite failures or malfunctions.

In conclusion, recent research on advanced tracking methods, on-board autonomous systems, and cooperative monitoring strategies has led to significant advancements in the field of chasing target detection and monitoring. These advancements could increase the effectiveness and dependability of satellite rendezvous operations, the cleanup of space debris, and the management of space traffic, ultimately enhancing the security and sustainability of the rapidly developing space environment.

This includes research on the challenge of periodically tracking targets or Periodic Close Encounters (PCE). PCE occurs when two satellites cross paths in space on a periodic basis. Clocchiatti and Mortari were the first to study the problem. [1]. Figure 1.1 depicts the PCE problem, where the chaser, which is in orbit "**1**" transfers to the PCE orbit "**3**". The target of orbit "**2**" is periodically met by the PCE orbit "**3**" which the chaser achieves.

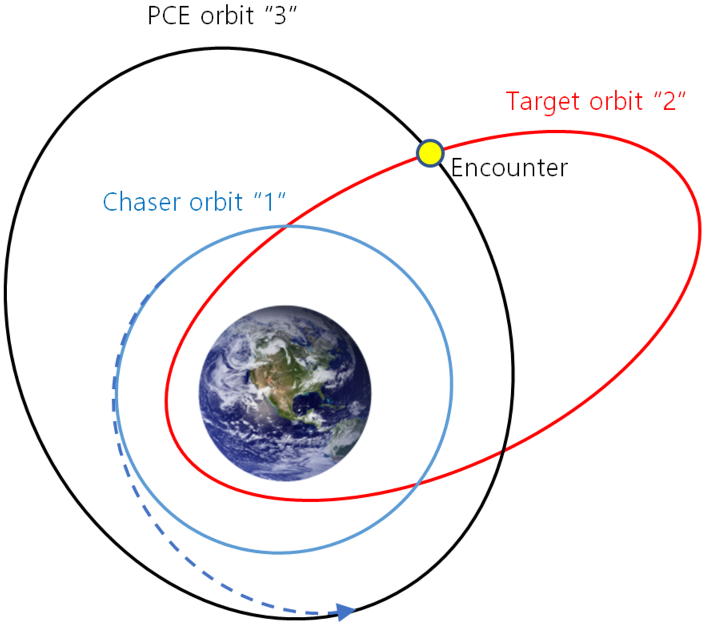

Figure 1.1: Geometry and Definitions of a 2-impulse PCE mission (adapted from : [1])

The PCE problem presents unique challenges in terms of orbital maneuvering, fuel efficiency, and the reliability of tracking systems. For the chaser satellite to maintain periodic encounters with the target, it must perform complex orbital transfers and adjustments. These maneuvers can be resource-intensive, both in terms of fuel consumption and the computational power needed to optimize the chaser's trajectory. Additionally, ensuring the accuracy and reliability of tracking systems is critical to avoid collisions or misidentification of targets. To address these challenges, researchers have explored various strategies, including optimizing orbital transfer maneuvers, utilizing advanced tracking algorithms, and improving the efficiency of propulsion systems. These efforts aim to minimize the chaser's fuel consumption, reduce the reliance on ground stations, and enhance the overall performance of the PCE-based tracking systems.

According to the original PCE theory [8], a potential PCE needs to be:

1. sufficiently close in terms of $\Delta v_{\text{tot}}$ to orbit "$\mathbf{1}$",
2. compatible (or resonant) with orbit "$\mathbf{2}$", and
3. encounter the target spacecraft with the prescribed distance and observation time.

To satisfy the specified criteria above, the PCE theory uses the Genetic Algorithm (GA) to locate an $n$-impulse PCE orbit. In prior research, the $n$-impulse PCE orbit was determined by using constraint (Minimum encounter distance, Minimum dwell time, Time constraints, and Illumination requirements) satisfaction to optimize results using GA. Several other studies have demonstrated that GA is also suitable for solving orbital mechanics problems like space rendezvous [9], Earth-surveillance problems [10, 11], and trajectory optimization problems [12, 13].

There is research on target tracking, but the majority of solutions require a large number of orbital manipulations and rely heavily on ground stations. In addition, the PCE method can only conduct periodic checks on the target satellite. The research in this study explores Persistent Space Surveillance (PSS) where targets can be tracked continually by using a new set of constraints to locate the orbit. In contrast to a PCE orbit, which can only be periodically monitored, PSS utilizes a chaser which is designed to consistently monitor its targets. The geometry depicted in Figure 1.2 defines PSS.

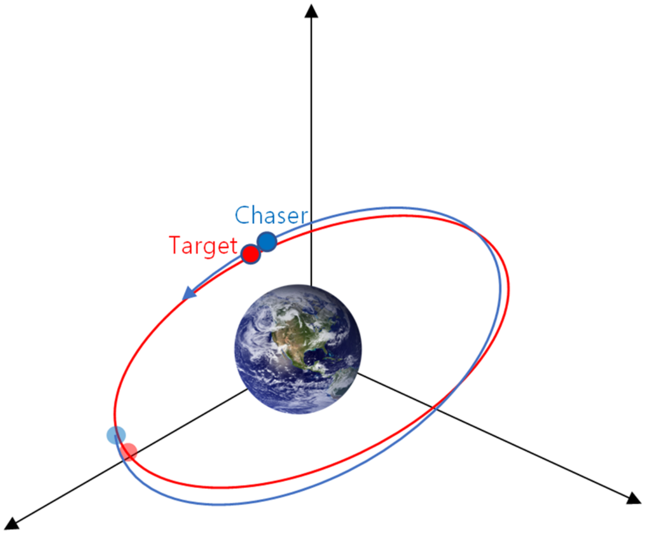

Figure 1.2: PSS Geometry and Definitions

PSS orbits are a novel approach to address the limitations of traditional target tracking methods. The PSS orbit aims to provide a more efficient and effective means of continuous monitoring for satellites and other space objects, which is crucial for maintaining the safety and integrity of space operations. One of the key features of PSS orbits is the use of a chaser satellite, which is designed to closely follow and monitor the target satellite in real-time. This is achieved through a carefully designed trajectory that optimizes the relative motion between the chaser and the target, ensuring that the chaser satellite maintains a consistent line of sight and remains within a close distance from the target. The PSS orbit relies on a new set of constraints to maintain the chaser's proximity to the target. These constraints account for various factors, such as the gravitational effects of the Earth, the Moon, and the Sun, as well as the atmospheric drag and solar radiation pressure. By considering these factors, the PSS orbit is designed to be more stable and robust against perturbations, ensuring that the chaser satellite can maintain its position relative to the target without the need for frequent orbital adjustments. In addition, the PSS orbit reduces the dependence on ground stations for tracking and maintaining the chaser's position. By utilizing on-board sensors and advanced algorithms, the chaser can autonomously maintain its orbit and make necessary adjustments when required, reducing the need for continuous ground station support. This not only makes the PSS orbit more cost-effective but also increases its resilience in the face of communication disruptions or other operational challenges. Furthermore, the PSS orbit offers a high degree of flexibility and adaptability, making it suitable for various mission scenarios. For example, it can be used for tracking non-cooperative targets, such as space debris or malfunctioning satellites, as well as cooperative targets that require continuous monitoring, such as servicing or refueling missions. In summary, PSS orbit represents a significant advancement in satellite tracking and monitoring capabilities. By utilizing a chaser satellite and a new set of constraints, the PSS orbit provides continuous, real-time tracking of target satellites without relying heavily on ground stations or frequent orbital adjustments. This innovative approach not only improves the efficiency and effectiveness of space operations but also contributes to the long-term sustainability and safety of the space environment.

This research uses two primary methods for PSS. The application of these methods enhances the capability to establish PSS orbits and increases the accuracy of orbit propagation :

1. Genetic Algorithm (GA)
2. Theory of Functional Connection (TFC)

Firstly, the GA is employed to find the optimal solution that satisfies all three constraints (orbit shape, orbit orientation, and observation distance) necessary for determining the PSS orbital elements of the chaser satellite. In order to establish a PSS orbit, the orbital elements of the chaser and target satellites must be similar. The semi-major axis ($\boldsymbol{a}$) and semi-minor axis ($\boldsymbol{b}$) define the orbit shape, so the chaser's $\boldsymbol{a}$ and $\boldsymbol{b}$ values must be comparable to those of the target. However, to ensure continuous monitoring, the orbital periods of the chaser and target must be identical, leading to equal $\boldsymbol{a}$ values for both satellites, as shown in Equation (1.1).

$$T = 2\pi\sqrt{\frac{a^3}{\mu}} \tag{1.1}$$

The inclination ($\boldsymbol{i}$), right ascension of the ascending node ($\boldsymbol{\Omega}$), and argument of periapsis ($\boldsymbol{\omega}$) determine the orientation of the chaser's orbit and should also have values similar to those of the target. This can be represented by an equation for the orbit's principal axis. Lastly, the true anomaly ($\boldsymbol{\varphi}$) determines the chaser's position, which can be calculated by selecting an observation distance from the target as a constraint. In essence, the PSS orbit is established by selecting the aforementioned chaser orbital elements as GA genes and optimizing them within the range that satisfies the specified constraints.

The second step involves conducting orbit propagation using the TFC. Previous research has shown that the solutions for linear and nonlinear differential equations obtained through TFC are more accurate than those derived from general integrators. In this study, orbit propagation is performed using TFC, solving the differential equation with nonlinear least squares, and expressing the perturbed orbit propagation equation as follows:

$$\ddot{\boldsymbol{r}}(t) = -\frac{\mu}{r^3(t)}\boldsymbol{r}(t) + \boldsymbol{a}_p(t, \boldsymbol{r}, \dot{\boldsymbol{r}}) \tag{1.2}$$

By solving the perturbed multi revolution Lambert problem with TFC, this research aims to achieve better control over orbits for space surveillance and more precise trajectory adjustments in response to perturbations. This innovative approach contributes to the effectiveness and accuracy of Persistent Space Surveillance, addressing critical challenges in the field of space engineering.

# 2. PERSISTENT SPACE SURVEILLANCE*

## 2.1 Motivation

Bae and Mortari pioneered the study of the PSS problem, aiming to enable continuous monitoring of space targets—an achievement that was not possible with the previously studied PCE problem. The PSS orbit optimally satisfies the given constraints, ensuring that the target and chaser satellites do not collide while providing uninterrupted observation of the target. This capability offers significant military and commercial benefits.

In the context of military operations, PSS emerges as a superior space intelligence warfare tool, as it enables the continuous monitoring of potential threats. This advantage allows for timely elimination of hostile targets in emergency situations, significantly enhancing the strategic capabilities of the military in space. On the commercial front, PSS can be employed for various purposes, such as repairing or inspecting high-cost satellites, as well as facilitating rendezvous operations for spacecraft resupply missions. This continuous monitoring capability not only extends the lifespan of valuable space assets but also improves overall mission safety and efficiency.

Furthermore, the Low Earth Orbit (LEO) region is becoming increasingly congested, with numerous nations actively launching satellites, leading to a proliferation of space debris. With more than 8,000 satellites already in LEO and over 20,000 planned launches in the next decade, effective space traffic management and debris removal are critical to preventing satellite collisions. By leveraging the PSS problem, it is possible to address these concerns and maintain the safety and sustainability of the space environment.

In summary, the PSS problem offers a range of advantages in both military and commercial applications, including enhanced space intelligence warfare capabilities, satellite maintenance and inspection, and efficient rendezvous operations. Moreover, PSS plays a crucial role in addressing

[*] Parts of this section have been reprinted from 1) H. Johnston, D. Mortari, *Orbit propagation via the theory of functional connections*, Proceedings of the 2019 AAS/AIAA Astrodynamics Specialist Conference, Portland, ME, USA, pp. 11–15, 2019 : [14] and 2) F. Criscola, D. Canales, D. Mortari, *Solution of the Perturbed Lambert's Problem Using the Theory of Functional Connections*, 33rd AAS/AIAA Astrodynamics Specialist Conference, Big Sky (MT), 2023 : [15]

the challenges of space traffic management and debris removal in the increasingly congested LEO region.

## 2.2 GA and Constraints

The GA is a powerful heuristic optimization technique that draws inspiration from the principles of natural selection and genetics. The algorithm operates by generating a population of potential solutions and iteratively selecting the fittest individuals to produce offspring. These offspring inherit the most desirable traits from their parents, and then undergo genetic operations such as mutation and crossover to form new individuals. As the process repeats, the population evolves, converging towards an increasingly optimized solution.

GA has been successfully applied to a diverse range of optimization problems, including feature selection, image processing, financial forecasting, and many more. The algorithm's widespread popularity is attributable to its ability to handle intricate, high-dimensional search spaces and to efficiently identify globally optimal solutions.

In this study, GA is employed to achieve Persistent Space Surveillance by optimizing three constraints: orbit shape, orbit orientation, and observation distance. By utilizing GA, the initial PSS orbital elements of the chaser satellite can be optimized. The initial position and velocity vector are then derived by converting the optimized initial PSS orbital elements. This iterative process enables the chaser satellite to maintain optimal proximity to the target, satisfying the necessary constraints while continually monitoring the target. The procedures involved in this optimization process are depicted in Figure 2.1.

Incorporating GA into the PSS problem allows for the efficient and accurate optimization of the chaser satellite's orbit, ensuring the effective monitoring of targets in space. This approach demonstrates the robustness of GA in solving complex optimization problems within the realm of space engineering.

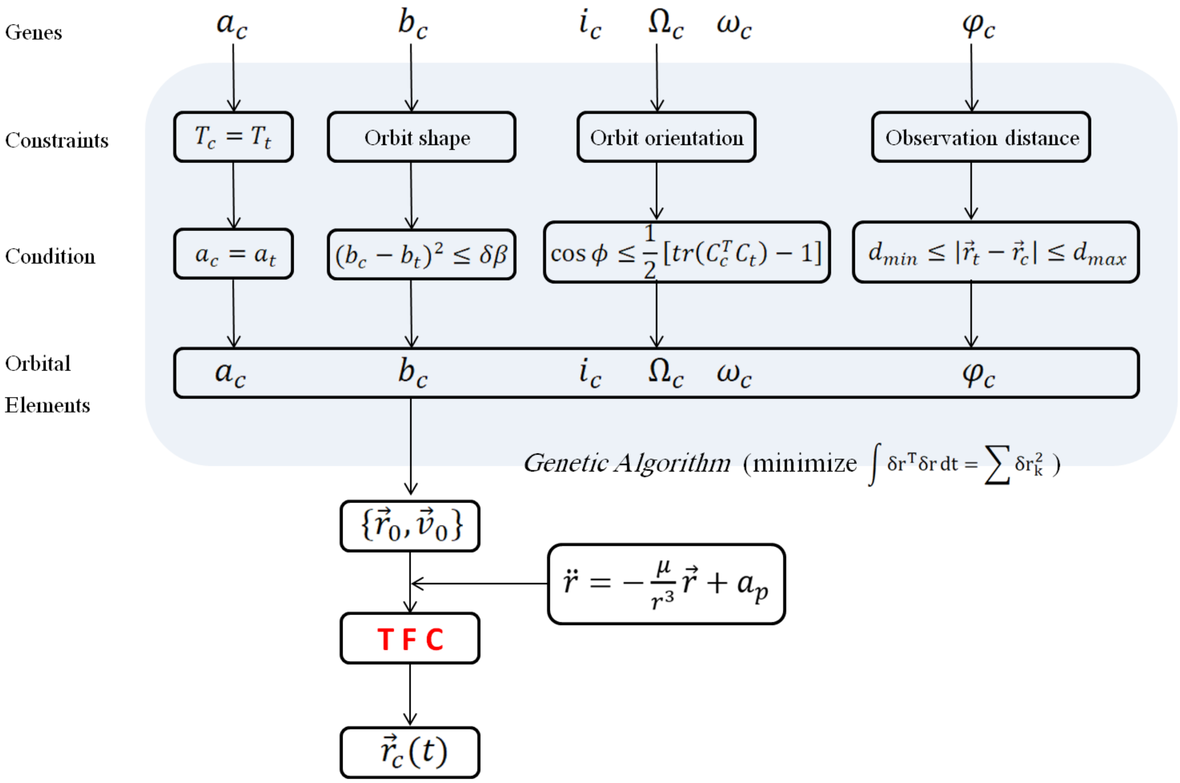
Genes
$a_c$
$b_c$
$i_c$
$\Omega_c$
$\omega_c$
$\varphi_c$
Constraints
$T_c = T_t$
Orbit shape
Orbit orientation
Observation distance
Condition
$a_c = a_t$
$(b_c - b_t)^2 \leq \delta\beta$
$\cos\phi \leq \frac{1}{2}[tr(C_c^T C_t) - 1]$
$d_{min} \leq |\vec{r}_t - \vec{r}_c| \leq d_{max}$
Orbital Elements
$a_c$
$b_c$
$i_c$
$\Omega_c$
$\omega_c$
$\varphi_c$
*Genetic Algorithm* (minimize $\int \delta r^T \delta r \, dt = \sum \delta r_k^2$ )
$\{\vec{r}_0, \vec{v}_0\}$
$\ddot{\vec{r}} = -\frac{\mu}{r^3}\vec{r} + a_p$
TFC
$\vec{r}_c(t)$

Figure 2.1: Genetic Algorithm flow chart

### 2.2.1 Orbit shape

In order to determine the chaser's PSS orbit, it is essential to identify an orbit that closely aligns with the target's orbit. Continuous monitoring without collisions necessitates that the orbits be as similar as possible, ideally identical. To achieve the most precise orbit selection, the approximate shape of all orbits can be inferred from the values of the semi-major axis ($\boldsymbol{a}$) and semi-minor axis ($\boldsymbol{b}$). Finding a chaser's $\boldsymbol{a}$ and $\boldsymbol{b}$ in close proximity to the target's orbit shape is crucial to ensure that the PSS orbit maintains constant surveillance of the target. This requirement can be expressed using the following equation:

$$(a_t - a_c)^2 + (b_t - b_c)^2 \leq \delta\beta^2 \tag{2.1}$$

Equation (2.1) represents a circle equation with a radius of $\delta\beta$ centered on the target's $\boldsymbol{a}$ and $\boldsymbol{b}$ values in a two-dimensional plane. The chaser's $\boldsymbol{a}$ and $\boldsymbol{b}$ serve as variables representing the location, while $x$ and $y$ denote the respective axes. The closer the chaser's orbital shape is to the target's, the nearer the values of $\boldsymbol{a}_c$ and $\boldsymbol{b}_c$ will be to the circle's center. To establish the appropriate PSS orbit shape, the constraints of $\delta\beta$ must first be defined, after which the values of $\boldsymbol{a}_c$ and $\boldsymbol{b}_c$ can be optimized.

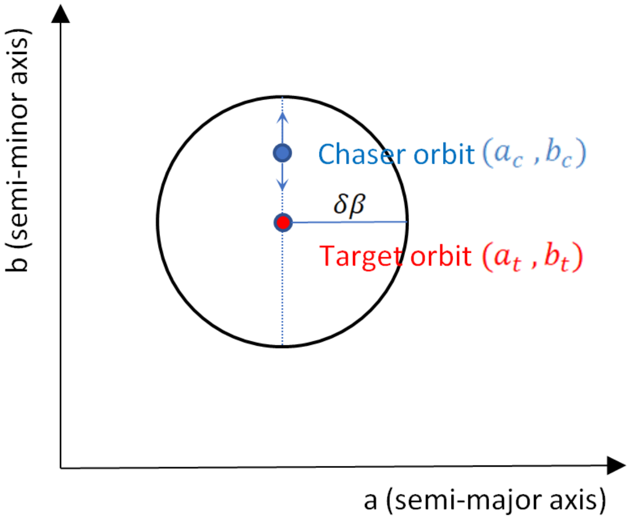

Figure 2.2: Orbit shape constraint

Nonetheless, for the chaser to maintain continuous monitoring of the target, both satellites must have the same time period, as indicated in Equation (1.1), which necessitates that both $\boldsymbol{a}$ elements be equal. Consequently, only the value of $\boldsymbol{b}_c$ needs to be optimized to define the orbit's shape. This optimization process ensures that the chaser's orbit is accurately aligned with the target's orbit, enabling effective and continuous surveillance in a PSS configuration.

### 2.2.2 Orbit orientation

Even if the orbit shape is comparable, continuous monitoring is impossible if the direction of the principal axis differs significantly between the chaser and the target. The principal axis defines the rotational direction, and the principal angle represents the amount of rotation around this axis from the initial attitude to the final attitude. To achieve a PSS orbit, the principal axis must not deviate considerably from the target's principal axis. The ($\phi$) shown in the Equation (2.2) signifies the principal angle, which is the difference, or distance, between the axes of the two orbits.

$$\phi = \cos^{-1}\left(\frac{1}{2}[\mathrm{tr}(C_c^{\mathrm{T}}C_t) - 1]\right) < \phi_{tol} \tag{2.2}$$

where $C_c$ is the coordinate transforming matrix of the chaser and $C_t$ is the coordinate transforming matrix of the target.

$$C = \begin{bmatrix} \cos\omega & \sin\omega & 0 \\ -\sin\omega & \cos\omega & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos i & \sin i \\ 0 & -\sin i & \cos i \end{bmatrix} \begin{bmatrix} \cos\Omega & \sin\Omega & 0 \\ -\sin\Omega & \cos\Omega & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

These matrices are made up of the argument of periapsis, inclination, right ascension of the ascending node of the chaser and target respectively. By setting the tolerance of the principal angle, the range of values for the above three elements can be determined. Of course, the lower the tolerance of the principal angle, the more similar the orientations of the two orbits are. Through this constraint, the three orbital elements can be obtained as the GA output. A GA can then be employed to optimize the chaser's orbit orientation, ensuring that the chaser's orbit is closely aligned with the target's orbit in both shape and orientation. This alignment is crucial for enabling continuous monitoring of the target in a PSS configuration.

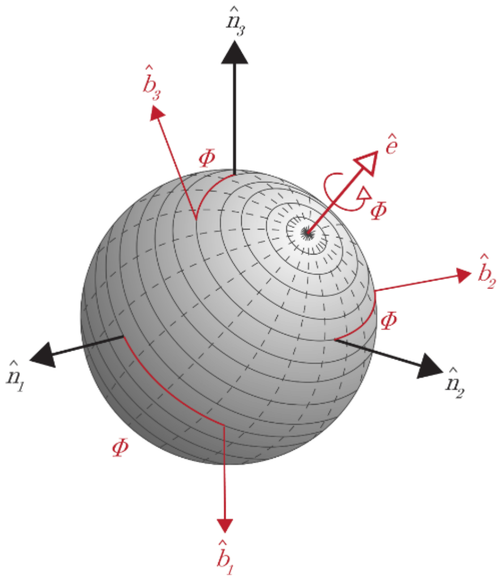

Figure 2.3: Euler's principal rotation

### 2.2.3 Observation distance

The observation distance is a critical parameter for establishing and optimizing the initial true anomaly ($\varphi_c$) of the PSS orbit. The difference between the position vectors of the target and chaser must be maintained within a specified range, ensuring that the distance is neither too close (risking collision) nor too far (compromising observation capabilities). The minimum observation distance serves as a safety buffer to prevent collisions between the two satellites, while the maximum observation distance is determined by the observation system's capabilities.

$$d_{\min} \leq \left| \boldsymbol{r}_t - \hat{\boldsymbol{r}}_c \right|_2 \leq d_{\max} \tag{2.3}$$

To satisfy the constraint expressed in Equation (2.3), the $\varphi_c$ can be represented by Equation (2.4), in which $\alpha$ denotes the maximum angle between the target and chaser (i.e., the maximum true anomaly difference).

$$\varphi_t - \alpha \leq \varphi_c \leq \varphi_t + \alpha \tag{2.4}$$

where,

$$\sin\alpha = \frac{d_{\max}}{r_t}$$

To prevent collisions, it is essential to consider the angle occupied by $d_{\min}$ within the $\boldsymbol{\varphi}_c$ range. However, since $d_{\min}$ is typically a very small value (e.g., $d_{\max} = 10km$, $d_{\min} = 10m$), its impact on the overall $\boldsymbol{\varphi}_c$ range is negligible, and thus not accounted for in this analysis. By carefully selecting the $\boldsymbol{\varphi}_c$, the PSS orbit can ensure continuous monitoring of the target satellite while maintaining a safe and effective observation distance throughout the mission.

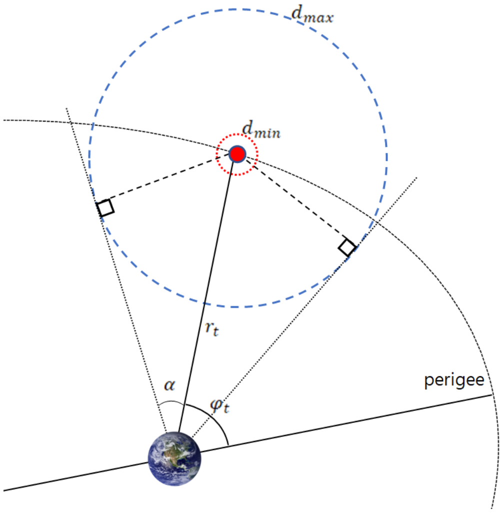

Figure 2.4: Observation distance

## 2.3 Theory of Functional Connections

Differential equations play a crucial role in physics, assisting in the comprehension of natural laws. These are particularly vital in space engineering, where precise solutions are needed for tasks such as modifying a satellite's location and following its path in space. Throughout time, various strategies have been devised to solve these equations, each with its own advantages. However, a technique known as the Theory of Functional Connections (TFC) [16] has demonstrated to be particularly efficient. This document will elucidate how this strategy operates, its notable features, and its relevance in space engineering. The TFC-based methods, detailed in previous research [17, 18], have demonstrated remarkable effectiveness in dealing with differential equations. In contrast to traditional methodologies, TFC-based methods possess several notable features:

1. Exceptional machine-level accuracy
2. Superior computational speed
3. Robust solutions that maintain a low condition number

1) Exceptional machine-level accuracy: The answers obtained using TFC exhibit extraordinary precision, reaching a level of detail in line with what contemporary computing devices can deliver. This heightened degree of precision guarantees that the outcomes are dependable and can be utilized for additional scrutiny and implementation. 2) Superior computational speed: Another asset of TFC techniques is their enhanced speed regarding calculations. Relative to other strategies, such as the well-recognized Runge-Kutta, TFC techniques are much more swift. 3) Robustness and stability: TFC answers remain steady and exhibit minimal fluctuations with slight alterations in input. This renders them more resilient against errors in computations or input, which is exceedingly critical for precise depiction of satellite trajectories. Collectively, these features render TFC techniques highly effective for resolving issues involving differential equations. These perks ensure that the solutions are reliable and perform well, even under rigorous testing, particularly in the demanding field of space engineering.

TFC is a well-known theory and its features show that it's a really good way to solve problems with differential equations, especially in making models of satellite motion in space engineering. This paper will look into how this method works, how it's been used in real situations, and how it could help make better solutions in space engineering.

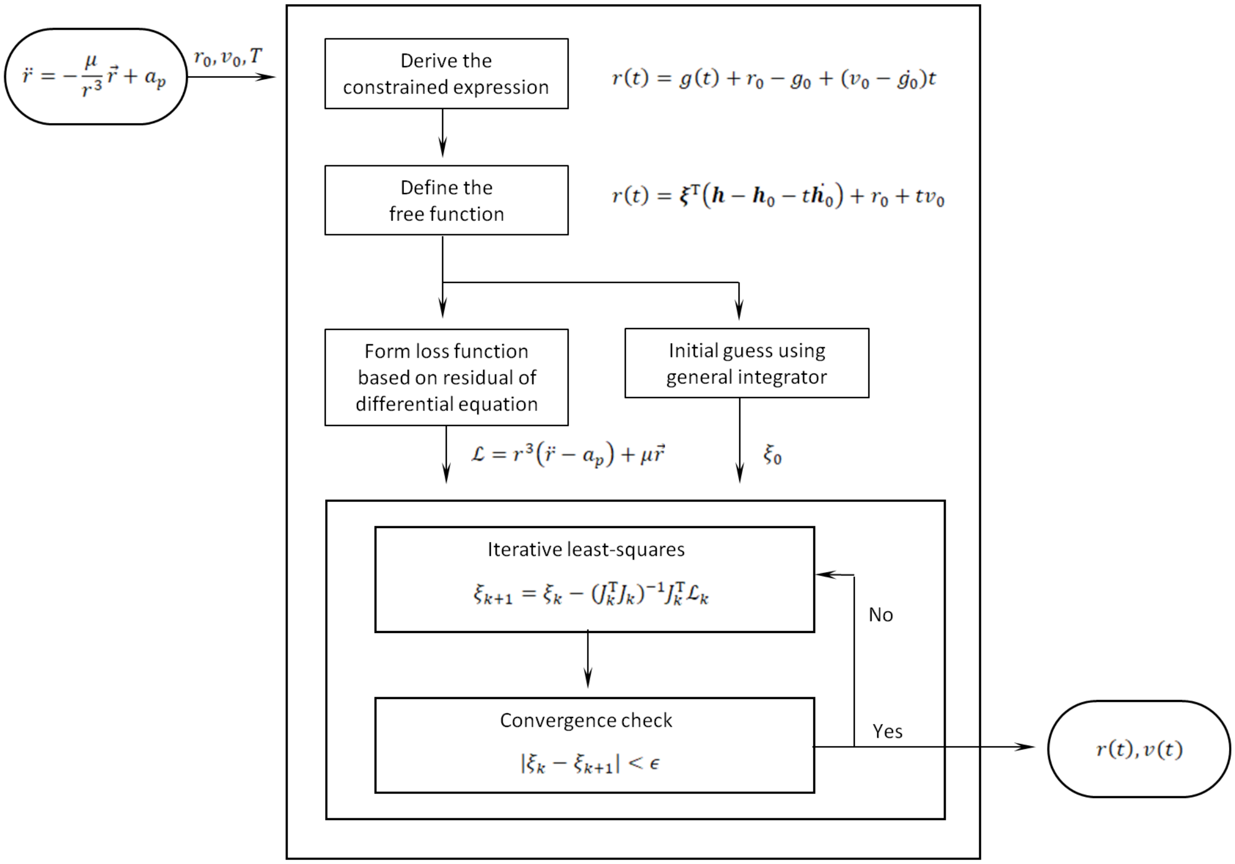
$\ddot{r} = -\frac{\mu}{r^3}\vec{r} + a_p$
$r_0, v_0, T$
Derive the constrained expression
$r(t) = g(t) + r_0 - g_0 + (v_0 - \dot{g}_0)t$
Define the free function
$r(t) = \xi^T(h - h_0 - t\dot{h}_0) + r_0 + tv_0$
Form loss function based on residual of differential equation
Initial guess using general integrator
$\mathcal{L} = r^3(\ddot{r} - a_p) + \mu\vec{r}$
$\xi_0$
Iterative least-squares
$\xi_{k+1} = \xi_k - (J_k^T J_k)^{-1} J_k^T \mathcal{L}_k$
No
Convergence check
$|\xi_k - \xi_{k+1}| < \epsilon$
Yes
$r(t), v(t)$

Figure 2.5: TFC algorithm flow chart

Johnston's study [14] gives a thorough explanation of a model for predicting changes in satellite orbits using TFC. This model is very useful for understanding and predicting how satellites move, especially when their orbits change slightly. Figure 2.5 shows a flow chart that outlines Johnston's study, making it easier to understand how TFC is used to model and analyze satellite motion. We'll talk more about Johnston's study below.

### 2.3.1 Background on the TFC

#### *2.3.1.1 The origin of functional interpolation*

The roots of functional interpolation are closely linked with "Lagrange interpolation". Equation (2.5) illustrates the Lagrange interpolation formula, showing a line directly crossing two points, $(x_1, y_1)$ and $(x_2, y_2)$.

$$y(x) = \frac{x - x_2}{x_1 - x_2} y_1 + \frac{x - x_1}{x_2 - x_1} y_2 \tag{2.5}$$

Taking a cue from Equation (2.5), functional interpolation is a calculation that can depict any line crossing through two points, and this is demonstrated in Equation (2.6).

$$y(x, g(x)) = g(x) + \frac{x - x_2}{x_1 - x_2}(y_1 - g(x_1)) + \frac{x - x_1}{x_2 - x_1}(y_2 - g(x_2)) \tag{2.6}$$

The $g(x)$ appearing in this formula is a free function, meaning it can represent any kind of equation. Consequently, it can represent any line passing through the two points $(x_1, g(x_1))$ and $(x_2, g(x_2))$. The detailed derivation process will be explained below. Equation (2.6) is generalized, it is as follows.

$$y(x, g(x)) = g(x) + \sum_{k=1}^{n} (y_k - g(x_k)) \prod_{i \neq k} \frac{x - x_i}{x_k - x_i} \tag{2.7}$$

In other words, the above equation includes constraints that pass through $n$ points and is called a constrained expression. Two well-known types of constrained expressions are Initial Value Problem (IVP) and Boundary Value Problem (BVP), which are explained in detail below.

#### *2.3.1.2 The $\eta$ constrained expression for Initial value problem*

Equation (2.7) can be transformed as follows. Here, $\eta$ is an expression related to $x$ and $g(x)$ that includes each constraint, and $s(x)$ is the support function.

$$y(x, g(x)) = g(x) + \sum_{j=1}^{n} \eta_j(x, g(x)) s_j(x) \tag{2.8}$$

To solve the Initial Value Problem (IVP), when two constraints are given, Equation (2.8) can be represented as follows.

$$y(x, g(x)) = g(x) + \eta_1 s_1(x) + \eta_2 s_2(x) \qquad \text{subject to :} \quad \begin{cases} y(x_1) = y_1 \\ \dot{y}(x_1) = \dot{y}_1 \end{cases} \tag{2.9}$$

When the given constraints are applied, the $\eta$ constrained expression can be represented as follows.

$$\begin{aligned} y_1 &= g(x_1) + \eta_1 s_1(x_1) + \eta_2 s_2(x_1) \\ \dot{y}_1 &= \dot{g}(x_1) + \eta_1 \dot{s}_1(x_1) + \eta_2 \dot{s}_2(x_1) \end{aligned}$$

Assuming that the support function as follows,

$$s_1(x) = 1, \quad s_2(x) = x$$

The constrained expression can be represented as follows,

$$y_1 = g(x_1) + \eta_1 + \eta_2 x_1 \tag{2.10}$$

$$\dot{y}_1 = \dot{g}(x_1) + \eta_2 \tag{2.11}$$

To obtain $\eta_1$ and $\eta_2$, they are expressed in matrix form as follow,

$$\begin{Bmatrix} y_1 - g(x_1) \\ \dot{y}_1 - \dot{g}(x_1) \end{Bmatrix} = \begin{bmatrix} 1 & x_1 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} \eta_1 \\ \eta_2 \end{Bmatrix}$$

When the above matrix is calculated, $\eta_1$ and $\eta_2$ are as follows.

$$\eta_1 = y_1 - g(x_1) - x_1(\dot{y}_1 - \dot{g}(x_1))$$
$$\eta_2 = \dot{y}_1 - \dot{g}(x_1)$$

By substituting $\eta_1$ and $\eta_2$ into Equations (2.10) and (2.11), we can obtain the following constrained expression.

$$y(x, g(x)) = g(x) + (y_1 - g(x_1)) + (x - x_1)(\dot{y}_1 - \dot{g}(x_1)) \tag{2.12}$$

In other words, Equation (2.12) is a constrained expression related to the Initial Value Problem (IVP), and it can represent all equations that satisfy the initial value constraints mentioned in Equation (2.9).

#### *2.3.1.3 The $\eta$ constrained expression for Boundary value problem*

Similar to the Initial Value Problem (IVP), the $\eta$ constrained expression for the Boundary Value Problem (BVP) can also be represented as follows,

$$y(x, g(x)) = g(x) + \eta_1 s_1(x) + \eta_2 s_2(x) \qquad \text{subject to :} \begin{cases} y(x_1) = y_1 \\ y(x_2) = y_2 \end{cases} \tag{2.13}$$

When the given two constraints are substituted into Equation (2.13), it is expressed as follows,

$$\begin{aligned} y_1 &= g(x_1) + \eta_1 s_1(x_1) + \eta_2 s_2(x_1) \\ y_2 &= g(x_2) + \eta_1 s_1(x_2) + \eta_2 s_2(x_2) \end{aligned}$$

Assuming that the support function as follows,

$$s_1(x) = 1, \quad s_2(x) = x$$

We obtain the constrained expression in the following form.

$$y_1 = g(x_1) + \eta_1 + \eta_2 x_1 \tag{2.14}$$

$$y_2 = g(x_2) + \eta_1 + \eta_2 x_2 \tag{2.15}$$

By transforming $\eta_1$ and $\eta_2$ into matrix form,

$$\begin{Bmatrix} y_1 - g(x_1) \\ y_2 - g(x_2) \end{Bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \end{bmatrix} \begin{Bmatrix} \eta_1 \\ \eta_2 \end{Bmatrix}$$

We can obtain the values of $\eta_1$ and $\eta_2$ as shown below.

$$\begin{aligned} \eta_1 &= \frac{1}{x_2 - x_1}[x_2(y_1 - g(x_1)) - x_1(y_2 - g(x_2))] \\ \eta_2 &= \frac{1}{x_2 - x_1}[-(y_1 - g(x_1)) + (y_2 - g(x_2))] \end{aligned}$$

By substituting $\eta_1$ and $\eta_2$ into Equations (2.14) and (2.15), we can obtain the constrained expression for the BVP that satisfies the two constraints mentioned in Equation (2.13).

$$y(x, g(x)) = g(x) + \frac{x - x_2}{x_1 - x_2}(y_1 - g(x_1)) + \frac{x - x_1}{x_2 - x_1}(y_2 - g(x_2))$$

#### *2.3.1.4 Linear differential equations*

This section discusses how to solve linear differential equations using the TFC. For this, we use the Initial Value Problem (IVP) as an example. Equation (2.16), used as an example, represents a second-order differential equation and its initial values.

$$f(x, y, \dot{y}, \ddot{y}) = 0 \qquad \text{subject to :} \quad \begin{cases} y(x_1) = y_1 \\ \dot{y}(x_1) = \dot{y}_1 \end{cases} \tag{2.16}$$

To solve Equation (2.16), we need to substitute the constrained expression mentioned in Equation (2.12) according to the form of the differential equation. Since the above formula is a second-order differential equation, the constrained expressions can be represented as follows.

$$\begin{aligned} &y(x, g(x)) = g(x) + (y_1 - g(x_1)) + (x - x_1)(\dot{y}_1 - \dot{g}(x_1)) \\ &\dot{y}(x, g(x)) = \dot{g}(x) - \dot{g}(x_1) + \dot{y}_1 \\ &\ddot{y}(x, g(x)) = \ddot{g}(x) \end{aligned}$$

Moreover, to solve a differential equation using TFC, we need to use a “Basis function” (e.g., Chebyshev, Legendre orthogonal polynomials). For this, we need to discretize the domain and represent it as collocation points. Here, $x$ should be mapped to the domain $\omega \in [-1, 1]$.

$$\frac{\omega + 1}{2} = \frac{x - x_0}{x_f - x_0} \quad \rightarrow \quad \frac{\mathrm{d}\omega}{\mathrm{d}x} = \frac{2}{x_f - x_0} = c \tag{2.17}$$

To discretize, we can use the following equation.

$$\omega_k = -\cos\frac{(k-1)\pi}{n-1}$$

$\dot{y}$ and $\ddot{y}$, when expressed as derivatives with respect to $\omega$, can be represented as follows, and the value of $c$ is indicated in Equation (2.17).

$$\dot{y} = \frac{\mathrm{d}y}{\mathrm{d}x} = \frac{\mathrm{d}y}{\mathrm{d}\omega} \cdot \frac{\mathrm{d}\omega}{\mathrm{d}x} = y'c$$
$$\ddot{y} = \frac{\mathrm{d}(y'c)}{\mathrm{d}x} = \frac{\mathrm{d}y'}{\mathrm{d}\omega} \cdot \frac{\mathrm{d}\omega}{\mathrm{d}x} = y''c^2$$

In other words, if we express the constrained expression that changes the domain regarding $x$ to the domain regarding $\omega$, it can be represented as follows.

$$\begin{aligned} y(\omega, g(\omega)) &= g(\omega) + (y_1 - g(\omega_1)) + (\omega - \omega_1)(y_1' - g'(\omega_1)) \\ y'(\omega, g(\omega)) &= c(g'(\omega) - g'(\omega_1)) + y_1' \\ y''(\omega, g(\omega)) &= c^2 g''(\omega) \end{aligned}$$

$g(\omega)$ can be represented as follows. This is to solve the basis function in the form of least squares. Here, $\xi$ is an unknown value, which can be obtained by solving the least squares.

$$g(\omega) = \boldsymbol{\xi}^{\mathrm{T}} \boldsymbol{h}(\omega) \tag{2.18}$$

$$g'(\omega) = \boldsymbol{\xi}^{\mathrm{T}} \boldsymbol{h}'(\omega) \tag{2.19}$$

$$g''(\omega) = \boldsymbol{\xi}^{\mathrm{T}} \boldsymbol{h}''(\omega) \tag{2.20}$$

$g(\omega)$ is represented in the form of a basis function and least squares. In this context, the free function must be linearly independent from the support functions. In other words, since we assumed the support function as $x$ and $1$ in this section, the free function must be at least a quadratic term. This is because if it is not independent of the support function, the free function will be eliminated along with the support function during the calculation. For this reason, when applying the basis function below, we include from $L_2$, excluding $L_0$ and $L_1$ in the calculations.

$$\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{h}(\omega) = \begin{bmatrix} L_2(\omega_1) & L_3(\omega_1) & \dots & L_m(\omega_1) \\ L_2(\omega_2) & L_3(\omega_2) & \dots & L_m(\omega_2) \\ \vdots & \vdots & \vdots & \vdots \\ L_2(\omega_n) & L_3(\omega_n) & \dots & L_m(\omega_n) \end{bmatrix} \begin{Bmatrix} \xi_2 \\ \xi_3 \\ \vdots \\ \xi_m \end{Bmatrix} = \begin{Bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{Bmatrix} = g(\omega)$$

When we substitute Equation (2.18 – 20) into the constrained expression, it becomes as follows.

$$y(\omega, g(\omega)) = \boldsymbol{\xi}^{\mathrm{T}}(\boldsymbol{h} - \boldsymbol{h}_0 - \omega\boldsymbol{h'}_0) + y_1 + \omega y_1' \tag{2.21}$$

$$y'(\omega, g(\omega)) = \boldsymbol{\xi}^{\mathrm{T}}(\boldsymbol{h}' - \boldsymbol{h'}_0) + y_1' \tag{2.22}$$

$$y''(\omega, g(\omega)) = \boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{h}'' \tag{2.23}$$

After substituting Equation (2.21 – 23) into Equation (2.16), we express the left-hand side as $\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{h}$, and the right-hand side as the other terms.

$$f_1(\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{h}, \boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{h}', \boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{h}'') = f_2(\omega)$$

Then, by using the least squares in the form below,

$$A\boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{a}_2(\omega_1) & \boldsymbol{a}_3(\omega_1) & \dots & \boldsymbol{a}_m(\omega_1) \\ \boldsymbol{a}_2(\omega_2) & \boldsymbol{a}_3(\omega_2) & \dots & \boldsymbol{a}_m(\omega_2) \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{a}_2(\omega_n) & \boldsymbol{a}_3(\omega_n) & \dots & \boldsymbol{a}_m(\omega_n) \end{bmatrix} \begin{Bmatrix} \xi_2 \\ \xi_3 \\ \vdots \\ \xi_m \end{Bmatrix} = \begin{Bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{Bmatrix} = \boldsymbol{b}$$

we can obtain the unknown value as follows, and by substituting it into the differential equation and calculating, we can get the solution.

$$\boldsymbol{\xi} = (A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}\boldsymbol{b} \tag{2.24}$$

#### *2.3.1.5 Nonlinear differential equations*

This section discusses how to solve nonlinear differential equations using the TFC. To solve a nonlinear differential equation, we convert the equation into a residual form and then use the Newton method to find the unknown value, $\boldsymbol{\xi}$. First, we convert the given equation into a residual form and then substitute the constrained expression in the same way as mentioned above to make it into the form shown below.

$$\begin{aligned}\mathcal{L}(x, y, \dot{y}, \ddot{y}) &= 0 \\ \mathcal{L}(\omega, y, cy', c^2y'') &= 0 \\ \mathcal{L}(\omega, g, cg', c^2g'') &= 0 \\ \mathcal{L}(\omega, \boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{h}, c\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{h}', c^2\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{h}'') &= 0\end{aligned}$$

Equation (2.25) is the Newton method, and at this time, the initial guess for $\boldsymbol{\xi}$ can use the $\boldsymbol{\xi}$ obtained by solving the linear differential equation. Also, the Newton method is set to stop when the difference $(\boldsymbol{\xi}_{k+1} - \boldsymbol{\xi}_k)$ is less than a user-specified tolerance, $\epsilon$.

$$0 \approx \mathcal{L}_k + \frac{\mathrm{d}\mathcal{L}}{\mathrm{d}\boldsymbol{\xi}}\bigg|_k (\boldsymbol{\xi}_{k+1} - \boldsymbol{\xi}_k) \tag{2.25}$$

where,

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}\boldsymbol{\xi}} = \frac{\partial\mathcal{L}}{\partial y''} \cdot \frac{\partial y''}{\partial\boldsymbol{\xi}} + \frac{\partial\mathcal{L}}{\partial y'} \cdot \frac{\partial y'}{\partial\boldsymbol{\xi}} + \frac{\partial\mathcal{L}}{\partial y} \cdot \frac{\partial y}{\partial\boldsymbol{\xi}} \tag{2.26}$$

Equation (2.26) is an application of the multivariable chain rule. Here, $\mathcal{L}$ represents the residual form, $y$, $y'$, $y''$ represent the first and second derivatives of $y$ respectively, and $\boldsymbol{\xi}$ represents the parameters we are trying to optimize. This equation reflects the fact that $\mathcal{L}$ depends on $y$, $y'$, $y''$, and $y$, $y'$, $y''$ all depend on $\boldsymbol{\xi}$. Therefore, this equation shows a method for computing the total derivative of $\mathcal{L}$ with respect to $\boldsymbol{\xi}$.

$$A_k(\boldsymbol{\xi}_{k+1} - \boldsymbol{\xi}_k) \approx \boldsymbol{b}_k$$

If we transform Equation (2.25) for easier viewing, it becomes as above.
Here, $A_k = \dfrac{\mathrm{d}\mathcal{L}}{\mathrm{d}\boldsymbol{\xi}}$, $\boldsymbol{b}_k = -\mathcal{L}_k$. If we repeat the least squares until $\left|\boldsymbol{\xi}_{k+1} - \boldsymbol{\xi}_k\right|_2 < \epsilon$, we can obtain the solution.

### *2.3.1.6 System of 3 equations in 3 unknowns*

To solve $m$ equations containing $m$ variables through TFC, the following method is required. For example, to solve the Two-body problem, it needs to simultaneously solve three equations related to the Cartesian coordinates, namely $x$, $y$, and $z$. Equation (2.27) is the Two-body equation converted into a residual form.

$$\mathcal{L} = \ddot{\boldsymbol{r}}r^3 + \mu\boldsymbol{r} = 0 \tag{2.27}$$

When Equation (2.27) is solved into three equations for $x$, $y$, and $z$, it appears as follows.

$$\begin{aligned}
\mathcal{L}_1 &= \ddot{x}(x^2 + y^2 + z^2)^{3/2} + \mu x = 0 \\
\mathcal{L}_2 &= \ddot{y}(x^2 + y^2 + z^2)^{3/2} + \mu y = 0 \\
\mathcal{L}_3 &= \ddot{z}(x^2 + y^2 + z^2)^{3/2} + \mu z = 0
\end{aligned}$$

Also, each of the initial unknown constants can be represented as follows.

$$\boldsymbol{\xi}_0 = \begin{Bmatrix} \xi_{x0} \\ \xi_{y0} \\ \xi_{z0} \end{Bmatrix}$$

When represented as Equation (2.25), it appears as follows.

$$
\begin{aligned}
0 &\approx \mathcal{L}_1(\xi_{x_0}) + \left.\frac{\mathrm{d}\mathcal{L}_1}{\mathrm{d}\boldsymbol{\xi}}\right|_0 (\boldsymbol{\xi}_1 - \boldsymbol{\xi}_0) = \mathcal{L}_1(\xi_{x_0}) + \left\{ \frac{\partial \mathcal{L}_1}{\partial \xi_x} \quad \frac{\partial \mathcal{L}_1}{\partial \xi_y} \quad \frac{\partial \mathcal{L}_1}{\partial \xi_z} \right\}_0 (\boldsymbol{\xi}_1 - \boldsymbol{\xi}_0) \\
0 &\approx \mathcal{L}_2(\xi_{y_0}) + \left.\frac{\mathrm{d}\mathcal{L}_2}{\mathrm{d}\boldsymbol{\xi}}\right|_0 (\boldsymbol{\xi}_1 - \boldsymbol{\xi}_0) = \mathcal{L}_2(\xi_{y_0}) + \left\{ \frac{\partial \mathcal{L}_2}{\partial \xi_x} \quad \frac{\partial \mathcal{L}_2}{\partial \xi_y} \quad \frac{\partial \mathcal{L}_2}{\partial \xi_z} \right\}_0 (\boldsymbol{\xi}_1 - \boldsymbol{\xi}_0) \\
0 &\approx \mathcal{L}_3(\xi_{z_0}) + \left.\frac{\mathrm{d}\mathcal{L}_3}{\mathrm{d}\boldsymbol{\xi}}\right|_0 (\boldsymbol{\xi}_1 - \boldsymbol{\xi}_0) = \mathcal{L}_3(\xi_{z_0}) + \left\{ \frac{\partial \mathcal{L}_3}{\partial \xi_x} \quad \frac{\partial \mathcal{L}_3}{\partial \xi_y} \quad \frac{\partial \mathcal{L}_3}{\partial \xi_z} \right\}_0 (\boldsymbol{\xi}_1 - \boldsymbol{\xi}_0)
\end{aligned}
$$

Each of the above equations is included in the calculation formula of the Jacobian matrix,

$$
J_0(\boldsymbol{\xi}_1 - \boldsymbol{\xi}_0) = \begin{bmatrix} \frac{\partial \mathcal{L}_1}{\partial \xi_x} & \frac{\partial \mathcal{L}_1}{\partial \xi_y} & \frac{\partial \mathcal{L}_1}{\partial \xi_z} \\ \frac{\partial \mathcal{L}_2}{\partial \xi_x} & \frac{\partial \mathcal{L}_2}{\partial \xi_y} & \frac{\partial \mathcal{L}_2}{\partial \xi_z} \\ \frac{\partial \mathcal{L}_3}{\partial \xi_x} & \frac{\partial \mathcal{L}_3}{\partial \xi_y} & \frac{\partial \mathcal{L}_3}{\partial \xi_z} \end{bmatrix}_0 \begin{Bmatrix} \xi_{x_1} - \xi_{x_0} \\ \xi_{y_1} - \xi_{y_0} \\ \xi_{z_1} - \xi_{z_0} \end{Bmatrix} = - \begin{Bmatrix} \mathcal{L}_1 \\ \mathcal{L}_2 \\ \mathcal{L}_3 \end{Bmatrix}_0 = -\mathcal{L}_0
$$

And it can be represented using the method of least squares as shown below,

$$
J_0(\boldsymbol{\xi}_1 - \boldsymbol{\xi}_0) = -\mathcal{L}_0
$$

Using $\boldsymbol{\xi}_0$, the initial guess obtained from the process of solving a linear differential equation,

$$
\boldsymbol{\xi}_1 = \boldsymbol{\xi}_0 - J_0^{-1}\mathcal{L}_0
$$

The solution is obtained by iterating until it falls below the tolerance set by the user.

$$
\boldsymbol{\xi}_{k+1} = \boldsymbol{\xi}_k - J_k^{-1}\mathcal{L}_k
$$

### 2.3.2 Perturbed orbit propagation using TFC

*2.3.2.1 Solving unperturbed orbit propagation*

As previously noted, the effectiveness of TFC was clearly demonstrated in Johnston's study [14]. That research evaluated TFC's performance by comparing it with MATLAB's numerical propagator, `ode113`. But the study at hand takes a different approach, investigating how well TFC performs when compared with Python's `solve_ivp` function.

Python's `SciPy` library includes a helpful function, `solve_ivp`, which is an effective tool for solving ordinary differential equations (ODEs). It works by performing a process known as "integration" over a specified time period. To put it simply, `solve_ivp` uses special techniques to find an approximate solution to the ODEs. Typically, it employs the "RK45" method, which efficiently finds a solution without overly straining computer resources. However, the problem's specifics might demand the use of alternative methods like "RK23", "Radau", "BDF", and "LSODA".

Table 2.1 showcases all the different methods compatible with the `solve_ivp` function in Python's `SciPy` library. It's essential to note that the chosen method should match the problem's specific requirements, like the required accuracy of the solution, available computational resources, and whether the system is "stiff" or "non-stiff".

This study is not simply a reiteration of what's been done before but instead expands the scope of understanding of TFC by testing it in a novel way. This experiment isn't only aimed at gaining deeper insights about TFC, but also gauging its performance against a diverse range of tools. The rationale behind choosing `solve_ivp` for this comparison is because of its widespread use in Python for solving ODEs. Comparing TFC straight up with `solve_ivp` can help us understand how TFC works better and show us where it might do a better job than the usual methods.

Furthermore, this broader perspective on TFC offers a chance to more fully appreciate its potential and applicability. It's not just about proving that TFC can hold its own against established techniques, but also about exploring where and how it can offer new possibilities and advantages.

The evaluation provides an opportunity to potentially uncover unique strengths and promising avenues for future exploration and application of TFC.

| Method | Description |
|---|---|
| RK45 | Default method. It's a Runge-Kutta method of order 5(4) and a good all-around choice for most problems. |
| RK23 | This is also a Runge-Kutta method but of order 3(2). It's less accurate and requires fewer computational resources than "RK45". |
| Radau | This is an implicit Runge-Kutta method of the Radau IIA family of order 5. It's best suited for stiff systems, and performs well on problems requiring high accuracy. |
| BDF | This method is an implicit multi-step method suited for stiff systems, especially when the problem involves long-term integration. |
| LSODA | This method switches between "stiff" and "non-stiff" solvers as needed, making it versatile for a variety of problems. "LSODA" stands for "Livermore Solver for Ordinary Differential equations with Automatic method switching". |

Table 2.1: Methods of "solve_ivp" function in Python's SciPy library

In this section, the performance of TFC and solve_ivp is compared using unperturbed orbit propagation. Under perturbation conditions, a true solution does not exist, thus making it impossible to compare the performance of these two tools. However, under unperturbed conditions, comparison becomes possible. This is due to the theoretical identical nature of the initial satellite data and the data following one full orbit rotation.

### Monte Carlo test setting

This experiment is about propagating the Two-body problem equation, using both TFC and `solve_ivp` methods.

$$\ddot{\boldsymbol{r}}(t) = -\frac{\mu}{r^3(t)}\boldsymbol{r}(t) \qquad \text{subject to :} \quad \begin{cases} \boldsymbol{r}(t_0) = \boldsymbol{r}_0 \\ \\ \boldsymbol{v}(t_0) = \boldsymbol{v}_0 \end{cases} \tag{2.28}$$

The accuracy of each technique is determined by comparing the initial position & velocity vector with the final position & velocity vector after propagating the satellite for one orbit.

$$\text{Position Error} \quad = \quad \left|\boldsymbol{r}_0 - \hat{\boldsymbol{r}}_f\right|_2 \tag{2.29}$$

$$\text{Velocity Error} \quad = \quad \left|\boldsymbol{v}_0 - \hat{\boldsymbol{v}}_f\right|_2 \tag{2.30}$$

Additionally, a Monte Carlo test is performed, which entails carrying out 10,000 tests and examining the outcomes. For this task, 10,000 sets of orbital elements are randomly generated based on the conditions outlined in Table 2.2. Essentially, this means creating 10,000 orbits at random.

| Orbit elements | Range |
|---|---|
| $\boldsymbol{r}_p$ : Perigee (km) | $R_E + 500$ ; $R_E + 5000$ |
| $\boldsymbol{r}_a$ : Apogee (km) | $\boldsymbol{r}_p + 10$ ; $\boldsymbol{r}_p + 5000$ |
| $\boldsymbol{i}$ : Inclination (rad) | $0$ ; $\pi$ |
| $\boldsymbol{\omega}$ : Argument of periapsis (rad) | $0$ ; $2\pi$ |
| $\boldsymbol{\Omega}$ : Right ascension of the ascending node (rad) | $0$ ; $2\pi$ |
| $\boldsymbol{\phi}$ : True anomaly (rad) | $0$ ; $2\pi$ |

Table 2.2: Randomly generated orbit elements

### Accuracy results

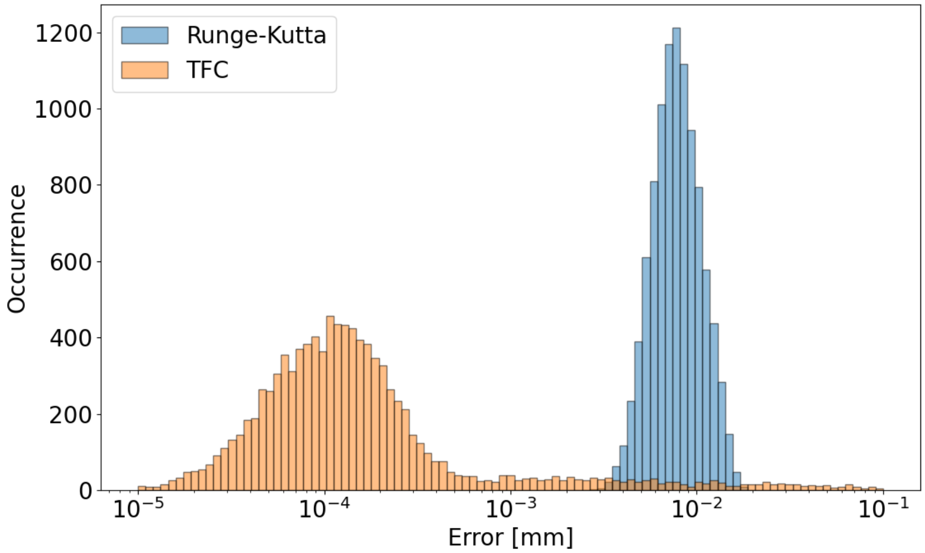

Figure 2.6: Position Error

Looking at Figure 2.6, the $x$-axis is the error between the initial position of the orbit and the position after one orbit, measured in millimeters. The $y$-axis is the occurrence of each result. The orange histogram bars present the outcomes for TFC, while the blue bars depict results from `solve_ivp`. Overall, it looks like TFC is a lot more accurate than `solve_ivp`. However, it also shows a wider spread, and a very small number of tests even show less accuracy. The reason is that Runge-kutta (RK45) is an integrator with variable (optimized) step while TFC is not. TFC can also distribute the discretization points in an optimized way, that is, by following the change in dynamics. This has not been implemented in these Monte Carlo tests. Nevertheless, TFC seems to be a lot more accurate than `solve_ivp`, which mostly uses the Runge_Kutta method. If TFC were to use an optimized way similar to `solve_ivp`, it could probably get even better results.

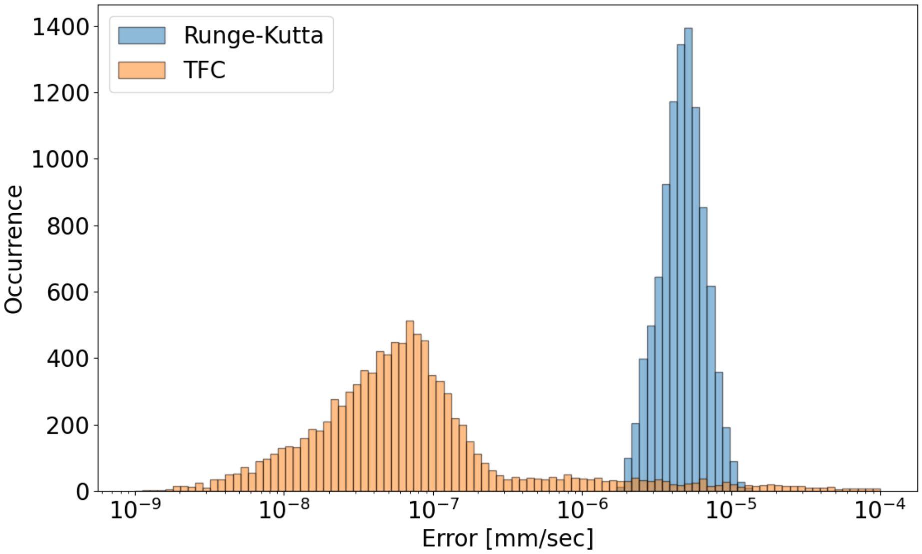

Figure 2.7: Velocity Error

In Figure 2.7, the $x$-axis shows the difference between the initial and final velocity of the orbit, measured in millimeters per second. The $y$-axis counts how often each result happens. Orange bars in the graph show TFC's results, and blue bars show results from `solve_ivp`. Generally, TFC is more exact than `solve_ivp`, similar to what we saw in Figure 2.6. Likewise, there's more variety in TFC's results, and a tiny number of tests are less exact. Another interesting point is that each operator shows about 1000 times more accuracy compared to the results in Figure 2.6. This is because the velocity vector's values are much lower than those of the position vector. This could suggest that position vectors with lower numerical values (like Low Earth Orbit satellites) may have higher accuracy than those with higher values (like Geostationary Orbit satellites). The wide spread of TFC results is due to differences in each satellite orbit's eccentricity, suggesting that accuracy might decrease as eccentricity increases.

### Computational results

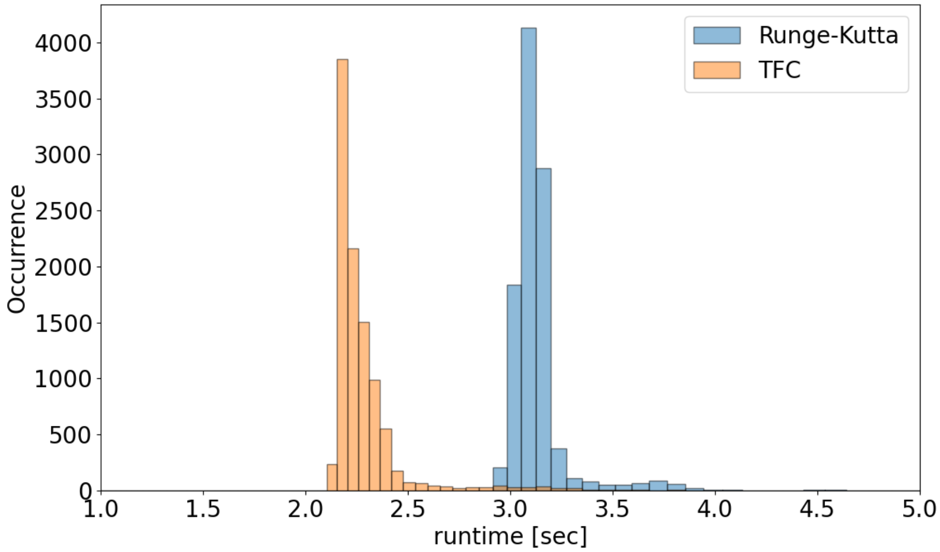

Figure 2.8: Runtime comparison

Figure 2.8 shows the runtime results for the experiment above. The $x$-axis shows the time it takes to propagate one sample, or one orbit, and the $y$-axis shows the occurrence. According to the results, it takes an average of about 2 seconds for TFC to propagate one orbit, while it takes about 3 seconds for `solve_ivp`. This means there's a speed difference of about 1.5 times. This speed difference is clearly noticeable in a Monte Carlo test that runs 10,000 times. In other words, TFC is much faster than `solve_ivp`, and this advantage is clearly seen, especially in environments where many tests need to be run.

#### *2.3.2.2 Differential equation for perturbed orbit propagation*

The differential equation for perturbed orbit propagation is typically formulated as:

$$\ddot{\boldsymbol{r}}(t) = -\frac{\mu}{r^3(t)}\boldsymbol{r}(t) + \boldsymbol{a}_p(\boldsymbol{r}, \boldsymbol{r}_m) \qquad \text{subject to :} \quad \begin{cases} \boldsymbol{r}(t_0) = \boldsymbol{r}_0 \\ \\ \dot{\boldsymbol{r}}(t_0) = \dot{\boldsymbol{r}}_0 \end{cases} \tag{2.31}$$

where $\boldsymbol{a}_p(\boldsymbol{r}, \boldsymbol{r}_m)$ is the perturbing acceleration. For instance, using the third body perturbation model for the Moon described by Vallado [19].

$$\boldsymbol{a}_p = -\frac{\mu_m}{r_m^3}\left[\boldsymbol{r} - \left[3\frac{\boldsymbol{r}^{\mathrm{T}}\boldsymbol{r}_m}{r_m^2} - \frac{15}{2}\left(\frac{\boldsymbol{r}^{\mathrm{T}}\boldsymbol{r}_m}{r_m^2}\right)^2\right]\boldsymbol{r}_m\right]$$

#### *2.3.2.3 Deriving the constrained expressions*

To get the IVP constrained expression for an orbit propagation that has to follow $r_i(0) = r_{0_i}$ and $v_i(0) = v_{0_i}$ ( for $i = 1, 2, 3$ ), this formula is used:

$$r_i(t) = g_i(t) + \eta_{1_i} + \eta_{2_i} t$$

Applying these constraints forms a set of equations which can be presented as follows:

$$\begin{Bmatrix} r_{0_i} - g_{0_i} \\ \\ v_{0_i} - \dot{g}_{0_i} \end{Bmatrix} = \begin{bmatrix} 1 & 0 \\ \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} \eta_{1_i} \\ \\ \eta_{2_i} \end{Bmatrix}$$

The solution for $\eta$ coefficients is found from these equations, which gives the final form of the constrained expression:

$$r_i(t) = g_i(t) + r_{0_i} - g_{0_i} + (v_{0_i} - \dot{g}_{0_i})t$$

By defining the free function $g_i(t)$ as basis function,

$$g_i(t) = \boldsymbol{\xi}_i^{\mathrm{T}} \boldsymbol{h}(x(t))$$

The constrained expression and its derivatives as functions of $\boldsymbol{\xi}_i$ are,

$$\begin{aligned}
r_i(t) &= \boldsymbol{\xi}_i^{\mathrm{T}}(\boldsymbol{h} - \boldsymbol{h}_0 - t\dot{\boldsymbol{h}}_0) + r_{0_i} + tv_{0_i} \\
v_i(t) &= \boldsymbol{\xi}_i^{\mathrm{T}}(\dot{\boldsymbol{h}} - \dot{\boldsymbol{h}}_0) + v_{0_i} \\
a_i(t) &= \boldsymbol{\xi}_i^{\mathrm{T}}\ddot{\boldsymbol{h}}
\end{aligned}$$

*2.3.2.4 Perturbed orbit propagation using the TFC method*

The process of incorporating the problem's constraints into a constrained expression allows transforming the main differential equation provided by Equation (2.31) into a residual form. This residual form comes in the following form where,

$$\mathcal{L}_i = \left(\sum_{j=1}^{3} r_j^2\right)^{3/2} (\ddot{r}_i - a_{p_i}) + \mu r_i = 0, \quad i = 1, 2, 3. \tag{2.32}$$

The Jacobian is expressed as follows using the given residual form.

$$\frac{\partial \mathcal{L}_i}{\partial \boldsymbol{\xi}_j} = \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\xi}_i}\left(\sum_{j=1}^{3} r_j^2\right)^{3/2} (\ddot{r}_i - a_{p_i}) + \left(\sum_{j=1}^{3} r_j^2\right)^{3/2} \left[-\frac{\mathrm{d}a_{p_i}}{\mathrm{d}\boldsymbol{\xi}_j}\right]$$

After the domain has been discretized, the Jacobian terms combine to form $N \times m$ matrices, where $N$ is the total number of points and $m$ is the total number of basis functions. The $\mathcal{L}_i$ terms are $N \times 1$. Finally, by combining these terms into an enhanced matrix and the residual forms into an augmented vector,

$$J = \begin{bmatrix} \frac{\partial \mathcal{L}_1}{\partial \boldsymbol{\xi}_1} & \frac{\partial \mathcal{L}_1}{\partial \boldsymbol{\xi}_2} & \frac{\partial \mathcal{L}_1}{\partial \boldsymbol{\xi}_3} \\ \frac{\partial \mathcal{L}_2}{\partial \boldsymbol{\xi}_1} & \frac{\partial \mathcal{L}_2}{\partial \boldsymbol{\xi}_2} & \frac{\partial \mathcal{L}_2}{\partial \boldsymbol{\xi}_3} \\ \frac{\partial \mathcal{L}_3}{\partial \boldsymbol{\xi}_1} & \frac{\partial \mathcal{L}_3}{\partial \boldsymbol{\xi}_2} & \frac{\partial \mathcal{L}_3}{\partial \boldsymbol{\xi}_3} \end{bmatrix}_{(3N \times 3m)} , \quad \mathcal{L} = \begin{Bmatrix} \mathcal{L}_1 \\ \mathcal{L}_2 \\ \mathcal{L}_3 \end{Bmatrix}_{(3N \times 1)}$$

Using a nonlinear least squares approach, the update equation for the $\boldsymbol{\xi}_i$ terms is provided by,

$$\begin{Bmatrix} \boldsymbol{\xi}_1 \\ \boldsymbol{\xi}_2 \\ \boldsymbol{\xi}_3 \end{Bmatrix}_{k+1} = \begin{Bmatrix} \boldsymbol{\xi}_1 \\ \boldsymbol{\xi}_2 \\ \boldsymbol{\xi}_3 \end{Bmatrix}_{k} - \left(J_k^{\mathrm{T}} J_k\right)^{-1} J_k^{\mathrm{T}} \mathcal{L}_k$$

where $k$ defines the current step in the nonlinear least squares method. This section details the process of solving the differential equation associated with perturbed orbit propagation. What's left is to find the initial guess, $\boldsymbol{\xi}_0$. At this point, initial guess depends on the problem, but there is no problem as it converges even if it's applied 0.

### 2.3.3 Perturbed multi revolution Lambert problem using TFC

The orbit propagation mentioned above is an Initial Value Problem (IVP) that solves orbital differential equation with initial conditions. On the other hand, the Lambert problem is a way of finding the solution of the fundamental orbital differential equation given the position vectors $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ at different times $t_1$ and $t_2$. That is, the Lambert problem is a Boundary Value Problem (BVP) that solves differential equations including boundary conditions. The Lambert problem is basically a problem related to two position vectors and orbital flight time, which can be interpreted and used in three ways as follows.

1. orbit determination : position vectors $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ and flight time $\Delta T$ are interpreted as observations of an unknown spacecraft and used to determine the orbit of the spacecraft,

2. orbit transfer : the position vector $\boldsymbol{r}_1$ is used to interpret the current orbit, $\boldsymbol{r}_2$ as the position in the final orbit, and the flight time $\Delta T$ as variables, and to change the orbit of the spacecraft,

3. rendezvous : the position vector $\boldsymbol{r}_1$ is interpreted as the current position of the interceptor, $\boldsymbol{r}_2$ as the position of the target spacecraft, and $\Delta T$ as the flight time, and is used for intercept or rendezvous problems.

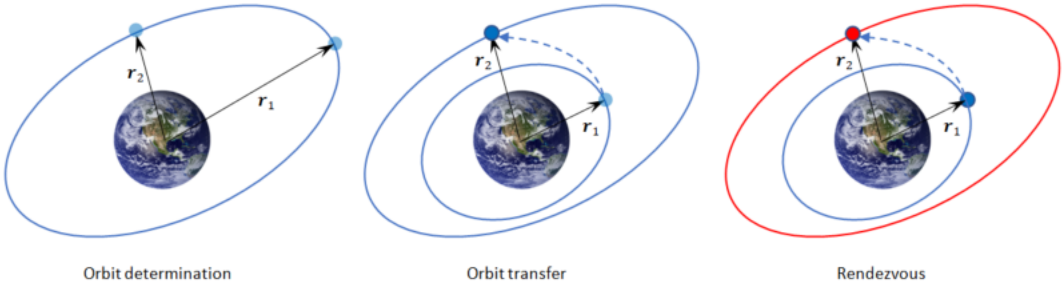

Figure 2.9: Lambert problem to solve 3 distinct problems

*2.3.3.1 Deriving the constrained expressions*

The perturbed Lambert problem consists of the following Boundary Value Problme (BVP):

$$\ddot{\boldsymbol{r}} = -\frac{\mu}{r^3}\boldsymbol{r} + \boldsymbol{a}_p(\boldsymbol{r}, \dot{\boldsymbol{r}}) \qquad \text{subject to :} \quad \begin{cases} \boldsymbol{r}(0) = \boldsymbol{r}_0 \\ \boldsymbol{r}(\Delta T) = \boldsymbol{r}_f \end{cases} \tag{2.33}$$

In this equation, $\boldsymbol{a}_p$ is the perturbed acceleration, $\boldsymbol{r}_0$ and $\boldsymbol{r}_f$ are the initial and final position vectors of the Lambert arc, and $\Delta T$ indicates the Time of Flight (ToF). Based on Criscola's study [15], the TFC is utilized to resolve the Lambert problem by expressing the position vector, $\boldsymbol{r}(t)$, with a model that overcomes the Cartesian coordinates issues with these variables: $p(t)$, $\vartheta(t)$, and $h(t)$. They define the spacecraft's position in the Lambert arc as a function of time, as follows:

$$\boldsymbol{r}(t) = p(t)[\cos\vartheta(t)\hat{\boldsymbol{r}}_0 + \sin\vartheta(t)\hat{\boldsymbol{t}}_0] + h(t)\hat{\boldsymbol{h}}_0 \tag{2.34}$$

where,

$$\hat{\boldsymbol{r}}_0 = \frac{\boldsymbol{r}_0}{|\boldsymbol{r}_0|}, \quad \hat{\boldsymbol{r}}_f = \frac{\boldsymbol{r}_f}{|\boldsymbol{r}_f|}, \quad \hat{\boldsymbol{h}}_0 = \frac{\hat{\boldsymbol{r}}_0 \times \hat{\boldsymbol{r}}_f}{|\hat{\boldsymbol{r}}_0 \times \hat{\boldsymbol{r}}_f|}, \quad \text{and} \quad \hat{\boldsymbol{t}}_0 = \hat{\boldsymbol{h}}_0 \times \hat{\boldsymbol{r}}_0. \tag{2.35}$$

Specifically, [$\hat{\boldsymbol{r}}_0$,$\hat{\boldsymbol{t}}_0$,$\hat{\boldsymbol{h}}_0$] form the three directions of an orthogonal reference frame which is defined as long as the cross product $\hat{\boldsymbol{r}}_0 \times \hat{\boldsymbol{r}}_f$ exists, i.e., $\hat{\boldsymbol{r}}_0$ and $\hat{\boldsymbol{r}}_f$ are not parallel. The variable $p(t)$ is the projection of the radius vector on the [$\hat{\boldsymbol{r}}_0$,$\hat{\boldsymbol{t}}_0$] plane, $\vartheta(t)$ is a parametric angle with no inherent physical meaning (besides satisfying the bounds), and $h(t)$ denotes the orthogonal projection of the position vector with respect to the [$\hat{\boldsymbol{r}}_0$,$\hat{\boldsymbol{t}}_0$] plane. This formal representation to represent the evolution of the radius vector is particularly apt when adopted by TFC. The first two derivatives of

$\boldsymbol{r}(t)$ are:

$$\dot{\boldsymbol{r}}(t) = [\dot{p}\cos\vartheta - p\dot{\vartheta}\sin\vartheta]\hat{\boldsymbol{r}}_0 + [\dot{p}\sin\vartheta + p\dot{\vartheta}\cos\vartheta]\hat{\boldsymbol{t}}_0 + \dot{h}\hat{\boldsymbol{h}}_0 \tag{2.36}$$

$$\ddot{\boldsymbol{r}}(t) = [(\ddot{p} - p\dot{\vartheta}^2)\cos\vartheta - (2\dot{p}\dot{\vartheta} + p\ddot{\vartheta})\sin\vartheta]\hat{\boldsymbol{r}}_0 + [(\ddot{p} - p\dot{\vartheta}^2)\sin\vartheta + (2\dot{p}\dot{\vartheta} + p\ddot{\vartheta})\cos\vartheta]\hat{\boldsymbol{t}}_0 + \ddot{h}\hat{\boldsymbol{h}}_0 \tag{2.37}$$

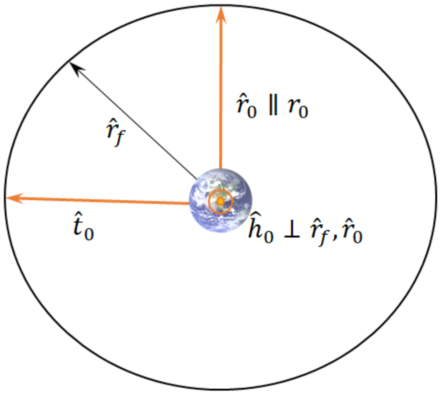
$\hat{r}_0 \parallel r_0$
$\hat{r}_f$
$\hat{t}_0$
$\hat{h}_0 \perp \hat{r}_f, \hat{r}_0$

Figure 2.10: Three directions of an orthogonal reference frame

As a result, the position, velocity, and acceleration of the spacecraft are determined solely with the three functionals across the entirety of the Lambert arc. The boundary constraints of the variables are,

$$\begin{cases} p(0) = r_0, \\ \vartheta(0) = 0 \\ h(0) = 0 \end{cases} \quad \text{and} \quad \begin{cases} p(\Delta T) = r_f \\ \vartheta(\Delta T) = \vartheta_r + 2k\pi \\ h(\Delta T) = 0 \end{cases} \tag{2.38}$$

where,

$$\hat{\boldsymbol{r}}_0^{\mathrm{T}}\hat{\boldsymbol{r}}_f = \cos\vartheta_r, \quad k : \text{the number of revolutions} \tag{2.39}$$

Additionally, the three functionals, $p(t)$, $\vartheta(t)$, and $h(t)$, oscillate with the following approximate mean frequency :

$$\omega \approx \frac{2k\pi + \vartheta_r}{\Delta T} \tag{2.40}$$

These expected orbital oscillations are captured / absorbed by including in a vector of basis functions the following additional term:

$$b(t) = A\cos\omega t + B\sin\omega t \tag{2.41}$$

$$\boldsymbol{s}(t) = \begin{Bmatrix} \boldsymbol{h}(z) \\ \cos\omega t \\ \sin\omega t \end{Bmatrix}^{\mathrm{T}} \tag{2.42}$$

where $\boldsymbol{s}(z)$ is the basis functions vector containing $\boldsymbol{h}(z)$, which corresponds to a set of orthogonal polynomials of degree $m$, and the harmonic function presented in Equation (2.41). The vector $\boldsymbol{h}(z)$ can be a set of polynomials, e.g. Legendre, Chebyshev, etc. This study chooses Legendre orthogonal polynomials to represent the basis function. The vector in Equation (2.42) is used to define a trio of free functions $g_i(z)$, which are in turn used to form the functionals. Necessarily, there must be three independent free functions to represent $p(t)$, $\vartheta(t)$, and $h(t)$. These three functions, $g_p(z)$, $g_\vartheta(z)$, and $g_h(z)$, are expressed as follows :

$$g_p(z) = \boldsymbol{\xi}_p^{\mathrm{T}}\boldsymbol{s}(z), \quad g_\vartheta(z) = \boldsymbol{\xi}_\vartheta^{\mathrm{T}}\boldsymbol{s}(z), \quad \text{and} \quad g_h(z) = \boldsymbol{\xi}_h^{\mathrm{T}}\boldsymbol{s}(z). \tag{2.43}$$

and,

$$\begin{cases} \dot{g}_p = \boldsymbol{\xi}_p^{\mathrm{T}}\dot{\boldsymbol{s}}, \quad \dot{g}_\vartheta = \boldsymbol{\xi}_\vartheta^{\mathrm{T}}\dot{\boldsymbol{s}}, \quad \dot{g}_h = \boldsymbol{\xi}_h^{\mathrm{T}}\dot{\boldsymbol{s}} \\ \ddot{g}_p = \boldsymbol{\xi}_p^{\mathrm{T}}\ddot{\boldsymbol{s}}, \quad \ddot{g}_\vartheta = \boldsymbol{\xi}_\vartheta^{\mathrm{T}}\ddot{\boldsymbol{s}}, \quad \ddot{g}_h = \boldsymbol{\xi}_h^{\mathrm{T}}\ddot{\boldsymbol{s}} \end{cases} \tag{2.44}$$

where,

$$\dot{\boldsymbol{s}} = \begin{Bmatrix} c\boldsymbol{h}' \\ -\omega\sin(\omega t) \\ \omega\cos(\omega t) \end{Bmatrix}^{\mathrm{T}} \quad \text{and} \quad \ddot{\boldsymbol{s}} = \begin{Bmatrix} c^2\boldsymbol{h}'' \\ -\omega^2\cos(\omega t) \\ \omega^2\sin(\omega t) \end{Bmatrix}^{\mathrm{T}} \tag{2.45}$$

Note that $\boldsymbol{\xi}_p$, $\boldsymbol{\xi}_\vartheta$, and $\boldsymbol{\xi}_h$ are the three unknown coefficient vectors associated to $p(t)$, $\vartheta(t)$, and $h(t)$. The variable $z$ corresponds to the domain basis functions, represented in the form of collocation points due to the presence of the orthogonal polynomials in $\boldsymbol{h}(z)$. A linear mapping is introduced between $z$ and the time $t$,

$$z(t) = \frac{2}{\Delta T}t - 1 \in [-1, +1] \quad \rightarrow \quad t(z) = \frac{\Delta T}{2}(z+1) \in [0, \Delta T] \tag{2.46}$$

from which a mapping constant $c$ is determined as follows:

$$\dot{g} = \frac{\mathrm{d}g}{\mathrm{d}t} = \frac{\mathrm{d}g}{\mathrm{d}z}\cdot\frac{\mathrm{d}z}{\mathrm{d}t} = g'c = g'\frac{2}{\Delta T} \qquad \text{and} \qquad \frac{\mathrm{d}^k g}{\mathrm{d}t^k} = \frac{\mathrm{d}^k g}{\mathrm{d}z^k}c^k \tag{2.47}$$

In order to carry out this work, the approach takes advantage of the three constrained expressions $p(t)$, $\vartheta(t)$, and $h(t)$. These are functionals that always satisfy the boundary constraints given in Equation (2.38). They represent the whole set of functions satisfying the boundary constraints. This is obtained by spanning all possible expressions of the free functions. The bounds given in

Equation (2.38) allow TFC to derive the following expressions :

$$\begin{cases} p(t, g_p) & = g_p(z) + \dfrac{\Delta T - t}{\Delta T}[p_0 - g_p(-1)] + \dfrac{t}{\Delta T}[p(\Delta T) - g_p(+1)] \\ \vartheta(t, g_\vartheta) & = g_\vartheta(z) - \dfrac{\Delta T - t}{\Delta T} g_\vartheta(-1) + \dfrac{t}{\Delta T}[\vartheta_r + 2k\pi - g_\vartheta(+1)] \\ h(t, g_h) & = g_h(z) - \dfrac{\Delta T - t}{\Delta T} g_h(-1) - \dfrac{t}{\Delta T} g_h(+1) \end{cases} \tag{2.48}$$

whose time derivatives, setting $c = 2 \,/\, \Delta T$, are,

$$\begin{cases} \dot{p}(t, g_p) & = c g_p'(z) + \dfrac{1}{\Delta T}[p_0 - g_p(-1)] + \dfrac{1}{\Delta T}[p(\Delta T) - g_p(+1)] \\ \dot{\vartheta}(t, g_\vartheta) & = c g_\vartheta'(z) + \dfrac{1}{\Delta T} g_\vartheta(-1) + \dfrac{1}{\Delta T}[\vartheta_r + 2k\pi - g_\vartheta(+1)] \\ \dot{h}(t, g_h) & = c g_h'(z) + \dfrac{1}{\Delta T} g_h(-1) - \dfrac{1}{\Delta T} g_h(+1) \end{cases} \tag{2.49}$$

where $g_p'(z)$, $g_\vartheta'(z)$, and $g_h'(z)$, indicate the first derivatives with respect to the basis functions variable, $z$. The second derivativeds are,

$$\ddot{p}(t, g_p) = c^2 g_p''(z), \quad \ddot{\vartheta}(t, g_\vartheta) = c^2 g_\vartheta''(z), \quad \text{and} \quad \ddot{h}(t, g_h) = c^2 g_h''(z) \tag{2.50}$$

where $\dot{f}$=$\dfrac{\mathrm{d}f}{\mathrm{d}t}$ and $f'$=$\dfrac{\mathrm{d}f}{\mathrm{d}z}$. It is important to remove the constant and linear terms in $\boldsymbol{h}(z)$ given that they have been already used to derive the constrained expressions in the above equations. The set of Equation (2.48) constitute the TFC constrained expressions. It is trivial to verify that thesse equations satisfy the boundary constraints for any expression of the free functions $g_p(z)$, $g_\vartheta(z)$, and $g_h(z)$. Section 2.3.1 describes the theory to derive the constrained expressions associated to a given set of constraints. These constraints are defined in terms of points, derivatives, integrals, limits, and any linear combination of them in uni-variate and multi-variate cases. The complete formulation has now been developed, and so the goal becomes to solve for $\boldsymbol{\xi}_p$, $\boldsymbol{\xi}_\vartheta$, and $\boldsymbol{\xi}_h$. The problem is an iterative one that is solved via least-squares.

#### *2.3.3.2 Nonlinear least-squares solution*

The solution of the Lambert problem is represented by the trajectory described by $\boldsymbol{r}(t)$ satisfying the BVP defined by $\boldsymbol{r}(0) = \boldsymbol{r}_0$ and $\boldsymbol{r}(T) = \boldsymbol{r}_f$, of the differential equation,

$$\mathcal{L} = \ddot{\boldsymbol{r}} + \frac{\mu}{r^3}\boldsymbol{r} - \boldsymbol{a}_p(\boldsymbol{r}, \dot{\boldsymbol{r}}) = 0 \tag{2.51}$$

Using Equation (2.34), (2.36), (2.37), and Equation (2.48) - (2.50), this problem is solved by estimating the unknown coefficients vectors, $\boldsymbol{\xi}_p$, $\boldsymbol{\xi}_\vartheta$, and $\boldsymbol{\xi}_h$, by nonlinear least-squares. Therefore, linearization around an estimated solution is obtained as follows for every coordinate

$$\begin{cases} 0 \approx \mathcal{L}_{xk} + \left[\dfrac{\partial \mathcal{L}_x}{\partial \boldsymbol{\xi}_p}, \dfrac{\partial \mathcal{L}_x}{\partial \boldsymbol{\xi}_\vartheta}, \dfrac{\partial \mathcal{L}_x}{\partial \boldsymbol{\xi}_h}\right]_k \begin{Bmatrix} \boldsymbol{\xi}_p \\ \boldsymbol{\xi}_\vartheta \\ \boldsymbol{\xi}_h \end{Bmatrix}_k = \mathcal{L}_{xk} + J_{xk}\boldsymbol{\xi}_k \\ 0 \approx \mathcal{L}_{yk} + \left[\dfrac{\partial \mathcal{L}_y}{\partial \boldsymbol{\xi}_p}, \dfrac{\partial \mathcal{L}_y}{\partial \boldsymbol{\xi}_\vartheta}, \dfrac{\partial \mathcal{L}_y}{\partial \boldsymbol{\xi}_h}\right]_k \begin{Bmatrix} \boldsymbol{\xi}_p \\ \boldsymbol{\xi}_\vartheta \\ \boldsymbol{\xi}_h \end{Bmatrix}_k = \mathcal{L}_{yk} + J_{xk}\boldsymbol{\xi}_k \\ 0 \approx \mathcal{L}_{zk} + \left[\dfrac{\partial \mathcal{L}_z}{\partial \boldsymbol{\xi}_p}, \dfrac{\partial \mathcal{L}_z}{\partial \boldsymbol{\xi}_\vartheta}, \dfrac{\partial \mathcal{L}_z}{\partial \boldsymbol{\xi}_h}\right]_k \begin{Bmatrix} \boldsymbol{\xi}_p \\ \boldsymbol{\xi}_\vartheta \\ \boldsymbol{\xi}_h \end{Bmatrix}_k = \mathcal{L}_{zk} + J_{xk}\boldsymbol{\xi}_k \end{cases} \tag{2.52}$$

from which the nonlinear least-squares solution is solved via an iterative process :

$$\boldsymbol{\xi}_{k+1} = \boldsymbol{\xi}_k - \left(J_k^{\mathrm{T}} J_k\right)^{-1} J_k^{\mathrm{T}} \mathcal{L}_k \tag{2.53}$$

Note that, in the above formulation, $k$ represents the current iteration, $\boldsymbol{\xi}_k$ is a vector that includes $\boldsymbol{\xi}_p$, $\boldsymbol{\xi}_\vartheta$, and $\boldsymbol{\xi}_h$, and $J_k$ is the Jacobian matrix of the system. Such a matrix requires the evaluation of the partial derivatives with respect to the three unknown vectors of coefficients. Starting with the partial derivatives of the functionals with respect to $\boldsymbol{\xi}_p$, $\boldsymbol{\xi}_\vartheta$, and $\boldsymbol{\xi}_h$,

$$\frac{\partial p}{\partial \boldsymbol{\xi}_p} = \frac{\partial \vartheta}{\partial \boldsymbol{\xi}_\vartheta} = \frac{\partial h}{\partial \boldsymbol{\xi}_h} = \boldsymbol{s} - \frac{\Delta T - t}{\Delta T}\boldsymbol{s}_0 - \frac{t}{\Delta T}\boldsymbol{s}_f \tag{2.54}$$

$$\frac{\partial \dot{p}}{\partial \boldsymbol{\xi}_p} = \frac{\partial \dot{\vartheta}}{\partial \boldsymbol{\xi}_\vartheta} = \frac{\partial \dot{h}}{\partial \boldsymbol{\xi}_h} = \dot{\boldsymbol{s}} + \frac{\boldsymbol{s}_0}{\Delta T} - \frac{\boldsymbol{s}_f}{\Delta T} \tag{2.55}$$

$$\frac{\partial \ddot{p}}{\partial \boldsymbol{\xi}_p} = \frac{\partial \ddot{\vartheta}}{\partial \boldsymbol{\xi}_\vartheta} = \frac{\partial \ddot{h}}{\partial \boldsymbol{\xi}_h} = \ddot{\boldsymbol{s}} \tag{2.56}$$

which are used in the derivatives of $r$ and $\boldsymbol{r}$,

$$\frac{\partial r}{\partial \boldsymbol{\xi}_p} = \frac{p}{\sqrt{p^2+h^2}}\frac{\partial p}{\partial \boldsymbol{\xi}_p} \tag{2.57}$$

$$\frac{\partial r}{\partial \boldsymbol{\xi}_h} = \frac{h}{\sqrt{p^2+h^2}}\frac{\partial h}{\partial \boldsymbol{\xi}_p} \tag{2.58}$$

$$\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\xi}_p} = \frac{\partial p}{\partial \boldsymbol{\xi}_p}(\hat{\boldsymbol{r}}_0 \cos\vartheta + \hat{\boldsymbol{t}}_0 \sin\vartheta) \tag{2.59}$$

$$\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\xi}_\vartheta} = p\frac{\partial \vartheta}{\partial \boldsymbol{\xi}_\vartheta}(-\hat{\boldsymbol{r}}_0 \sin\vartheta + \hat{\boldsymbol{t}}_0 \cos\vartheta) \tag{2.60}$$

$$\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\xi}_h} = \frac{\partial h}{\partial \boldsymbol{\xi}_h}\hat{\boldsymbol{h}}_0 \tag{2.61}$$

Finally, the partials that correspond to the acceleration of the spaceraft are :

$$\frac{\partial \ddot{\boldsymbol{r}}}{\partial \boldsymbol{\xi}_p} = \left[\left(\frac{\partial \ddot{p}}{\partial \boldsymbol{\xi}_p} - \frac{\partial p}{\partial \boldsymbol{\xi}_p}\dot{\vartheta}^2\right)\cos\vartheta - \left(2\frac{\partial \dot{p}}{\partial \boldsymbol{\xi}_p}\dot{\vartheta} + \frac{\partial p}{\partial \boldsymbol{\xi}_p}\ddot{\vartheta}\right)\sin\vartheta\right]\hat{\boldsymbol{r}}_0 \tag{2.62}$$

$$+\left[\left(\frac{\partial \ddot{p}}{\partial \boldsymbol{\xi}_p} - \frac{\partial p}{\partial \boldsymbol{\xi}_p}\dot{\vartheta}^2\right)\sin\vartheta - \left(2\frac{\partial \dot{p}}{\partial \boldsymbol{\xi}_p}\dot{\vartheta} + \frac{\partial p}{\partial \boldsymbol{\xi}_p}\ddot{\vartheta}\right)\cos\vartheta\right]\hat{\boldsymbol{t}}_0 \tag{2.63}$$

$$\frac{\partial \ddot{\boldsymbol{r}}}{\partial \boldsymbol{\xi}_\vartheta} = \left[-2p\dot{\vartheta}\frac{\partial \dot{\vartheta}}{\partial \boldsymbol{\xi}_\vartheta}\cos\vartheta - \left(2\dot{p}\frac{\partial \dot{\vartheta}}{\partial \boldsymbol{\xi}_\vartheta} + p\frac{\partial \ddot{\vartheta}}{\partial \boldsymbol{\xi}_\vartheta}\right)\sin\vartheta\right]\hat{\boldsymbol{r}}_0 \tag{2.64}$$

$$+\left[-2p\dot{\vartheta}\frac{\partial \dot{\vartheta}}{\partial \boldsymbol{\xi}_\vartheta}\sin\vartheta - \left(2\dot{p}\frac{\partial \dot{\vartheta}}{\partial \boldsymbol{\xi}_\vartheta} + p\frac{\partial \ddot{\vartheta}}{\partial \boldsymbol{\xi}_\vartheta}\right)\cos\vartheta\right]\hat{\boldsymbol{t}}_0 \tag{2.65}$$

$$-\frac{\partial \vartheta}{\partial \boldsymbol{\xi}_\vartheta}\left[\left(\ddot{p} - p\dot{\vartheta}^2\right)\sin\vartheta + \left(2\dot{p}\dot{\vartheta} + p\ddot{\vartheta}\right)\cos\vartheta\right]\hat{\boldsymbol{r}}_0 \tag{2.66}$$

$$+\frac{\partial \vartheta}{\partial \boldsymbol{\xi}_\vartheta}\left[\left(\ddot{p} - p\dot{\vartheta}^2\right)\cos\vartheta - \left(2\dot{p}\dot{\vartheta} + p\ddot{\vartheta}\right)\sin\vartheta\right]\hat{\boldsymbol{t}}_0 \tag{2.67}$$

$$\frac{\partial \ddot{\boldsymbol{r}}}{\partial \boldsymbol{\xi}_h} = \frac{\partial \ddot{h}}{\partial \boldsymbol{\xi}_h}\hat{\boldsymbol{h}}_0 \tag{2.68}$$

Thus, the Jacobian is finally computed with the above results:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\xi}_p} = \frac{\partial \ddot{\boldsymbol{r}}}{\partial \boldsymbol{\xi}_p} - 3\mu\frac{\boldsymbol{r}}{r^4}\frac{\partial r}{\partial \boldsymbol{\xi}_p} + \frac{\mu}{r^3}\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\xi}_p} \tag{2.69}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\xi}_\vartheta} = \frac{\partial \ddot{\boldsymbol{r}}}{\partial \boldsymbol{\xi}_\vartheta} - 3\mu\frac{\boldsymbol{r}}{r^4}\cancel{\frac{\partial r}{\partial \boldsymbol{\xi}_\vartheta}} + \frac{\mu}{r^3}\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\xi}_\vartheta} \tag{2.70}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\xi}_h} = \frac{\partial \ddot{\boldsymbol{r}}}{\partial \boldsymbol{\xi}_h} - 3\mu\frac{\boldsymbol{r}}{r^4}\frac{\partial r}{\partial \boldsymbol{\xi}_h} + \frac{\mu}{r^3}\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\xi}_h} \tag{2.71}$$

where $\frac{\partial r}{\partial \boldsymbol{\xi}_\vartheta}$ = 0 because $r^2 = p^2 + h^2$. That is, $r$ does not depend on $\boldsymbol{\xi}_\vartheta$. Note that Equation (2.69) - (2.71) have missing partials of $\boldsymbol{a}_p(\boldsymbol{r}, \dot{\boldsymbol{r}})$. This is due to the fact that the perturbation applied in this investigation are merely constant and random stresses added to the equations of motion in order to prove flexibility of convergence. Future work includes expanding upon the formulation to include perturbations that are time and position dependent. Once the formulation is completed, the equations must be evaluated iteratively due to the indirect and direct dependence on $\boldsymbol{\xi}_p$, $\boldsymbol{\xi}_\vartheta$, and $\boldsymbol{\xi}_h$.

#### 2.3.3.3 Initial guess

In this investigation, the initial guess for $\boldsymbol{\xi}_{p0}$, $\boldsymbol{\xi}_{\vartheta 0}$, and $\boldsymbol{\xi}_{h0}$ to start the nonlinear iterative process is defined in two distinct ways :

1. If no prior knowledge is known about the particular problem, setting $\boldsymbol{\xi}_{p0} = \boldsymbol{\xi}_{\vartheta 0} = \boldsymbol{\xi}_{h0} = 0$. This means that the nonlinear iterative approach begins with an initial trajectory that linearly changes the values of $p(t)$, $\vartheta(t)$, and $h(t)$, from their initial values to their final values.

2. If prior knowledge to the problem is known, and the ToF is desired to be varied or perturbations are added, then the initial known trajectory using the constrained expressions. These equations provide the guarantee that the boundary conditions are analytically satisfied.

Given an initial guess for $\boldsymbol{\xi}_p$, $\boldsymbol{\xi}_\vartheta$, and $\boldsymbol{\xi}_h$ is obtained, the nonlinear least-squares problem is solved.

#### 2.3.3.4 Final solution : computation of the initial velocity

The solution of the perturbed / unperturbed Lambert problem is determined by the velocity at the initial position, $\dot{\boldsymbol{r}}_0$. Equation (2.36) evaluated at initial time is :

$$\dot{\boldsymbol{r}}_0 = \dot{p}_0 \hat{\boldsymbol{r}}_0 + p_0 \dot{\vartheta}_0 \hat{\boldsymbol{t}}_0 + \dot{h}_0 \hat{\boldsymbol{h}}_0 \tag{2.72}$$

where

$$\begin{cases} p_0 &= r_0 \\ \dot{p}_0 &= c g_p'(-1) + \dfrac{1}{\Delta T}\left[r_f - r_0 + g_p(-1) - g_p(+1)\right] \\ \dot{\vartheta}_0 &= c g_\vartheta'(-1) + \dfrac{1}{\Delta T}\left[g_\vartheta(-1) + \vartheta_r + 2k\pi - g_\vartheta(+1)\right] \\ \dot{h}_0 &= c g_h'(-1) + \dfrac{1}{\Delta T}\left[g_h(-1) - g_h(+1)\right] \end{cases} \tag{2.73}$$

Given this initial velocity, it is possible to propagate the resulting Lambert arc with traditional orbit integrators.

# 3. NUMERICAL EXAMPLES

This section demonstrates several numerical examples of PSS orbits based on constraints (orbit shape, orbit orientation, and observation distance). The LEO satellite (ECHO-1), GEO satellite (COSMOS-2350), and HEO satellite (FALCON) were used as examples to confirm the difference in PSS orbit based on orbital altitude; the Two Line Elements (TLE) of the three satellites are listed below.

1. Echo-1 (SatID : 00049), a satellite in LEO, whose TLE are

```
1 00053U 60009E   21027.13978946 -.00000093  00000-0 -70004-4 0  9992

2 00053  47.2764 289.1802 0100430  22.0030 338.5028 12.17252257695817
```

2. Cosmos-2350 (SatID : 25315), a satellite in GEO, whose TLE are

```
1 25315U 98025A   21026.52501575 -.00000120  00000-0  00000-0 0  9995

2 25315  13.2193  18.0058 0005455  68.2888 306.1394  1.00259395 83289
```

3. Falcon (SatID : 55684), a satellite in HEO, whose TLE are

```
1 55684U 23022B   23079.04434928  .00014710  00000+0  26348-2 0  9999

2 55684  27.1322  43.3082 7580232 180.5463 177.6030  1.92960517   555
```

And converting a given TLE into orbital elements is as follows.

| | $\boldsymbol{a}$ (km) | $\boldsymbol{b}$ (km) | $\boldsymbol{i}$ (deg) | $\boldsymbol{\Omega}$ (deg) | $\boldsymbol{\omega}$ (deg) | $\boldsymbol{\varphi}$ (deg) |
|---|---|---|---|---|---|---|
| Echo-1 (LEO) | 7982.66 | 7982.26 | 47.27 | 289.18 | 22.00 | -0.80 |
| Cosmos-2350 (GEO) | 42168.20 | 42168.19 | 13.21 | 18.00 | 68.28880 | -1.73 |
| Falcon (HEO) | 27253.53 | 17775.50 | 27.13 | 43.30 | 180.54 | 2.79 |

Table 3.1: Orbital elements of LEO, GEO, and HEO satellites

The success of GA depends on the proper selection of its parameters. In this regard, the five key parameters to consider are shown in Table 3.2. The number of generations specifies the number of iterations during which the algorithm runs. The number of parents determines the number of individuals that are selected to create the next generation. The number of chromosomes defines the number of solutions that coexist in each generation, while the number of genes represents the number of variables in each chromosome that define the solution space. Finally, the mutation percent is the probability that each gene mutates during the evolution process.

| GA Parameter | Value | TFC Parameter | Value |
|---|---|---|---|
| Number of generations | 50 | Number of points $(N)$ | 200 |
| Number of parents | 10 | | |
| Number of chromosome | 18 | Number of basis functions $(m)$ | 60 |
| Number of genes | 5 | | |
| Mutation percent | 20 | Convergence criteria $(\epsilon)$ | $10^{-13}$ |

Table 3.2: Input GA & TFC parameters for Monte Carlo test

The number of points of discretization and the number of basis functions are both important TFC parameters that affect the accuracy and computational complexity of numerical methods such as least squares. Discretization refers to the process of approximating a continuous function with a set of discrete points. The number of points of discretization refers to the number of discrete points used to represent the continuous function. Basis functions are used to represent the approximation of a function in terms of a linear combination of a set of functions. The number of basis functions refers to the number of functions used to represent the approximation. A higher number of these typically results in a more accurate approximation, but also increases the computational complexity of the method. The convergence criteria refers to a small positive number that is used to determine when the iteration has converged to a solution. It is used as the stopping criterion for the TFC nonlinear least square. The values of the TFC parameters used in this test are shown in Table 3.2.

## 3.1 Monte Carlo Test

For the LEO, GEO, and HEO cases without perturbations, we first do a Monte Carlo test. In order to determine the PSS orbit, it is necessary to confirm the parameter ranges for the constraints (orbit shape, orbit orientation, and observation distance). There are two steps to the Monte Carlo test. In the first stage, 10,000 random constraint values, specifically the orbit shape parameter and orbit orientation parameter, are assigned to investigate the observation percentage within the range of these two variables. (In essence, each variable is randomly selected 10,000 times within the range values presented in Table 3.3.) The percentage of an orbit cycle that the chaser can observe the target is referred to as the observation percentage. It stands for the proportion of time that the distance between the two satellites is within the parameters for observational distance. (Note that the parameters for the observation distance, $d_{\max}$ and $d_{\min}$, are fixed at 10 km and 0.01 km, respectively, for all tests.)

| First Test | LEO | GEO | HEO |
|---|---|---|---|
| $N_{trial}$ | 10,000 | 10,000 | 10,000 |
| $\delta\beta$ (km) | 0–100 | 0–1 | 0–100 |
| $\phi$ (deg) | 0–1 | 0–0.3 | 0–1 |

| Second Test | LEO | GEO | HEO |
|---|---|---|---|
| $N_{trial}$ | 10,000 | 10,000 | 10,000 |
| $\delta\beta$ (km) | 0–5 | 0–0.05 | 0–1 |
| $\phi$ (deg) | 0–0.05 | 0–0.0005 | 0–0.01 |

Table 3.3: Input test parameters for Monte Carlo test

The second stage involves identifying the zone where the observation percentage exhibits 100%, which corresponds to the range of the two parameters that yield a 100% result, based on the findings from the first stage. This test is conducted by adjusting the parameter range for each orbit and subsequently carrying out an additional 10,000 tests to obtain the results. The adjusted parameter ranges are depicted in Table 3.3.

### 3.1.1 Examples #1 : Echo-1, a satellite in LEO

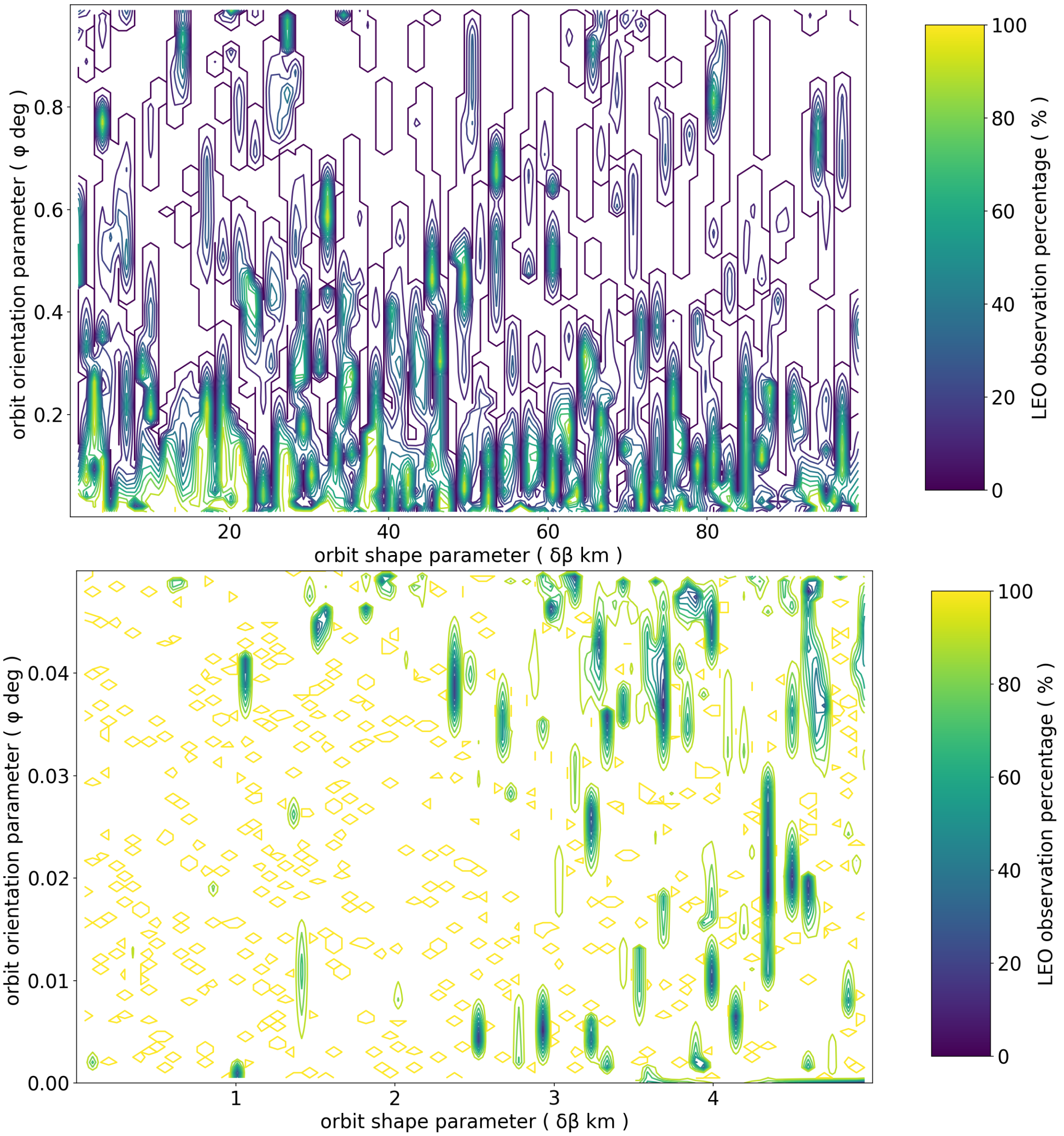

Figure 3.1: 2D interpolation contour plot for LEO observation percentage

Figure 3.1 illustrates the results for LEO using a 2D interpolation contour plot to display the outcomes based on the two parameters. The first plot represents the results of the initial Monte Carlo test, while the second plot shows the results of the subsequent Monte Carlo test. Upon examining the first plot, it can be observed that lower values for both parameters correspond to higher observation results, whereas higher values for both parameters yield lower observation results. Additionally, it is evident that the $\phi$ (orbit orientation parameter) has a more significant impact on the observation results than the $\delta\beta$ (orbit shape parameter). The second test was carried out with further adjustments to the parameter ranges, based on the findings from the first test. Examining the second plot, clusters of yellow shapes can be seen, indicating that within this range, the observation results reach 100%. In conclusion, the LEO example demonstrates that the optimal observation results can be obtained within the range of $\delta\beta$ (0–5 km) and $\phi$ (0–0.05 deg).

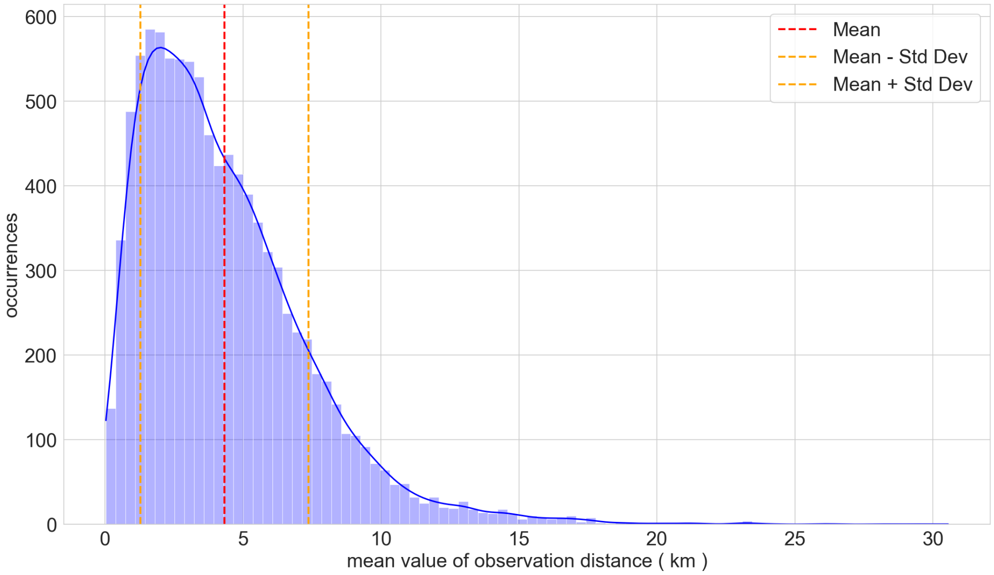

Figure 3.2: Monte Carlo test histogram for obs distance mean values (LEO)

Figure 3.2 presents the results of the second test in the form of a histogram. The x-axis represents the mean value of observation distance between the chaser and the target during one orbit cycle, while the y-axis displays the occurrences of the Monte Carlo test results. The red dashed line indicates the mean value of all the results, and the orange dashed line represents the standard deviation. Consequently, the overall mean value for this test is approximately 5 km, and the majority of the results fall within the observation distance range of 0.01–10 km. This implies that most of the Monte Carlo test outcomes satisfy the 100% observation result criterion.

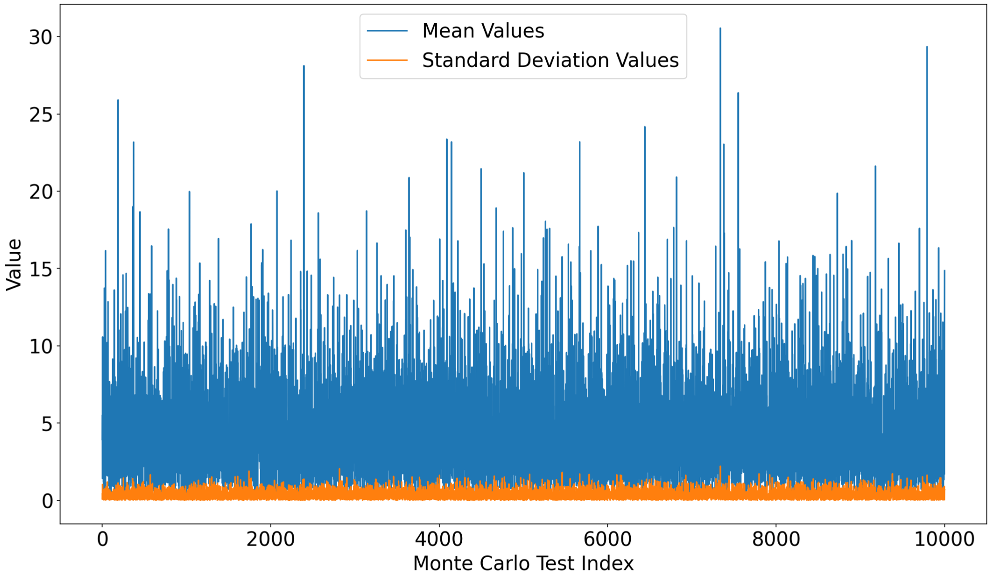

Figure 3.3: Monte Carlo test plot for Mean & STD values (LEO)

Figure 3.3 depicts a plot of the mean value of the distance between the two satellites and the standard deviation for each sample in the Monte Carlo test. One observation that can be derived from this plot is that the distance between the chaser and the target during one orbit cycle remains relatively consistent and does not deviate significantly from the mean value. This characteristic can be attributed to the near-zero eccentricity of the LEO.

### 3.1.2 Examples #2 : Cosmos-2350, a satellite in GEO

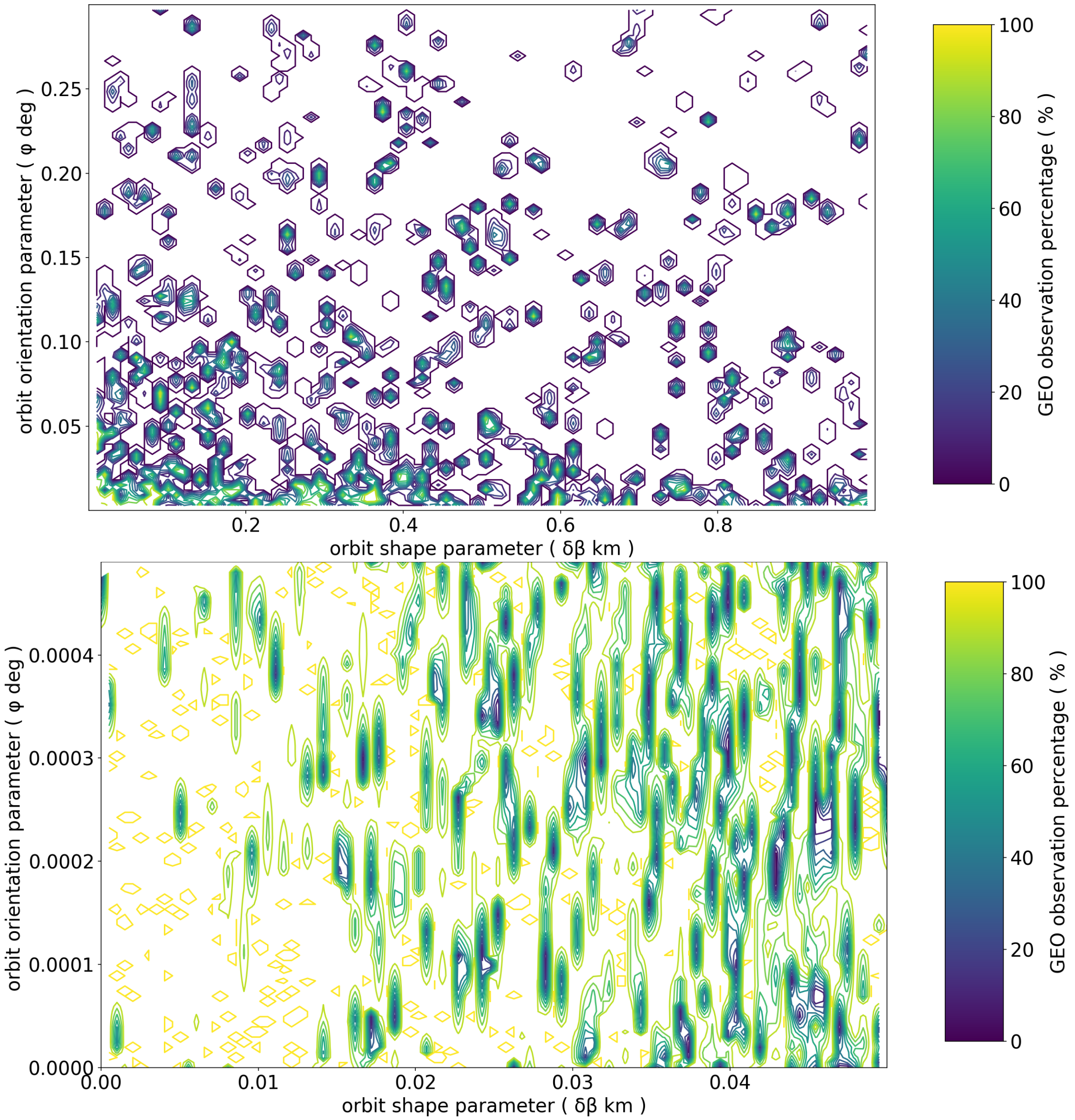

Figure 3.4: 2D interpolation contour plot for GEO observation percentage

Aside from two parameter ranges, which are the same as the previous LEO tests, Figure 3.4 illustrates the results obtained for GEO. It can be deduced that the parameter ranges must be significantly smaller in the GEO case because of the GEO's noticeably larger size compared to LEO. Examining the first plot, similar to the LEO results, it can be observed that lower values of the two parameters correspond to higher observation outcomes, while higher values of the parameters result in lower observation outcomes. Likewise, it is evident that the $\phi$ (orbit orientation parameter) value has a greater impact on the observation outcome compared to the $\delta\beta$ (orbit shape parameter) value, consistent with the LEO findings. The second test is conducted by adjusting the two parameter ranges to lower values, based on the results of the first test. It is clear that the parameter range values are much smaller for GEO compared to LEO. In the second plot, similarly, clusters of yellow shapes can be observed, which indicate that within this range, a 100% observation outcome is achieved. The optimal observation results in GEO are found to be within the parameter ranges of $\delta\beta$ (0–0.05 km) and $\phi$ (0–0.0005 degree), it can be concluded.

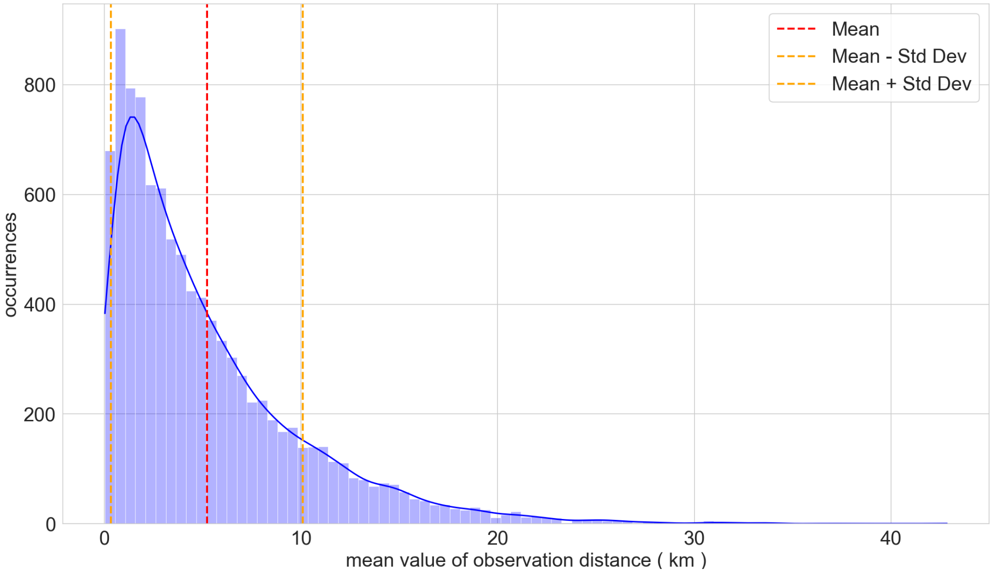

Figure 3.5: Monte Carlo test histogram for obs distance mean values (GEO)

Figure 3.5 illustrates the results of the second Monte Carlo test for the GEO in the form of a histogram. Similar to the LEO results, the overall average value is approximately 5 km, and the majority of the results fall within the observation distance range of 0.01–10 km. This again implies that most of the test outcomes satisfy the 100% observation result criterion.

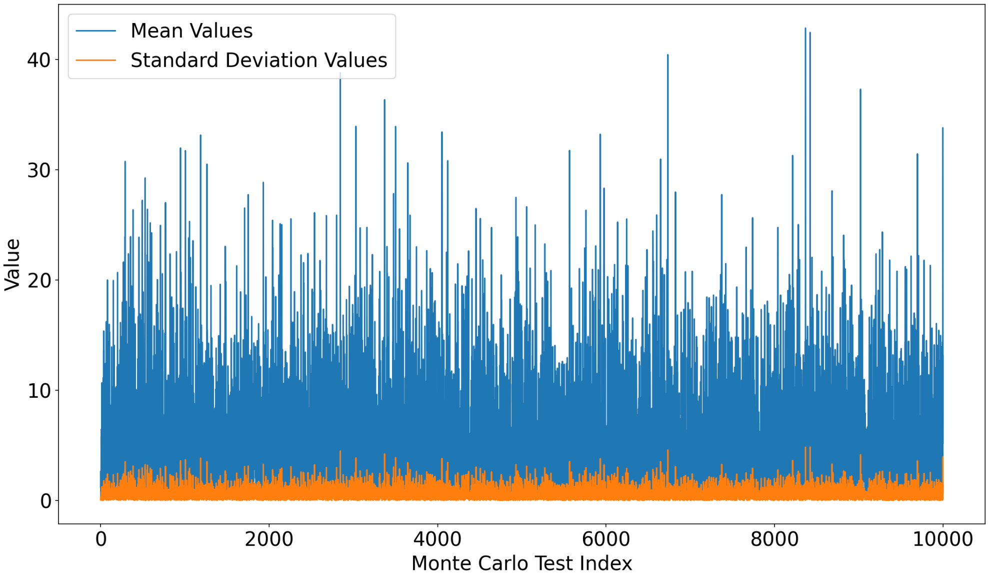

Figure 3.6: Monte Carlo test plot for Mean & STD values (GEO)

Figure 3.6 presents the mean value and standard deviation results for each sample in the GEO Monte Carlo test. When compared to the LEO results, there are no significant differences except for slightly higher values. Similarly, the GEO test demonstrates that the distance between the chaser and the target during one orbit cycle remains relatively consistent and does not deviate significantly from the mean value. This is also due to the near-zero eccentricity of GEO, similar to LEO.

### 3.1.3 Examples #3 : Falcon, a satellite in HEO

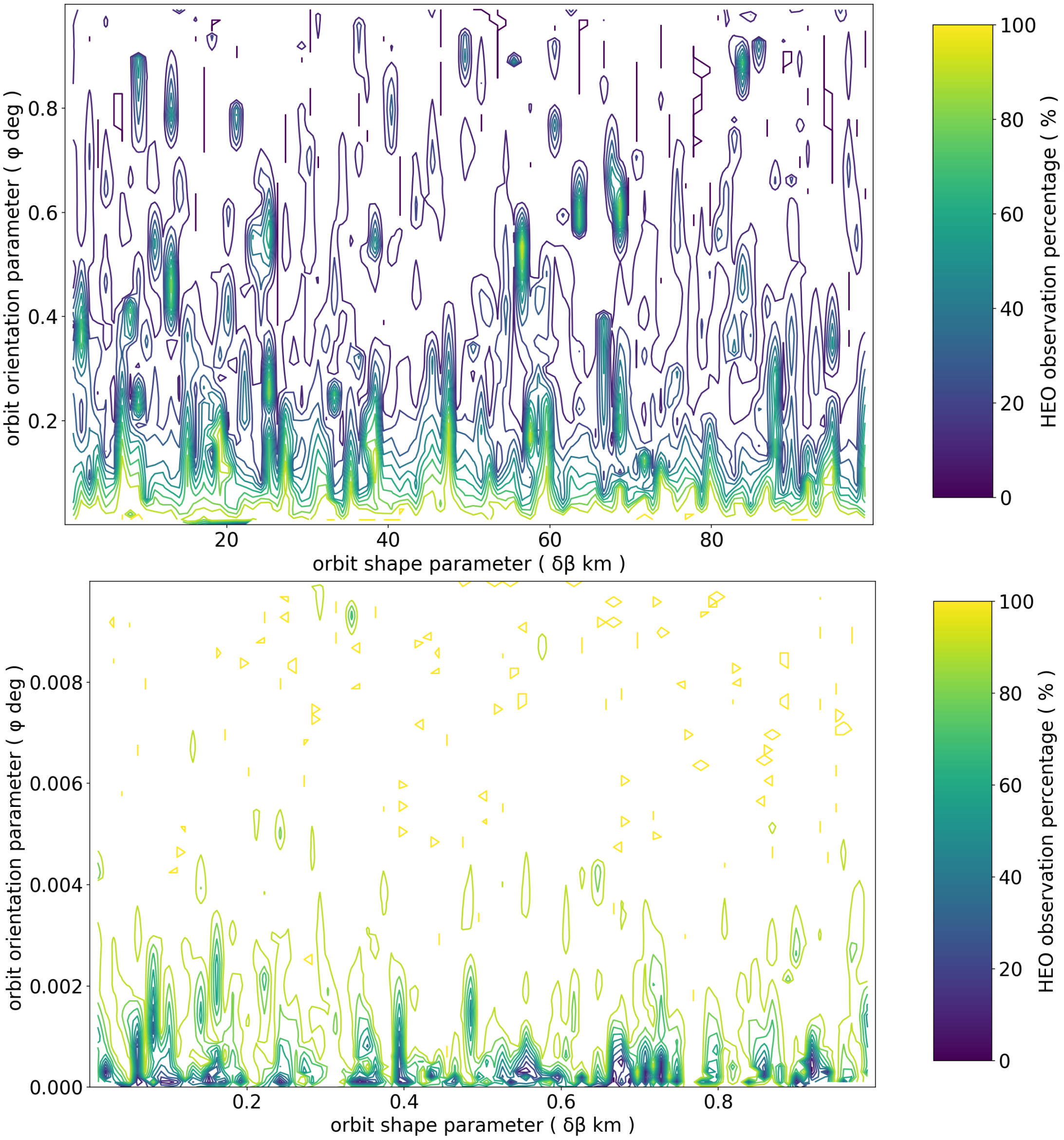

Figure 3.7: 2D interpolation contour plot for HEO observation percentage

Figure 3.7 presents the results for HEO. Unlike LEO and GEO, HEO is characterized by a significantly high eccentricity. Consistent with the LEO and GEO findings, lower values of the two parameters yield higher observation outcomes, and the $\phi$ (orbit orientation parameter) value has a greater impact on the observation outcome compared to the $\delta\beta$ (orbit shape parameter) value. The two parameter ranges are changed to lower values for the second test. Clusters of yellow shapes can be seen on the second plot, indicating that within this range, a 100% observation outcome is attained. There is a small variation from the first two findings, though. As the $\phi$ value approaches 0, the observation outcome becomes considerably lower. This phenomenon occurs because the observation distance between the chaser and the target decreases significantly, increasing the likelihood of collision. Understanding the impact of these parameters in HEO is crucial for optimizing observation performance and minimizing the risk of collisions, which is essential for the safe and efficient operation of satellites in highly elliptical orbits.

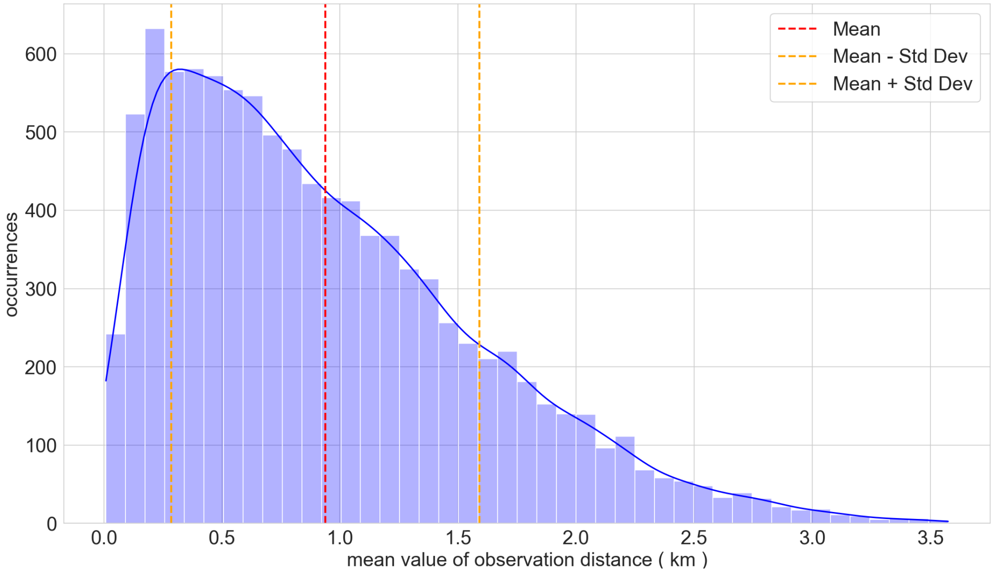

Figure 3.8: Monte Carlo test histogram for obs distance mean values (HEO)

Figure 3.8 displays the results of the second Monte Carlo test for HEO in the form of a histogram. With an average observation distance value of around 1 km, it can be inferred that the

HEO results have been optimized to be closer to the target compared to the LEO and GEO results. Nevertheless, the majority of the results still fall within the observation distance range of 0.01–10 km, which implies that the outcomes satisfy the 100% observation result criterion.

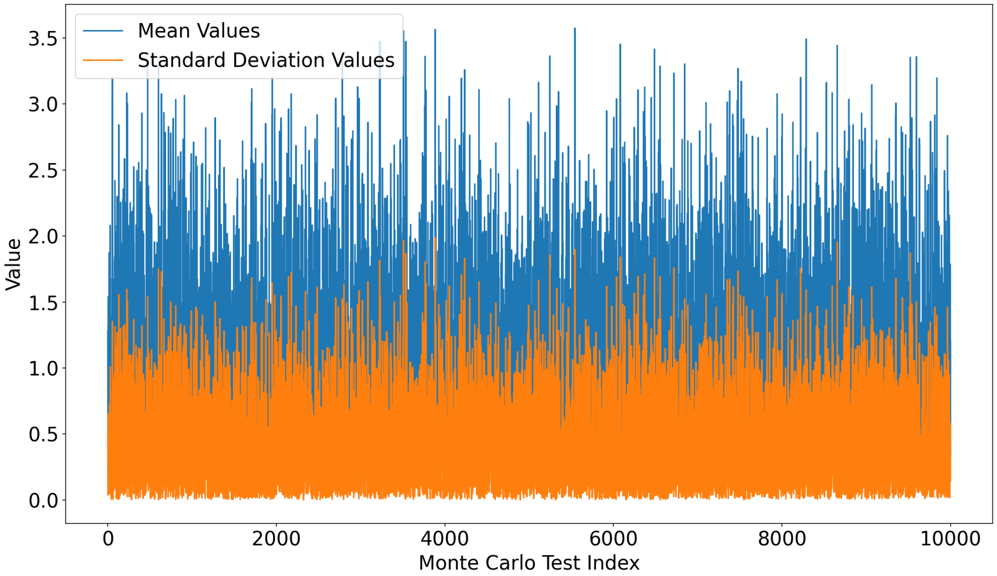

Figure 3.9: Monte Carlo test plot for Mean & STD values (HEO)

Figure 3.9 presents the mean value and standard deviation results for each sample in the HEO Monte Carlo test. As previously mentioned, it can be confirmed once again that the optimization has been achieved to maintain an average distance of approximately 1 km from the target. Unlike the LEO and GEO results, a larger standard deviation value can be observed relative to the mean value. This is due to the higher eccentricity of HEO, which sets it apart from LEO and GEO.

## 3.2 Sample Test

In this study, we obtained 10,000 datasets for each of the Low Earth Orbit (LEO), Geostationary Earth Orbit (GEO), and Highly Elliptical Orbit (HEO) examples through Monte Carlo simulations. We aim to examine the relative distance between the chaser and target spacecraft over one orbital period by extracting four samples from each example. The criterion for selecting these samples is to randomly choose four instances where the average relative distance between the chaser and target over one orbital period lies between 10 and 100 meters from the pool of 10,000 datasets in each example. The sample data for each example can be found in Table 3.4. Additionally, we incorporate the effects of third-body perturbations to investigate their impact on each orbit. In this context, the third body is assumed to be the Moon, while the lunar orbital motion is not considered. This is because the change in the Moon's position over the duration of one orbital period for each example does not significantly influence the experimental results.

| LEO Scenario I–IV | Orbit shape | Orbit orientation | Observation distance |
|---|---|---|---|
| (Echo-1) | $\delta\beta$ (km) | $\phi$ (deg) | $d_{\max}$, $d_{\min}$ (km) |
| I | 4.83343 | 0.01694 | 10, 0.01 |
| II | 1.01045 | 0.0006 | 10, 0.01 |
| III | 0.11282 | 0.00353 | 10, 0.01 |
| IV | 0.03542 | 0.00019 | 10, 0.01 |
| GEO Scenario I–IV | Orbit shape | Orbit orientation | Observation distance |
| (Cosmos-2350) | $\delta\beta$ (km) | $\phi$ (deg) | $d_{\max}$, $d_{\min}$ (km) |
| I | 0.04189 | 0.00035 | 10, 0.01 |
| II | 0.03459 | 0.0003 | 10, 0.01 |
| III | 0.02162 | 0.00022 | 10, 0.01 |
| IV | 0.00939 | 0.00045 | 10, 0.01 |
| HEO Scenario I–IV | Orbit shape | Orbit orientation | Observation distance |
| (Falcon) | $\delta\beta$ (km) | $\phi$ (deg) | $d_{\max}$, $d_{\min}$ (km) |
| I | 0.70717 | 0.00034 | 10, 0.01 |
| II | 0.63895 | 0.00009 | 10, 0.01 |
| III | 0.40296 | 0.00004 | 10, 0.01 |
| IV | 0.21765 | 0.00074 | 10, 0.01 |

Table 3.4: Constraints for each scenario

### 3.2.1 Examples #1 : Echo-1, a satellite in LEO

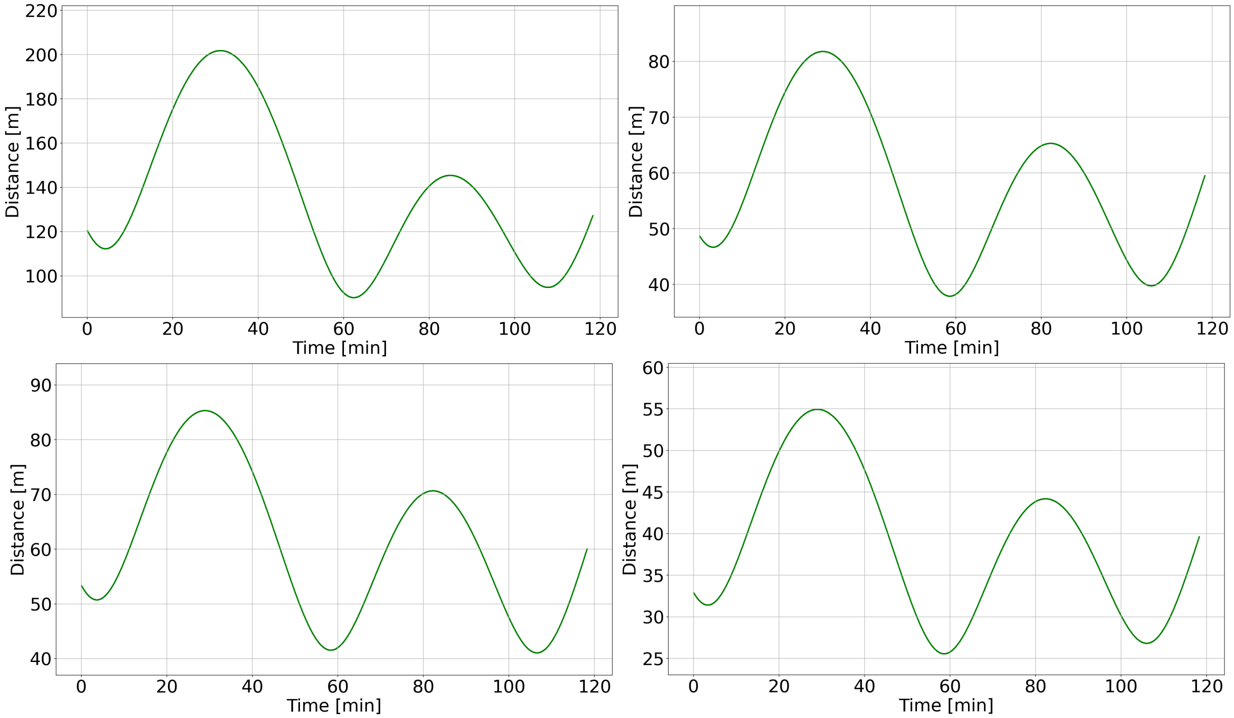

Figure 3.10: Observable distance with third body (Moon) perturbations in LEO

Figure 3.10 presents a plot of the relative distance between two satellites over one orbital period, taking into consideration the Moon's perturbation effects on the orbits, based on four scenarios extracted from a LEO example. The x-axis of the plot represents the time duration of one orbit of Echo-1, while the y-axis illustrates the relative distance in meters between the two satellites, namely the chaser and target satellites. For the Monte Carlo test, only samples with average relative distances ranging from 10 to 100 meters were randomly selected. However, it is important to note that some discrepancies, as seen in Scenario I, may occur. This is attributed to the fact that the GA was executed again, taking into account the perturbation effects. On the other hand, Scenarios II through IV demonstrate that the observed distances are consistently maintained within the 10 to 100 meter range, and it can be concluded that 100% observation is possible for all scenarios during one orbital period. In spite of perturbations, the stability of the relative distance between the two satellites is also seen. This can be explained by the lower altitude of LEO compared to other orbits, which makes the satellites only little affected by the Moon's gravitational pull.

**Scenario** I

| | | | |
|---|---|---|---|
| Position (km) | $R_x = 4251.69000592$ | $R_y = -6321.70327886$ | $R_z = 2099.30266777$ |
| Velocity (km/s) | $V_x = 3.41585014$ | $V_y = 3.92198011$ | $V_z = 4.88852825$ |

**Scenario** II

| | | | |
|---|---|---|---|
| Position (km) | $R_x = 4251.753816$ | $R_y = -6321.73178791$ | $R_z = 2099.32382177$ |
| Velocity (km/s) | $V_x = 3.41580696$ | $V_y = 3.92195867$ | $V_z = 4.88849368$ |

**Scenario** III

| | | | |
|---|---|---|---|
| Position (km) | $R_x = 4251.76064$ | $R_y = -6321.72851888$ | $R_z = 2099.30962918$ |
| Velocity (km/s) | $V_x = 3.41580828$ | $V_y = 3.92195735$ | $V_z = 4.88849736$ |

**Scenario** IV

| | | | |
|---|---|---|---|
| Position (km) | $R_x = 4251.76462759$ | $R_y = -6321.74191929$ | $R_z = 2099.32930621$ |
| Velocity (km/s) | $V_x = 3.41579639$ | $V_y = 3.92195271$ | $V_z = 4.88848576$ |

**Target (Echo-1)**

| | | | |
|---|---|---|---|
| Position (km) | $R_x = 4251.78804498$ | $R_y = -6321.76187171$ | $R_z = 2099.34099662$ |
| Velocity (km/s) | $V_x = 3.41577373$ | $V_y = 3.92194092$ | $V_z = 4.88846971$ |

Table 3.5: Initial position & velocity vectors in LEO

Table 3.5 presents the results of the initial position vectors and initial velocity vectors for each scenario. It can be observed that all the obtained values are in close proximity to the target's initial values. This indicates that by propagating the initial values of these scenarios, it is possible to satisfy the requirements for the PSS orbit.

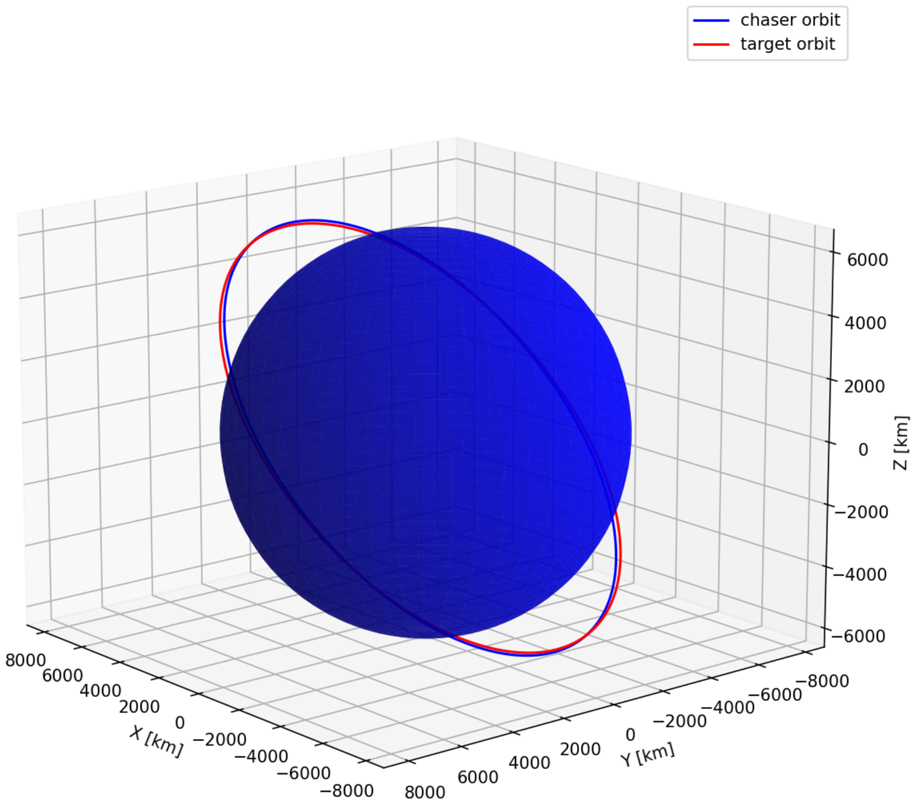

Figure 3.11: 3D view of PSS orbit to ECHO-1 in LEO

In conclusion, the analysis presented in Figure 3.10 and Table 3.5 demonstrates the importance of considering Moon perturbations when evaluating the relative distances between satellites in LEO. The results show that, despite some discrepancies, the majority of the scenarios maintain the observed distances within the desired range of 10 to 100 meters, enabling 100% observation during one orbital period. The stability of the relative distance between the two satellites, even in the presence of perturbations, highlights the resilience of LEO orbits due to their lower altitude, which results in satellites experiencing only minor effects from the Moon's gravitational forces. Furthermore, the close proximity of the initial position and velocity vectors to the target's initial values indicates the potential for achieving the requirements of the PSS orbit by propagating these initial values.

### 3.2.2 Examples #2 : Cosmos-2350, a satellite in GEO

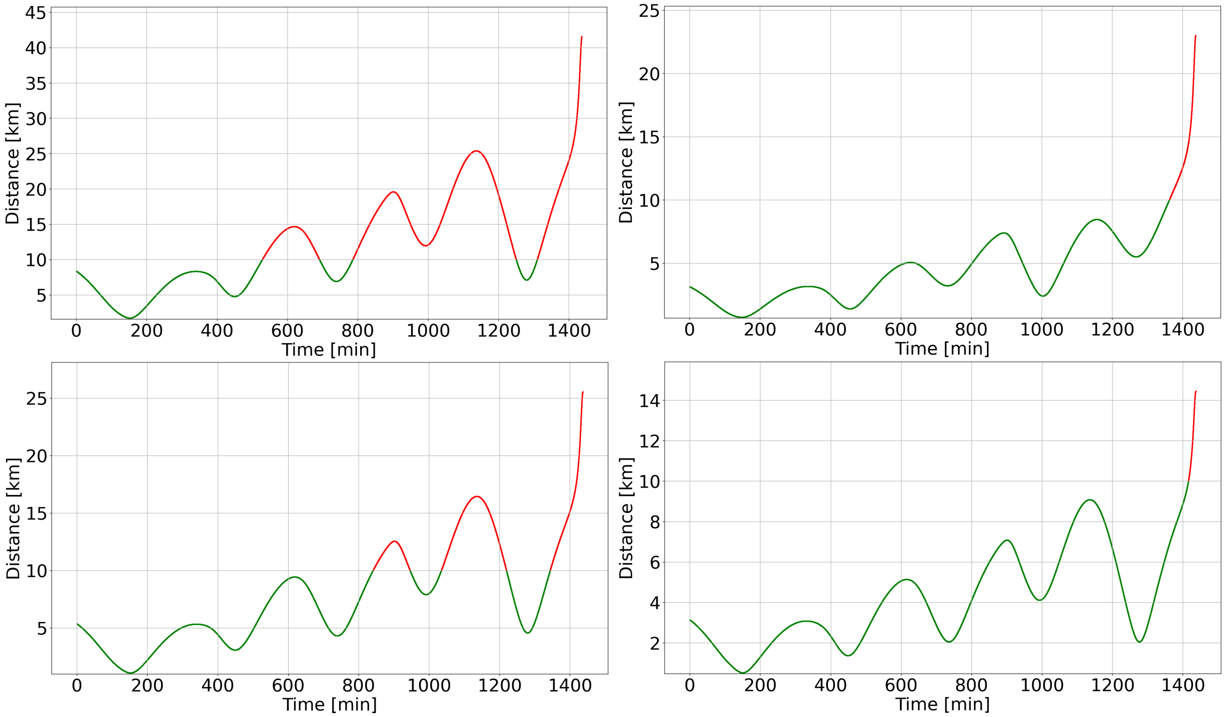

Figure 3.12: Observable distance with third body (Moon) perturbations in GEO

Figure 3.12 depicts a plot of the relative distance between two satellites over one orbital period, considering the Moon's perturbation effects on the orbits, based on four scenarios extracted from a GEO example. The x-axis of the plot signifies the time duration of one orbit of Cosmos-2350, while the y-axis represents the relative distance in kilometers between the two satellites, namely the chaser and target satellites. The green line indicates the observable range where the chaser can monitor the target, whereas the red line represents the unobservable range. The observation percentage for each scenario during one orbital period is displayed in Table 3.6. Similar to the LEO example, only samples with average relative distances ranging from 10 to 100 meters were randomly selected in the Monte Carlo test. However, some discrepancies in the results are evident. This is due to the re-execution of the GA while considering the perturbation effects. The

| Scenario | Observation percentage (%) |
|---|---|
| I | 47.49 |
| II | 94.80 |
| III | 73.31 |
| IV | 98.53 |

Table 3.6: Observation percentage for each scenario in GEO

GEO's higher altitude compared to LEO results in a greater influence from the Moon's gravitational forces. As a result, after one orbital period, the impact of Moon perturbations causes the distance between the two satellites to quickly rise, as seen in all cases. These results highlight the necessity of taking perturbations into account when calculating satellite-relative distances, particularly in higher-altitude orbits like GEO.

**Scenario** I

| | | | |
|---|---|---|---|
| Position (km) | $R_x = 4318.93720574$ | $R_y = 40973.90459632$ | $R_z = 8840.11256105$ |
| Velocity (km/s) | $V_x = -3.05273857$ | $V_y = 0.26129291$ | $V_z = 0.28004449$ |

**Scenario** II

| | | | |
|---|---|---|---|
| Position (km) | $R_x = 4316.29558763$ | $R_y = 40978.29072781$ | $R_z = 8841.1433572$ |
| Velocity (km/s) | $V_x = -3.05245631$ | $V_y = 0.26105819$ | $V_z = 0.27996538$ |

**Scenario** III

| | | | |
|---|---|---|---|
| Position (km) | $R_x = 4317.32723074$ | $R_y = 40976.36419284$ | $R_z = 8840.74775681$ |
| Velocity (km/s) | $V_x = -3.05258147$ | $V_y = 0.26115216$ | $V_z = 0.27999679$ |

**Scenario** IV

| | | | |
|---|---|---|---|
| Position (km) | $R_x = 4316.16191924$ | $R_y = 40978.15795131$ | $R_z = 8841.40438042$ |
| Velocity (km/s) | $V_x = -3.05246356$ | $V_y = 0.26104756$ | $V_z = 0.27996685$ |

**Target (Cosmos-2350)**

| | | | |
|---|---|---|---|
| Position (km) | $R_x = 4314.51562913$ | $R_y = 40980.80869905$ | $R_z = 8841.78544129$ |
| Velocity (km/s) | $V_x = -3.05229728$ | $V_y = 0.26090416$ | $V_z = 0.27991828$ |

Table 3.7: Initial position & velocity vectors in GEO

Table 3.7 displays the outcomes for both initial position vectors and initial velocity vectors across various scenarios. A close alignment between all the derived values and the target's initial values is evident.

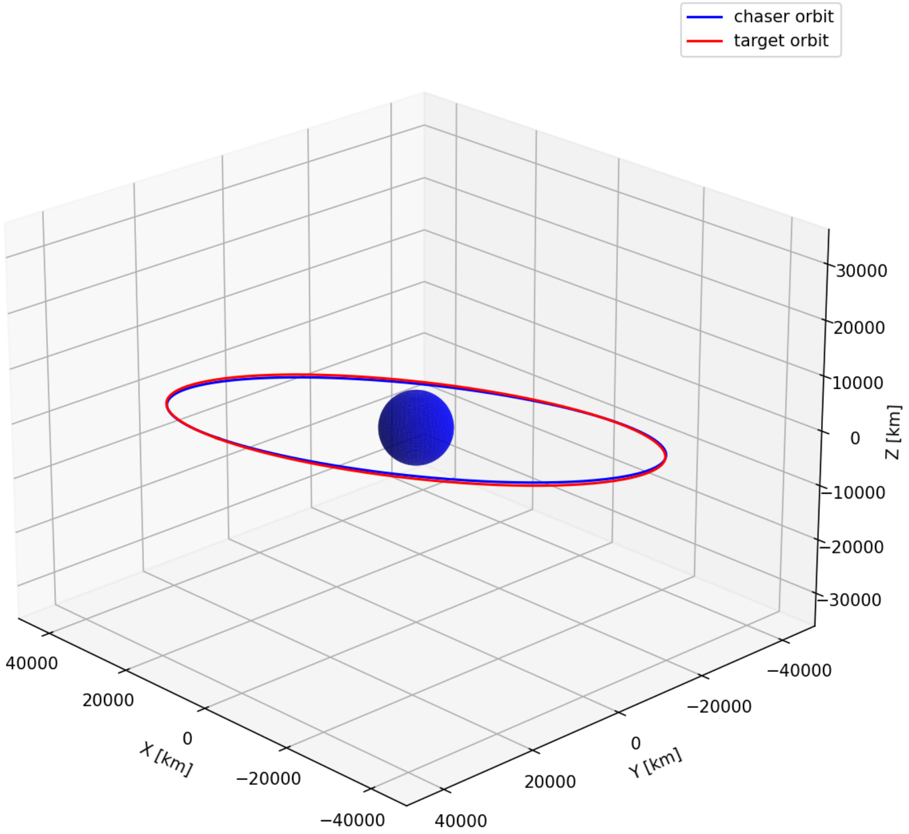

Figure 3.13: 3D view of PSS orbit to COSMOS-2350 in GEO

In summary, the analysis presented in Figure 3.12 and Table 3.7 highlights the significance of considering Moon perturbations when examining the relative distances between satellites in higher-altitude orbits such as GEO. Despite some discrepancies, the close alignment of the initial position and velocity vectors with the target's initial values indicates the potential for meeting the requirements of the PSS orbit by propagating these values. The rapid increase in the distance between the two satellites after one orbital period, as a result of Moon perturbations, emphasizes the importance of factoring in these effects when analyzing satellite-relative distances in GEO.

### 3.2.3 Examples #3 : Falcon, a satellite in HEO

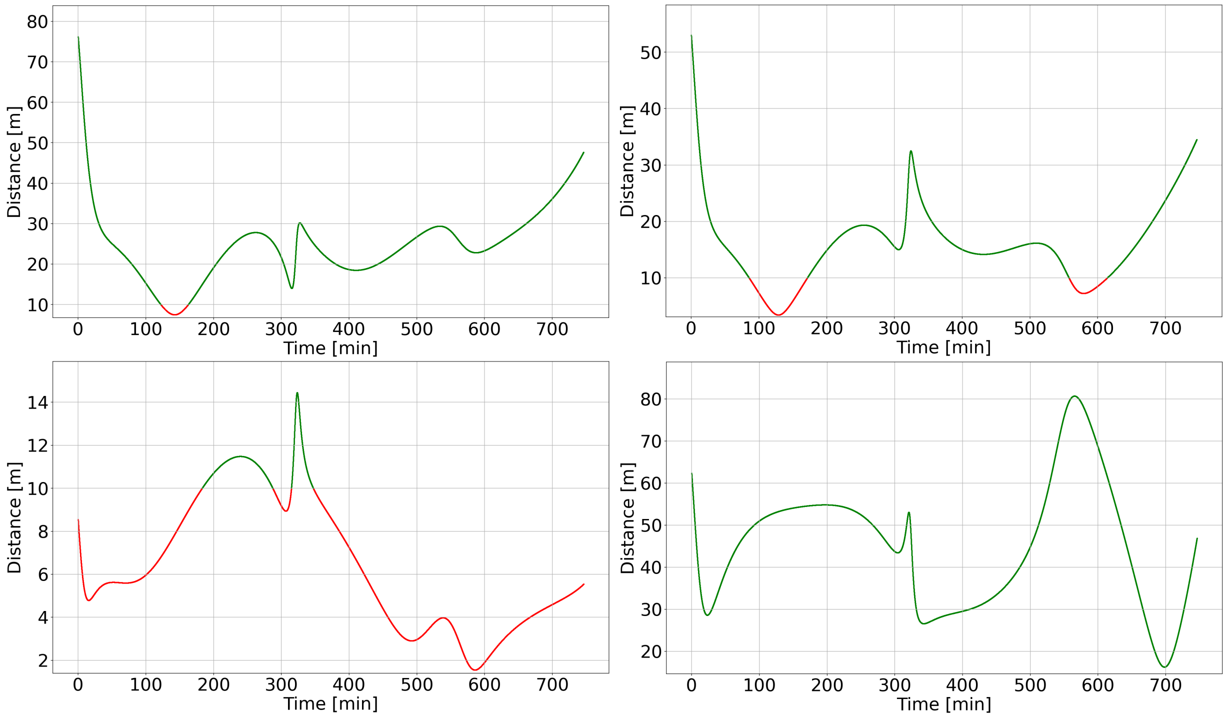

Figure 3.14: Observable distance with third body (Moon) perturbations in HEO

Figure 3.14 presents a plot of the relative distance between two satellites over one orbital period, taking into account the Moon's perturbation effects on the orbits, based on four scenarios extracted from a HEO example. The observation percentage for each scenario during one orbital period is displayed in Table 3.8. As with the LEO and GEO examples, only samples with average relative distances ranging from 10 to 100 meters were randomly selected in the Monte Carlo test. However, unlike the LEO and GEO examples, the relative distances between the two satellites are not consistently maintained. HEO orbits have a high eccentricity, resulting in significant velocity differences when the satellite is closer to Earth compared to when it is farther away. These factors contribute to the susceptibility of HEO orbits to the influence of Moon perturbations. Upon examining the results of Scenarios I to III, red lines can be observed, which can be interpreted

| Scenario | Observation percentage (%) |
|:---:|:---:|
| I | 94.56 |
| II | 80.88 |
| III | 18.31 |
| IV | 100 |

Table 3.8: Observation percentage for each scenario in HEO

as instances where the relative distance between the two satellites dropped below the minimum observation distance of 10 meters, leading to potential collisions. This further emphasizes the importance of considering the unique characteristics of different orbits and the impact of Moon perturbations on satellite-relative distances for efficient space mission planning and satellite design.

**Scenario** I

| | | | |
|---|---|---|---|
| Position (km) | $R_x = -4558.4164869$ | $R_y = -4767.05735843$ | $R_z = -175.33095614$ |
| Velocity (km/s) | $V_x = 6.56580509$ | $V_y = -6.40558939$ | $V_z = -4.69638582$ |

**Scenario**II

| | | | |
|---|---|---|---|
| Position (km) | $R_x = -4558.40392023$ | $R_y = -4767.07533844$ | $R_z = -175.33908559$ |
| Velocity (km/s) | $V_x = 6.56581682$ | $V_y = -6.40557503$ | $V_z = -4.69638019$ |

**Scenario** III

| | | | |
|---|---|---|---|
| Position (km) | $R_x = -4558.38258604$ | $R_y = -4767.11065221$ | $R_z = -175.35788301$ |
| Velocity (km/s) | $V_x = 6.56583951$ | $V_y = -6.40554229$ | $V_z = -4.69637114$ |

**Scenario** IV

| | | | |
|---|---|---|---|
| Position (km) | $R_x = -4558.40065543$ | $R_y = -4767.07007372$ | $R_z = -175.32665345$ |
| Velocity (km/s) | $V_x = 6.56581284$ | $V_y = -6.40558007$ | $V_z = -4.69639133$ |

**Target (Falcon)**

| | | | |
|---|---|---|---|
| Position (km) | $R_x = -4558.382533$ | $R_y = -4767.11799905$ | $R_z = -175.36225963$ |
| Velocity (km/s) | $V_x = 6.5658416$ | $V_y = -6.40553472$ | $V_z = -4.69636804$ |

Table 3.9: Initial position & velocity vectors in HEO

Table 3.9 showcases the results for both initial position vectors and initial velocity vectors across various scenarios. A close alignment between all the derived values and the target's initial values is evident. This suggests that by propagating the initial values from these scenarios, the PSS orbit's conditions can be successfully met.

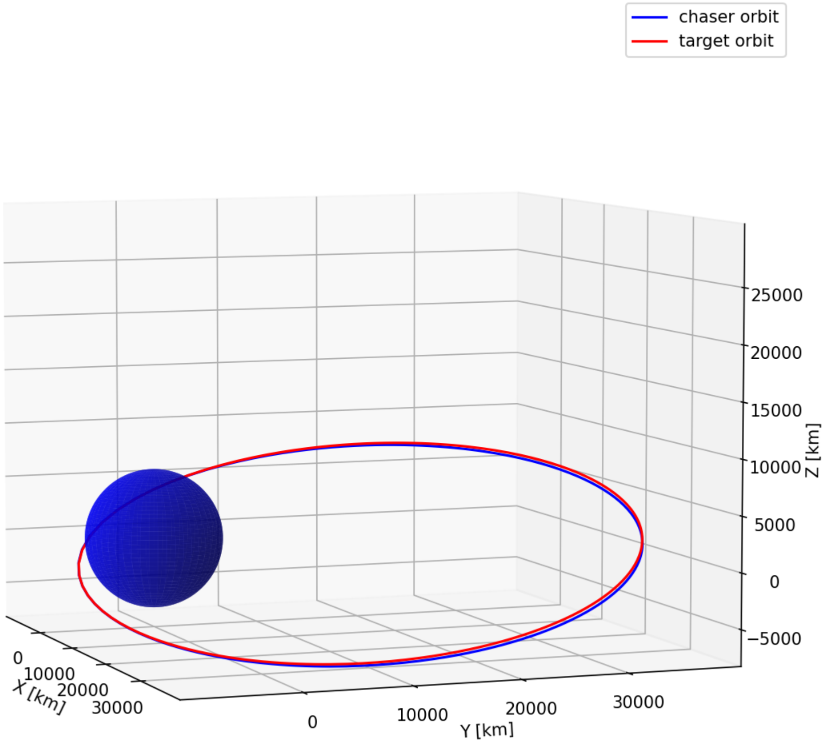

Figure 3.15: 3D view of PSS orbit to FALCON in HEO

The HEO example demonstrates the varying behavior of satellite-relative distances in comparison to LEO and GEO examples, highlighting the importance of considering orbital eccentricity and the influence of Moon perturbations on satellite behavior. Ultimately, this study emphasizes the importance of comprehensive analysis and accurate initial value determination in achieving mission objectives and efficiently utilizing PSS orbits for space engineering applications.

## 3.3 $n$-Impulse Persistent Space Surveillance Orbit

In this chapter, we aim to conduct experiments assuming the transition from the current orbit to another without perturbations. In other words, it could be said that this is an experiment to achieve the PSS orbit through an orbit transfer. The main goal of this is to more practically use the findings regarding the PSS orbit. We believe that our experiment's findings will lead to more real-world solutions to issues like clearing space of debris and spacecraft rendezvous.

The core problem we address involves reaching the PSS orbit through $n$-impulse from the chaser spacecraft. The key is to maneuver with the least amount of fuel possible, achieved through the utilization of a GA. Further explanation on this approach can be found in the PCE research conducted by Clocchiattis [1]. The distinguishing feature of our study compared to Clocchiattis's is that while their research targeted finding the PCE orbit through $n$-impulse, our study focuses on achieving the PSS orbit. In designing the experiment, we select one of the three satellite TLEs mentioned earlier as the chaser, and another one as the target. The experimental design consists of three scenarios, each of which is conducted with varying impulses (as depicted in Table 3.10).

| Scenario | Example | $n$-impulse |
|---|---|---|
| I | LEO (Echo-1) → GEO (Cosmos-2350) | 5 |
| II | LEO (Echo-1) → HEO (Falcon) | 4 |
| III | HEO (Falcon) → GEO (Cosmos-2350) | 3 |

Table 3.10: Scenario for $n$-impulse

The successful transition from one orbit to another, specifically to a PSS orbit, represents a significant advancement in space exploration. This could potentially provide us with the ability to optimize and manage satellite trajectories more efficiently. Additionally, the use of GA reduce fuel consumption. It might open the door for better and more effective maneuvering techniques, which would greatly increase the cost-effectiveness of both scientific and commercial space missions.

### 3.3.1 Examples #1 : 5-impulses from LEO (Echo-1) to GEO (Cosmos-2350)

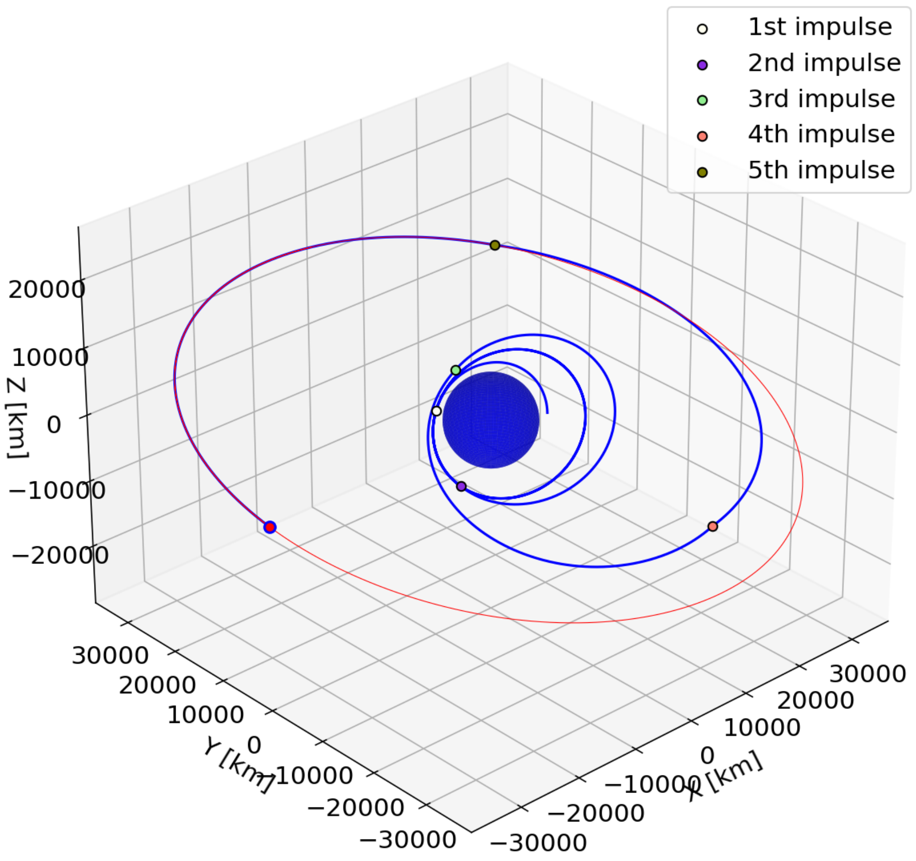

Figure 3.16: 5-impulses PSS orbit

The results displayed in Figure 3.16 represent the achievement of the PSS orbit through a 5-impulses orbit transfer from LEO with Echo-1 to GEO with Cosmos-2350. The blue dots, which appear overlaid with the red dots, represent the chaser satellite, Echo-1, and the blue line illustrates its trajectory. Similarly, the red dots (which appear overlaid with the blue dots) represent the target satellite, Cosmos-2350, and the red line traces its trajectory.

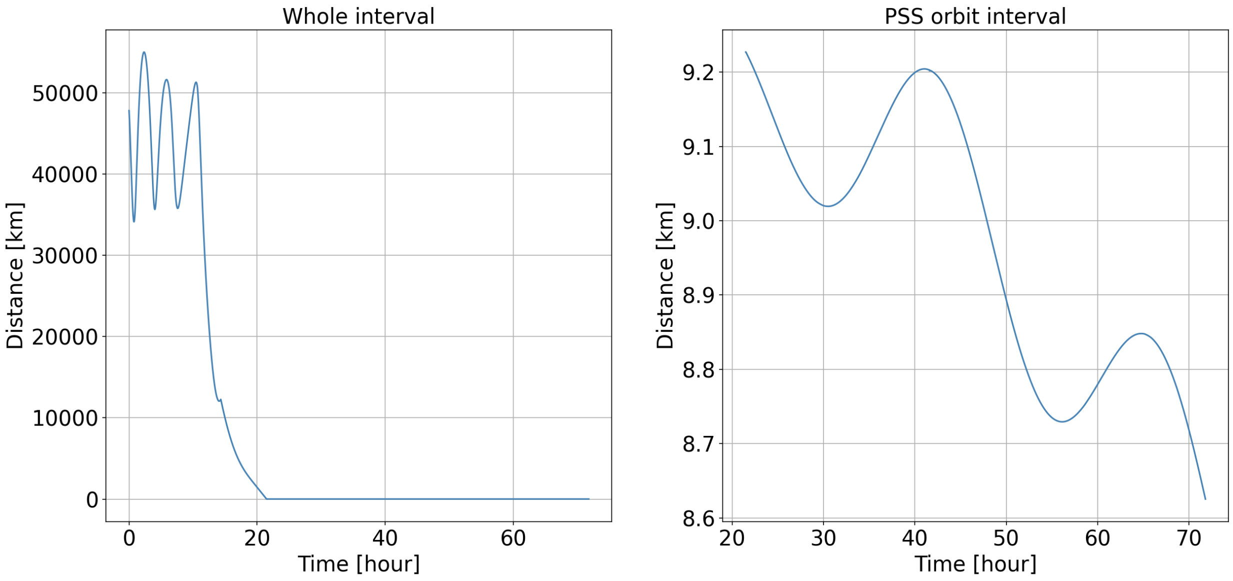

Figure 3.17: Distance between chaser and target (from LEO to GEO)

The outcomes in Figure 3.17 show that it is possible to calculate the relative distance between the chaser and the target. The plot on the left represents the relative distance during three revolutions (or three periods) of the target around the Earth, while the plot on the right illustrates the relative distance following the chaser's proximity to the target (i.e., immediately after achieving the PSS orbit). Observing the relative distance after achieving the PSS orbit, it can be noted that the distance throughout the entire section is less than 10km, indicating that it is observable. However, as time progresses, the distance is seen to gradually narrow, implying a closer approach. Depending on the direction of the problem we wish to apply, this can be adjusted through additional impulses.

Moreover, Table 3.11 presents detailed information at each impulse stage. It shows the time and position vector at each impulse point, along with the subsequent velocity vector determined by the amount of fuel used. In other words, it can be inferred that the chaser approaches the target using a minimum amount of fuel (Total $\Delta v$ = $5.3765$ km/s) over the course of five impulses, and achieves the PSS orbit using the final impulse at the position of the initial encounter with the target.

| **Impulse 1** | $\lvert\Delta v_1\rvert = 0.9013$ km/s | Time = 3219 sec | |
|---|---|---|---|
| Position (km) | $R_x = -3073.1390$ | $R_y = 7433.5708$ | $R_z = -498.4336$ |
| $\Delta v$ (km/s) | $\Delta v_x = -0.74243771$ | $\Delta v_y = -0.3188351$ | $\Delta v_z = -0.39936084$ |
| Velocity (km/s) | $V_x = -5.02582412$ | $V_y = -2.41054436$ | $V_z = -5.52403852$ |

| **Impulse 2** | $\lvert\Delta v_2\rvert = 0.6356$ km/s | Time = 26823 sec | |
|---|---|---|---|
| Position (km) | $R_x = -6723.0586$ | $R_y = -1453.2412$ | $R_z = -6816.9354$ |
| $\Delta v$ (km/s) | $\Delta v_x = -0.04284056$ | $\Delta v_y = -0.2719486$ | $\Delta v_z = -0.5729612$ |
| Velocity (km/s) | $V_x = 0.64167716$ | $V_y = -6.78282266$ | $V_z = -2.03134288$ |

| **Impulse 3** | $\lvert\Delta v_3\rvert = 1.2446$ km/s | Time = 38485 sec | |
|---|---|---|---|
| Position (km) | $R_x = 948.7371$ | $R_y = 8016.2030$ | $R_z = 3988.6705$ |
| $\Delta v$ (km/s) | $\Delta v_x = -1.18211592$ | $\Delta v_y = -0.38412085$ | $\Delta v_z = 0.06434305$ |
| Velocity (km/s) | $V_x = -6.85098483$ | $V_y = 0.7658522$ | $V_z = -4.7632754$ |

| **Impulse 4** | $\lvert\Delta v_4\rvert = 2.3195$ km/s | Time = 51598 sec | |
|---|---|---|---|
| Position (km) | $R_x = 9501.7957$ | $R_y = -32093.7709$ | $R_z = -6112.5128$ |
| $\Delta v$ (km/s) | $\Delta v_x = 1.6457271$ | $\Delta v_y = 1.13774507$ | $\Delta v_z = -1.17368843$ |
| Velocity (km/s) | $V_x = 3.5079447$ | $V_y = 0.70413153$ | $V_z = 0.02865225$ |

| **Impulse 5** | $\lvert\Delta v_5\rvert = 0.2753$ km/s | Time = 77337 sec | |
|---|---|---|---|
| Position (km) | $R_x = 28596.2227$ | $R_y = 30598.1357$ | $R_z = 4759.1705$ |
| $\Delta v$ (km/s) | $\Delta v_x = -0.19060618$ | $\Delta v_y = 0.03865562$ | $\Delta v_z = 0.19491166$ |
| Velocity (km/s) | $V_x = -2.24991289$ | $V_y = 2.00614418$ | $V_z = 0.61154057$ |

| **Target (Cosmos-2350)** | | | |
|---|---|---|---|
| Position (km) | $R_x = 28596.2533$ | $R_y = 30598.1685$ | $R_z = 4759.1756$ |
| Velocity (km/s) | $V_x = -2.2499087$ | $V_y = 2.00614395$ | $V_z = 0.61154021$ |

Table 3.11: The information of each impulse (from LEO to GEO)

### 3.3.2 Examples #2 : 4-impulses from LEO (Echo-1) to HEO (Falcon)

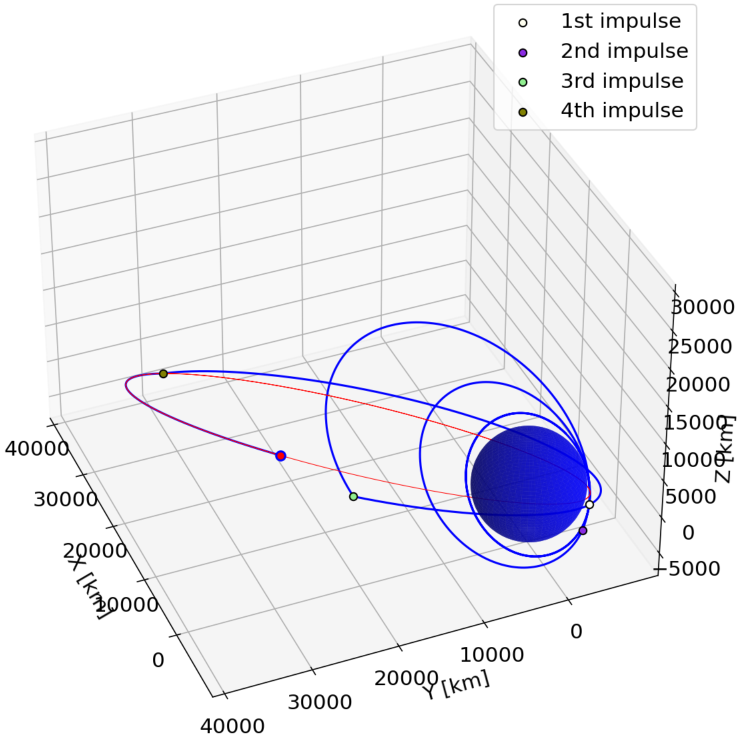

Figure 3.18: 4-impulses PSS orbit

The results shown in Figure 3.18 demonstrate the achievement of the PSS orbit through a 4-impulses orbit transfer from LEO with Echo-1 to HEO with Falcon. By examining the results in Figure 3.19, we can observe the relative distance between the chaser and the target. After achieving the PSS orbit, the relative distance indicates less than 10km in some sections, presenting a slightly different result compared to the first example. This variation is due to the orbit characteristics of the HEO. The speed of an HEO satellite is fast when it is close to Earth and slow when it is further away, thus leading to the observed pattern in relative distance. Also, Table 3.12 provides detailed information at each impulse stage. It can be confirmed that the chaser approaches the target using a minimal amount of fuel (Total $\Delta v$ = $5.5236$ km/s) over the course of 4-impulses, achieving the PSS orbit using the final impulse at the position of the initial encounter with the target.

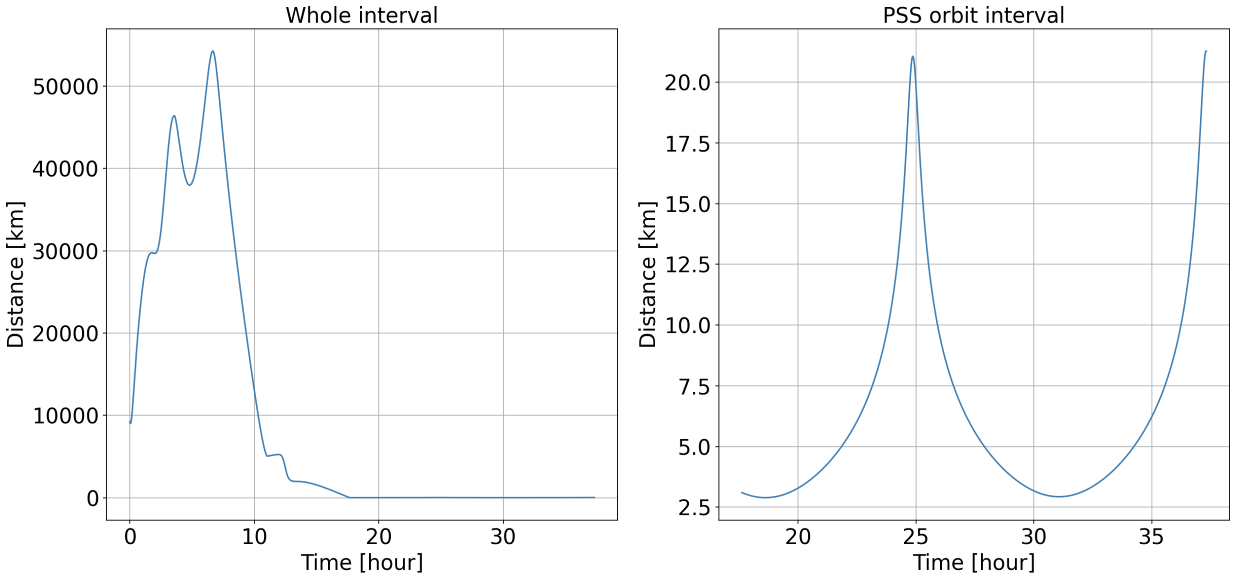

Figure 3.19: Distance between chaser and target (from LEO to HEO)

| **Impulse 1** | $\lvert\Delta v_1\rvert = 1.2663$ km/s | Time = 12981 sec | |
|---|---|---|---|
| Position (km) | $R_x = -1410.2144$ | $R_y = -6800.2025$ | $R_z = -3861.3247$ |
| $\Delta v$ (km/s) | $\Delta v_x = 1.1935191$ | $\Delta v_y = -0.3363606$ | $\Delta v_z = -0.25700177$ |
| Velocity (km/s) | $V_x = 6.15881989$ | $V_y = -3.53014315$ | $V_z = 3.68479273$ |

| **Impulse 2** | $\lvert\Delta v_2\rvert = 0.8385$ km/s | Time = 23983 sec | |
|---|---|---|---|
| Position (km) | $R_x = -3856.9711$ | $R_y = -4864.0070$ | $R_z = -5113.5174$ |
| $\Delta v$ (km/s) | $\Delta v_x = 0.7388613$ | $\Delta v_y = -0.30378155$ | $\Delta v_z = 0.25483467$ |
| Velocity (km/s) | $V_x = 6.01158464$ | $V_y = -5.80367199$ | $V_z = 2.42682585$ |

| **Impulse 3** | $\lvert\Delta v_3\rvert = 3.2429$ km/s | Time = 39661 sec | |
|---|---|---|---|
| Position (km) | $R_x = -2661.9535$ | $R_y = 21898.8134$ | $R_z = 6930.3380$ |
| $\Delta v$ (km/s) | $\Delta v_x = -0.41878513$ | $\Delta v_y = -2.55623787$ | $\Delta v_z = 1.95113139$ |
| Velocity (km/s) | $V_x = -2.64723063$ | $V_y = -3.61736569$ | $V_z = -0.27890955$ |

| **Impulse 4** | $\lvert\Delta v_4\rvert = 0.1758$ km/s | Time = 63418 sec | |
|---|---|---|---|
| Position (km) | $R_x = 36941.0468$ | $R_y = 28974.8429$ | $R_z = -2180.1239$ |
| $\Delta v$ (km/s) | $\Delta v_x = 0.08433895$ | $\Delta v_y = -0.12608582$ | $\Delta v_z = 0.0889722$ |
| Velocity (km/s) | $V_x = -0.38285021$ | $V_y = 1.33742811$ | $V_z = 0.63327669$ |

| **Target (Falcon)** | | | |
|---|---|---|---|
| Position (km) | $R_x = 36941.0467$ | $R_y = 28974.8429$ | $R_z = -2180.1239$ |
| Velocity (km/s) | $V_x = -0.38285021$ | $V_y = 1.33742811$ | $V_z = 0.63327669$ |

Table 3.12: The information of each impulse (from LEO to HEO)

### 3.3.3 Examples #3 : 3-impulses from HEO (Falcon) to GEO (Cosmos-2350)

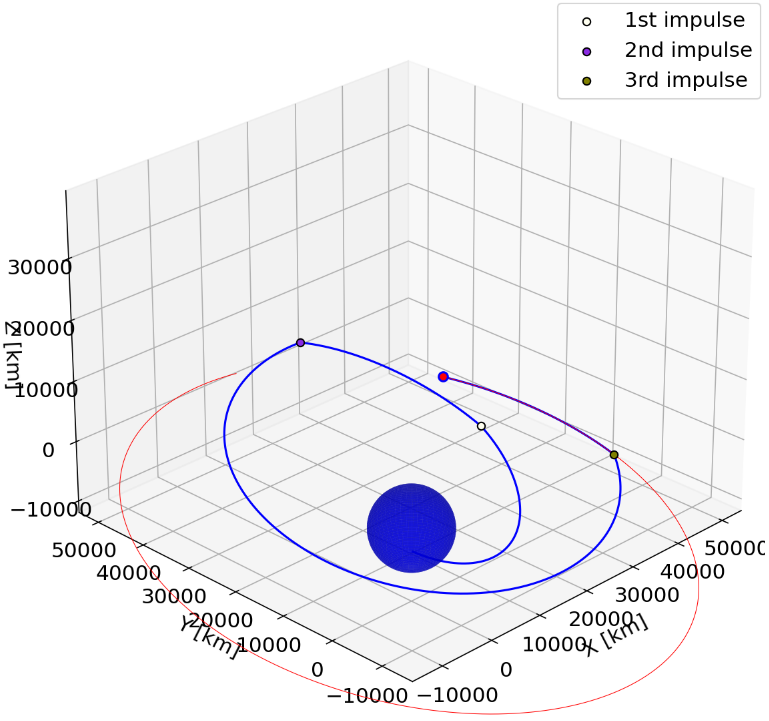

Figure 3.20: 3-impulses PSS orbit

The results displayed in Figure 3.20 represent the achievement of the PSS orbit through a 3-impulses orbit transfer from HEO with Falcon to GEO with Cosmos-2350. Looking at the results in Figure 3.21, we can determine the relative distance between the chaser and the target. After achieving the PSS orbit, the relative distance throughout the entire section is less than 10km, and similar to the first example, we can observe that the distance gradually decreases over time. For consistent observation, the relative distance must maintain a certain range; thus, this issue can be addressed as future work. Also, Table 3.13 provides detailed information at each impulse stage. Consistently, it can be confirmed that the chaser approaches the target using a minimum amount of fuel (Total $\Delta v$ = 3.1470 km/s) over the course of three impulses and achieves the PSS orbit using the final impulse at the position of the first encounter with the target.

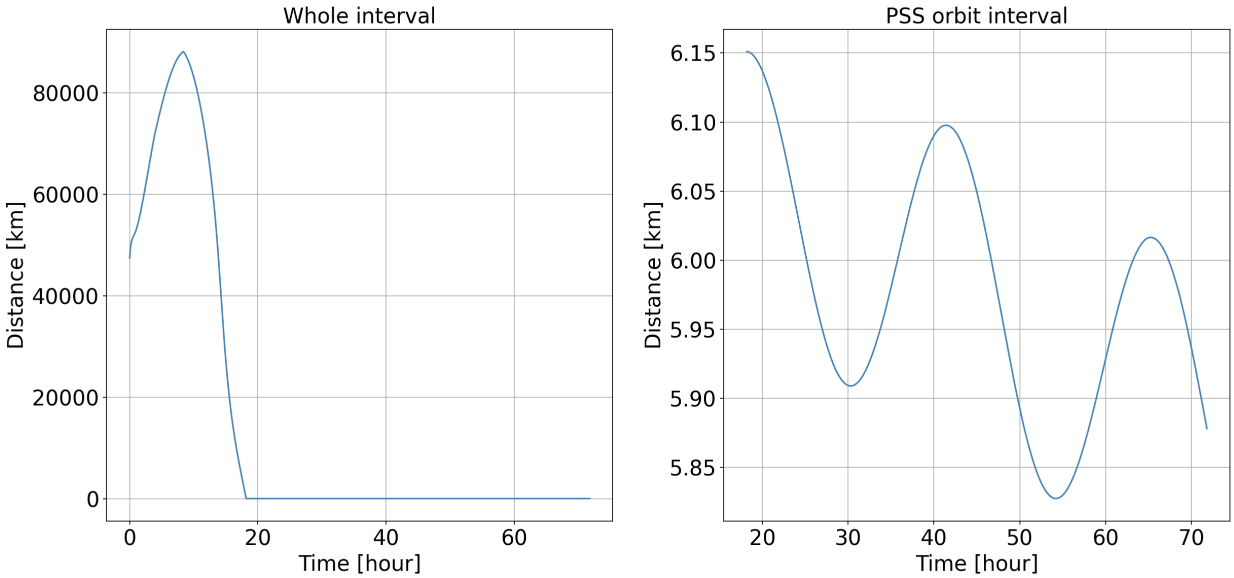

Figure 3.21: Distance between chaser and target (from HEO to GEO)

| **Impulse 1** | $\lvert\Delta v_1\rvert = 0.6179$ km/s | Time = 14042 sec | |
|---|---|---|---|
| Position (km) | $R_x = 37083.9520$ | $R_y = 21666.8390$ | $R_z = -4955.4068$ |
| $\Delta v$ (km/s) | $\Delta v_x = -0.27171912$ | $\Delta v_y = 0.54118044$ | $\Delta v_z = 0.12333019$ |
| Velocity (km/s) | $V_x = 0.0801171$ | $V_y = 2.37815288$ | $V_z = 0.68464445$ |

| **Impulse 2** | $\lvert\Delta v_2\rvert = 0.8221$ km/s | Time = 30257 sec | |
|---|---|---|---|
| Position (km) | $R_x = 21193.7215$ | $R_y = 44789.5200$ | $R_z = 6833.6512$ |
| $\Delta v$ (km/s) | $\Delta v_x = -0.44719415$ | $\Delta v_y = -0.46781232$ | $\Delta v_z = -0.507007$ |
| Velocity (km/s) | $V_x = -2.19460529$ | $V_y = -0.0813905$ | $V_z = 0.14625894$ |

| **Impulse 3** | $\lvert\Delta v_3\rvert = 1.7069$ km/s | Time = 65458 sec | |
|---|---|---|---|
| Position (km) | $R_x = 42021.9464$ | $R_y = -1147.2777$ | $R_z = -3307.5865$ |
| $\Delta v$ (km/s) | $\Delta v_x = -1.49297784$ | $\Delta v_y = 0.74612908$ | $\Delta v_z = 0.35782932$ |
| Velocity (km/s) | $V_x = 0.13220866$ | $V_y = 2.99985167$ | $V_z = 0.66056383$ |

| **Target (Cosmos-2350)** | | | |
|---|---|---|---|
| Position (km) | $R_x = 42021.9481$ | $R_y = -1147.2777$ | $R_z = -3307.5867$ |
| Velocity (km/s) | $V_x = 0.13221286$ | $V_y = 2.99985143$ | $V_z = 0.66056347$ |

Table 3.13: The information of each impulse (from HEO to GEO)

# 4. DISCUSSION

## Comparison between TFC and solve_ivp

In this research, a comparison was drawn between the performances of TFC and solve_ivp in the propagation of unperturbed orbits. An initial selection of 10,000 random orbits provided the basis for propagation over one time period. Findings revealed differences in both the initial and final position vectors and the initial and final velocities.

The experiment focused on the propagation of the Two-body problem equation, deploying both TFC and solve_ivp methods. The efficacy of each approach was gauged by contrasting the initial position and velocity vector with the final position and velocity vector post-orbit.

Results demonstrated that TFC displayed more precision than solve_ivp. However, a broader array of outcomes and a handful of less accurate results were observed. Such results could stem from the adaptability of solve_ivp, which can modify its methodology for each random orbit, as opposed to TFC's largely fixed approach. Interestingly, if TFC was to incorporate a more adaptive strategy, akin to solve_ivp, outcomes could potentially improve further.

The experiments also exhibited an intriguing discrepancy between the accuracy of position and velocity vectors. Operators showed approximately 1000 times more accuracy with velocity vectors. This might be due to the lower numerical values of velocity vectors compared to position vectors, suggesting that orbits with lower numerical position vectors could have higher accuracy.

Further investigation showed a difference in runtime between TFC and solve_ivp, with TFC completing tasks in approximately two-thirds the time taken by solve_ivp. This efficiency was especially noticeable in the Monte Carlo test, where TFC's speed proved a clear advantage, underscoring its suitability for environments requiring multiple test runs.

In conclusion, while both TFC and solve_ivp have their respective strengths and weaknesses, TFC demonstrated an edge in terms of accuracy and speed in this series of tests. However, room for potential improvement in TFC's adaptability suggests a dynamic future for the propagation of unperturbed orbits.

### Monte Carlo Test for PSS orbit

This study focused on a Monte Carlo test for Low Earth Orbit (LEO), Geostationary Orbit (GEO), and Highly Elliptical Orbit (HEO) examples. It sought to ascertain the range of parameter values for orbit shape, orbit orientation, and observation distance constraints to discover the possible satellite-to-satellite Persistent Space Surveillance (PSS) orbit. The test took place in two stages, with 10,000 random constraint values assigned in the first stage to assess the observation percentage within the range of two variables: orbit shape and orbit orientation parameters.

Lower values of both parameters linked to higher observation results, suggesting an inverse relationship between parameter values and observation results. Notably, the orbit orientation parameter appeared to have a more profound impact on observation results than the orbit shape parameter. Upon modifying parameter ranges based on first test results, the second test revealed zones where observation results reached 100%, confirming successful findings.

Observation results for GEO were consistent with those of LEO, with the same inverse relationship between parameter values and observation results. However, given the significantly larger size of GEO compared to LEO, it was inferred that parameter ranges had to be smaller in the GEO case. Interestingly, GEO results confirmed the importance of the orbit orientation parameter over the orbit shape parameter, as seen in LEO findings.

Results for HEO showed similarity to both LEO and GEO findings, with lower parameter values resulting in higher observation outcomes. However, as the orbit orientation parameter approached zero, observation outcomes diminished significantly, likely due to a decrease in observation distance and a subsequent increase in collision risk. This insight emphasizes the need to optimize observation performance and minimize collision risks for satellite operations in highly elliptical orbits.

On average, all three cases showed most of the Monte Carlo test results falling within the observation distance range of 0.01–10 km, satisfying the 100% observation result criterion. Histogram results revealed that most outcomes fell within the distance parameters, highlighting consistency and reliability in the testing process.

This research not only demonstrates the relative importance of different parameters in achieving optimal observation outcomes but also provides valuable insights into the unique characteristics of different orbital types. While LEO and GEO exhibit near-zero eccentricity, HEO is characterized by significantly higher eccentricity, which underscores the need for specialized strategies for different orbit types. Overall, the findings provide a robust basis for future work in satellite-to-satellite observation optimization.

### Sample Test for PSS orbit

This research study has meticulously examined the relative distances between a target spacecraft and a chaser over one orbital period. The datasets were created for each of the Low Earth Orbit (LEO), Geostationary Earth Orbit (GEO), and Highly Elliptical Orbit (HEO) instances, and the effect of third-body perturbations was incorporated to add real-world applicability.

For LEO, despite occasional discrepancies, it was found that most scenarios could maintain the observed distances between satellites within the desired range of 10 to 100 meters, allowing full observation during one orbital period. This consistent behavior and resilience of LEO orbits were attributed to their lower altitude. The analysis showed that the minor effects of the Moon's gravitational forces have no significant influence on the relative distance between satellites in LEO orbits.

However, for GEO, due to its higher altitude compared to LEO, the impact of the Moon's gravitational forces is more pronounced. Consequently, there was a rapid increase in the distance between the satellites after one orbital period, emphasizing the need for accounting for perturbations when dealing with higher altitude orbits like GEO.

When it came to HEO, the dynamics changed. The high eccentricity of these orbits caused significant velocity differences when the satellite is closer to or farther from Earth, making them more susceptible to the Moon's perturbations. In some scenarios, the relative distance between satellites fell below the minimum observation distance, hinting at potential collision risks. This underlines the necessity of considering the unique characteristics of different orbits for efficient space mission planning.

The initial position vectors and initial velocity vectors were found to be closely aligned with the target's initial values in all the orbit examples. This implies the possibility of satisfying the Persistent Space Surveillance (PSS) orbit requirements by propagating these initial values.

In conclusion, understanding the effect of third-body perturbations, especially from the Moon, on the relative distances between satellites in different orbits is critical. The varied behaviors of satellites in LEO, GEO, and HEO orbits highlight the importance of comprehensive analysis and accurate initial value determination for the successful execution of space missions. This research underlines the significance of these factors for efficient space mission planning and satellite design, particularly in the context of PSS orbits.

### $n$-impulses PSS orbit

This chapter sets out to investigate the transition from a current orbit to another orbit. This has important implications for practical applications in space engineering, such as spacecraft rendezvous and space debris removal.

A crucial problem discussed is reaching the PSS orbit from a chaser spacecraft using the smallest number of impulses, known as $n$-impulses. To minimize fuel usage, a Genetic Algorithm (GA) is applied, as outlined in the PCE research by Clocchiattis. However, this research distinguishes itself by focusing on achieving the PSS orbit, rather than the PCE orbit.

The process involved the selection of two satellites, with one acting as the chaser and the other as the target. The experiment was designed with three different scenarios, each using a varying number of impulses. This was done to find the optimal number of impulses for the transition.

The results of these experiments have potential far-reaching benefits. Achieving a successful orbit transfer, specifically to a PSS orbit, represents a significant step forward in space exploration. This could lead to more efficient trajectory optimization and management for satellites. Additionally, using a GA to minimize fuel consumption could revolutionize sustainable space travel by making maneuvers more cost-effective, a critical consideration for both scientific and commercial space missions.

This experiment featured three different orbit transitions: from LEO to GEO, from LEO to HEO, and from HEO to GEO. In all three scenarios, the experiments resulted in the chaser satellite successfully reaching the PSS orbit with the target satellite. The relative distance between the chaser and the target was observed after achieving the PSS orbit. In most cases, the distance remained within an observable range of less than 10km, suggesting that the chaser could continually monitor the target.

However, it was noted that over time, the distance between the two satellites gradually decreased. This could be adjusted with additional impulses, depending on the direction of the problem at hand. Moreover, detailed information about the impulses was recorded, including the time and position vector at each impulse point, along with the resulting velocity vector. This demonstrated that the chaser approached the target using a minimal amount of fuel over the course of several impulses, ultimately achieving the PSS orbit at the point of the initial encounter with the target.

Overall, the results underscore the feasibility of achieving the PSS orbit through an $n$-impulse orbit transfer. Furthermore, they highlight the importance of using GAs in minimizing fuel consumption. This could greatly enhance the sustainability and efficiency of future space missions, ultimately leading to significant advancements in the field of space exploration.

# 5. CONCLUSIONS

In recent years, research into tracking targets in space has grown in importance due to its significant commercial and military applications, especially considering the increasing number of satellites orbiting Earth. Among these challenges associated with target-chaser dynamics, Periodic Close Encounter (PCE) problems have been extensively studied to understand how to periodically encounter targets. The question of how to transition to a PCE orbit was also explored and confirmed in this research.

We extended the scope of this research by augmenting the PCE problem to facilitate persistent surveillance. Persistent Space Surveillance (PSS) orbits, which continuously track targets while mitigating the risk of collision, were introduced. The investigation of these PSS orbits was conducted in two phases. Initially, Genetic Algorithm (GA) was employed with three constraints to obtain optimal orbital elements proximate to the target's orbit. Subsequently, the results were verified through Theory of Functional Connections (TFC) propagation based on the orbital elements influenced by third-body perturbations.

The findings of this study confirm that persistent surveillance of a target over multiple orbit cycles is not feasible due to the influence of perturbations. Therefore, future research related to PSS orbits will need to address the challenge of perturbation effects by applying impulses on the objects, considering the additional costs incurred.

The concept of $n$-impulses PSS orbit introduced and tested in this study contributes not only to our understanding of PCE and PSS orbits but also provides a practical approach for achieving PSS orbits with minimal fuel consumption. This $n$-impulses PSS orbit approach could potentially enhance the efficiency and feasibility of maintaining persistent surveillance of a target in the space environment, thereby significantly advancing the field of space tracking and surveillance.

In conclusion, this study is expected to lay a solid foundation for the developmental trajectory of tracking and maintaining surveillance of objects in space, thereby contributing to the broader endeavour of space exploration and satellite tracking.

# REFERENCES

[1] A. Clocchiatti and D. Mortari, “Responsive space surveillance using periodic close encounters,” in *7th Dynamics and Control of Systems and Structures in Space Conference*, 2006.

[2] B. Lal, A. Balakrishnan, B. M. Caldwell, R. S. Buenconsejo, and S. A. Carioscia, “Global trends in space situational awareness (ssa) and space traffic management (stm),” tech. rep., INSTITUTE FOR DEFENSE ANALYSES WASHINGTON DC, 2018.

[3] S.-I. Nishida, S. Kawamoto, Y. Okawa, F. Terui, and S. Kitamura, “Space debris removal system using a small satellite,” *Acta Astronautica*, vol. 65, no. 1-2, pp. 95–102, 2009.

[4] T. J. Muelhaupt, M. E. Sorge, J. Morin, and R. S. Wilson, “Space traffic management in the new space era,” *Journal of Space Safety Engineering*, vol. 6, no. 2, pp. 80–87, 2019.

[5] M. Rezaee, M. Mahdianpari, Y. Zhang, and B. Salehi, “Deep convolutional neural network for complex wetland classification using optical remote sensing imagery,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 9, pp. 3030–3039, 2018.

[6] Y. Xie, Y. Lei, J. Guo, B. Meng, Y. Xie, Y. Lei, J. Guo, and B. Meng, “Autonomous guidance, navigation, and control of spacecraft,” *Spacecraft Dynamics and Control*, pp. 371–422, 2022.

[7] C. Sun, X. Wang, H. Qiu, and Q. Zhou, “Game theoretic self-organization in multi-satellite distributed task allocation,” *Aerospace Science and Technology*, vol. 112, p. 106650, 2021.

[8] A. D’Ambrosio, T. Henderson, A. Clocchiatti, and D. Mortari, “Constrained optimization of n-impulse periodic close encounter orbits for inspection missions,” *Advances in Space Research*, vol. 70, no. 11, pp. 3393–3404, 2022.

[9] Y. H. Kim and D. B. Spencer, “Optimal spacecraft rendezvous using genetic algorithms,” *Journal of Spacecraft and Rockets*, vol. 39, no. 6, pp. 859–865, 2002.

[10] O. Abdelkhalik and D. Mortari, “Orbit design for ground surveillance using genetic algorithms,” *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 5, pp. 1231–1235, 2006.

[11] O. Abdelkhalik and D. Mortari, “Reconnaissance problem using genetic algorithms,” *Paper AAS*, pp. 05–184, 2005.

[12] G. A. Rauwolf and V. L. Coverstone-Carroll, “Near-optimal low-thrust orbit transfers generated by a genetic algorithm,” *Journal of Spacecraft and Rockets*, vol. 33, no. 6, pp. 859–862, 1996.

[13] P. Cage, I. Kroo, and R. Braun, “Interplanetary trajectory optimization using a genetic algorithm,” in *Astrodynamics Conference*, p. 3773, 1994.

[14] H. Johnston and D. Mortari, “Orbit propagation via the theory of functional connections,” in *Proceedings of the 2019 AAS/AIAA Astrodynamics Specialist Conference, Portland, ME, USA*, pp. 11–15, 2019.

[15] F. Criscola, D. Canales, and D. Mortari, “Solution of the perturbed lambert’s problem using the theory of functional connections,” in *33rd AAS/AIAA Astrodynamics Specialist Conference*, pp. 23–429, August, 13-17, 2023.

[16] D. Mortari, H. Johnston, and C. Leake, *The Theory of Functional Connections*. Elsevier, 2022.

[17] D. Mortari, “Least-squares solution of linear differential equations,” *Mathematics*, vol. 5, no. 4, p. 48, 2017.

[18] D. Mortari, H. Johnston, and L. Smith, “High accuracy least-squares solutions of nonlinear differential equations,” *Journal of computational and applied mathematics*, vol. 352, pp. 293–307, 2019.

[19] D. A. Vallado, *Fundamentals of astrodynamics and applications*, vol. 12. Springer Science & Business Media, 2001.