UNIVERSIDADE D
COIMBRA

Afonso Fernandes Simões

SPACE DEBRIS MONITORING AND SUPERVISION SYSTEMS

Dissertation in the context of the Master in Astrophysics and Instrumentation for Space oriented by Doctor Alexandre Correia and Doctor Miguel Bergano and presented to the Department of Physics of the University of Coimbra

June 2023

Acknowledgements

First, I would like to thank to Doctor Bruno Coelho and Doctor Rui Silva for accepting being member of the jury.

Second, I would like to pay my special regards to Doctor Miguel Bergano for giving me the opportunity to develop my work at Atlar Innovation and for my coordinator in the university, Doctor Alexandre Correia, for the help given during the development of my master thesis.

On a more personal note, I would like to thank my parents, João Simões and Sonia Simões. To my father for always remembering me that the day has 24 hours, there is time for everything. To my mother for being my emotional support during my ups and downs.

To my grandmother, avó Mimi, I want to thank for always taking care of me since I was little. From the point of trying to ask water by dragging a gallon of water to the point of telling stories of my academic life.

A special thanks to my friends Javier Hernandez and Rafael Vitoria for all the support during my academic life.

Last but not least I would to thank to my best friend Pedro Carrasco, for the great moments during the year of the development of the master thesis and for the relaxing evenings after a long day of work.

Abstract

With the increasing density of space debris in the Earth's skies and the consequences carried with this growth, it was necessary to create a system to monitor the debris. The organization Europe Union Space Surveillance and Tracking (EUSST) was established to unite several countries in Europe, being one of those countries Portugal, to work together through this problem. To be able to supervise and monitor the debris, several techniques of observation had to be used. With the observational data, the orbit determination of the debris can be calculated. Each one of the debris is subject to different types of forces that can influence its trajectory and different statistical algorithms exist to deal with the measurements obtained by the observations. Hence, an equilibrium between the correct choice of forces applied to the debris and the appropriate statistical algorithm for a correct orbit determination is necessary. This master thesis explores all these variables applied to the orbit determination and orbit prediction, although the main focus will be the debris in Low Earth Orbit (LEO), since the high density in the number of debris and its high chances of reentrance brings great risk to human life. Modifications were made to the software in order to represent different calculations of orbit determination and a couple of scripts were developed for the representation of the generated data by the software. As a result, a representation of the data, an analysis and an understating of the different results are accomplished.

Keywords: Orbit determination, space debris, orbit prediction

Resumo

Com o aumento da quantitade de detritos espaciais nos céus e as consequências que este crescimento acarreta, foi necessário criar um sistema pronto a monitorizar esses detritos. A organização Europe Union Space Surveillance and Tracking (EUSST) foi estabelecida para unir vários países da Europa, sendo um desses países Portugal, para trabalharem em conjunto para resolver este problema. Para poder supervisionar e monitorizar os detritos, várias técnicas de observação são utilizadas. Com os dados dos resultados das observações, a determinação da órbita dos detritos pode ser calculada. Cada um dos detritos está sujeito a diferentes tipos de forças que podem influenciar a sua trajectória e existem diferentes algoritmos estatísticos para lidar com as medições obtidas pelas observações. Assim, existe um balanço entre a escolha correcta das forças aplicadas aos detritos e o adequado algoritmo estatístico para uma determinação correcta da órbita. Esta tese de mestrado explora todos estes variáveis aplicadas à determinação e previsão de orbita, embora o foco principal sejam os detritos em (Low Earth Orbit) LEO, uma vez que a grande quantidade de detritos e o seu elevado risco de reentrada traz grandes riscos para a vida humana. As alterações foram efectuados ao software a fim de representar diferentes cálculos de determinação da órbita e diversos scripts foram desenvolvidos para a representação dos dados gerados pelo software. Como resultado, foi desenvolvido uma representação dos dados, uma análise e uma subavaliação dos diferentes resultados.

Palavras-chave: Determinação de orbita, detritos espaciais, previsão de orbita

Table of contents

List of figures

List of tables

Abbreviations and Acronyms

API Application Programming Interface

ASAT Anti-satellite weapons

ASCII American Standard Code for Information Interchange

COSPAR Committee on Space Research

CPF Consolidated Prediction Format

CRD Consolidated laser ranging Data Format

ECEF Earth-fixed coordinate system

ECI Earth-centered inertial

EKF Extend Kalman Filter

ESO European Southern Observatory

EUSST Europe Union Space Surveillance and Tracking

FITS Flexible Image Transport System

GCRF Geocentric Celestial Reference Frame

GEO Geostationary orbit

GLONASS Global Navigation Satellite System

GPS Global Positioning System

IOD Initial Orbit Determination

ITRF International Terrestrial Reference System and Frame

JSON JavaScript Object Notation

LEO Low Earth Orbit

LRA Laser retroreflector arrays

LS Least Square

MEO Medium Earth orbit

NED North East Down

NPI NASA Predictive Information

OD Orbit determination

OMM Orbit Mean-Elements Message

RCS Radar Cross Section

RSO Resident Space Object

RSW Radial-track, along-track and cross-track

SDSS Sloan Digital Sky Survey

SLR Satellite laser ranging

SPG4 Simplified General Perturbations 4

STS Space Transportation System

TCA Time of Closest Approach

TDM Tracking Data Message

TEME True Equator Mean Equinox

TLE Two-line element set

Chapter 1

Introduction

Since the beginning of mankind, humans have looked into the skies to search for the answers to the unknown. From being capable to define the calendar year to being able to navigate through our vast seas with the help of the stars, mankind always tries to find answers to the unknown.

In the year 1957, the frontiers of the skies were crossed and with the first artificial satellite being launched, Sputnik, was created a new era. Following this first artificial satellite launched by the Soviet Union, many countries challenged this barrier. There have been thousands of more objects, including active satellites, inactive satellites, and debris, launched since the first satellite was sent into orbit around the Earth. Among these objects, the greater number of objects orbiting the Earth is debris. Debris are any kind of objects made by humans that orbit the Earth and doesn't have a purpose anymore. One example of debris is a space rocket that was left orbiting the Earth after its launch.

The rise in the number of debris create several problems, such as collisions that can damage active satellites and atmospheric reentry. For most cases of atmospheric reentry, the debris will burn down before they reach the surface of the Earth. The size of the debris can range from lower than 1 mm to higher than 10 cm. The bigger debris and inactive satellites, can reach the surface of the Earth and provoke damage to humans. Looking at the case of the uncontrolled reentry of the Chinese satellite ([42]) in the year 2022, a big part of the satellite reach the surface of the Earth but with some luck, this remaining part fell into the sea, not causing any damage to the human population. This recent case was a big recall of how increasing efforts between countries have to be accomplished to solve this problem.

The prevision and determination of the trajectory of these objects is essential for a calculus of the risk assessment the objects can present. If world cooperation is achieved and the data captured by the observation is shared between the countries, debris and orbits can be calculated much faster. This work studies methods for supervising and monitoring objects orbiting Earth

1.1 Objectives

The main goal of this master's thesis is to study the feasibility of the systems used to supervise and monitor the debris in orbit of the Earth. In order to do this, Initial Orbit Determination (IOD), Orbit determination (OD) and propagation are carried out, and the viability of the outcomes is examined. As for the main focus of the type of object chosen in this master thesis, an example of LEO is analyzed.

The reason for the main focus being Low Earth Orbit (LEO) objects is due to the higher density of active, derelict satellites and debris on this kind of orbit.

As for the analysis of the correct propagation of the orbit of the object, a comparison of the predicted position with the position in the Consolidated Prediction Format (CPF) will be performed to study the feasibility of the orbit determination done and the propagator offered by the software used. An explanation of what a CPF file is may be found in the introduction 3 of chapter 3.

In figure 1.1 a simplified version of the steps done to produce the results is represented.

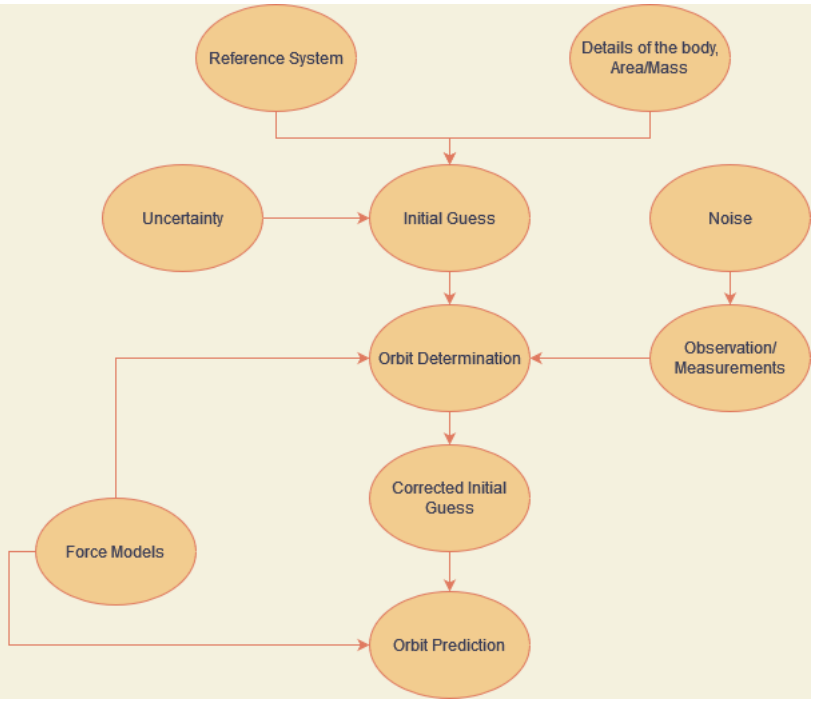


Fig. 1.1 Flow Diagram for orbit prediction simplified steps

1.2 Motivation

The rise of the population of space debris is transforming space exploration into a more dangerous activity, in the present, and in the future. If this scenario does not change, it can jeopardize access to space for future generations. Supervising and monitoring have become one of the main steps to change this scenario. Knowing the trajectory of the space debris gives the chance of taking action to fix the problem, like removing debris from orbit, having a controlled reentry, or just changing them to a graveyard orbit.

1.3 Thesis Content

The first chapter is devoted to a better understanding of the space debris situation and its rapid increase, followed by the available orbit propagation and orbit determination methods along with the work developed during the internship. The accuracy and computational requirements for the chosen methods

were thoroughly examined to see if they were met. The following list with a brief description of the contents of the chapters is presented:

- `Chapter 2`: Monitor and Supervising Systems provide the essential background knowledge about the present issue, its features, available methods for obtaining information for a catalog maintenance system. It also describes the essential astrodynamics concepts need it for a better understanding of the following chapters. This is done while showcasing cutting-edge approaches for uncertainty propagation, orbit determination, and orbit propagation.
- `Chapter 3`: Explanation of the process to produce an orbit determination, as the final result.
- `Chapter 4`: Explanation of how to accomplish a prediction of orbit and the showcase of the results produced for the object used in this master thesis.
- `Chapter 5`: Conclusion and Future Work while summarizing the text discussed in this master thesis.

Chapter 2

Monitor and Supervising Systems

The concepts needed to understand the issues brought by space debris are introduced in this chapter. The chapter starts with a brief discussion of the space objects population, along with the primary observing methods used in space surveillance and monitoring. Orbit definition and prediction algorithms for satellites, as for other space objects, are presented in this chapter. The main objective of this chapter is to provide a succinct overview of the key concepts required for a better understanding of the following chapters.

2.1 Population of Space Objects

With space exploration accelerating in the last few years, the number of objects orbiting the Earth has increased. The increasing population of space objects leads to congestion of the space area around the Earth and with such congestion, can preclude future space missions. Understanding the evolution and distribution of these objects will benefit not just future space missions but the avoidance of future collisions between the objects. One collision, regardless of the type of collision, releases multiple fragments. These fragments can provoke another collision that will release another group of fragments and so on. The fragments can vary in size as well as in the density of their population. There are about 23,000 pieces ([27]) of debris larger than 28 cm orbiting the Earth, half a million of size up to 1 cm or larger, and about 100 million pieces of debris of about 1 mm. The main events of collisions are shown in table 2.1. This event sequence has to be avoided.

Another benefit of the understating of the distribution of space objects is the risk analysis of reentries of space objects in the Earth's atmosphere. When an object of a certain size starts reentering, it can reach the surface of the Earth and create damage to human life or to infrastructures. So for this specific case, a tracking plan and observation of the debris takes a great responsibility so it can prevent this event of reentry.

Figure 2.1 displays the progression over time of all the space objects orbiting the Earth, including debris, and fragments related to past collisions.

Kosmos 2251 collision with Iridium 33 [79] was the principal event that shows how the lack of management of satellite traffic can result in a collision that can generate a huge amount of fragments.

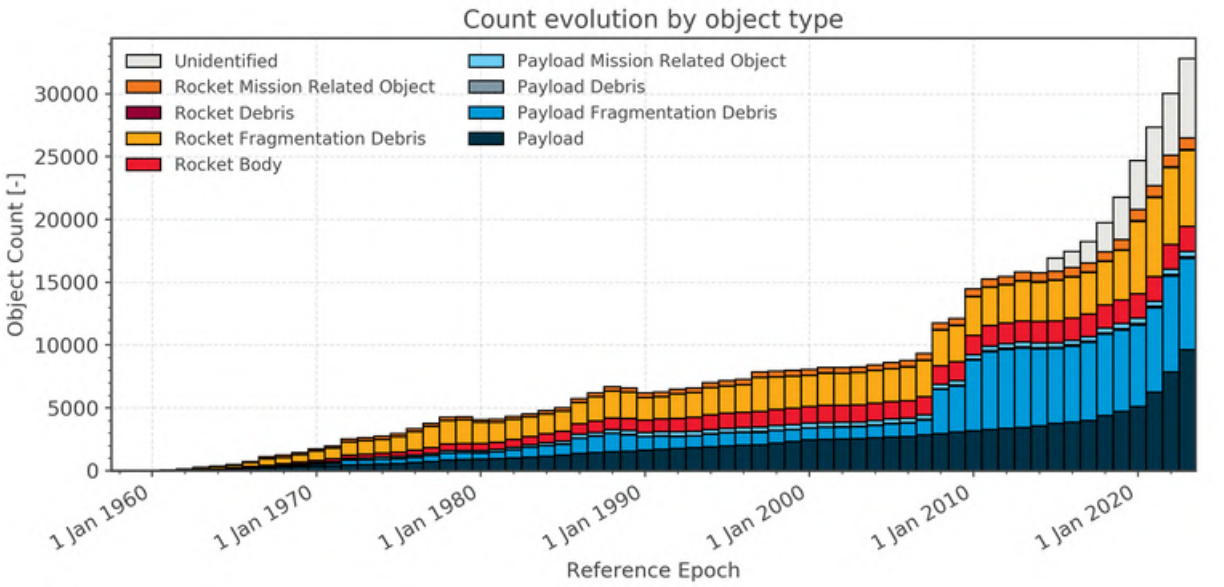


Fig. 2.1 Increasing number of debris taken from ([20])

Table 2.1 Top 10 Events Collisions ([81])

Top instances of debris creation, March 2023			
Name of the satellite	Year	Number of debris	Type of collision
Fengyun-1C	2007	3549	Intentional collision (Anti-satellite weapons (ASAT))
Kosmos 2251	2009	1716	Accidental collision with Iridium 33
Kosmos 1408	2021	1562	Intentional collision (ASAT)
STEP 2 Rocket Body	1996	756	Residual propellant explosion
Iridium 33	2009	659	Accidental collision with Kosmos 2251
Kosmos 2421	2008	511	Disintegrated
SPOT 1 Rocket Body	1986	506	Accidental explosion
Parus	1981	482	Battery explosion
OV2-1 Rocket Body	1965	473	Engine explosion
Nimbus 4 Rocket Body	1970	465	Residual propellant explosion

From all these fragments created and the loss of control in some satellites, the reentries of these objects may occur and may bring damage to human life. Table 2.2 shows the main occurrences of reentries.

Table 2.2 Top 10 Reentries of objects ([80])

Largest Objects to Reenter, November 2022				
Mir	Russia	120,000 kg	23 MAR 2001	Controlled
Skylab	USA	69,000 kg	11 JUL 1979	Uncontrolled
Salyut 7/Cosmos 1686	USSR	40,000 kg	07 FEB 1991	Uncontrolled
S-II Stage / Skylab	USA	36,200 kg	11 JAN 1975	Uncontrolled
Space Transportation System (STS) external	USA	35,000 kg	1981 (1981–83)	Partially Controlled
Salyut 6/Cosmos 1267	USSR	35,000 kg	29 JUL 1982	Controlled
Long March 5B core (5B-Y1 flight)	China	21,600 kg	11 May 2020	Uncontrolled
Long March 5B core (5B-Y2 flight)	China	21,600 kg	9 May 2021	Uncontrolled
Long March 5B core (5B-Y3 flight)	China	21,600 kg	30 July 2022	Uncontrolled
Long March 5B core (5B-Y4 flight)	China	21,600 kg	NOV 2022	Uncontrolled

2.2 Space catalog

A space object catalog is a group of orbital elements (or states) that have been generated through measurements. Measurements obtained by observations made by a space surveillance network are used to update the status of space objects on a regular basis. The process of updating an object's orbit information based on observations or adding a new object is referred to as catalog maintenance.

For operational and scientific purposes, several space organizations maintain their own databases. EUSST organization also maintains its own database. Typical tasks done to keep updated the space object catalog are:

- processing of observations: object correlation, identification of tracks with objects, and update of cataloged objects (orbit determination);
- evaluation of objects conjunctions: propagation of covariance and uncertainty;
- collision risk analysis;
- sensor tasking (cooperation on tracking objects and distribution of objects to observe);
- lifetime predictions.

2.3 Orbital space

The majority of human space activities occur between 100 km and 36,000 km in altitude. This is generally divided into three major zones ([41]).

- Low Earth Orbit (LEO): is widely understood to reside between 100 and 2,000 kilometers above Earth's surface, with several anthropogenic space objects (ASOs) at about 500 kilometers. Military, earth observation, and scientific missions have traditionally dominated LEO, along with some communications systems. The satellites cross the entire sky in a few minutes when viewed from Earth, the orbital period at these heights is approximately 90–120 minutes. The atmospheric drag in this region causes orbits to degrade, although the rate of degradation varies greatly over a wide range of altitudes. The degradation of the orbit implies that the objects in orbit gradually lose altitude and eventually reenter the Earth´s atmosphere. The rate of deterioration indicates that the speed of decay of the item might vary depending on the altitude of the satellite.

- Medium Earth orbit (MEO): is found at elevations ranging from 2000 kilometers to 35,786 kilometers. This regime governs the operation of global navigation satellite systems such as Global Positioning System (GPS), Global Navigation Satellite System (GLONASS) and Galileo. Each MEO satellite can communicate with a significant portion of the Earth's surface and circles the planet in around 12 hours. An object in MEO will move much more slowly across the sky and will be visible for an hour from the perspective of a person on Earth.

- Geosynchronous Orbit Geostationary orbit (GEO): is 35,786 kilometers above mean sea level, where the orbital period coincides with the rate of rotation of the Earth around its spin axis. A GEO object will appear to a person on Earth as a weak, steady point source in the sky. A significant portion of the Earth's surface can observe GEO satellites. Historically, communications satellites, such as those that offer internet or phone services to far-flung areas, have been positioned in this orbit. Sending a signal from Earth to a satellite in GEO and back takes at least 0.24 seconds.

A representation of the Earth orbital classes is seen in figure 2.2.

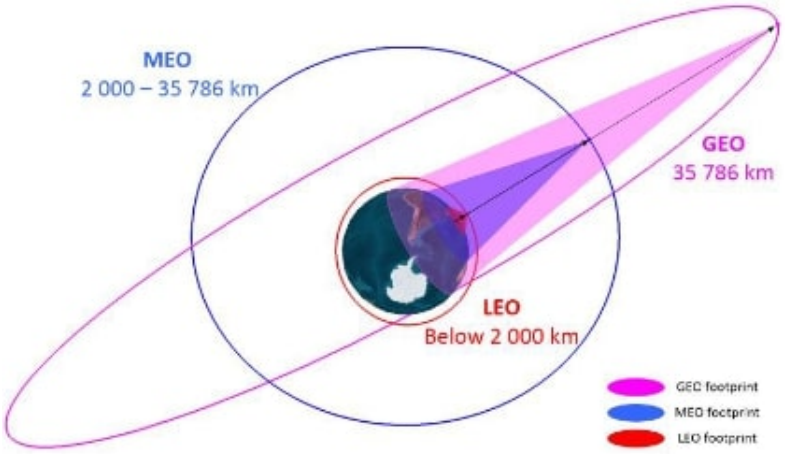


Fig. 2.2 Earth Orbital classes taken from ([34])

2.4 Techniques of observations

There are suitable techniques of observation, for each kind of orbit. At the present time, space debris are mainly monitored and supervised by ground-based stations. In this category, the types of observations can be divided into optical, radar and satellite laser ranging. The radar is preferable for objects in LEO, due to the high speed of these objects. This high speed can lead to optical observations not being able to track the objects in LEO. The radar observations are not limited by the weather and visibility conditions to perform observations. However, due to the target signal loss being proportional to the fourth power of the distance ([3]), the detection range of the radar is usually limited to low orbit. Satellite laser ranging has higher accuracy than radar although SLR observations are limited by weather and visibility conditions. For MEO and GEO satellites, SLR, and optical are the appropriate techniques to use due to their higher range. These techniques are divided into tracking and surveying when doing the observations. Tracking consists in following an object through its orbit and survey is characterized as pointing to the sky and trying to locate a new object. Different types of observation result in different types of files with the result of information of the observation. SLR observations come in the format of Consolidated laser ranging Data Format (CRD) files. These files contain information on the measurements and the bias applied to each measurement depending on the station doing the observation.

Figure 2.3 shows an example of how an SLR observation works on a satellite with retro-reflectors. Retro-reflectors is a device implemented in some satellites that reflects the radiation back to its source with the minimum scattering. An explanation of the procedures mentioned above is provided in the following sections.

2.4.1 Method SLR

Satellite laser ranging stations keep track of how long it takes for a signal to travel from the station to the target satellite and back again. Alternatively, the timing measurement's output of the entire ("2-way") time of flight can be reported as a range value as shown in equation 2.1

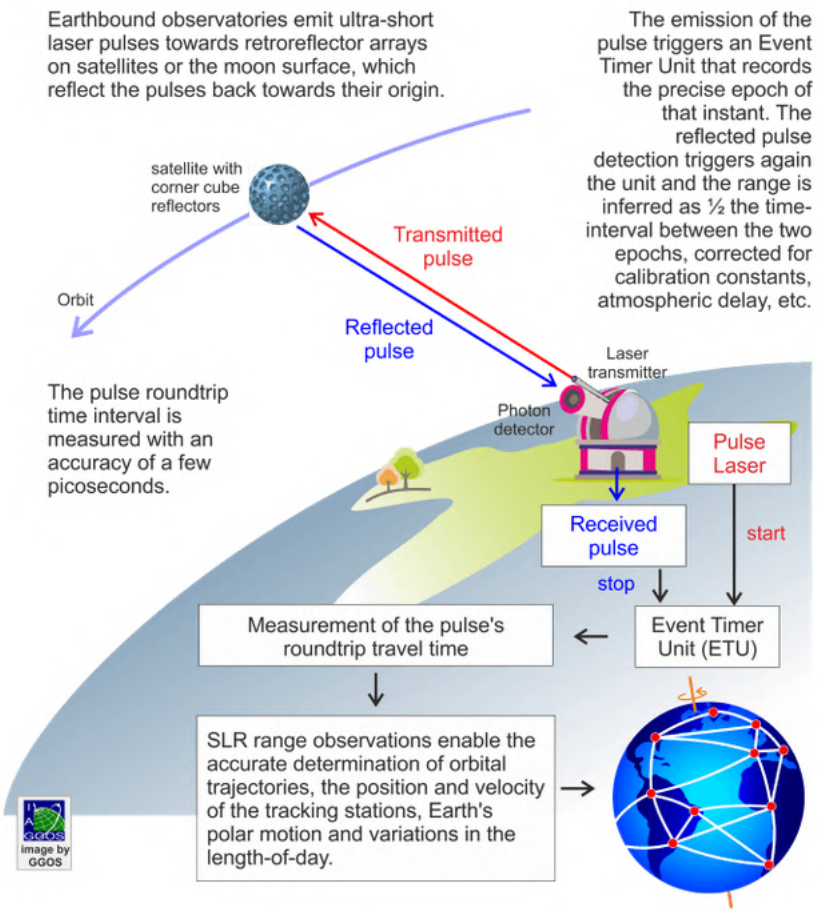


Fig. 2.3 Satellite laser Ranging observation taken from ([71])

$$\rho = \frac{1}{2} * c\tau \tag{2.1}$$

where c stands for light's vacuum speed, ρ for distance and τ for travel time. This distance is effectively the "1-way" distance between the object and the SLR ground station, although more accurate modeling requires taking into account other signal propagation factors.

Range

Given the CRD, the range measurement can be described by the following equation ([6]):

$$\rho = \frac{1}{2} * (\rho_{up}(t) + \rho_{down}(t) + \Delta\rho_{rel} + \Delta\rho_{trop} + \Delta\rho_{LRA} + \varepsilon) \tag{2.2}$$

where ρ_{up} and ρ_{down} are the (light-time corrected) up and downlink ([72]) distances between the station and the satellite. The relativistic correction $\Delta\rho_{rel}$ accounts for the range delay caused by the gravitational deflection of light near the Earth, where Laser retroreflector arrays (LRA) is a specific range correction that accounts for the difference in distance between the station and the actual signal route within the LRA ([69]). The variable ρ_{trop} represents the tropospheric range delay. The variable $\Delta\rho_{LRA}$ is an LRA-specific range correction that considers the distance between the station and the LRA reference point as well as the actual signal route within the LRA. The ε includes satellite and station-specific systematic imperfections that impact the measured range, such as orbit offsets, range biases, and range errors introduced by station timing offsets

Having the correct range values the position vector can be obtained. When the range-rate measurements are made available, the velocity vector can be obtained. In the case of just range measurements, different methods can be applied to obtain the full state vector, being the state vector the vector with the position and the velocity vectors.

However, when the observations with this technique are made to non-cooperative targets, being objects without retro-reflectors, the way of processing the reflected radiation is modified. For this diffuse reflection laser ranging is used ([70]). This technique uses a bigger amount of energy in the laser and brings a lower accuracy with a precision ranging between 50cm-250cm.

If the observation are made to targets with retro-reflectors, it can reach a precision of 3 mm ([66]).

2.4.2 Radar method

Although the accuracy of SLR is of higher precision, radar observations are still the most common and the ones that can work independently of the weather conditions and are prepared to work with any kind of debris or objects.

The antenna of the radar sends a beam of electromagnetic energy in the direction of a target ([26]).

If any energy is reflected, the reflected beam will be detected by a receiver, which will calculate the time between the sending signal and the received signal and the direction of the signal received ([45]). Radar observations are based on azimuth, elevation, range, and Doppler measurements that are obtained after the signal has been processed ([13]). The distance between the radar and the target can be calculated using the equation 2.1.

Fig. 2.4 Telescope Portugal taken from ([1])

Due to the relative velocity between the wave's source and the observer, a phenomenon known as the Doppler effect causes an increase or reduction in the apparent frequency of a wave and so the radial velocity of the object can be calculated. The siren of a passing ambulance is the most known example of this frequency shift, although it can also happen when receiving light and radio signals ([35]). Additionally, using post-processing methods ([47]), angle rates and range rates can be determined.

2.4.3 Optical method

Optical observations return angular measurements and not range measurements as the previous methods. This kind of observation works better for MEO and GEO satellites because in LEO the satellites and debris pass too fast([55]) in most of the cases, giving a limited time window that could give a not as accurate results. Optical observations are limited to weather conditions ([21]) and are performed only during nighttime. Figure 2.4 shows the telescope used by Portugal to perform the observation of space debris, located in Terceira Island.

2.4.4 Advantages and Disadvantages of the Observation methods

Due to the radar and SLR observations not being limited by the time window of the debris passing in LEO, they are preferred for LEO. In the case of laser ranging that is not limited by distance, which is not the case for radar measurements, for MEO and GEO are preferable to do laser observation when tracking an object in further orbits. Despite being constrained by weather conditions, optical observation can yield more accurate angular measurements than radar.

Nevertheless, as mentioned before, laser observation work really well for an object with retro-reflectors but not perfectly with objects that don't have them. Then, depending on the types of objects

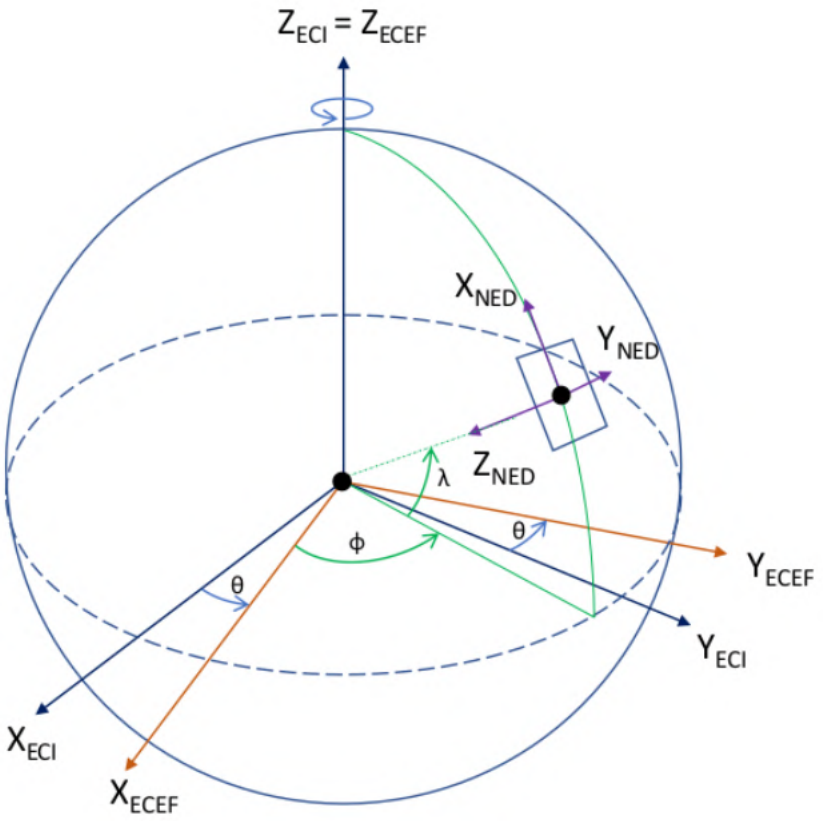


Fig. 2.5 North East Down (NED), Earth-centered Earth-fixed (ECEF), and Earth-centered inertial (ECI) reference frames taken from ([23])

which are needed to be monitored, rethinking of a choice of which observations to use, has to be made.

All of the observation types discussed above will yield coordinates in the position of the objects observed. The coordinates of the objects can be normally represented in Cartesian coordinates or elliptical elements. In both coordinates, we can express them in several orbital reference frames depending on the application. The section that follows will discuss some of the orbital reference frames in which the objects can be represented.

2.4.5 Orbital Reference Frames

There are two types of reference frames, Inertial and non-Inertial frames ([78]). A reference frame that is not accelerating is said to be inertial. The non-inertial reference frame accelerates. For satellites orbiting the Earth, the main reference frames used are the Earth-centered inertial (ECI), the Earth-fixed coordinate system (ECEF) and the North East Down (NED) systems. In figure 2.5 an illustration of the frames is represented where the coordinates followed by ECEF correspond to the non-Inertial frame and ECI to the inertial.

Because the equations of motion are most easily solved in an inertial frame, whereas force models or measurements are applied with respect to non-inertial frames, such as a co-rotating Earth-fixed frame, the usage of several coordinates and time systems is inevitable in orbit propagation.

To deal with the necessary coordinates and time systems at an early stage of development, this process is crucial. It is only advised to use standardized procedures at this point because the orbit propagator is designed to receive input states connected to a specific coordinate frame and output the result again with respect to that frame.

In orbital mechanics and spaceflight dynamics, a common coordinate system for describing how an object moves in relation to a central body is called the Radial-track, along-track and cross-track (RSW) reference frame. With mutually orthogonal axes and an origin of the reference system at the center of the object, it is a three-dimensional Cartesian coordinate system.

The following definitions apply to the RSW frame's three axes:

- Radial axis: The direction pointing away from the center of the central body is referred to as the radial axis, and it is typically aligned with the position vector of the object with respect to the central body. As it points away from the central body, the radial direction is also occasionally referred to as the "outward" direction;
- Along-track axis: The along-track axis is the direction that is tangent to an object's orbit and that, at a specific point in the object's orbit, points in the direction of the object's velocity vector; The object's orbital period and shape are both influenced by the along-track orientation, making it crucial;
- Cross-track axis: To complete the mutually orthogonal RSW frame, the cross-track axis is defined as the direction that is perpendicular to both the radial and along-track directions; It points in the direction of the radial and along-track directions' cross product; The direction of the cross-track is crucial because it dictates how the orbit of the object is tilted with regard to the central body and how extraneous perturbations, such as gravitational pulls from other objects, affect the orbit.

In figure 2.6 represents a sketch of this coordinate system.

2.5 Initial orbit determination

When an orbit is not known, Initial Orbit Determination (IOD) generates position and velocity using satellite tracking data. The results of the IOD method, which presuppose two-body dynamics, are quite susceptible to observational mistakes. There are numerous methods to calculate the initial orbit. Some of these methods are Gibbs, Gooding, Laplace, Herrick-Gibbs, and SITE-TRACK([54]) ([22]). For the estimation of an initial orbit, some of the techniques need a certain amount of information pertinent to the object. Position vectors or angles of the observations are some of the information needed. The angles can be azimuth and elevation (local coordinates) or right ascension and declination, (equatorial coordinates) of the object. Further explanation of the algorithms can be found in the book of Vallado ([74]).

One method for IOD commonly used for radar measurements is the Herrick-Gibbs method.

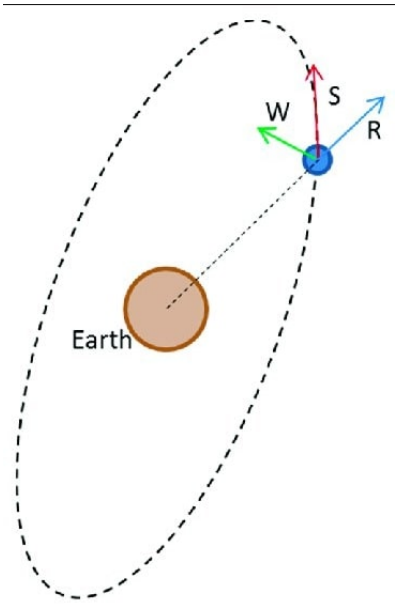


Fig. 2.6 Sketch of the RSW frame taken from ([49])

2.5.1 Herrick-Gibbs Method

The Herrick-Gibbs Method is a variant of the standard Gibbs Method that use a Taylor-series approximation to calculate the velocity vector associated with the second of three successive position vectors.

When the three position vectors are closely separated in space, the Gibbs Method becomes unstable, and the Herrick-Gibbs Method is better suited for such scenarios. More about the description of the Herrick-Gibbs being more appropriate for measurements with a close time between can be found in the book of Vallado ([74]).

The first step to do when using the Herrick-Gibbs method for a IOD is to calculate the time difference between the position vectors$(\vec{r}_1, \vec{r}_2 and \vec{r}_3)$:

$$\Delta t_{31} = JD_3 - JD_1 \tag{2.3}$$

$$\Delta t_{32} = JD_3 - JD_2 \tag{2.4}$$

$$\Delta t_{21} = JD_2 - JD_1 \tag{2.5}$$

where the JD is the Julian days associated with each position. The Herrick-Gibbs Method is similar to the Gibbs Method at this stage in that it also tests if the position vectors are coplanar and checks the angular separation:

$$\vec{Z}_{23} = \vec{r}_2 \times \vec{r}_3 \quad \alpha_{cop} = 90^\circ - \cos^{-1}\left(\frac{\vec{Z}_{23} \cdot \vec{r}_1}{\left|\vec{Z}_{23}\right| \, |\, \vec{r}_1}\right) \tag{2.6}$$

$$\cos(\alpha_{12}) = \frac{\vec{r}_1 \cdot \vec{r}_2}{|\vec{r}_1|\,|\vec{r}_2|} \quad \cos(\alpha_{23}) = \frac{\vec{r}_2 \cdot \vec{r}_3}{|\vec{r}_2|\,|\vec{r}_3|} \tag{2.7}$$

If the coplanarity and spacing are adequate, the velocity vector is:

$$\vec{v}_2 = -\Delta t_{32}\left(\frac{1}{\Delta t_{21}\Delta t_{31}} + \frac{\mu}{12r_1^3}\right)\vec{r}_1 + (\Delta t_{32} - \Delta t_{21})\left(\frac{1}{\Delta t_{21}\Delta t_{32}} + \frac{\mu}{12r_2^3}\right)\vec{r}_2 \tag{2.8}$$

$$+\Delta t_{21}\left(\frac{1}{\Delta t_{32}\Delta t_{31}} + \frac{\mu}{12r_3^3}\right)\vec{r}_3 \tag{2.9}$$

where the μ is the Earth's gravitational parameter.

2.6 Two-Line Element Set

A Two-Line Element Set (TLE) is a type of data format used to keep track of known Resident Space Object (RSO). By reducing periodic variations in a particular method, the TLE encodes a list of an object's mean Keplerian orbital elements for a specified time period known as the epoch. The force model's periodic changes caused by the Earth's oblateness, atmospheric drag, and the gravitational forces of the moon and sun are removed to create the mean state values, which are referred to as an Simplified General Perturbations 4 (SPG4) state. So the information contained in a TLE is for the specific use of the SPG4 propagator. The TLEs are published every day and can be retrieved from the platforms SpaceTrack ([59]). In Appendix A.1 a description of the elements in a TLE and a description is represented.

More about the SPG4 propagator will be explained in section 2.9.

Because the TLE format data contain an object's mean Keplerian orbital elements, a definition of Keplerian orbital elements is provided. Keplerian orbital elements are a set of six parameters that describe the orbit of a celestial object, such as a planet or a satellite, around another celestial object, such as a star or a planet. The six Keplerian orbital elements are:

- Semi-major axis *(a)*: Is the sum between the periapsis and the apoapsis divided by two;
- Eccentricity *(e)*: This parameter explains how an elliptical orbit looks. With values ranging from 0 (a circular orbit) to 1 (a parable orbit) or greater than 1 (a hyperbolic), it serves as a gauge for how far the orbit veers from a perfect circle;
- Inclination *(i)*: This is the angle between the plane of the orbit and a reference plane, usually the plane of the ecliptic (the plane of Earth's orbit around the Sun). As for the case of satellite in LEO, the reference plane is the Earth equator;
- Longitude of the ascending node (Ω): This is the angle, measured from a set reference direction, between the reference plane and the line connecting the plane of the orbit with the reference plane (line of nodes);
- Argument of periapsis (ω): This angle, measured in the plane of the orbit, lies between the ascending node and the point of the orbit's closest approach to the central body;
- True anomaly (v, f): Defines the orbiting body's position along the ellipse at a given moment measured from the position of the periapsis.

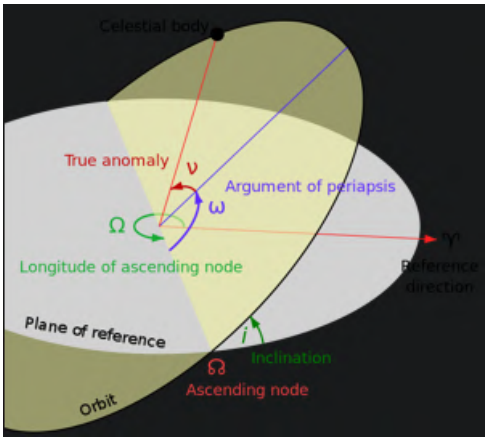


Fig. 2.7 Keplerian orbital elements taken from ([40])

Figure 2.7 shows a representation of the Keplerian orbital elements.

With the help of the TLE, the object can be located with a few kilometers of accuracy. The more away from the time, the elements set is disseminated, the less accurate it becomes. As a result, it's critical to routinely maintain and update the TLEs with fresh information.

The TLE is merely a rough approximation of the satellite's actual position and velocity at the epoch due to the inherent uncertainty in the underlying measurements and models used to construct it and the impact of disturbances from other objects in orbit. Additionally, the TLE's accuracy will decline over time as the satellite's orbit changes. As a result, rather than accurately representing a satellite's position and velocity, TLEs are frequently employed as a starting point for further in-depth orbital study and prediction.

2.6.1 SpaceTrack

Spacetrack is an online catalog run by the United States Department of Defense and controlled by the United States Space Command. Its primary function is to track and categorize things in orbit, such as active satellites, rocket bodies, debris, and other space objects. The Spacetrack database provides orbital components and other data required for following and forecasting the future locations of these objects. These orbital elements are typically represented in the TLE format.

2.7 Orbit Determination Methods

The process of orbit determination involves the use of an imperfect mathematical model to determine the best approximation of the state of an object, whose initial state is unknown or imperfect, using data influenced by random and systematic errors. A typical orbit determination is a sum of an IOD (initial guess) and a correction done to this initial guess through past or previous measurements. There would, in theory, be no need for additional re-observations of an identifiable or observed space object if it was feasible to precisely model the forces operating on the item along with the precise initial conditions. Due to the lack of precise initial conditions, a statistical/estimation approach has to be made. The OD procedure estimates an orbit in the form of an orbital state or elements at an epoch,

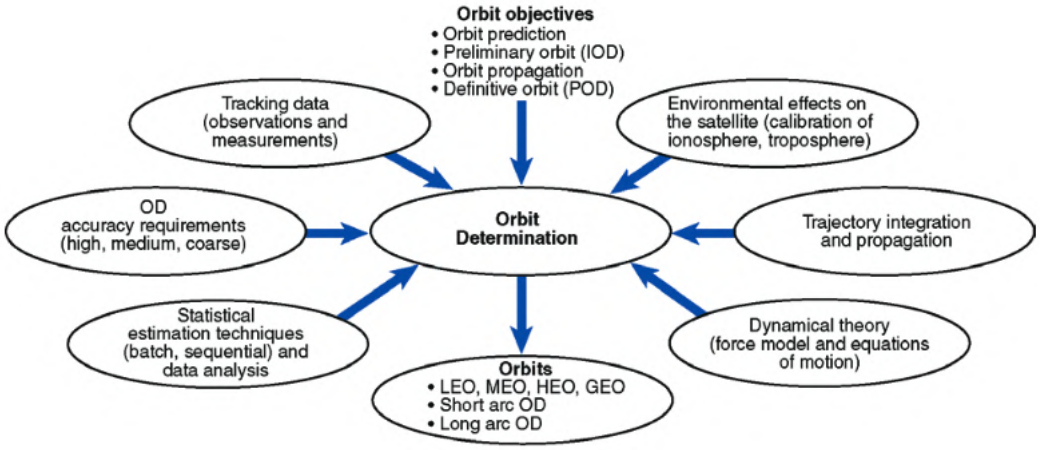


Fig. 2.8 Typical Orbit determination system taken from ([75])

or at the present moment, based on a collection of observations. Force models, measurements, and estimating methods are all part of an OD process.

In figure 2.8 the composition of an orbit determination system is represented.

Orbit determination methods can be organized into two main categories ([53]) based on their operation principle and application:

- Batch Least Squares Orbit determination;
- Sequential Batch Least Squares Orbit Estimation.

2.7.1 Batch Least Squares Orbit determination

The batch least squares method collects and processes all the data for a set period at once. Since the tracking data is normally retrieved once every rotation, the batch least squares approach is frequently used for offline processing of spacecraft trajectory data.

For a given satellite, orbit motion is modeled using a nonlinear dynamic equation, so the filtering technique used in the OD system should be suitable for nonlinear systems. A typical algorithm to resolve non-linear Least Square (LS) solutions to statistical OD is the Levenberg-Marquardt algorithm([28]). The Levenberg-Marquardt approach, which offers benefits over the Gauss-Newton method and the steepest gradient method, is the most used nonlinear fitting technique. Since the Levenberg-Marquart method is a combination of the other two methods, a description of the Gradient Descend Method, the Gauss-Netwon method and the Levenberg-Marquart method are as follows:

Gradient Descent Method

In the gradient descent method, the sum of squared errors is minimized by updating the parameters in the descending direction ([28]).

Gauss-Newton Method

Gauss-Newton method, the sum of the squared errors is reduced by assuming the least squares function is locally quadratic, and finding the minimum of the quadratic ([28]).

Levenberg-Marquardt Method

The Levenberg-Marquardt ([83]) ([77]) method alternates the parameters update between the gradient descent and the Gauss-Newton. When the current solution is not the close to the ideal one, the algorithm functions choose the steepest descent approach, which is slow but ensures convergence. When the present solution is close to the ideal one, it becomes a Gauss-Newton algorithm.

2.7.2 Sequential Batch Least Squares Orbit determination

Using process noise information, the sequential estimation approach sequentially updates the state vector to generate a better estimate at each epoch. Applications involving real-time onboard spaceship navigation often use Sequencing Processing ([32]), with Kalman filter as a common approach, as the estimation algorithm. When the entire collection of observations is divided into statistically independent lots that are each composed of a single measurement, the sequential processing carried out by the filter can be considered as a recursive version of the least square method.

Kalman Filter

A method for estimating unknown variables based on measurements made over time is the Kalman filter. It is used to produce a minimum-variance estimate of state variables in a nonstationary linear system using the measurement covariance matrix, initial state error covariance matrix, dynamic vehicle model, and model noise statistics. The state vector is established by integrating the nonlinear equation of motion, and errors in the state vectors are estimated and corrected using the Kalman Filter theory to improve orbit estimates.

The key to nonlinear Kalman filtering is to expand the nonlinear elements of the system equation in a Taylor series expansion around a nominal point ([65]) because the Kalman filter cannot be applied directly to nonlinear issues.

Extended Kalman Filter

A nonlinear dynamic system's state variables can be estimated using the Extend Kalman Filter (EKF) ([57]), a Kalman Filter extension.

The nonlinear dynamics of the system are initially modeled by the EKF using a set of differential equations. The EKF linearizes the nonlinear equations around an operating point using a Taylor series expansion in order to estimate the state variables.

The EKF can be utilized to estimate the state variables and their errors after the nonlinear equations have been linearized. By combining the sensor readings and model predictions, the EKF calculates the state variables. The state and covariance estimations are then updated by the EKF using these estimates at each time step.

Prediction and update are the two stages of the EKF's operation. Based on the system model and the prior state and covariance values, the EKF predicts the state and covariance estimates during the prediction phase. The state and covariance estimations are updated by the EKF during the update phase using the new measurements. The Kalman gain, a weighting factor that defines how much to trust the measurements in relation to the model predictions, must also be calculated during the update phase.

The prediction and update phases of the EKF are carried out iteratively up until the state and covariance estimations reach a stable value. The final state estimate and covariance matrix can be used to assess the precision of the predicted state and propagate the uncertainty forward in time once the EKF has converged. The state vector elements' variances are represented by the covariance matrix's diagonal members, while their covariances are represented by the off-diagonal elements. The calculated state vector's uncertainty increases with increasing values in the covariance matrix.

The following equation([38]) presents the summary of how the EKF works explained above.

$$\begin{aligned} &\text{Input } \left\{\widehat{X}_k, \widehat{P}_k, z, Q, R\right\} \\ &\text{Output } \{X_{k+1}, P_{k+1}\} \\ \mathbf{H}_{k+1} &= \frac{\partial z}{\partial \widehat{X}_{k+1}} \end{aligned} \tag{2.10}$$

Prediction

$$\begin{aligned} &\overline{X}_{k+1} && \text{Predicted with Cowell's Method} && \text{Predicted State} \\ &F = \frac{\partial \dot{X}_{k+1}}{\partial X_{k+1}} \\ &\dot{\Phi} = F(t)\Phi \\ &\delta\bar{x}_{k+1} = 0 && && \text{Predicted State Update} \\ &\overline{P}_{k+1} = \Phi \widehat{P}_k \Phi^{\mathrm{T}} + \mathbf{Q} && && \text{Predicted Error Covariance} \end{aligned} \tag{2.11}$$

Update

$$\begin{aligned} &\tilde{b} = z - H_{k+1}\overline{X}_{k+1} \\ &K_{k+1} = \overline{P}_{k+1} H_{k+1}^T \left[H_{k+1}\overline{P}_{k+1}H_{k+1}^T + R\right]^{-1} && && \text{Kalman Gain} \\ &\delta\hat{x}_{k+1} = K_{k+1}\tilde{b}_{k+1} && && \text{State Update Estimate} \\ &\widehat{X}_{k+1} = \overline{\overline{X}}_{k+1} + \delta\hat{x}_{k+1} && && \text{State Estimate} \\ &\widehat{P}_{k+1} = [I - K_{k+1}H_{k+1}]\overline{P}_{k+1} && && \text{Error Covariance Estimate} \end{aligned} \tag{2.12}$$

where $\hat{X}_k$ is the initial state, $\hat{P}_k$ the initial state error, Q the covariance of the process noise, R the covariance of the observation noise, z the measurements, Φ the state transition matrix, H the observation matrix, F the state matrix and the $\tilde{b}$ represents the difference between the predicted measurement and the actual measurement.

The observation noise covariance matrix R represents the noise or uncertainty in the measurements. It detects errors, inaccuracies, or noise in sensor measurements. The process noise covariance matrix Q

captures the uncertainty in the predicted evolution of the state variables based on the dynamics model. In other words, a greater process noise covariance matrix takes into consideration that measurements are more dependable than model predictions. The state transition matrix relates the state between t_k and t_{k+1} and is calculations is one of the highest computational cost during the process of orbit determination with the EKF. More about the calculation of the state transition matrix can be found in the article ([12]).

2.7.3 Comparison between Batch Least Square and Sequential Least Square

When measurements are not available all at once during the orbit determination process, Sequential Least Square is a helpful technique to update the orbit. In this approach, the algorithm can use the most recent observations to improve the orbit estimate in real-time. Batch Least Square can be applied as a post-processing step to improve the Sequential Least Square estimates.

The main disadvantages of an approach with the Batch Least Square are its computational cost and the lack of convergence that can occur. For the case of Sequential Least Square with the Extend Kalman filter, the main disadvantage is the difficulty in defining the initial state error and the initial process noises. In terms of computational cost, the Sequential Batch least Square demands a lower one compared to the Batch Least Square ([17]).

2.8 Uncertainty in the Orbit determination

The method of calculating covariances associated with propagating states is known as orbit uncertainty quantification.

The reasons for orbit uncertainties are initial state uncertainties, force model errors, and model-specific simplifications. The covariance matrix's diagonal represents the uncertainty in the position and velocity of values for the Cartesian coordinates to define the state vector of the object. As a result, if the covariance matrix is employed to describe the state vector, the element of the first row and column on the x-axis will include the position vector's uncertain range. When attempting to predict if there will be a conjunction of orbits and then determining whether a collision between two or more objects will occur, the uncertainty in the state vector will be important for measuring the collision probability.

Propagation is the process of predicting the future position and velocity of an object, such as a satellite or a planet, based on its current state and the laws of motion.

2.9 Propagators

A prediction of the future positions of the objects can be made using an initial orbit estimate, regardless of whether the representation is on orbital elements or a state vector. There are three basic types of orbit propagation mechanisms: numerical, semi-analytical, and analytic ([67]). If a numerical integrator is used to reach the solution, the approach is known as numerical propagator theory or Special Perturbation. It is referred to as General Perturbation or analytical approach if the solution was obtained using analytical approximations. One such common analytical propagator is the SPG4.

The SPG4 orbital propagator technique is commonly used for forecasting the locations and velocities of Earth-orbiting objects. It is intended particularly for two-line element TLE data, which offers orbital information in a compact manner.

The SPG4 propagator is based on a simplified version of the SPG4 model. It uses a variety of perturbation effects to simulate the gravitational forces and air drag experienced by the orbiting object.

One can also combine analytical and numerical methods, these are known as Semi-analytical Satellite Theories.

Each technique has benefits and drawbacks ([24]) of its own, being the main ones, the accuracy and computer cost it takes to solve the problem. The numerical approach will take more time and computational effort but will come out with a solution with higher accuracy. On the other hand, the Semi-analytical and analytical will have a higher speed of resolution with lower accuracy. More about the comparison between the analytical and numerical precision and computational cost can be found in the article ([61]).

Before propagation can be resolved, several variables have to be defined. These fickle variables for a simple propagation are the force model used for the object, an initial state vector, and the type of integrator ([7]) used to resolve the equations of motions of the object.

In the two-body problem, resolving Kepler's equation is sufficient to advance the orbit in time. This equation is represented as:

$$M = E - e\sin E \tag{2.13}$$

where M is the true anomaly, e is the eccentricity of the orbit and E is the eccentric anomaly, which is the angle between the pericenter and a fictitious point on a circle that has the same radius as the orbit and passes through the planet's current position. More about the solving of the equation 2.13 to obtain the position and the velocity vector of the object can be found in ([5]).

When accounting with perturbations that depart from the Keplerian forces, Cowell's formulation ([73]) is used. The equation 2.14 presents the two-body differential equation separating the Keplerian and the perturbations.

$$\ddot{\mathbf{r}} = -\frac{\mu}{r^3}\mathbf{r} + \mathbf{a_p} \tag{2.14}$$

where $\ddot{\mathbf{r}}$ is the acceleration brought on by perturbations, $\mathbf{r}$ is the geocentric position vector, μ is the standard gravitational parameter, r^3 is the distance between the two bodies and $\mathbf{a_p}$ the perturbation accelerations. These perturbations are defined by force models that have a different impact depending on the altitude of the monitored objects.

2.9.1 Integrators

After the definition of the equations of motion with the perturbations included, the type of integration is chosen. As for the case of a more accurate propagation, a numerical approach is the best method. There are numerous techniques for numerical integration, and many of them are frequently applied in orbit propagation. Integrators can be classified into several groups: single-step multi-step, fixed-step size, variable-step size, single integration or double integration:

- Single/Multi step: Single-step techniques, predict an object's future position and velocity over a brief time interval using only its present location and speed. The multi-step makes use of a number of past positions and velocities to predict future positions and velocities across a longer time step. Although multi-step methods can offer greater accuracy, they also demand more computational power. There are two types of multi-step integrators, summed and non-summed, depending on whether the integration is carried out from an epoch or step-by-step;
- Fixed/Variable step size: Fixed-step methods employ a defined time step for each iteration of the numerical integration. On the other hand, variable-step approaches use smaller time steps in regions of rapid change and larger time steps in regions of sluggish change, adjusting the time step to the behavior of the solution at each step.
- Single/Double integration: In some instances, the position and velocity of the object over time can be integrated using a single set of differential equations to solve the equations of motion. A single integration approach is what this is. However, it may be necessary to integrate both the position and velocity of the object over time using a system of coupled differential equations in more challenging situations, such as those involving the effects of atmospheric drag or other perturbations. A double integration method is what this is;

Numerical integrators give the most accurate orbital states for computing short arcs of satellite orbits along with the complete collection of force models. When precision is crucial, a numerical technique is most suited.

Runge-Kutta 4th and higher-order algorithms ([43]) are the most popular solutions for satellite motion equations. To get a minimum computational effort an adaptive/multi-step-size method can be used. The adaptive methods use an estimate of the local truncation error of a single integration step to regulate the step size.

In figure 2.9, some types of integrators are shown. More about each type of integrator represented in figure 2.9 can be found in ([8]).

2.9.2 Forces

The space object is typically assumed to be affected by a variety of external forces, including third-body perturbations, Earth tidal effects, irregular gravity fields, atmospheric drag, solar radiation pressure, general relativity, and satellite propelling maneuvers ([60]). The terms above, excluding intentional moves, are together referred to as perturbing accelerations. In figure 2.10 one can see that the magnitude of these forces depends on the Semi-major axis of the object.

A description of the forces represented in the figure can be made in the following way:

- Third body problem (Jupiter, Moon, Sun, and Venus): resulting from the gravitational pull of celestial body other than Earth, the main contribution comes from the Moon and the Sun;
- Atmospheric Drag: The friction exerted by the atmosphere depending on the altitude of the object;
- Solar Radiation Pressure: This perturbation comes from the Sun radiation acting on the surface of the satellite;

Integration Methods

Method	Single / Multi	Fixed / Variable	Single / Double	Non-Summed / Summed
Runge-Kutta	Single	Fixed	Single	NA
Runge-Kutta-Fehlberg	Single	Variable	Single	NA
Adams (non-summed)	Multi	Fixed	Single	Non-Summed
Summed Adams	Multi	Fixed	Single	Summed
Shampine-Gordon	Multi	Variable	Single	Non-Summed
Störmer-Cowell	Multi	Fixed	Double	Non-Summed
Gauss-Jackson	Multi	Fixed	Double	Summed
New: var. S-C	Multi	Variable	Double	Non-Summed

Fig. 2.9 Some Types of Integrators taken from ([9])

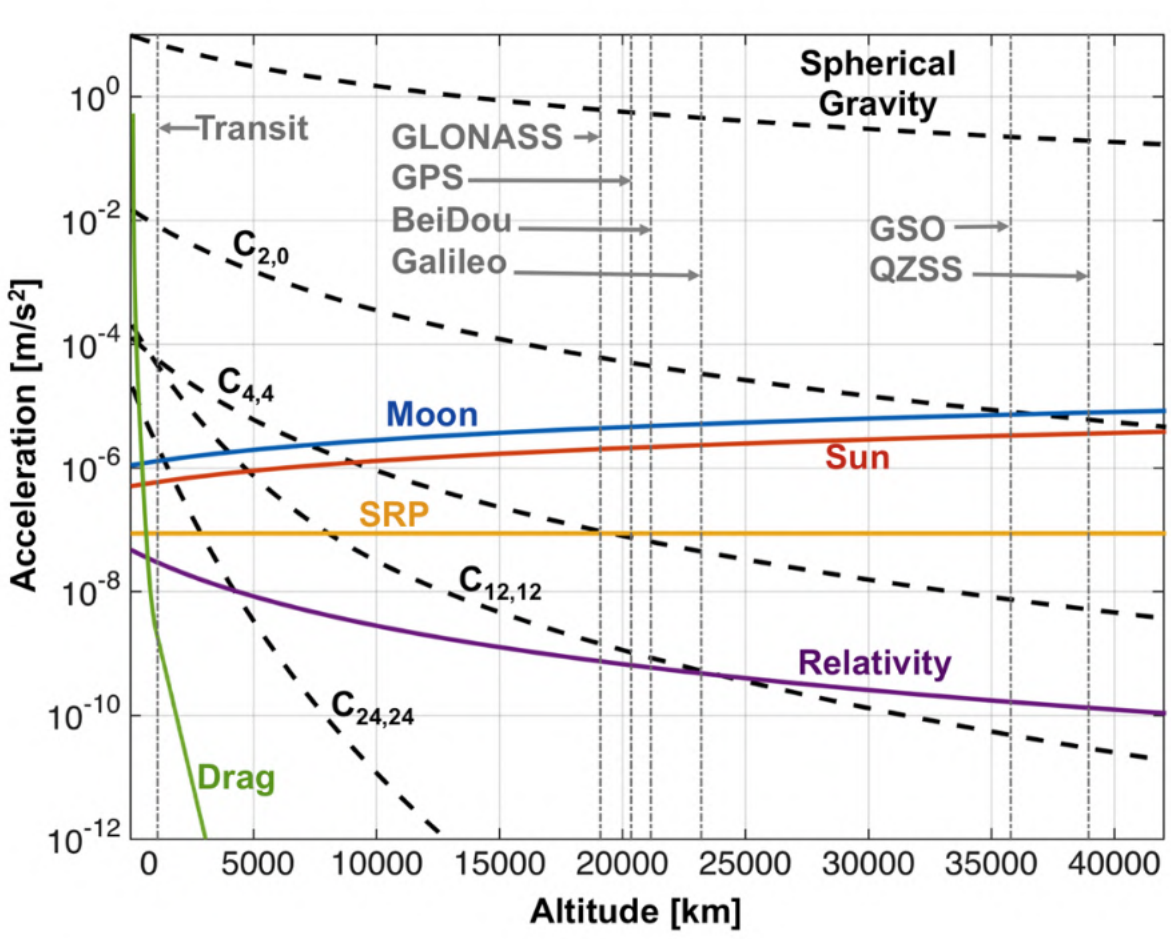


Fig. 2.10 Perturbation of the forces depending on the altitude of the object taken from ([56])

- Geopotential: Caused by Earth oblateness being J_2 and the rest of J the coefficient reflecting the inhomogeneities in the Earth gravity field;
- Dynamic Solid Tide: Is the movement of the solid earth's surface brought on by the Moon's and Sun's gravitational pull;
- Relativity: General relativity's post-Newtonian correcting acceleration ([64]).

One extra physical effect that not shown in the figure is the Albedo irradiation force.

Small bodies in the Solar System, including comets, asteroids, and dust specks, are subject to Albedo irradiation forces, a sort of non-gravitational force. These forces develop as a result of these bodies' surfaces reflecting and absorbing solar light.

A tiny body receives solar radiation and some of it is reflected back into space. The remainder is absorbed and re-emitted as thermal radiation. The small body receives a net momentum transfer from the solar radiation as a result of this absorption and re-emission process, which can lead it to accelerate or decelerate. As for LEO objects the main force that impacts the trajectory is the Drag effect.

Atmospheric Drag Uncertainty

The estimation of air drag is one of the main causes of uncertainty in spacecraft orbit propagation, particularly at LEO altitudes. Furthermore, a thorough understanding of the atmosphere and interactions between the gaseous surface and atmosphere is necessary. A lack of information in the drag coefficient calculation may therefore make it more difficult to arrive at accurate forecasts. However, errors in atmospheric density modeling also play a significant role in the overall level of uncertainty. The output of these models is subject to uncertainty due to inherent differences in the various state-of-the-art modeling techniques that are currently available, in addition to the challenges in obtaining accurate long-term predictions of solar and geomagnetic activity and their indices used in the atmospheric models. However, aerodynamic drag is the most significant perturbing force for Earth-orbiting spacecraft in LEO.

Particles interacting with the satellite body and causing an acceleration in the opposite direction of its velocity result in aerodynamic drag. Most satellites in LEO must take into account the Earth's atmosphere because it can be felt up to an altitude of 1000 km or more, especially for spacecraft with enormous surface areas. Accurate knowledge of the atmosphere, the surfaces of the spacecraft, and how the spacecraft interacts with the atmosphere are necessary to comprehend aerodynamic drag. The common equation for the acceleration caused by aerodynamic drag is([58]):

$$\mathbf{a_{drag}} = -\frac{1}{2}\frac{c_D A}{m}\rho \dot{\mathbf{r}}_{\mathbf{rel}}^{\mathbf{2}}\frac{\dot{\mathbf{r}}_{\mathbf{rel}}}{||\dot{\mathbf{r}}_{\mathbf{rel}}||} \tag{2.15}$$

where c_D is the drag coefficient, A is the cross-sectional area of the satellite normal to the incident flow, m is the spacecraft's mass, ρ is the atmospheric density, and $\dot{r}_{rel}$ is the satellite's relative velocity to the atmosphere. The drag coefficient, or c_D, can be one of the main causes of inaccuracies in the estimation of aerodynamic drag, depending on the shape of the object. Currently, the most common models are the NRLMSISE-00, JB2008, and DTM2000 ([10]). Depending on the altitude of the object

certain models are more appropriate than others. For altitudes below 500 km, the model JB2008 and DTM2000 are the more appropriate ones although DTM2000 is too dependent on high solar activity, above 500 km of altitude the model NRMSISE-00 and DTM2000 ([82]) are the best models.

Chapter 3

Orbit Determination

This chapter discusses the outcomes and also outlines how to produce the results received by the Orbit Determination (OD) algorithm.

Therefore throughout it, preparation of the tools required to produce the OD results are covered as well as the method's final outcomes. The EKF algorithm is utilized for the OD for which a detailed description was previously provided in section 2.7.2. The starting state $\hat{X}_k$, initial state error $\hat{P}_k$, the covariance of the process noise Q, the covariance of the observation noise R, and the measurements z are the five inputs that the EKF algorithm needs in order to run, see equations 2.10, 2.11 and 2.12. In this chapter, we thus explain how to obtain them and hence determine the orbits.

The measurements that were used and how they were characterized are two more crucial details shown in this chapter. This chapter also discusses the method used to deal with outliers (section 3.4).

As for the type of object to apply the OD, the selection was another important decision. This selection was made based on the measurements available, the kind of satellite orbiting in LEO, and having a corresponding CPF file. The accessible measurements were related to the satellite Jason-3 which orbits in LEO around an altitude of 1300 kilometers.

CPF files contain accurately predicted positions of an object. Their exceptional accuracy is a result of the use of measurements made with extremely precise lasers for the process of OD. Due to their highly accurate predicted positions, CPF files are ideal for comparing the results of propagation. More about propagation will be discussed in the next chapter. CPF files are calculated daily by NASA and can be found online on the website [48].

The software used to produce the orbit determination is also explained in this chapter.

3.1 Software

The software used for the study of OD and Propagators was Orekit([30]). This is a free Java toolkit created to assist programmers in creating simulations of space missions with the highest degree of accuracy. It offers a variety of tools and models for determining, propagating, and analyzing orbits as well as support for common data formats and coordinate systems used in spaceflight. In more detail, some of the features offered by Orekit are the following:

- The methods and models in Orekit can be used to propagate and determine the orbits of satellites. It supports numerous force models, including gravity, drag, and radiation pressure, and it can handle a range of orbital types, including Keplerian and Cartesian;
- Data handling: TLE is just one of the few common data formats supported by Orekit. It also features support for managing data caches and tools for reading and writing data in a variety of formats;
- Orekit comes with an extensive collection of tools for dealing with space dynamics and geometry. This comprises atmospheric and ionospheric models as well as models for coordinate transformations, Earth orientation parameters, and other variables.

More about the features presented above can be found on the website for the platform Orekit ([30]).

For the case of this work, it was necessary to employ the tools provided by Orekit for orbit determination, orbit propagation, and auxiliary needs. To perform an Orbit determination and Orbit propagation, it was used an input file with all the information needed where the content of the input will be explained in the next section.

3.2 Software Input Data

As for the input necessary for the correct processing of the information, there are some variables to consider. Some of these variables are the forces that will influence the trajectory of the object, the initial state of the orbit to be propagated and the reference system it is represented. Other variables are:

- Location of the ground station making the observations;
- Bias corresponding to each type of measurement;
- Which corrections to apply to the measurements due to the atmosphere;
- Mass of the observed satellite;
- Initial covariance matrix and Initial noise matrix is given when using EKF for OD;
- Central body definition, in this case, the planet Earth;
- Type of Propagator to use.

Chapter 2 contains descriptions of all the variables listed above.

3.3 Measurements

Since radar measurements are not affected by weather conditions, they can be made both day and night, unlike optical measurements, radar measurements were elected. Although satellite laser ranging observations have better accuracy, they are nevertheless subject to weather considerations, as are

optical observations. SLR measurements are also restricted to objects with retro-reflectors to have such high accuracy.

The total number of the measurements in the Tracking Data Message (TDM) file used is 27. The measurements are organized as follows:

- Range: Range is the straight-line distance, the line-of-sight measurement between the sensor and the target object. Time-of-flight measurements can be used to compute this distance, as explained in section 2.4.2;

- Azimuth: As measured clockwise around the observer's horizon, the azimuth is the angle between North and a body(satellite). The range of azimuth is 0° to 360°. North is at 0° when it begins. We will face East (which is 90°), South (which is 180°), West (which is 270°), and then North (which is 360° and also 0°) as we move to the right (in a clockwise motion). Hence, if the satellite's azimuth is, say, 45 degrees, it is northeast of us. But there are exceptions to measuring the azimuth of an object. The Sloan Digital Sky Survey (SDSS) and the Flexible Image Transport System (FITS) protocols used by European Southern Observatory (ESO), which measure it from the south and rise westward and eastward respectively, are examples of exceptions. More about the measuring of the azimuth with the SDSS and FITS protocols can be found in ([63]) and ([68]);

- Elevation: Elevation angle is the angle measured in the vertical plane between the horizontal plane and the line of sight. It is typically measured in degrees or radians, and is measured from the horizon;

- Range-Rate: Radar and remote sensing systems utilize the term "range rate" to define how quickly a target object is traveling towards or away from the sensor. It is described as the rate of variation in the sensor-target object distance over time. One method for figuring out an object's range rate is using the Doppler effect, as described in the section 2.4.2.

In figure 3.1 we present sketches representing the measurements mentioned above. It should be noted that the azimuth shown in the sketch corresponds to the angle measured from the North. (As previously stated, there are exceptions to measuring the azimuth angle).

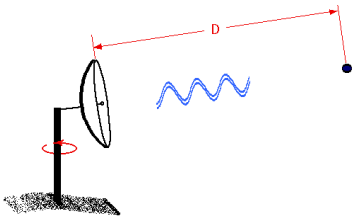
The observations are presented in a TDM [11] format with the information mentioned above and the epoch of each measurement.

The TDM is implemented as a series of basic American Standard Code for Information Interchange (ASCII) text lines, which can be in the form of a file or a real-time stream and it contains in a single file the tracking data necessary to utilized in OD.

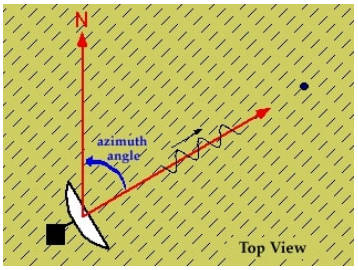
Notice that not all TDM files contain the same information.

In our example, the study measurements are from 2022-10-26, 00:38:28.97703 to 2022-10-26, 00:39:22.364074UTC and have a total of 27 points, where each point contains Range, Range-Rate, Azimuth, Elevation and the epoch associated.

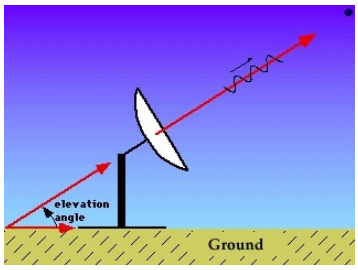
As the data in the TDM file doesn't contain any adjustments to the measurements, the OD method applies the following correction:



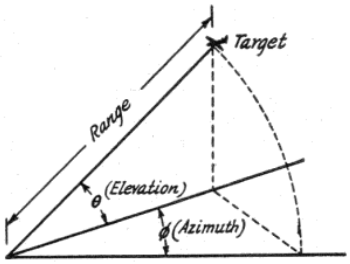
(a) Range measurement Sketch ([51])



(b) Azimuth measurement Sketch([51])



(c) Elevation measurement Sketch ([51])



(d) Sketch of Azimuth, Elevation and range measurements ([19])

Fig. 3.1 Sketches of radar measurements

- The Shapiro effect, which affects range measurements. The Shapiro effect refers to the delay of radio signals passing through the gravitational field of a massive object, such as a planet or a star. In other words, it can cause errors in measuring the distance between two objects;
- The ionosphere model, which causes a delay in electromagnetic signals sent between the ground station and the satellite;
- The troposphere model, which causes the same delay in the electromagnetic signal as the ionosphere model;
- An atmospheric refraction model, that affects the azimuth and elevation measurements. The bending of light as it travels through the atmosphere of the Earth is known as atmospheric refraction. Astronomical objects appear higher in the sky than they actually are because to this bending, especially when they are close to the horizon;
- Station displacements due to the ocean tides. Tides are caused by the gravitational differential attraction of the Moon and the Sun on the Earth's oceans. This causes the oceans to rise and fall in a periodic pattern. This periodic rise and fall of the ocean surface can cause the position of ground-based observatories to shift slightly, which is known as station displacement.

All these extra models to correct the measurement were left in default, meaning that it just need it to choose if the user wanted it to use them or not. These corrections were made available by the software Orekit.

3.4 Outliers

An outlier is a data point in statistics that dramatically deviates from other observations. An outlier may be caused by measurement variability or an experimental error, and it is occasionally removed from the data set. While an outlier may signal an intriguing possibility, it can also seriously impair statistical analyses. The "3 times the standard deviation" rule, also referred to as the "3σ rule of thumb", is a typical technique for defining outliers([84]). The way of working with outliers offered by Orekit is with the standard deviation of the values, so this technique is one that we used. According to the criterion, any measurement that deviates by more or less than three standard deviations from the mean of the data is regarded as an outlier, and is not taken into account for determining the orbit. This lessens the effect of any errors or abnormalities on the final orbit determination and ensures that the resulting orbital solution is based on the most precise and trustworthy measurements. In this case, the outlier will remove the values of a specific measurement taken by the radar that is too far apart from the estimated values by the orbit determination algorithm. Figure 3.2 illustrates an example of an outlier in a set of points.

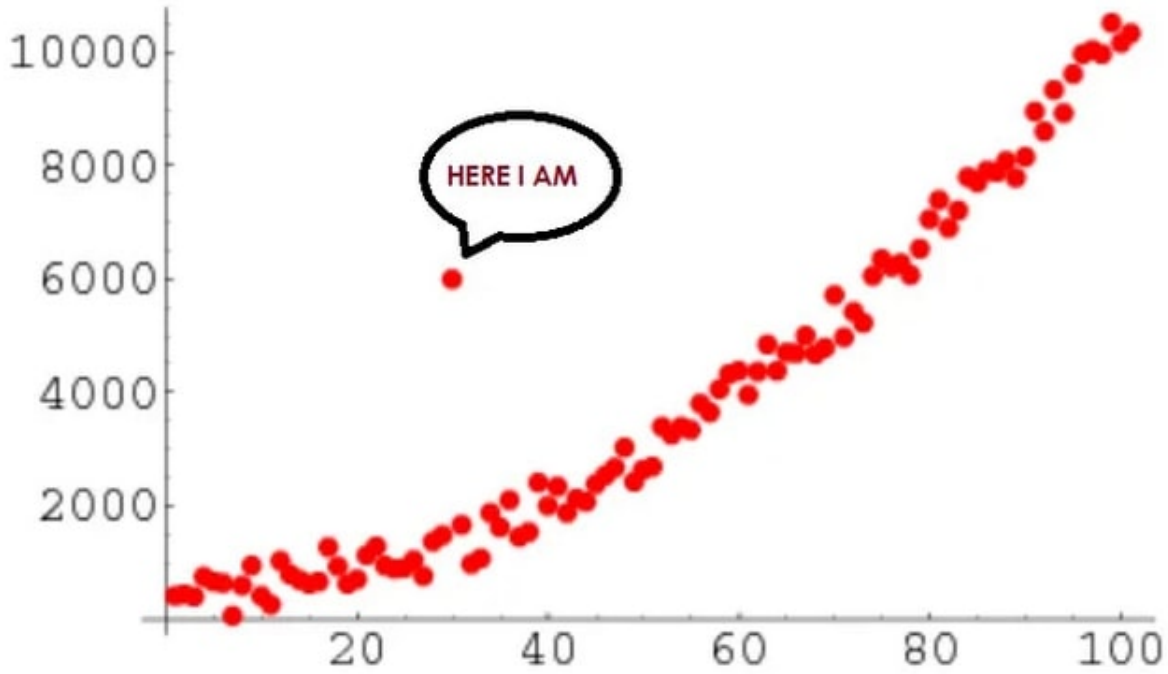


Fig. 3.2 Illustration of an outlier in a set of points taken from ([62])

3.5 Initial State

An estimate of an object's position and speed in space at a specific time is known as the initial guess or initial state. The initial estimation is crucial because it gives orbit determination algorithms a starting point to compute the object's trajectory and predict its future position and velocity. The orbit determination procedure typically entails iteratively improving the initial guess by incorporating new observations and adjusting the orbit model's parameters until the calculated trajectory matches the

observed data as closely as possible. The need to improve the initial state is because it is unlikely that the initial state is enough accurate.

One such method to calculate the initial state is the Herrick-Gibbs algorithm and is the method used in this master thesis to calculate an initial state.

Before calculating the starting state, a frame to represent the coordinates of the coordinates is chosen, and the frame used was the J2000.

The J2000 frame was picked as the reference system because it is a widely used ECI frame. The J2000 is defined with the Earth's Mean Equator and Mean Equinox (MEME) at 12:00 Terrestrial Time on 1 January 2000. It can be referred to as J2000 or EME2000. The mean equinox is parallel to the x-axis and the Earth's rotation axis, or alternatively, the celestial North Pole, is lined up with the z-axis as it was at that time. The y-axis is rotated 90 degrees east of the celestial equator.

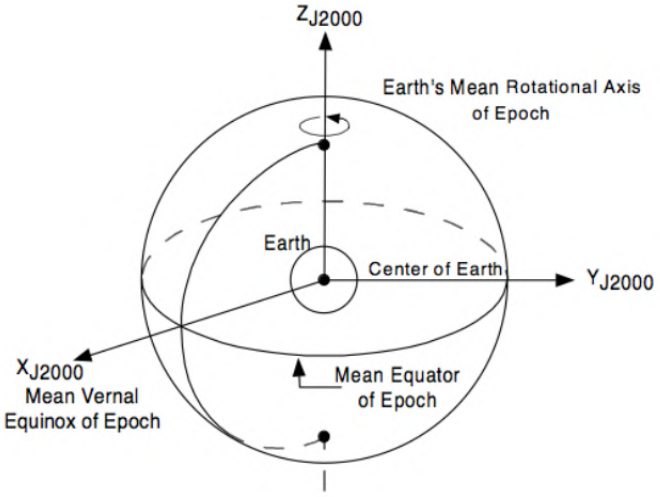


Fig. 3.3 Sketch of the geometry of J2000 frame taken from ([76])

3.5.1 Initial State Herrick-Gibbs Method

When using the algorithm of Herrick-Gibbs, three position vectors are needed. So the range, azimuth, and elevation of the first three measurements were used to get the position vectors. For this, we wrote a script in Python that depends on the location of the ground station and the measurements of the range, azimuth and elevation, which returns the position vector in the ECEF reference system. After having the position vector in this frame, it is converted to the ECI frame with a function given by Orekit. The reason for this conversion is that for the process of the orbit determination the initial guess/state is expected to be in the ECI frame. Having the three position vectors in the correct reference system, the algorithm will return the velocity vector at the epoch of the middle position vector. The script developed is represented in Appendix A.4.

To study the behavior of the algorithm itself, the position vectors of the CPF file for the same epochs of the measurements were used. Although the predicted positions in a CPF file are more commonly used for comparing results of propagation, they were used for comparing the results of the Herrick-Gibbs method.

Table 3.1 Positions CPF (in meters) in the J2000 frame

Pos x-axis	Pos y-axis	Pos z-axis	Epoch
5322021.993468885	3658037.078303746	4224083.075208987	2022-10-26T00:39:06.749777
5350111.262297582	3652606.921133936	4193216.83612897	2022-10-26T00:39:00.894421
5500390.808637966	3621111.474802437	4022807.046982691	2022-10-26T00:38:28.977031

Table 3.2 Positions through measurements (in meters) in the J2000 frame

Pos x-axis	Pos y-axis	Pos z-axis	Epoch
5324336.525298025	3652893.202188624	4223699.776597945	2022-10-26T00:39:06.749777
5350810.113175424	3650100.551896426	4194782.47860386	2022-10-26T00:39:00.894421
5502168.677709107	3611959.673312061	4031463.388644845	2022-10-26T00:38:28.977031

Interpolating the CPF file to the necessary epochs, allowed for the retrieval of these position vectors.

This was done by the interpolation library, `Skyfield` More about the type of interpolation applied will be referred in the section 4.2.

The position vectors of the CPF file related to the epoch of the measurements are shown in table 3.1 and the position vector through the measurements are presented in table 3.2.

In table 3.3, we show the difference between the position vectors inserted in the algorithm, given by the transformation of the range, azimuth and elevation, and the CPF interpolated position vectors at the same epoch and in the table 3.4 the relative error of the residuals.

In table 3.5 we show the results of the velocity vector when using the three position vectors obtained from the measurements and the three position vectors from the interpolation of the CPF file, with the Herrick-Gibbs method and in table 3.6 the relative error of the residuals.

Table 3.3 Residuals between CPF and position vector through the measurements (in meters) in the J2000 frame

Residuals in x-axis	Residuals in y-axis	Residuals in z-axis	Epoch
-2314.5318291395	5143.8761151224	84.397990727	2022-10-26T00:39:06.749777
-698.850877842	2506.36923751	-1565.64247489	2022-10-26T00:39:00.894421
-1777.869071141	9151.8014903765	-8656.3416621541	2022-10-26T00:38:28.977031

Table 3.4 Relative error Position vectors Herrick-Gibbs method

Pos x-axis	Pos y-axis	Pos z-axis	Epoch
0.043%	0.14%	0.009%	2022-10-26T00:39:06.749777
0.013%	0.069%	0.037%	2022-10-26T00:39:00.894421
0.032%	-0.25%	0.215%	2022-10-26T00:38:28.977031

Table 3.5 Residual in velocity vector (in meters per second) in the J2000 frame

	Velocity in x-axis	Velocity in y-axis	Velocity in z-axis	Epoch
CPF	-4783.54649	936.61479	5282.07263	2022-10-26T00:39:00.894421
Measurements	-4555.61907	588.27499	4966.38225	2022-10-26T00:39:00.894421
Residual	-227.92742	348.3398	315.69038	2022-10-26T00:39:00.894421

Table 3.6 Relative error Velocity vectors Herrick-Gibbs method

Pos x-axis	Pos y-axis	Pos z-axis	Epoch
4.8%	37.2%	5.9%	2022-10-26T00:39:00.894421

3.5.2 Initial State from the SpaceTrack platform

As explained in section 2.6, the TLE is a data format that contains orbital information about an object, therefore the TLE is used as an initial state. But, before using it as an initial state to be used in the algorithm of EKF, the same procedure for the testing of the Herrick-Gibbs algorithm with the comparison with the position vectors in the CPF file, is done to the orbit presented in the TLE. The difference between position vectors of the interpolate CPF to the epoch of the TLE and the difference between velocity vector is represented in tables 3.7 and 3.8, respectively. The relative error between the position vectors interpolate from the CPF and the measurements are represented in table 3.9. For the velocity the vector, the relative error is represented in the table 3.10

The TLE was retrieved from the platform `SpaceTrack` and the one chosen was the closest to the measurements, explained in section 3.3, from the epoch 2022-10-25T18:50:17. The TLE bellow is the one used:

line1: "1 41240U 16002A 22298.78492738 -.00000029 00000-0 13810-3 0 9999"
line2: "2 41240 66.0444 18.3708 0007836 265.1830 94.8291 12.80929142316600"

The orbital elements represented in the TLE, are represented in a different frame than the J2000, the True Equator Mean Equinox (TEME) frame. In figure 3.4 an illustration of the TEME frame is shown. The Y-axis completes the right-handed coordinate system in the TEME reference frame, with

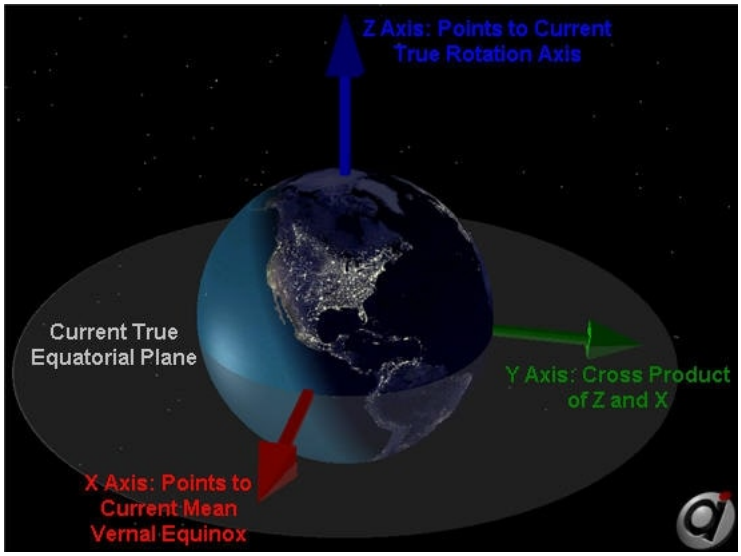


Fig. 3.4 True Mean Equinox Reference System sketch taken from ([2])

the X-axis pointing towards the date's mean equinox, and the Z-axis pointing towards the celestial north pole. The mean equinox of the date serves as the reference point for the TEME frame, which has its beginning at the planet's center. The TEME frame spins in accordance with the Earth's precession and nutation. It is a non-inertial frame.

To extract the mean orbital elements from the TLE and convert them from TEME frame to the J2000 frame using Orekit, the following steps are typically taken:

- Parse the TLE file: To begin, Orekit parses the TLE (Two-Line Element) file, which contains data about the satellite's orbit. The TLE contains information on the satellite's ID, orbital characteristics, and the moment these parameters were measured;
- Extract the orbital elements: The SPG4 algorithm takes the mean orbitals elements and converts them to a state vector represented in a TEME frame. More about this transformation is denoted in the SpaceTrack report ([39]) and in the master thesis of Mario Comini ([14]).;
- Convert to J2000: Finally, Orekit uses rotation matrices to translate the satellite's position and velocity from the TEME frame to the J2000 frame. The Earth's rotation axis's precession and nutation cause a discrepancy between the two frames that are taken into account by the rotation matrices.

The International Earth Rotation and Reference Systems Service (IERS) Conventions constitute the foundation for these rotation matrices describe above.

When comparing the relative error in the initial state vector calculated by the Herrick-Gibbs method and the initial state vector represented in the TLE, the TLE initial contains a lower relative error. This might be owing to some bias already present in the measurements. Because the orbit represented in a TLE contains less inaccuracy, it will be used to define the initial state for the remaining of this work.

The additional inputs required by the EKF algorithm are defined in the following sections.

Table 3.7 Residuals between position vectors of the interpolate CPF and position vectors of TLE (in meters) in the J2000 frame

	Pos x-axis	Pos y-axis	Pos z-axis
CPF	7337184.10712	2395190.93384	-16007.58901
TLE	7337245.24473	2395137.95862	-16139.20336
Residual	-61.13761	52.97523	131.61435

Table 3.8 Residuals between velocity vectors of the interpolate CPF and velocity vectors of TLE (in meters per second) in the J2000 frame

	Vel x-axis	Vel y-axis	Vel z-axis
CPF	-891.72368	2773.00797	6571.21892
TLE	-891.56861	2773.06832	6571.21199
Residual	-0.15508	-0.06035	0.00693

Table 3.9 Relative error Position vectors TLE

Pos x-axis	Pos y-axis	Pos z-axis	Epoch
0.0008%	0.002%	0.82%	2022-10-25T18:50:17

Table 3.10 Relative error Velocity vectors TLE

Pos x-axis	Pos y-axis	Pos z-axis	Epoch
0.017%	0.002%	0.0001%	2022-10-25T18:50:17

3.6 Initial covariance matrix

The covariance of the measurement noise are given values close to zero due to the lack of information of the radar used. For the process noise covariance matrix are given values close to zero meaning, that during the process of orbit determination, we trust the estimation values. Finally, the initial state error is given in form of the covariance matrix. Since the TLE does not have information about the covariance of the state vector, another technique was applied. The covariance matrix is crucial data since it enables the satellite's operator to understand the level of confidence. These uncertainties will allow to conduct a conjunction analysis and determine the probability of collision. In this master thesis, the use of the algorithm for orbit determination is the Extend Kalman filter, and so the initial state error is fundamental for the proceeding of the algorithm.

3.6.1 Methods for Covariance Estimation in TLEs

The drawback of using TLEs to determine an object's position is that they are less accurate than state vectors obtained from more thorough numerical propagators such as Runge-Kutta numerical integration. Runge-Kutta integrator was already explained in section 2.9.1.

TLEs are employed in this thesis despite their low accuracy because they are the sole publicly accessible data format for locating all tracked space debris.

Three approaches can be used to gauge a TLE's precision and determine its covariance: TLE differencing, TLE comparison with tracking data, and TLE as pseudo-observations([18]. The initial one is employed in this thesis. The reason for this choice is that TLE comparison with tracking data and TLE used as pseudo-observations rely on extra knowledge and presumptions regarding the satellite's orbit and tracking data, which might add more sources of error and uncertainty. TLE differencing is another easy-to-automate method that may be used to compare a large number of TLE for many satellites. In the following paragraphs, we provide an explanation of how the algorithms TLE differencing, TLE comparison with tracking data, and TLE employed as pseudo-observations work.

Only historical TLEs are used in TLE differencing, which operates as follows: we take as many TLEs of an item as we can, propagate them to the same time instance, and then subtract the state vectors to get statistical data. Because certain orbits are more disrupted than others, the amount of TLE taken will depend on the nature of the orbit of the object, and so the accuracy of TLE gets deteriorated faster in different types of orbits. For very low earth objects, the time-window of TLE taken can go as low as four days[29] and in higher altitude orbit the time window can go as high as 40 days. But this number can change due to the precision of the TLE taken related to an object, and if a maneuver occurs, it can influence the size of the time window. By applying prior TLEs to the most recent TLE epoch or another desired time and comparing the values, also known as the residuals, the covariance matrix is recovered. This method's assumption is that TLE creation is error-free and that TLE generation does not produce errors. The main benefit is that it can be applied to any cataloged item quickly and easily because it simply relies on SPG4 theory and does not call for extra models or OD algorithms. For this reason, it was adopted in this thesis.

TLE comparison with tracking data is only used for specific objects and not for every sort of object contained in the catalog, because it requires highly precise tracking data, such as pseudo-ranges

and elevation angles or Global Positioning System (GPS) data. Since not all objects have publicly accessible high-precision tracking data from ground stations this technique is not used.

TLE as pseudo-observations treats generated TLE states as if they were ground station pseudo-observations, which means that an OD process will treat these states as measurements. The drawback is that using TLE as measurements considering the error unknown to them is not a viable way.

3.6.2 TLE Differencing

The covariance matrix is only be calculated for one time instant, the epoch of the most recent TLE, and the more recent TLE will be referred to as prime. SPG4 is used to propagate every TLE to the epoch of the most recent TLE. The most plausible assumption is that the prime TLE generates the "true" state vector known as the prime state vector.

First, we developed a script to download the TLEs from `SpaceTrack`. A second script reads the historical TLEs from a text file, and propagates them to the most recent TLE epoch. The propagated TLE's and respective state vectors are transformed from the TEME frame to the J2000 frame. TLE prime, the most recent TLE, is to transform to the ECI frame. After having all the TLEs propagated to the most recent epoch, the following algorithm([18]) is written in a Python script:

$$\delta\vec{X}_{ECI_i} = \vec{X}_{propagated_{ECI}} - \vec{X}_{prime_{ECI}}, \tag{3.1}$$

where $\vec{X}_{propagated_{ECI}}$ refers to the i-th TLE propagated to the most recent epoch, in a set of N TLEs, the $\vec{X}_{prime_{ECI}}$ to the prime TLE and the $\delta\vec{X_{ECI_i}}$ the residual between the propagated TLE and the prime TLE. Having the covariance matrix in a certain epoch defines as

$$P_{ECI} = E[(\delta\vec{X}_{ECI} - m)(\delta\vec{X}_{ECI} - m)^T], \tag{3.2}$$

being m the vector with the mean of the residuals, it follows that:

$$m = \frac{[\sum_{i=1}^{N-1}(\delta\vec{X}_{ECI})i]}{N-1}, \qquad P_{ECI} = \frac{[\sum_{i=1}^{N-1}(\delta\vec{X}_{ECIi} - m)(\delta\vec{X}_{ECIi} - m)^T]}{N-1},$$

The script developed to represent the equation written above is provided in Appendix A.3.

The time window of the TLEs that will be downloaded depends highly on the type of orbit of the object being observed.

3.6.3 Initial State Error Results

Following the implementation written in the section 3.6, in the case of 2σ criterion for the outliers filtering in the procedure of Orbit determination, the initial state error time window of TLEs will take over a period of two days, which corresponds to 6 TLEs.

For the use of the TLE prime associated with the satellite `Jason3` over a period of two days of the date 2022-10-25, the results for the initial covariance matrix of state error are presented in table 3.11.

The selection of the download of TLEs over a period of 11 days, which corresponds to 33 TLEs, was made for the scenario of choosing the 3σ criterion for the outlier filtering in the procedure of Orbit

Table 3.11 Diagonal values of the initial state error covariance matrix for the case use of Two-sigma rule

Pos x-axis	Pos y-axis	Pos z-axis	Vel x-axis	Vel y-axis	Vel z-axis
28.2635 m	676.8112 m	2.9520 m	0.0001 m/s	0.0001 m/s	0.0001 m/s

Table 3.12 Diagonal values of the initial state error covariance matrix for the case use of Three-sigma rule

Pos x-axis	Pos y-axis	Pos z-axis	Vel x-axis	Vel y-axis	Vel z-axis
679.6077 m	4086.8535 m	2793.6285 m	0.0002 m/s	0.0001 m/s	0.0016 m/s

Determination. The initial diagonal values of the initial state error covariance matrix are presented in table 3.12.

The quantity of TLEs that are retrieved to construct the initial state error and the value of the sigma rule that is selected to handle outliers are correlated. A smaller initial state error was observed to produce better results when using a lower sigma value, whereas a larger initial state error was observed to produce better results when using a higher sigma value. This may happen because when employing a lower sigma, the acceptance of the observed values will be more constrained, and for a 3-sigma situation it is the opposite.

The best propagation results, shown in chapter 4, were used to determine how much and for how many days to download TLE in order to achieve the desired initial state error. It is crucial to note that the size of the time window that produces better results, as indicated in the following chapter, is for the particular instance of these satellites as well as for the particular case of the quantity and precision of the measurements provided.

For tweaking the EKF, several time-window sizes and initial state errors were chosen based on the model of trial and error that produced the greatest results.

3.7 Orbit Determination Results

The outcomes of the OD using the EKF method are shown in this section. Due to the usage of two distinct criteria to deal with outliers, as explained in section 3.4, the procedure of OD provides two different results. Not forgetting that the initial state vector is defined by the TLE, the initial state vector is propagated and compared with the values of the observations. Depending on the observed value and the residual between the observed and the propagated value, different corrections are performed on the propagated state vector as explained in section 2.7.2.

The threshold associated with each of the measurements in the TDM file is:

- Range: 100 meters for Survey and 5 meters for tracking;
- Range-Rate: 4 meters/second for Survey and 2 meters/second for tracking;

The thresholds listed above were used since they correspond to the maximum values allowed in the EUSST([15]), and because there is no further information regarding the error connected to the

measurements. As the TDM consists of a survey of a satellite, the values of the threshold associated with the survey are used.

The standard deviation for each measurement is an additional crucial concept to describe. This is due to the 3σ rule and 2σ rule being three and two times the measurement's standard deviation, respectively. If the residual exceeds this count, it is considered an outlier and is removed. The standard deviation was calculated with the formula:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{3.3}$$

Resulting in the following standard deviation for each measurement:

- Range: 42.956065481356 meters;
- Range-Rate: 0.32142580141075 meters/second;
- Azimuth: 1.5095872675114 degrees;
- Elevation: 3.5271922019943 degrees;

Through the process of OD, with the EKF, the residuals between the estimated observed values and the observed value for the associated Three-sigma rule as for definition to remove the outliers and the associated initial error represented in table 3.12, is represented in figure 3.5 where the final state vector and associated standard deviation are shown in table 3.13.

Another important aspect to analyze is the evolution of the covariance matrix during the process. For the use case shown in figure 3.5, the evolution of the diagonal values containing the variance of the Cartesian coordinates, during the orbit determination, are represented in figure 3.6.

For the case of the use of the Two-sigma rule as a definition to deal with the outliers and the associated initial state error represented in table 3.11, the residuals between the observed and predicted, are represented in figure 3.7 where the final state vector and the associated standard deviation are shown in table 3.14. The evolution of the diagonal values of the covariance matrix, that correspond to the variance of Cartesian coordinates, for the case in figure 3.7 are represented in figure 3.8.

All the residuals, shown in the figures 3.7 and 3.5, that are above or below the upper and lower limit are removed. The main residuals seen are for the case of the range-rate values. When adopting a two-sigma rule, the range of possible values can be reduced, which may result in the removal of crucial values. In the context of the Three-sigma rule, a larger aperture is considered as something that could result in greater uncertainty if the measurements used contain an inaccuracy.

In Figure 3.9, a flow diagram illustrates a simplified step-by-step of the operations discussed in this chapter for more easy comprehension.

3.7.1 Radial-track, along-track and cross-track (RSW) frame

Another important way of representing the standard deviation is on the RSW coordinate system. As explained in section 2.4.5, the RSW is a local frame where the center of the reference system is the orbit of the object itself. Because the covariance matrix represented in this frame is used to calculate the collision probabilities, it is crucial to choose this frame when analyzing the results. Figure 3.10,

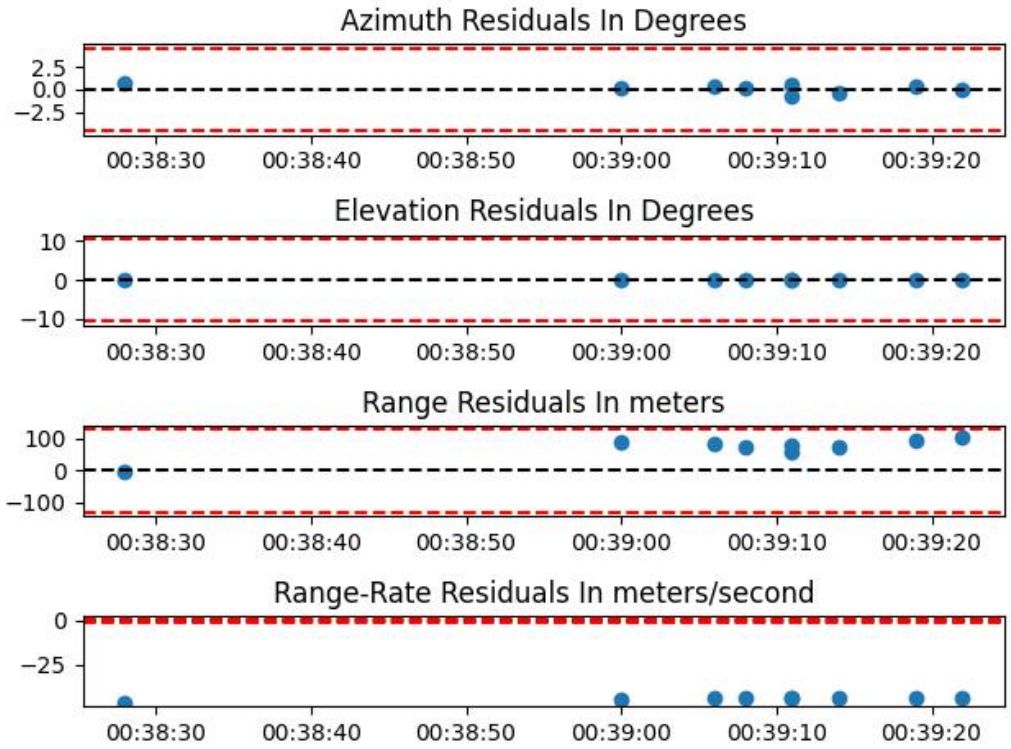


Fig. 3.5 Observed measurements minus predict by the model with the limit Three-sigma rule for dealing with outliers

Table 3.13 State vector and the corresponding standard deviation using the Three-sigma rule at epoch (2022-10-26 00:39:22.364074)

	Pos x-axis	Pos y-axis	Pos z-axis	Vel x-axis	Vel y-axis	Vel z-axis
State Vector	5246566 m	3672042 m	4305715 m	-4881.535 m/s	868.378 m/s	5202.513 m/s
σ	50 m	57 m	90 m	0.037 m/s	0.050 m/s	0.067 m/s

Table 3.14 State vector and the corresponding standard deviation using Two-sigma rule at epoch (2022-10-26 00:39:22.364074)

	Pos x-axis	Pos y-axis	Pos z-axis	Vel x-axis	Vel y-axis	Vel z-axis
State Vector	5246561 m	3672042 m	4305715 m	-4881.535 m/s	868.378 m/s	5202.513 m/s
σ	61 m	22 m	65 m	0.057 m/s	0.043 m/s	0.049 m/s

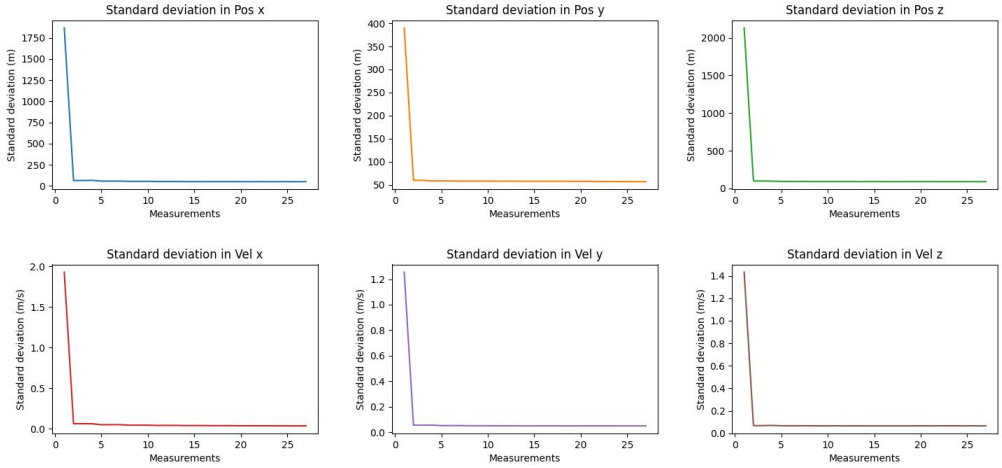


Fig. 3.6 Extended Kalman Filter State Error Propagation use case Three-sigma rule

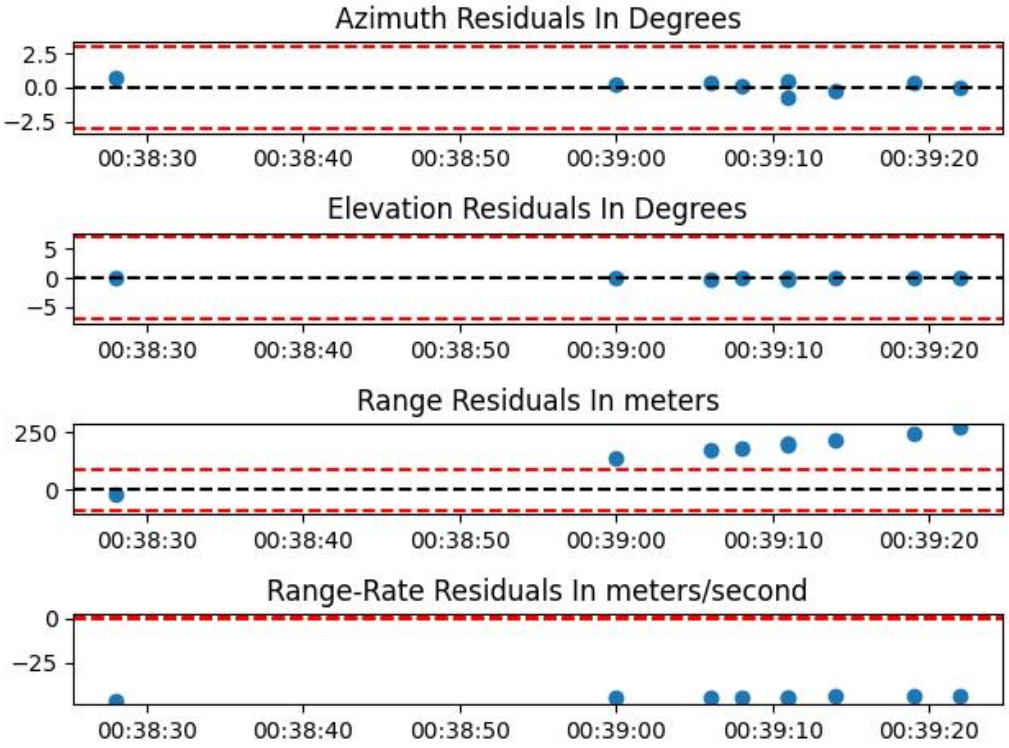


Fig. 3.7 Observed measurements minus predict by the model with the limit Two-sigma rule for dealing with outliers

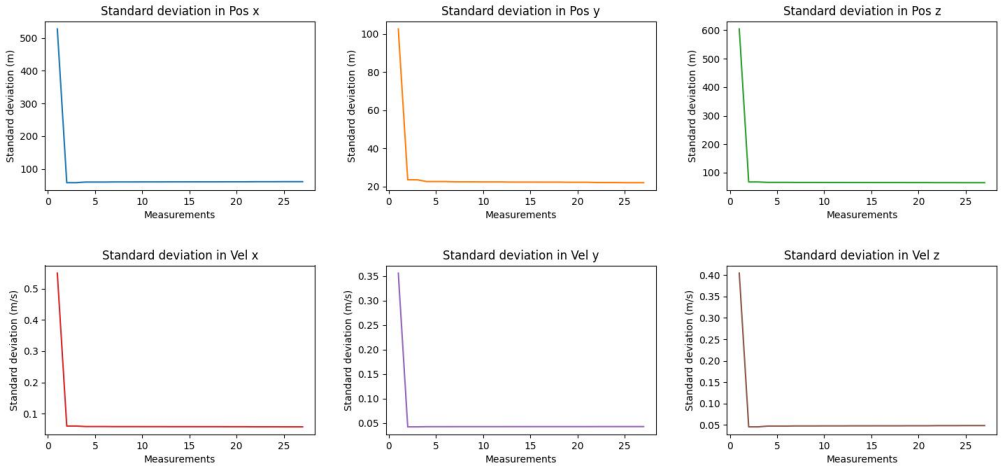


Fig. 3.8 Extended Kalman Filter State Error Propagation use case Two-sigma rule

we show a sketch of the covariance matrix effect in this frame, where the ellipsoid seen in the image is the standard deviation associated with the coordinates on that frame.

For the transformation between the coordinates represented in the ECI frame to the reference frame RSW it was developed a script following the formula ([16])([18]):

$$\hat{R} = \frac{\vec{r}}{|\vec{r}|}, \quad \hat{W} = \frac{\vec{r} \times \vec{v}}{|\vec{r} \times \vec{v}|}, \quad \hat{S} = \hat{W} \times \hat{R}$$

that enable transformation

$$\vec{r}_{ECI} = [\hat{R} : \hat{S} : \hat{W}]\vec{r}_{RSW}, \quad \vec{v}_{ECI} = [\hat{R} : \hat{S} : \hat{W}]\vec{v}_{RSW} \tag{3.4}$$

The matrix J that describes the change in covariance between ECI and RSW components becomes

$$\mathbf{J}_{\frac{RSW}{ECI}} = \begin{bmatrix} [\widehat{\mathbf{R}}|\widehat{\mathbf{S}}|\widehat{\mathbf{W}}]^T & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & [\widehat{\mathbf{R}}|\widehat{\mathbf{S}}|\widehat{\mathbf{W}}]^T \end{bmatrix} \quad \text{and} \quad \mathbf{P}_{RSW} = \mathbf{J}_{\frac{RSW}{ECI}} \cdot \mathbf{P}_{ECI} \cdot \mathbf{J}^T_{\frac{RSW}{ECI}}.$$

where P_{ECI} is the covariance matrix in the ECI frame and the J is the transformation matrix. This script is provided in appendix A.2.

The results obtained in the RSW frame are shown in table 3.15 for the standard deviation and for the state vector.

For the case use of the 2σ rule to deal with the outliers, the state vector and standard deviation the results are shown in table 3.16.

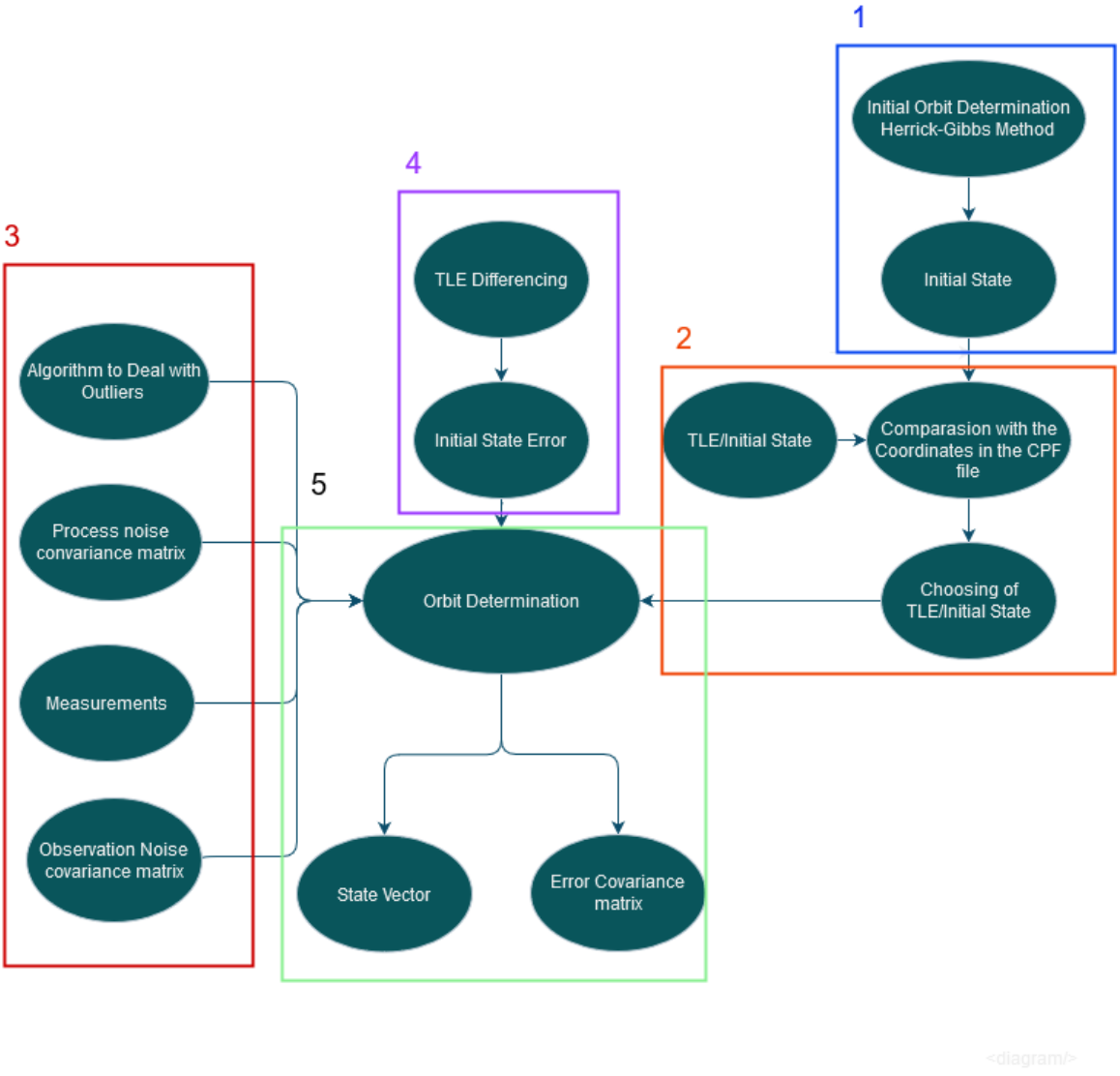


Fig. 3.9 Orbital Determination Flow Diagram

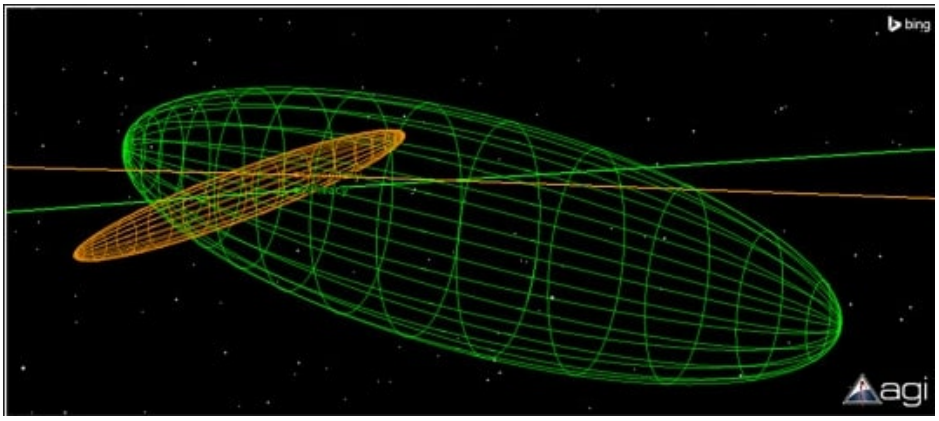


Fig. 3.10 RSW uncertainty sketch taken from ([49])

Table 3.15 State vector and associated Standard deviation use case 3σ rule, in the RSW frame

	Pos Radial-Track	Pos Along-Track	Pos Cross-Track	Vel Radial-Track	Vel Along-Track	Vel Cross-Track
State vector	2266753 m	-810017 m	7331795 m	-2467.114 m/s	-6750.011 m/s	14.012 m/s
σ	39 m	100 m	48 m	0.078 m/s	0.037 m/s	0.031 m/s

Table 3.16 State vector and associated Standard deviation use case 2σ rule, in the RSW frame

	Pos Radial-Track	Pos Along-Track	Pos Cross-Track	Vel Radial-Track	Vel Along-Track	Cross-Track
State vector	2266547 m	-810245 m	7331708 m	-2467.230 m/s	-6750.092 m/s	13.653 m/s
σ	11 m	90 m	16 m	0.085 m/s	0.011 m/s	0.014 m/s

3.8 Conclusion

The process of determining an orbit of an object is complex, from the point of retrieving enough data to start the process and understanding the values after the orbit determination.

This chapter presented the results produced by the algorithm EKF for the different contexts of the initial state error as for different contexts of dealing with the outliers, respectively the 2σ rule and 3σ rule applied to the measurements.

For both cases of use, the 2σ rule and 3σ rule, to deal with the outliers, we presented the results in two different reference systems, the RSW and the J2000. The choice for representing these two frames was due to the important usage of the representation of the covariance matrix in the case of the RSW frame to measure the probability of collisions, as explained in section 3.7.1. As seen in figures 3.8 and 3.6, the state error stabilizes, meaning that in both cases the EKF converge.

The quality of the measurements, the accuracy of the dynamic model, and the computational resources available are important factors that contribute to a higher or lower uncertainty on the final state vector.

The difference in the values of the state error at the end of the process of OD is due to the usage of fewer measurements in one case and in another case the use of more measurements which can lead to either more or less precise results, depending on the error associated to each kind of measurement. The precision of each final state vector will be reflected in the residuals of the propagation shown in the next chapter.

All of these choices were made accounting for the orbit of the satellite Jason3. If we aim to study other objects, we have to repeat the same steps for this object. We note that, the calculation of the appropriated initial state error can differ with respect to different objects and their different orbits.

Chapter 4

Prediction of Orbits

The prediction of the future orbit of the object leads to a choice between accuracy and computation cost. In other to predict the states of objects orbit in LEO, a more accurate approach is suitable, so in this master thesis, a numerical propagation is chosen to perform the propagation of the state vector. Although it cannot be propagated for a distant future like other types of propagators (as the semi-analytical and the analytical) due to the high computational cost of numerical propagation, the accuracy is far better and more precise.

The state vectors that we use for the propagation, are the ones that result from the orbit determination represented in the tables 3.14 and 3.13 shown in the chapter 3.

The type of numerical integrator to use in the propagation is the Dormand Prince. This integrator is a member of Runge-kutta family of integrators with a variable and multi step-size. The reason for this choice is due to the high perturbation in LEO so a step size that is ready to change depending on the situation of propagation, is preferred.

As a way of validating the propagation done, a comparison between the position Cartesian coordinates is done between the propagation performed by Orekit and the positions vectors in the CPF. Every propagation was tested during a four-day period because this is the range of dates offered by a CPF file. The forces to be employed that can modify the trajectory are selected even before the propagation process begins.

The sections 2.9.1 and the introduction to the chapter 3 describe what an integrator is and what an CPF file is, respectively.

4.1 Forces

The force models offered by Orekit are the Third body attraction force model, Atmospheric Drag, Solar Radiation Pressure, Geopotential, Ocean Tides, Albedo Irradiation Forces and General Relativity. All of them were used to maximize the precision of orbit determination related to the object propagating. To improve the prediction of the object's orbit, the perturbations are added to the equation of motion as described in section 2.9.

The third body attraction model takes into account the influence of the Sun and the Moon.

For the drag effect, it was chosen the coefficient with the value 2.2 due to the satellite orbiting in LEO([50]), and for the area of the satellite $20m^2$. The model used was the NRLMSISE-00. This model

was already described in section 2.9.2. For the solar radiation pressure, the area of the satellite was $20m^2$. Finally, the general relativity effect was left to default, meaning that the effect is accountable but no other information is to be given to the software.

All of these forces were explained in the section 2.9.2.

4.2 Validity of the Propagator

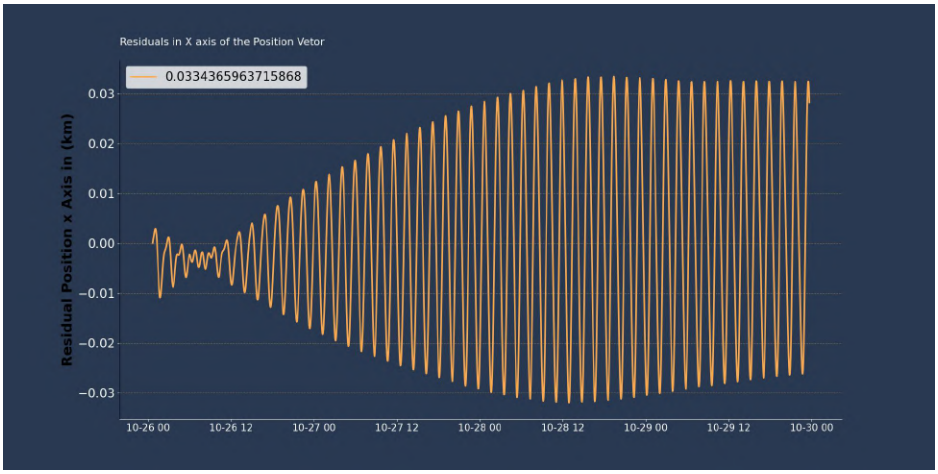
The first test performed was to analyze the accuracy of the propagator offered by Orekit. The object used for the propagation was again the Jason-3. The choice for this satellite is relevant to the fact that it orbits an altitude of 1336 km so it orbits in low orbit. To perform this evaluation the initial guess was the same as the first position of the CPF file. Since CPF file just contains the position vector, one needs to find a way to retrieve the velocity vector to add to the position vector of a certain epoch to a complete initial guess/initial state vector. To get the velocity vector, an interpolation of CPF file to the minimum possible time interval between positions is performed. The reason to have a minimum time interval from the position vector is to get the closest estimation of the instantaneous velocity. This was done with the help of the tool `SLRfield` ([25]) and the minimum time interval offered by the software is 0.001 seconds. After the interpolation, a transformation between reference systems is necessary. `SLRfield` will return the CPF file interpolate in the Geocentric Celestial Reference Frame (GCRF) reference system but the one used in the remaining of the software will be the J2000, so a conversion between coordinates is applied. This conversion was done with the development of the script shown in code listing below.

```
epoch = new  AbsoluteDate(2022,10,26,0,39,22.365,utc); // epoch
Frame emo20222 = FramesFactory.getEME2000(); // J2000 frame
Transform aux = FramesFactory.getGCRF()
.getTransformTo(eme2000,epoch);
//get the transform
Vector3D grcf = new Vector3D(PosX,PosY,PosZ);
//position vector
Vector3D grcf2 = new Vector3D(Velx,VelY,VelZ);
//velocity vector
PVCoordinates Coor = new PVCoordinates(grcf, grcf2);
//Join position and velocity vector
Coordinates=aux.transformPVCoordinates(Coor);
//Transformation
System.out.println("GRCF to ECI "+Coordinates);
```

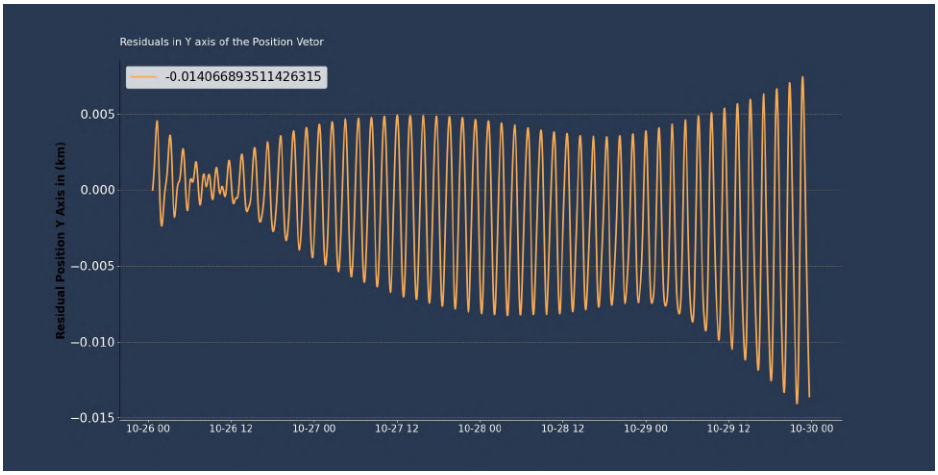
This script made use of the Orekit functions.

As the main goal of this section is to evaluate the propagator of Orekit, the initial state vector is the same as the one in the CPF file and a correction is not needed. Having the complete state vector a propagation for four days is performed. In the CPF file the position vector is represented in the reference frame International Terrestrial Reference System and Frame (ITRF), so another script was

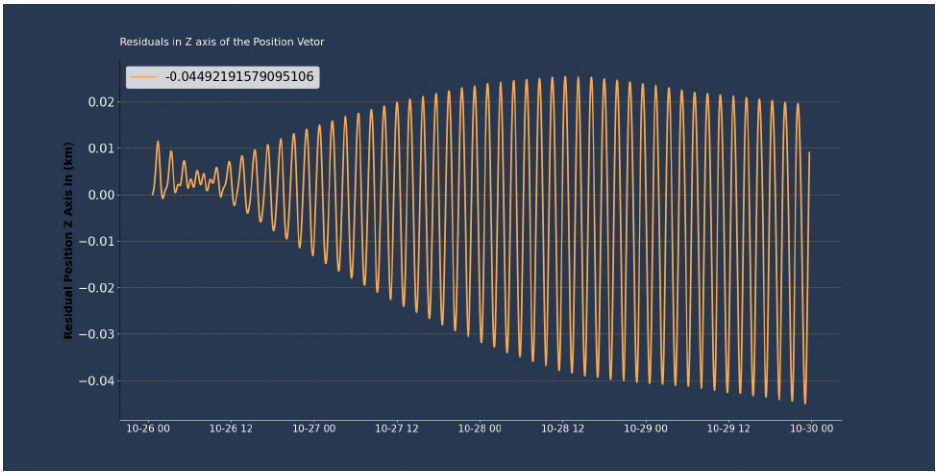
created to read the CPF file and convert it to J2000. With all the configuration done the residuals of prediction between the position vectors are shown in figure 4.1.



(a) Residuals in Position x-axis



(b) Residuals in Position y-axis



(c) Residuals in Position z-axis

Fig. 4.1 Residuals Position in Kilometers

Although the initial guess is assumed to be the same in the tests shown in figure 4.1, the velocity vector can contain errors. Consolidated Prediction Format (CPF) file does not give the velocity vector and the way it was calculated can contain uncertainties. Therefore, the few meters difference between the CPF file and the propagated state vector can be related to this uncertainty in the velocity vector but with just some meters in difference, the numerical propagator given by the software Orekit was presumed to be accurate. In the next section, a prevision of orbits is implemented. Propagation of the state vectors produced from the process of orbit determination demonstrated in the previous chapter is performed in the next section.

4.3 Orbit Prediction

After verifying the validity of the propagator, the real case can be implemented. Since the CPF data has much higher accuracy than the TLE data, the CPF data is used as the "true" state whereas the TLE data corresponds to the "estimate" state. Once again, there is a need to interpolate the CPF file for the times of propagation. Because the propagation of the satellite Jason-3 was performed to start at the time of the last measurements,2021-10-26T00:39:22.364074, to a period of four days with interval points of 3 minutes, the CPF file was interpolated from the time of the last measurement, 2021-10-26T00:39:22.364074, to a period of four days with interval points of 3 minutes. The choice for a period of 4 days is related to, the predicted positions on a CPF file to a duration of 4 days. In figures 4.2, the propagation is demonstrated over a period of 4 days, for the case of use of the state vector presented in the table 3.14. The relative error of the highest residual represented in the figures 4.2, is shown in table 4.2.

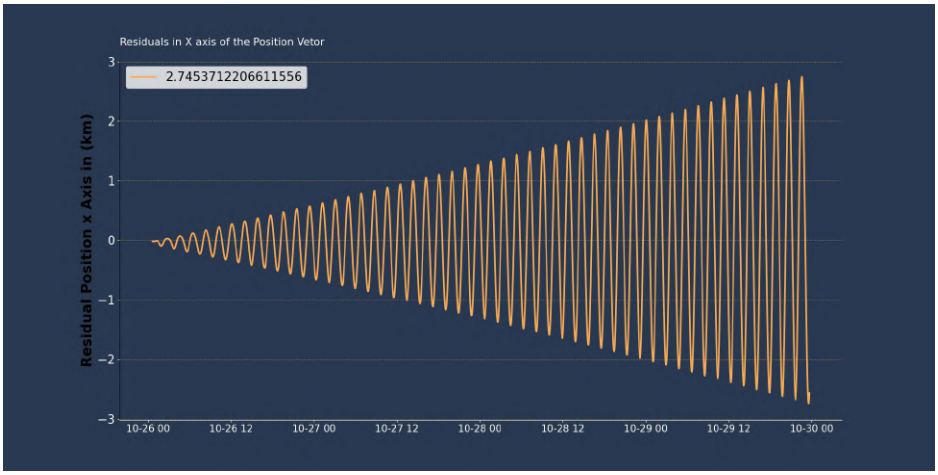
The same residual representations shown in figure 4.2 is shown in figures 4.3, but for the case of the state vector presented in the table 3.13. The relative error of the highest residual represented in the figures 4.3, is shown in table 4.1.

Table 4.1 Relative error higher residual for the Three-sigma rule in the J2000 frame

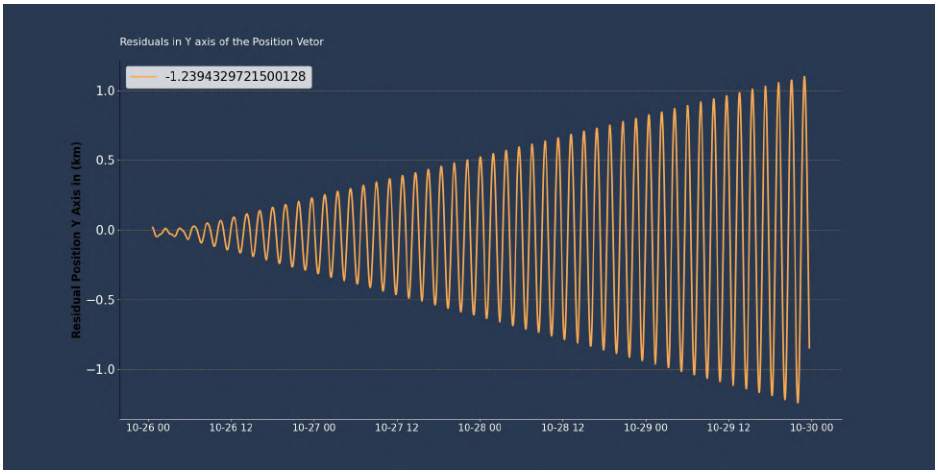
Pos x-axis	Pos y-axis	Pos z-axis
0.0446%	0.0029%	0.005%

Table 4.2 Relative error higher residual for the Two-sigma rule in the J2000 frame

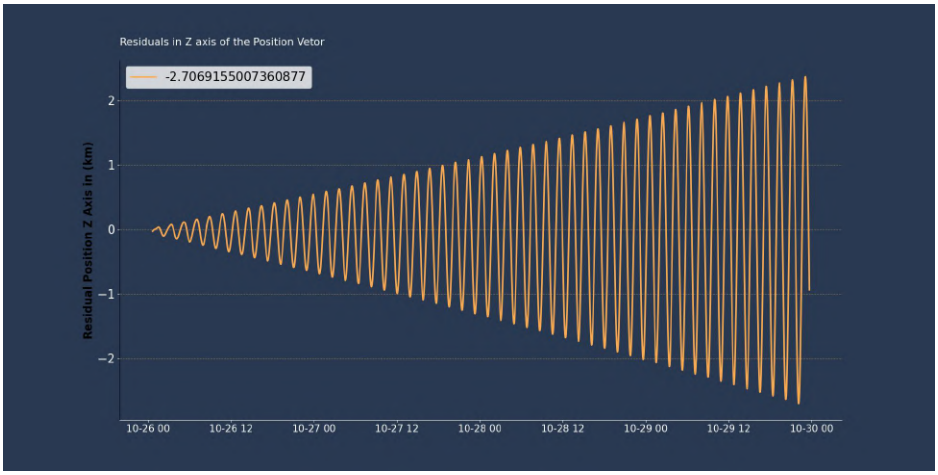
Pos x-axis	Pos y-axis	Pos z-axis
1.6793%	0.01549%	0.0313%



(a) Residuals in the Position X axis sigma 2

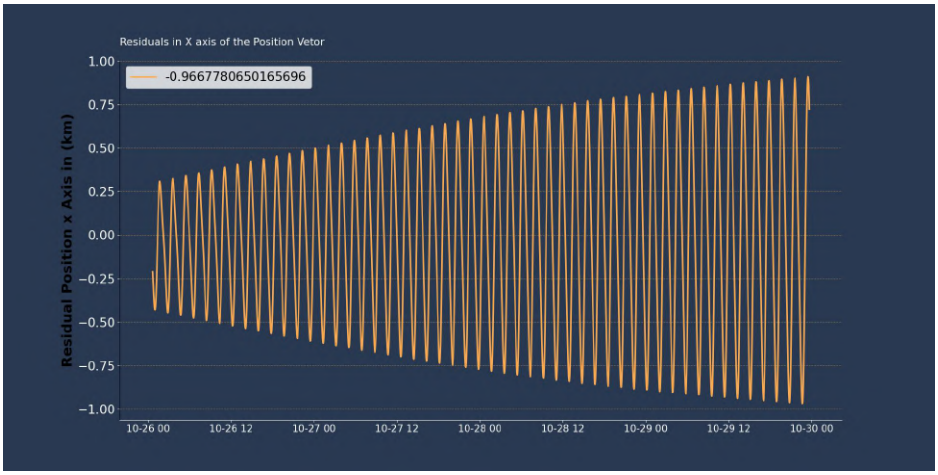


(b) Residuals in the Position Y axis sigma 2

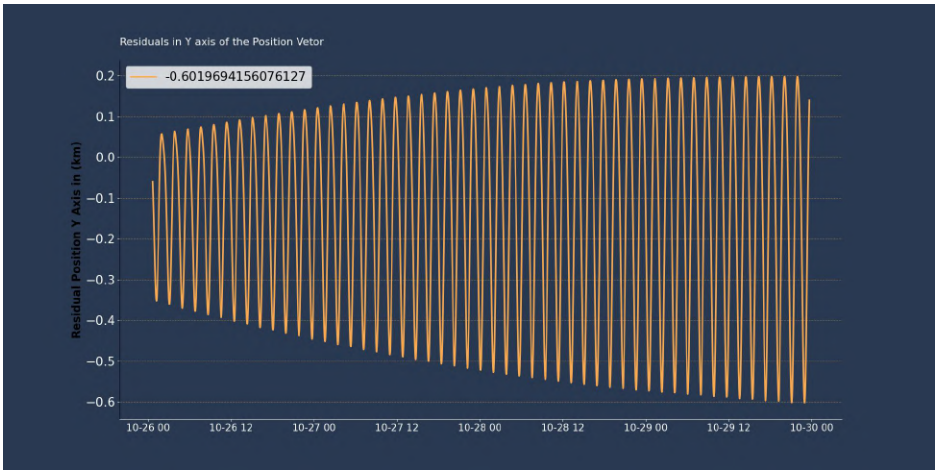


(c) Residuals in the Position Z axis sigma 2

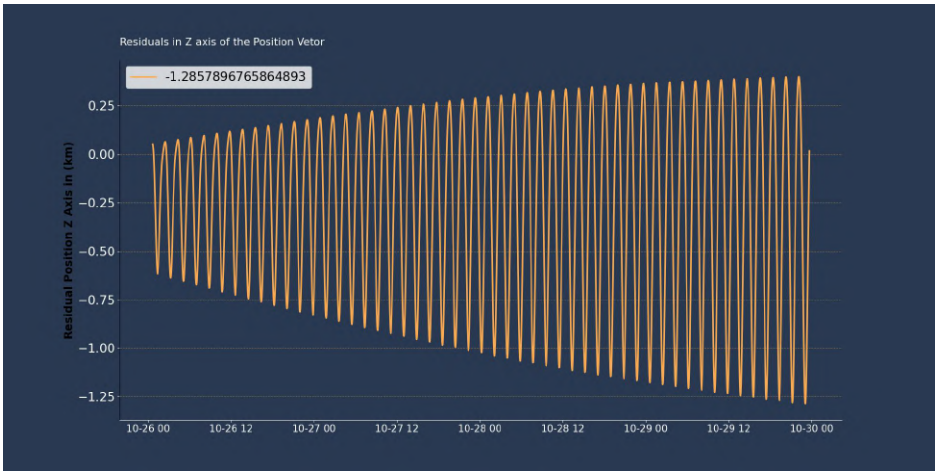
Fig. 4.2 Residuals in the Position Kilometers Two-sigma rule



(a) Residuals in the Position X axis sigma 3



(b) Residuals in the Position Y axis sigma 3



(c) Residuals in the Position Z axis sigma 3

Fig. 4.3 Residuals in the Position Kilometers Three-sigma rule

In both circumstances, the relative inaccuracy is less severe. For the case of state vector propagation shown in 4.3, the relative error is at the decimal level, indicating high propagation accuracy.

4.4 Orbit Prediction in the RSW frame

Next, a comparison between positions in a different reference system is done. This time the reference system is RSW. As mentioned before in the section 2.4.5, the RSW is a local frame where the center of the reference system is the object itself. When performing a transformation between the reference system J2000 and the RSW, the velocity vector is needed. Since the CPF files do not have the velocity vector, the state vector propagated by Orekit shown in the previous figure 4.1 is used as the true state of the object.

All the states propagated in the ECI frame as shown in section 4.2 are converted to the RSW frame. For the use case of the 3σ rule, the residuals are presented in figure 4.4 and the relative error of the highest residual in table 4.3.

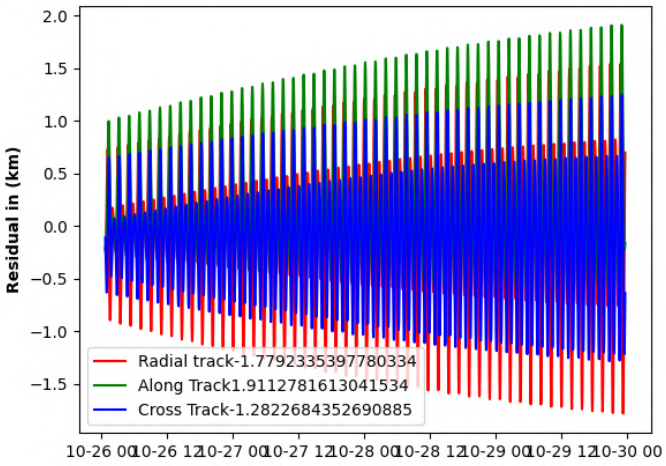


Fig. 4.4 Residuals in RSW frame for the case use of 3σ rule

Table 4.3 Relative error higher residual for the Three-sigma rule in RSW frame

Pos x-axis	Pos y-axis	Pos z-axis
0.0300%	0.0266%	0.01722%

The case of the 2σ rule is shown in figure 4.5 and the relative error of the highest residual in table 4.4.

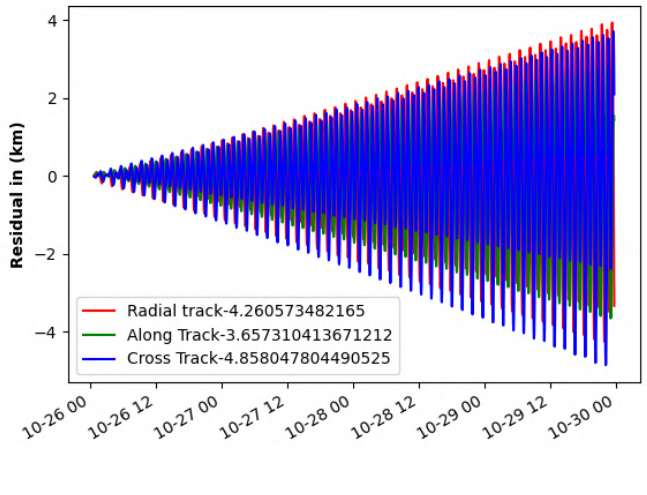


Fig. 4.5 Residuals in RSW frame for the case use of 2σ rule

Table 4.4 Relative error higher residual for the Three-sigma rule in RSW frame

Pos x-axis	Pos y-axis	Pos z-axis
0.0905%	0.0883%	0.4091%

The relative error is always on the decimal level in both scenarios of propagation, demonstrating high precision of propagation.

4.5 Discussion of Results

The different state vectors propagated in this chapter are connected to the various algorithms selected to handle outliers during the OD process. The chapter 3 described all of these various state vectors and algorithms to handle the outliers.

As seen in figure 4.2 the TLE propagation after 4 days reach residuals of the order of kilometers in the position vector. This error is mainly related to the differences in the initial state vector. This error can be related to the error in the initial state error in the TLE and the uncertainty in the force models and observations themselves. Because the algorithm used for orbit determination was the EKF, this error can be tuned with the modification of the initial state error covariance matrix. However even with that tuning, the best results were the ones shown in this chapter.

The residuals of the propagation represented in this chapter could be related to the consequence of the CPF file just containing the position vectors. Therefore the calculation of the velocity vector, as described in section 4.2, can bring uncertainties to this calculation.

Both the propagation residuals in the significant reference systems ECI and RSW, which are described in section 2.4.5, were shown in this chapter.

The fewer residuals in the state vector's propagation, as shown in figure 4.3, can be attributed to the acceptance of more observations taken into account during the orbit determination process. This

could have improved the final state vector's accuracy. The state vector which led to fewer residuals in the propagation is represented in the table 3.13.

Fewer observations were taken into account during the state vector's orbital determination process, which could have resulted in a less accurate state vector which consequently led to greater residuals in the propagation. The state vector that led to higher residuals in the propagation is represented in table 3.14 and the propagation results in figure 4.2.

Chapter 5

Internship at Atlar Innovation

Besides the work shown in chapters 3 and 4, other tasks were completed while doing the internship at Atlar Innovation. The complementary work conducted for this master's thesis during the internship was in the discipline of Astrodynamics, by aiding the Portuguese operations center located on Terceira Island.

The research required for the master's thesis was made easier by the knowledge obtained during the internship in the relevant components and algorithms used in Astrodynamics, such as the orbit determination and propagation algorithm.

5.1 Controlling the Sky

With Portugal's participation in the EUSST initiative, observations made by the ground stations in Portugal are filtered and sent to the EUSST database on the days of the observations. The handling of the big data produced by the observations demands a careful approach as explained in section 2.2.

Accounting with the process of receiving the data, storing it in a local database, and sending a copy to the European database, all this process can be facilitated with the automation of some steps.

Throughout the rest of this chapter, all work done during the internship related to the process of dealing with the software doing the supervision and monitoring of debris will be explained.

5.2 Sensor Tasks

Every ground station located throughout the numerous nations that make up the EUSST organization reports the findings. This observation can involve an object survey or object tracking. Depending on where the ground base is located, the organizing center will deliver the sensor execution plan on a daily basis. For each infrastructure, the sensor plans vary depending on the field of vision. When picking the observation to be performed the following day, accounting for the visibility situation is a crucial step.

Scripts were developed to acquire, arrange, and write this organized data in sorts of files that the sensors could understand in order to automate the process of obtaining the sensor execution plans and managing these plans. The majority of files are dictionary JavaScript Object Notation (JSON) files, and they had to be filtered by sensor and hour according to the desired ground station.

Dictionary JSON files are text files that store data in the form of key-value pairs using the JSON format. In a JSON file, each key is represented as a string and is followed by a colon, and then the corresponding value.

For example, a dictionary JSON file may look like this:

Listing 5.1 JSON Example

```
1
2 {
3   "name": "John Doe",
4   "age": 30,
5   "city": "New York"
6 }
```

When developing the script, it was necessary to take into account the hour that the systems could observe. According to the object's epoch, its right ascension, declination, azimuth, and elevation the sensor would know where to position itself.

Figure 5.1 shows an illustration of a sensor plan.

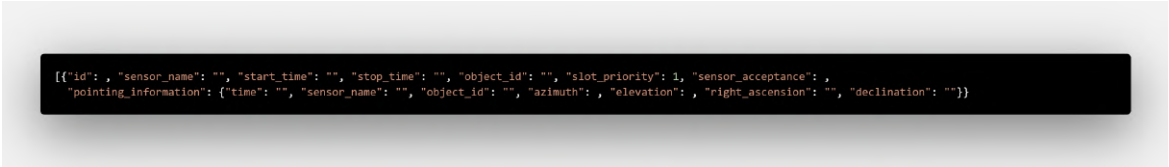


Fig. 5.1 Sensor Plans

The files containing the sensor plans may contain many more than just one plan of observation, so rather than selecting one by one and examining the data and downloading the sensor plans every day, a script was created to automate this process. By running the script, the process is completely automated and generates the filter plans to feed the sensor.

For instance, the plans can be filtered by selecting the hours that the ground station can observe.

5.3 Lifetime Predictions

A prediction of the lifetime of the objects orbiting the Earth can be made by having the objects' orbits around the Earth updated in the catalog. These lifetime predictions are used to predict how objects will behave as long as they stay in orbit.

When projecting the orbits of the objects to predict their orbital evolution, one of the factors to take into account is collision and reentries. Another thing to consider is that during the object's lifetime, if the object has not been observed for a long time or their reentry confirmed or a collision may have occurred. Then a decision related to the status of the object is done, to determine if the object is an account for being "alive" or "not alive".

When accounting for the status of the objects, the alive inquiry signifies whether or not the object still exists.

However, in the case of an object that has not been observed for a long time and has been deactivated, this object may reappear and activation of the presumably dead object may occur.

Other scripts and algorithms created during the internship were used to automate this process. One particular case was to download a particular type of file containing various information about an object, the script could filter the file and create another file with the object that could be deactivated and a sample commentary that could be given to explain the deactivation of the object, depending on the standards chosen to account for determination of the life of the object. These standards could be the following:

- Values in the diagonal of the Covariance Matrix to high (Variance);
- Epoch of the last observation.

This commentary would depend on the causes behind an object's deactivation.

5.4 Confirmation of Reentries

SpaceTrack offers all the most recent TLEs of all known objects orbiting the space region of Earth. Figure 5.2, illustrates an example of a table showing all the information of an object represented in SpaceTrack.

NORAD CAT ID	SATNAME	INTLDES	TYPE	COUNTRY	LAUNCH	SITE	DECAY	PERIOD	INCL	APOGEE	PERIGEE	RCS	LATEST ELSET
1	SL-1 R/B	1957-001A	ROCKET BODY	CIS	1957-10-04	TTMTR	1957-12-01	96.19	65.10	938	214	LARGE	TLE | OMM
2	SPUTNIK 1	1957-001B	PAYLOAD	CIS	1957-10-04	TTMTR	1958-01-03	96.10	65.00	1080	64		TLE | OMM
3	SPUTNIK 2	1957-002A	PAYLOAD	CIS	1957-11-03	TTMTR	1958-04-14	103.74	65.33	1659	211	SMALL	TLE | OMM
4	EXPLORER 1	1958-001A	PAYLOAD	US	1958-02-01	AFETR	1970-03-31	88.48	33.15	215	183		TLE | OMM
5	VANGUARD 1	1958-002B	PAYLOAD	US	1958-03-17	AFETR		132.71	34.24	3831	650	SMALL	TLE | OMM
6	EXPLORER 3	1958-003A	PAYLOAD	US	1958-03-26	AFETR	1958-06-28	103.60	33.50	1739	117		TLE | OMM
7	SL-1 R/B	1958-004A	ROCKET BODY	CIS	1958-05-15	TTMTR	1958-12-03	102.74	65.14	1571	206		TLE | OMM
8	SPUTNIK 3	1958-004B	PAYLOAD	CIS	1958-05-15	TTMTR	1960-04-06	88.43	65.06	255	139	LARGE	TLE | OMM
9	EXPLORER 4	1958-005A	PAYLOAD	US	1958-07-26	AFETR	1959-10-23	92.81	50.25	585	239		TLE | OMM
10	SCORE	1958-006A	PAYLOAD	US	1958-12-18	AFETR	1959-01-21	98.21	32.29	1187	159		TLE | OMM
NORAD CAT	SATNAME	INTLDES	TYPE	COUNTRY	LAUNCH	SITE	DECAY	PERIOD	INCL	APOGEE	PERIGEE	RCS	

Fig. 5.2 Space track taken from ([59])

The Radar Cross Section (RCS) and the Orbit Mean-Elements Message (OMM) are important descriptions of the an object.

The RCS is typically expressed in units of square meters (m^2). It represents the range of RCS values for which a particular tracking sensor is sensitive. Objects with an RCS within this range can be detected and tracked by the sensor, while objects outside of this range may go undetected. The large objects correspond to an RCS bigger than 1 (m^2). For Medium RCS it corresponds to a size between 0.1 and 1 (m^2) and a Small RCS corresponds to a site lower than 0.1 (m^2) size.

The mean orbital elements for one or more satellites are stored in data files called OMM files. The size, configuration, and orientation of a satellite's orbit around the Earth are all described by a collection of mean orbital elements included in the OMM files. These components, which are derived from a more comprehensive set of precise ephemeris data, are often updated frequently to account for

changes in the satellite's orbit over time. Applications for tracking and analyzing satellites employ OMM files to determine the locations and other orbital parameters of spacecraft based on their average orbital elements. OMM files are frequently employed in more sophisticated applications that demand a higher level of accuracy and precision in their calculations because they can represent a satellite's orbit more accurately over a longer period of time than TLEs. More about OMM files can be found in [46].

For the rest of the columns the describe goes as it follows:

- SATNAME: The name of the satellite or debris;
- INTLDES: Committee on Space Research (COSPAR) ID. The COSPAR ID is a unique identifier assigned to all objects launched into space, including satellites. It consists of the year of launch, a three-letter code representing the launch country, and a three-digit sequential number identifying the object within that year's launches.
- Type: Type of the object observed;
- Country: Owner Country of the object observed;
- Launch: Date of the Launch of the Object;
- Site: Place of launching;
- Decay: If it has already decayed, when was it;
- Period: A satellite's or debris period is the amount of time it needs to complete one orbit around its main target. Typically, it is expressed in terms of time units like seconds, minutes, or hours;
- Incl: Inclination already explain section 2.6;
- Apogee: A satellite's or debris apogee is the location in its orbit around the Earth where it is the furthest distance from the center planet.
- Perigee: The part of a satellite's or debris orbit that is closest to the planet's surface is known as the perigee. It is the opposite of the apogee.

Most of the orbit related to an object saved in the local catalog has a NORAD ID associated, this NORAD ID helps to know what object is being observed.

The NORAD ID is used to differentiate between various space objects, including satellites, rocket stages, and debris. Each object has a special NORAD ID that may be used to track it and project its future trajectory in space. Information about the object, including its launch date, owner, and mission, is also logged using the NORAD ID.

Sometimes an orbit can be still considered active in a database but it can have already reentred. Therefore, a script was developed that consists in retrieving all the objects of the local database and the NORAD ID associated with the object, then with an Application Programming Interface (API) of the SpaceTrack, download the TLE related to the objects associated with the ones in the database. This can be done by the NORAD ID that distinguishes the objects. Finally, having the local orbits and the TLEs downloaded, a simple comparison between NORAD ID and confirmation if the object has

reentered or not is performed. If the object was listed as active in the local database and there was evidence of its reentry in the SpaceTrack database, a deactivation of the object is performed and a comment is generated with the data of the reentry. This implies a direct injection of information in the database through a query.

Once the objects are deactivated, they are not accountable for the orbit determination and propagation, reducing the computational effort of the Software.

5.5 Field of Vision

When describing plans of observation determine, which objects will pass by the field of vision of a specific ground station is essential, such that the ground station will be able to catch them before anything else can even begin. As a result, a Python software was created to account for the orbit of various objects. It can predict an object's orbit based on the propagation of its most recent TLE in the SpaceTrack database, and it can filter data from multiple points in the sky to represent the object's various positions throughout its trajectory. The conjunction of numerous points from numerous objects is then to be linked to one file and sent to the ground station after being able to filter all of these constraints. The constraints that the points have to pass are the following:

- Speed of the motors of the instrument: if the object was an azimuth rate or elevation rate faster than the speed of the motors in the azimuth and elevation, is discarded because the instrument will not be able to follow the object, not being able to take multiple measurements of the object;

- Time of integration of the signal: The instrument needs to be able to process the signal of one point of the object and then still be able to "catch" the next points in the sky of the object;

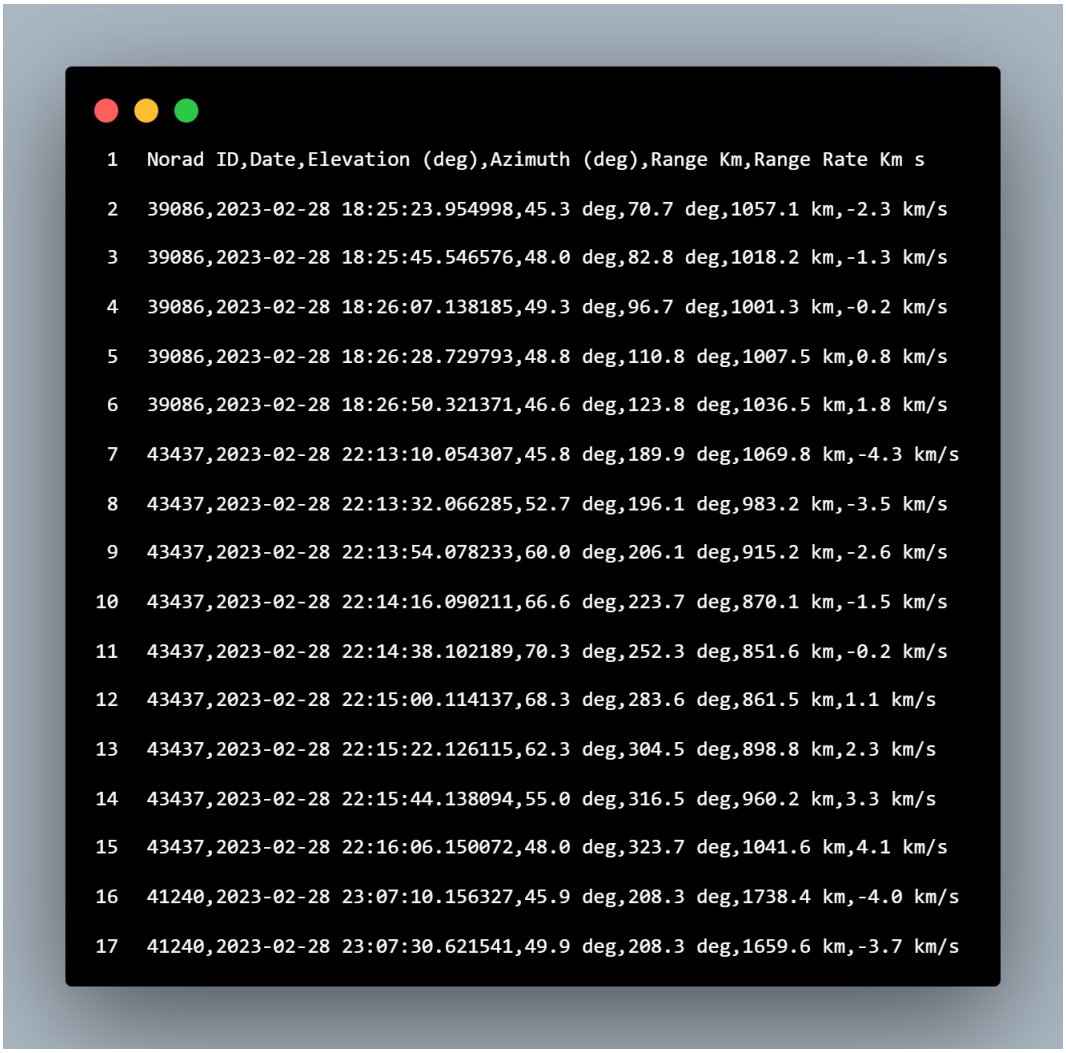
- Limits of observation: If using radar, and depending on the antenna, it can just detect objects between 45 and 90 degrees in elevation, in the case of the azimuth it can go from 0 to 360 degrees and depending on the power of the signal, the height until it can detect the object can depend to;

- Size of the Moon; Depending on the apparent size of the Moon in the night sky it can overshadow the object, limiting the possibility of seeing it and taking measurements, in the case of optical observations. In the case of radar and SLR, the light of the Sun reflecting on the moon can alter the signal;

- Size of the Sun; Depending on the apparent size of the Sun, it can alter the signal sent by the radar or the laser sent by the SLR.

The following figure 5.3 represents a use cast test of an output file after all the constraints were applied.

The file will contain the NORAD ID of the object, the epoch of observation, and the azimuth, elevation, range, and range rate of the object related to that epoch.



```
1   Norad ID,Date,Elevation (deg),Azimuth (deg),Range Km,Range Rate Km s
2   39086,2023-02-28 18:25:23.954998,45.3 deg,70.7 deg,1057.1 km,-2.3 km/s
3   39086,2023-02-28 18:25:45.546576,48.0 deg,82.8 deg,1018.2 km,-1.3 km/s
4   39086,2023-02-28 18:26:07.138185,49.3 deg,96.7 deg,1001.3 km,-0.2 km/s
5   39086,2023-02-28 18:26:28.729793,48.8 deg,110.8 deg,1007.5 km,0.8 km/s
6   39086,2023-02-28 18:26:50.321371,46.6 deg,123.8 deg,1036.5 km,1.8 km/s
7   43437,2023-02-28 22:13:10.054307,45.8 deg,189.9 deg,1069.8 km,-4.3 km/s
8   43437,2023-02-28 22:13:32.066285,52.7 deg,196.1 deg,983.2 km,-3.5 km/s
9   43437,2023-02-28 22:13:54.078233,60.0 deg,206.1 deg,915.2 km,-2.6 km/s
10  43437,2023-02-28 22:14:16.090211,66.6 deg,223.7 deg,870.1 km,-1.5 km/s
11  43437,2023-02-28 22:14:38.102189,70.3 deg,252.3 deg,851.6 km,-0.2 km/s
12  43437,2023-02-28 22:15:00.114137,68.3 deg,283.6 deg,861.5 km,1.1 km/s
13  43437,2023-02-28 22:15:22.126115,62.3 deg,304.5 deg,898.8 km,2.3 km/s
14  43437,2023-02-28 22:15:44.138094,55.0 deg,316.5 deg,960.2 km,3.3 km/s
15  43437,2023-02-28 22:16:06.150072,48.0 deg,323.7 deg,1041.6 km,4.1 km/s
16  41240,2023-02-28 23:07:10.156327,45.9 deg,208.3 deg,1738.4 km,-4.0 km/s
17  41240,2023-02-28 23:07:30.621541,49.9 deg,208.3 deg,1659.6 km,-3.7 km/s
```

Fig. 5.3 List of observations

5.6 Combination of Software

There may be issues due to the data overload and the daily analysis of thousands of orbits. a visit to Terceira Island Azores, analysis, and thorough comprehension of operations center activities help to develop a more refined understanding of the problem. During several days spent at the operations center, some issues were solved and new scripts were created.

With the understanding of the software used in the Terceira Island, inspiration to combine other software to try to reproduce the same purpose was developed.

Orekit, as software to do orbit determination and propagation, as demonstrated in the results in chapters 3.7 and 4, in conjunction with NASA Predictive Information (NPI) Ephemeris Propagation Tool could be a way of conjunction between software.

Software created by NASA's Navigation and Mission Design Branch, a unit of the Goddard Space Flight Center, is known as the NPI Ephemeris Propagation Tool. The tool is meant to offer accurate and effective satellite ephemeris data propagation over extended periods of time.

Ephemeris data is a term used to describe calculated positions and velocities of astronomical objects at particular times, including planets, moons, asteroids, comets and satellites.

Based on a set of initial conditions, the NPI Ephemeris Propagation Tool uses numerical integration techniques to determine the position and velocity of a satellite at a particular instant in time. More about the initial condition for the usage of NPI Ephemeris Propagation Tool software can be found in ([37]).

The choice of using NPI propagator or Orekit propagator would depend on the specific use case and implementation.

One example of a mix of this software is the OKAPI Platform ([52]). In this platform, both Orekit and NPI are used, always depending on the situation and on the decision of the user. This platform accounts for conjunction of orbits, probability of collision analysis, and collision avoidance strategy. Neither Orekit nor NPI Ephemeris Propagation Tool offers this kind of analysis so these algorithms could be built by the programmer or it could also use other free open-source packages such as CARA Analysis Tools.

5.7 Conjunction Assessment and Probability of Collision

The CARA Analysis Tools is a, software that can be used for conjunction analysis in addition to other sorts of analysis. Conjunction analysis is the process of locating and evaluating the danger of collision between probable collisions involving space objects, such as satellites or debris. Tools for data visualization, coordinate transformations and time series analysis are just a few of the elements available in the CARA Analysis Tools software that can be used for conjunction analysis. For instance, to visualize the orbits of multiple objects and pinpoint times when those orbits are in close proximity to each other. The software can assist in estimating the danger of collision and determining the likelihood of a collision by examining the relative locations and velocities of the objects. Figure 5.4 illustrates an example of a conjunction analysis and risk of collision analysis.

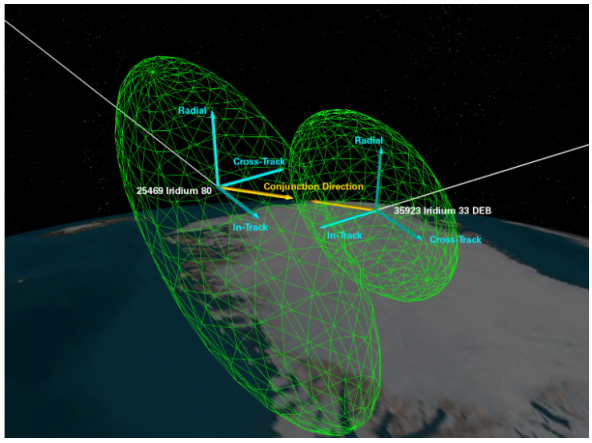


Fig. 5.4 Conjunction illustration taken from ([31])

As seen in the figure 5.4, the center point of the sphere represents the satellite in the RSW coordinate system and the sphere itself represents the uncertainly in the position of the satellite. The RSW coordinate system was already explained in section 2.4.5.

An alert is made when a miss distance between the satellites supervising is reached is minimum, during the conjunction analysis. The miss distance indicates how near the two items will be depending on the findings of the conjunction evaluation screening. When this happens, the risk of collision enters, giving a more realistic value of the probability of collision between the two satellites. The reason for the value being more realistic is due to the use of the covariance matrix of both satellites that contain uncertainly in the position of the satellites. When calculating the probability of collision it is also accounted the object's radius.

The probability of collision is not exactly what is determined while calculating it, but the probability at Time of Closest Approach (TCA) that two objects are closer than a predetermined distance.

The probability of collision can be defined as follows:

$$P = \frac{1}{2 \cdot \pi \cdot \sigma x \cdot \sigma y} \cdot \int_{-OBJ}^{OBJ} \int_{-\sqrt{OBJ^2 - x^2}}^{\sqrt{OBJ^2 - x^2}} \exp\left[\frac{-1}{2} \cdot \left[\left(\frac{x - xm}{\sigma x}\right)^2 + \left(\frac{y - ym}{\sigma y}\right)^2\right]\right] dx \tag{5.1}$$

where σx and σy are the respective standard deviations, OBJ is the combined radius of both objects, x is along the minor axis, y is along the main axis, xm is the miss distance along the minor axis, and ym is the miss distance along the major axis.

From the formula 5.1, there are algorithms that express the probability of collision numerically or analytically. The following are a few examples of algorithms for the probability of collision:

- Foster's (Numerical);
- Chan's (Analytical);
- Patera's (Numerical);
- Alfano's (Numerical).

All of these algorithms have their advantages and disadvantages, and the choice of which to use depends on the specific requirements of the simulation. The desired level of accuracy and the available processing resources determine which method is used. More information about each of the algorithms can be found in ([4]).

5.8 Manouvers

Depending on the likelihood of a collision, one can take measures to maneuver a particular satellite. The purpose of satellite collision avoidance software is to assist space agencies and satellite operators in avoiding collisions with debris. It operates by examining the orbital positions and trajectories of satellites and space debris to spot possible collisions. Various data sources, such as ground-based telescopes, radar systems, and other space-based sensors, are frequently used by the program. The orbits of the satellites and space debris are then predicted using this data, and potential collision risks are identified.

The software will notify satellite operators and suggest ways to prevent a collision if it detects a potential collision. These procedures could involve altering the satellite's orbit or its attitude (orientation) in space.

Software for avoiding satellite collisions is crucial for protecting satellites in orbit from harm or destruction due to collisions. Collision avoidance has become a top priority for the space industry due to the growing quantity of satellites and debris in orbit.

Figure 5.5, illustrates a sketch of a maneuver occurring when detected a future collision.

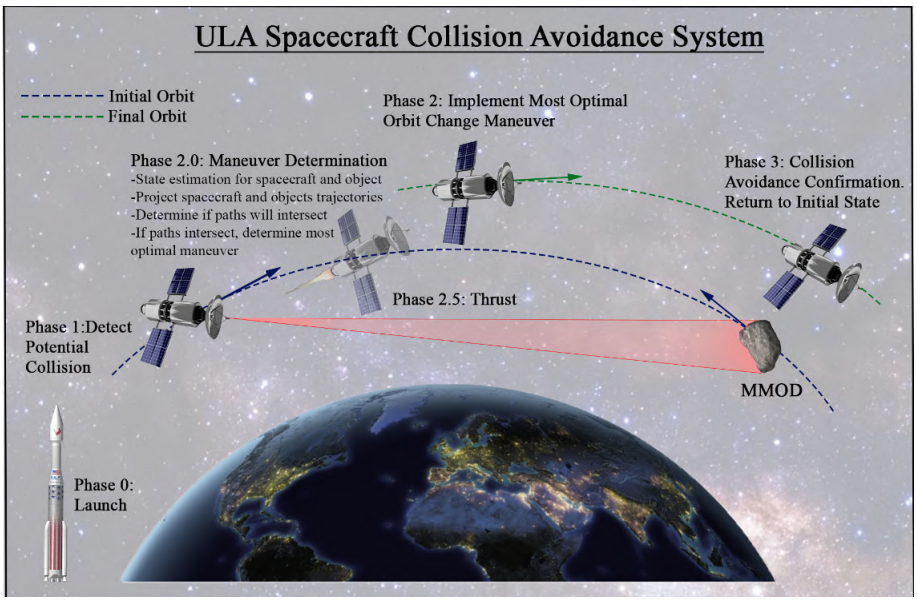


Fig. 5.5 Collision avoidance taken from ([33])

The inquiry into this type of software for performing manouvers was purely exploratory because, in the event that a collision is anticipated, the data will be forwarded to the EUSST organizing, which will then decide whether or not to carry out the necessary maneuver. Not all situations can warrant

moves, including collisions. Low-orbiting objects can perform maneuvers to extend their lifetime and keep them from reentering the atmosphere.

5.9 Conclusion

Each task completed during the internship at Atlar Innovation is described in this chapter. Several scripts and additional software were developed during this internship to aid the procedure in Portugal's operation center. Further research was conducted to gain a better understanding of the processes that came following the identification of the orbit determination and its propagation, namely the probability of collision and satellite maneuvers.

Also, certain scripts were created to complete various autonomous tasks, and other research was done to attempt and determine which algorithms could be more useful and the situations in which the algorithms might be more appropriate. For the aim of measuring the probability of a collision and the realization of maneuvers, the study was purely academic and no further scripts were developed to auxiliary these parts.

Chapter 6

Conclusion

The main goal of this work was to study the software and the systems available to follow the orbit of the space debris. Space awareness has become a real issue. With the rapid increase in the number of debris, systems for monitoring and supervising objects have to be improved. It has become challenging to filter and update these objects orbits while tracking the enormous amount of debris.

The algorithm to determine an initial orbit still presents a lot of issues. These issues are associated with the position vectors given as initial input containing too much error. This is due to the bias in the measurements. The bias associated with the measurements is still one of the main problems when detecting objects. This bias corrupts the data and gives wrong results regarding the measurements resulting in errors in the calculation of the position and velocity of the objects.

The orbit determination algorithm tries to reduce this detection error to the best result possible.

In the case of the Extended Kalman Filter, the results of the orbit determination can be "tuned" with the modification of the initial state error associated with the initial guess. As for the case of the Batch Least Square, modification of the convergence threshold chosen, delineate the quality of the output state vector the user wants and the quality of data that the user has. Sometimes, a high convergence threshold is preferred compared to a small lack of convergence threshold that can lead to a divergence of the data.

The type of propagators used for the prediction of the orbits of the object depends on the computational cost and accuracy of the results the user wants. Numerical propagators produce better results, but demand more computational effort than semi-analytical and analytical propagators. Depending on the task needed to be performed, the type of propagator can change. For example, when predicting the passages of objects, an analytical propagator is used but, when predicting orbits a numerical propagator is better.

Since the propagation of the orbit contain errors, accounting for the uncertainty of the position of the object is important. When estimating the likelihood of collisions, the covariance associated in the RSW frame is essential because it incorporates the uncertainty in the position between objects and it gives more realistic results of the probability of collision.

The work presented in this master's thesis can be used to examine the supervision and monitoring of debris processes and to offer solutions to perform real time predictions of debris orbits. This master's thesis also presents many scripts created to support the process of working with systems to observe and detect the future passage of debris as a real-world application.

6.1 Future Work

It would be important to investigate in further work the evolution of the covariance matrix propagation. In order to provide accurate estimations of probability, it is required to account for the uncertainty associated with each individual object. A more thorough examination of the conjunction assessment for the research of reentries of objects would be an additional significant component that would be intriguing for further inquiry.

Doing a conjunction assessment evaluation would be the first step before even taking the probability of a collision into consideration.

Given the current rapid increase in the number of LEO objects, the threat of reentries will increase along with the number of objects orbiting in LEO.

Since it is still unclear where an object will impact the Earth's surface when its reentry starts, further research into measuring a more accurate trajectory of reentries of LEO objects would be intriguing to explore.

With the rising risk to the human population as a result of the rapid growth in the number of debris, a more proactive strategy is required to begin. If more international collaboration to deal with debris is not established, future space missions may be imperiled, as may the usage of communication used by humans now, such as GPS, jeopardized.

References

[1] (2020). Primeiro telescópio português dedicado a operações de vigilância espacial entrou em funcionamento na ilha Terceira . https://www.portugal.gov.pt/pt/gc22/comunicacao/noticia?i=primeiro-telescopio-portugues-dedicado-a-operacoes-de-vigilancia-espacial-entrou-em-funcionamento-na-ilha-terce

[2] ai solutions (2022). Orbit Reference Frames . https://ai-solutions.com/_help_Files/orbit_reference_frames.htm.

[3] Al Sadoon, S. H. and Elias, B. H. (2013). Radar theoretical study: minimum detection range and maximum signal to noise ratio (snr) equation by using matlab simulation program. *American Journal of Modern Physics*, 2(4):234–241.

[4] Alfano, S. (2007). Review of conjunction probability methods for short-term encounters (aas 07-148). *Advances in the Astronautical Sciences*, 127(1):719.

[5] Almeida, D. S. G. (2018). Dynamics and attitute estimation and control of a spacecraft system.

[6] Arnold, D., Montenbruck, O., Hackel, S., and Sośnica, K. (2019). Satellite laser ranging to low earth orbiters: orbit and network validation. *Journal of geodesy*, 93(11):2315–2334.

[7] Atallah, A. M., Woollands, R. M., Elgohary, T. A., and Junkins, J. L. (2020). Accuracy and efficiency comparison of six numerical integrators for propagating perturbed orbits. *The Journal of the Astronautical Sciences*, 67(2):511–538.

[8] Berry, M. M. (2004). *A variable-step double-integration multi-step integrator*. Virginia Polytechnic Institute and State University.

[9] Berry, M. M. and Healy, L. M. (2005). *Speed and accuracy tests of the variable-step störmer-cowell integrator*. Univelt, Inc.

[10] Bruinsma, S. L., Sánchez-Ortiz, N., Olmedo, E., and Guijarro, N. (2012). Evaluation of the dtm-2009 thermosphere model for benchmarking purposes. *Journal of Space Weather and Space Climate*, 2:A04.

[11] CCSDS (2007). TRACKING DATA MESSAGE. https://public.ccsds.org/Pubs/503x0b1s.pdf.

[12] Chiaradia, A., Kuga, H. K., and Prado, A. (1999). Investigation of simplified models for orbit determination using single frequency gps measurements. *Journal of the Brazilian Society of Mechanical Sciences*, 21:165–172.

[13] Classroom, T. P. (2022). The Doppler Effect . https://www.physicsclassroom.com/class/waves/Lesson-3/The-Doppler-Effect.

[14] COMINI, M. (2017). Orbit determination with the simplified general perturbation model.

[15] COMMISSION, E. (2022). COMMISSION IMPLEMENTING DECISION (EU) 2022/1245. https://eur-lex.europa.eu/eli/dec_impl/2022/1245/oj.

[16] Cordelli, E., Schildknecht, T., and Vananti, A. (2017). *Improvement of Space Debris Orbits*. PhD thesis, Philosophisch-naturwissenschaftliche Fakultät der Universität Bern.

[17] Corrao, G. (2018). Batch vs. Sequential estimation methods in Orbit Determination. https://www.agi.com/articles/Batch-vs-Sequential-estimation-methods-in-Orbit-De.

[18] da Silva, J. P. R. (2021). Orbital collision warning and avoidance.

[19] DeSoto, C. B. (2019). Radar Techniques - Primer Principles . https://www.embedded.com/using-nonlinear-kalman-filtering-to-estimate-signals/.

[20] ESA (2022). ESA's Space Environment Report 2022 . https://www.esa.int/Space_Safety/Space_Debris/ESA_s_Space_Environment_Report_2022.

[21] ESO (2022). Observing conditions: definitions . https://www.eso.org/sci/observing/phase2/ObsConditions.html.

[22] Fernandez Uson, M. (2016). Geosar mission: Orbit determination methods and techniques.

[23] Finance, A., Dufour, C., Boutéraon, T., Sarkissian, A., Mangin, A., Keckhut, P., and Meftah, M. (2021). In-orbit attitude determination of the uvsq-sat cubesat using triad and mekf methods. *Sensors*, 21(21):7361.

[24] Flores, R., Burhani, B. M., and Fantino, E. (2021). A method for accurate and efficient propagation of satellite orbits: A case study for a molniya orbit. *Alexandria Engineering Journal*, 60(2):2661–2676.

[25] Foundation, P. S. (2022). Welcome to the SLRfield package . https://pypi.org/project/slrfield/.

[26] FURUNO ELECTRIC CO., L. (2014). Radar Basics. https://www.furuno.com/en/technology/radar/basic/.

[27] Garcia, M. (2021). Space Debris and Human Spacecraft. https://www.nasa.gov/mission_pages/station/news/orbital_debris.html.

[28] Gavin, H. P. (2019). The levenberg-marquardt algorithm for nonlinear least squares curve-fitting problems. *Department of Civil and Environmental Engineering, Duke University*, 19.

[29] Geul, J., Mooij, E., and Noomen, R. (2017). Tle uncertainty estimation using robust weighted differencing. *Advances in Space Research*, 59(10):2522–2535.

[30] GROUP, C. et al. (2021). Orekit . https://www.orekit.org/.

[31] Hall, R., Alfano, S., and Ocampo, A. (2010). Advances in satellite conjunction analysis. In *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference*.

[32] Hobbs, D. and Bohn, P. (2006). Precise orbit determination for low earth orbit satellites. *Annals of the Marie Curie Fellowships*, 4:128–135.

[33] HOLDRIDGE, A., VAN ANNE, G., VALADE, T., BAILEY, R., GOLDNER, I., HARTMAN, S., HOFFMAN, A., WARNER, R., BALKE, J., TURMAN, C., et al. (2020). Collision avoidance system testbed (cast).

[34] Höyhtyä, M., Boumard, S., Yastrebova, A., Järvensivu, P., Kiviranta, M., and Anttonen, A. (2022). Sustainable satellite communications in the 6g era: A european view for multi-layer systems and space safety. *IEEE Access*.

[35] Jain, V. and Heydari, P. (2013). Radar fundamentals. In *Automotive radar sensors in silicon technologies*, pages 5–11. Springer.

[36] Kaitlyn (2020). Two Line Elements (TLE) . https://kaitlyn.guru/projects/two-line-elements-tle/.

[37] Kebschull, C. (2023). NEPTUNE . https://github.com/Space-Systems/neptune.

[38] Keil, E. M. (2014). *Kalman Filter Implementation to Determine Orbit and Attitude of a Satellite in a Molniya Orbit*. PhD thesis, Virginia Tech.

[39] Kelso, T., Hoots, F., and Roehrich, R. (1988). Spacetrack report no. 3-models for propagation of norad element sets. *NASA, Tech. Rep*.

[40] Lakdawalla, E. (2012). Figuring out orbital positions from orbital elements . https://www.planetary.org/articles/3380.

[41] Lawrence, A., Rawls, M. L., Jah, M., Boley, A., Di Vruno, F., Garrington, S., Kramer, M., Lawler, S., Lowenthal, J., McDowell, J., et al. (2022). The case for space environmentalism. *Nature Astronomy*, 6(4):428–435.

[42] LLC, C. (2021). China says its rocket debris landed in the Indian Ocean. https://www.cnbc.com/2021/05/09/china-says-rocket-debris-landed-in-indian-ocean-west-of-maldives.html.

[43] Maciuk, K. (2016). Different approaches in glonass orbit computation from broadcast ephemeris. *Journal of Geodetski Vestnik*, 60(3):455–466.

[Mahdi] Mahdi, M. C. Tigrisat orbital motionsimulation and analysis.

[45] Mehrholz, D., Leushacke, L., Flury, W., Jehn, R., Klinkrad, H., and Landgraf, M. (2002). Detecting, tracking and imaging space debris. *ESA Bulletin(0376-4265)*, (109):128–134.

[46] Messages, O. D. (2009). Issue 2. recommendation for space data system standards (blue book). Technical report, CCSDS 502.0-B-2. Washington, DC: CCSDS.

[47] Minear, K. and Martin, G. P. (2018). System and method for detection and orbit determination of earth orbiting objects. US Patent 9,989,634.

[48] NASA (2021). Consolidated Prediction Format (CPF) . https://ilrs.gsfc.nasa.gov/data_and_products/formats/cpf.html.

[49] Nicolini, L. and Caporali, A. (2018). Investigation on reference frames and time systems in multi-gnss. *Remote Sensing*, 10(1):80.

[50] Nwankwo, V. U., Denig, W., Chakrabarti, S. K., Ajakaiye, M. P., Fatokun, J., Akanni, A. W., Raulin, J.-P., Correia, E., Enoh, J. E., and Anekwe, P. I. (2021). Atmospheric drag effects on modelled low earth orbit (leo) satellites during the july 2000 bastille day event in contrast to an interval of geomagnetically quiet conditions. In *Annales Geophysicae*, volume 39, pages 397–412. Copernicus GmbH.

[51] of Illinois Urbana-Champaign, U. (99). Locating a Target angles and distances used . http://ww2010.atmos.uiuc.edu/(Gh)/guides/rs/rad/basics/angl.rxml.

[52] OKAPI (2023). Space Situational Awareness . https://okapiorbits.space/.

[53] Park, E.-S., Park, S.-Y., Roh, K.-M., and Choi, K.-H. (2010). Satellite orbit determination using a batch filter based on the unscented transformation. *Aerospace Science and Technology*, 14(6):387–396.

[54] Pastor, A., Sanjurjo-Rivo, M., and Escobar, D. (2021). Initial orbit determination methods for track-to-track association. *Advances in Space Research*, 68(7):2677–2694.

[55] Potter, A. (1995). Ground-based optical observations of orbital debris: A review. *Advances in Space Research*, 16(11):35–45.

[56] Reid, T. G. R. (2017). *Orbital diversity for global navigation satellite systems*. PhD thesis, Stanford University.

[57] Ribeiro, M. I. (2004). Kalman and extended kalman filters: Concept, derivation and properties. *Institute for Systems and Robotics*, 43:46.

[58] Sagnieres, L. and Sharf, I. (2017). Uncertainty characterization of atmospheric density models for orbit prediction of space debris. In *7th European Conference on Space Debris*, volume 1, pages 18–21.

[59] saic (2023). SpaceTrack . https://www.space-track.org/.

[60] Shou, H.-N. (2014). Orbit propagation and determination of low earth orbit satellites. *International Journal of Antennas and Propagation*, 2014.

[61] Shuster, S. (2017). A survey and performance analysis of orbit propagators for leo. *GEO, and highly elliptical orbits. Utah State University*.

[62] Singh, R. (2020). It's all about Outliers . https://medium.com/analytics-vidhya/its-all-about-outliers-cbe172aa1309.

[63] Smale, D. A. P. (2021). FITS Documentation . https://fits.gsfc.nasa.gov/fits_documentation.html.

[64] Sośnica, K., Bury, G., Zajdel, R., Kazmierski, K., Ventura-Traveset, J., Prieto-Cerdeira, R., and Mendes, L. (2021). General relativistic effects acting on the orbits of galileo satellites. *Celestial Mechanics and Dynamical Astronomy*, 133:1–31.

[65] Staff, E. (2006). Using nonlinear Kalman filtering to estimate signals . https://www.embedded.com/using-nonlinear-kalman-filtering-to-estimate-signals/.

[66] Steindorfer, M. (2023). Satellite Laser Ranging . https://www.oeaw.ac.at/en/iwf/research/research-groups/satellite-laser-ranging.

[67] (STK), S. T. K. (2016). Orbit Propagators for Satellites . https://help.agi.com/stk/11.0.1/Content/stk/vehSat_orbitProp_choose.html.

[68] Stoughton, C., Lupton, R. H., Bernardi, M., Blanton, M. R., Burles, S., Castander, F. J., Connolly, A., Eisenstein, D. J., Frieman, J. A., Hennessy, G., et al. (2002). Sloan digital sky survey: early data release. *The Astronomical Journal*, 123(1):485.

[69] Strugarek, D., Sośnica, K., Zajdel, R., and Bury, G. (2021). Detector-specific issues in satellite laser ranging to swarm-a/b/c satellites. *Measurement*, 182:109786.

[70] Sun, H., Zhang, H.-F., Zhang, Z.-P., and Wu, B. (2015). Experiment on diffuse reflection laser ranging to space debris and data analysis. *Research in Astronomy and Astrophysics*, 15(6):909.

[71] System, G. G. G. O. (2023). SLR / LLR – Laser Ranging. https://ggos.org/item/slr-llr/.

[72] TechTarget (2007). downlink and uplink . https://www.techtarget.com/searchmobilecomputing/definition/downlink-and-uplink.

[73] Urrutxua, H., Bombardelli, C., Peláez, J., and Huhn, A. (2011). High fidelity models for orbit propagation: Dromo vs. störmer-cowell. In *European Space Surveillance Conference*, pages 7–9.

[74] Vallado, D. A. (2001). *Fundamentals of astrodynamics and applications*, volume 12. Springer Science & Business Media.

[75] Vetter, J. R. (2007). Fifty years of orbit determination. *Johns Hopkins APL technical digest*, 27(3):240.

[76] Vittaldev, V. (2010). The unified state model. derivation and applications in astrodynamics and navigation.

[77] Wang, S., Xu, M., Zhang, X., and Wang, Y. (2022). Fitting nonlinear equations with the levenberg–marquardt method on google earth engine. *Remote Sensing*, 14(9):2055.

[78] Weber, B. (2022). Reference Frames . https://orbital-mechanics.space/intro/reference-frames.html.

[79] Weeden, B. (2009). Iridium-cosmos collision fact sheet. *Secure World Foundation*.

[80] Wikipedia (2022). List of reentering space debris . https://en.wikipedia.org/wiki/List_of_reentering_space_debris.

[81] Wikipedia (2023). List of space debris producing events . https://en.wikipedia.org/wiki/List_of_space_debris_producing_events.

[82] Yin, L., Wang, L., Zheng, W., Ge, L., Tian, J., Liu, Y., Yang, B., and Liu, S. (2022). Evaluation of empirical atmospheric models using swarm-c satellite data. *Atmosphere*, 13(2):294.

[83] Ziegler, R. (2019). *Levenberg-Marquardt Filter for Orbit Estimation*. PhD thesis, San José State University.

[84] Zolg, S. (2020). 68–95–99.7 — The Three-Sigma Rule of Thumb Used in Power BI. . https://towardsdatascience.com/68-95-99-7-the-three-sigma-rule-of-thumb-used-in-power-bi-59cd50b242e2.

Appendix A

Scripts

This appendix presents the scripts developed for the master thesis as a description of the TLE format.

A.1 TLE description

With SPG4, TLEs can be used to propagate orbits. They have mean orbital elements that have been obtained by removing periodic perturbations. SPG4 theory then adds these perturbations to create state vectors (position and velocity) of objects with TLEs that are frequently updated. This means that orbital elements are publicly available and stored in TLEs format. The TLE format is shown in figure A.1

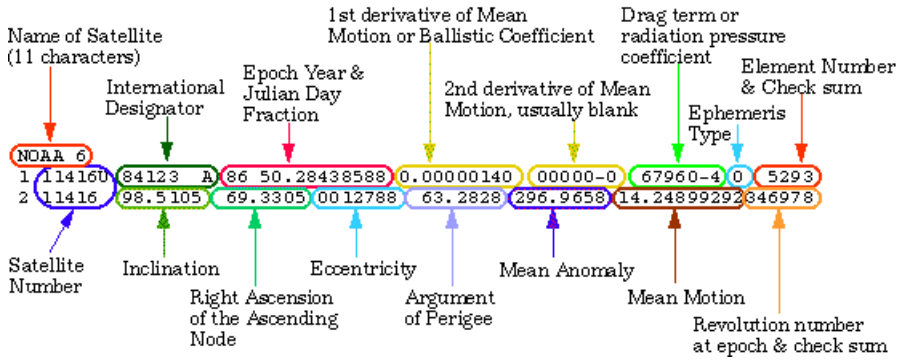


Fig. A.1 TLE elements ([Mahdi])

The elements represented in a TLE, as shown above, are([36]):

- Name of Satellite: (NOAA 6) This is merely the name given to the satellite;
- International Designator: (84123 A) The 84 denotes that the launch occurred in 84, the 033 indicates that it was the 123th launch of the year, and "A" indicates that it was the first object associated with this launch;
- Satellite number: (11416) Satellite catalog number follow by a classification (U: unclassified, C: classified, S: secret);

- Epoch Date and Julian Date Fraction: The Julian day fraction is simply the number of days in a certain year. The date shown in figure A.1, shows "86" as the epoch year 1986 and 50.28438588 as the Julian day fraction, telling that is a little more than 50 days after January 1, 1986;

- Ballistic Coefficient: (0.00000140) The ballistic coefficient, often known as the first derivative of mean motion, is the daily rate of change in the number of revolutions the item completes each day divided by 2. The units are revolutions per day;

- Second Derivative of Mean Motion: (00000-0 = 0.00000) It measures how much drag increases or decreases over time. Normally, this number is zero. The value is the mean motion's second derivative divided by six. This value is expressed in units of revolutions per day cubed;

- Drag Term: (67960-4 = 0.000067960) The parameter, also known as the radiation pressure coefficient, is another drag term in the SPG4 predictor. Earth $radii^{-}1$ are the units. The final two characters establish a power of ten that is relevant;

- Element Set Number and Check Sum: (5293) The element set number (529 in this case) is a running total of all 2 line element sets created for this object. Because this function is performed by multiple agencies, numbers are occasionally skipped to avoid ambiguity. The counter should always increase over time until it reaches 999, at which point it should reset to 1. The line's last number is the check sum of line 1;

- Inclination (degrees): Inclination representation in the TEME frame;

- Right Ascension of the Ascending Node (degrees): The angle formed between the vernal equinox and the point at which the orbit crosses the equatorial plane (as seen from the north). The value reported is the rising node's TEME mean right ascension;

- Eccentricity: A constant that defines the orbit's form (0=circular, less than 1=elliptical). The provided value is the mean eccentricity;

- Argument of Perigee (degrees): The angle formed by the ascending node and the orbit's closest approach to Earth (perigee) also represented in the TEME frame;

- Mean Anomaly (degrees): The angle measured from perigee of the satellite's orbital location in relation to a circular orbit with radius equal to the semi-major axis;

- Mean Motion: (14.24899292) The figure represents the object's average number of orbits each day. When the next element reaches 9999, there are 8 digits after the decimal, with no trailing space(s);

- Revolution Number and Check Sum: (346978) At Epoch Time, the orbit number. As a matter of routine, this period is picked quite close to the genuine ascending node transit. The final digit is the check sum on line 2.

A.2 ECI transformation to RSW

A simple script to perform the transformation between the covariance matrix represented in the ECI reference system to the RSW reference system is provided in this section.

To produce the transformation, the position vector, velocity vector and the associated covariance matrix have to be given.

```
import numpy as np
import math

r=np.array([5246377.377481407,3671965.2741612564,4305796.681117097])
v=np.array([-4881.784362190023,868.2759706025225,5202.444929161765])
# Reshape r and v vectors if input as column vector
r = np.reshape(r, (1, 3))
v = np.reshape(v, (1, 3))

# Setting up unit vectors in the RIC directions
h = np.cross(r, v)
rhat = r / np.linalg.norm(r)
chat = h / np.linalg.norm(h)
ihat = np.cross(chat, rhat)

# RIC to ECI rotation matrix
ECItoRIC = np.vstack((rhat, ihat, chat))
#this covariance is for 3 sigma
covariane_matrix = np.array([[2452.0212225104,
-2100.8757750813,-3933.3009840019,1.3300750179,
2.1852657341,2.9561574651],
[-2100.8757750813,3232.8248992747,
2542.2649086604,-0.6485784707,-2.623499088,
-2.2282022904],
[-3933.300984002,2542.2649086605,
8178.4510134462,-2.1576521247,-2.8897421931,
-5.7178981927],
[1.3300750179,-0.6485784707,-2.1576521247,
0.001376319,0.0010338891,
0.0019724473],
[2.1852657341,-2.623499088,-2.8897421931,
0.0010338891,
0.0025124143,0.0024186741],
[2.9561574651,-2.2282022904,-5.7178981927,0.0019724473,
0.0024186741,0.0044826728]])

ECI=covariane_matrix


# Rotating covariance matrix (6x6 case)
if ECI.shape[0] >= 6:

    # Get additional Transformation Term Diagonals
    if ECI.shape[0] > 6:
        additional_terms = ECI.shape[0] - 6
```

```
    else:
        additional_terms = 0

    # Creating rotation matrix that will work for 6x6 covariance
    ZERO = np.zeros((3, 3))
    ECItoRIC6x6 = np.block([[ECItoRIC, ZERO], [ZERO, ECItoRIC]])
    ECItoRIC6x6 = np.block([[ECItoRIC6x6, np.zeros((6, additional_terms))],
                            [np.zeros((additional_terms, 6)), np.eye(
    additional_terms)]])

    # Rotating covariance matrix from RIC to ECI coordinates
    RIC = ECItoRIC6x6 @ ECI @ ECItoRIC6x6.T

x=np.array(RIC).diagonal()
diagonal_values=[]
for i,values in enumerate  (x):
    diagonal_values.append(math.sqrt(x[i]))
print("diagonal values",diagonal_values)
r=np.reshape(r, (3, 1))
v=np.reshape(v, (3, 1))
print("Position",np.matmul(np.linalg.inv(ECItoRIC), r))
print("Velocity",np.matmul(np.linalg.inv(ECItoRIC), v)
)
```

Listing A.1 ECI to RSW transformation

A.3 TLE Differencing

TLE Differencing is a technique of obtaining a state error referent to a TLE. It consists on propagating previous TLE to the date of the prime TLE and accounting with the residuals of the different TLEs to the same epoch of the primeTLE.

```
import numpy as np
TLE_prime=[7337245.244729307, 2395137.9586154674, -16139.203361978713,
-891.5686063433857, 2773.068322265332, 6571.2119939881]
data=open(Path) #file for the propagated TLEs
lines=data.readlines()

xx=[]
yy=[]
zz=[]
vx=[]
vy=[]
vz=[]
for x in lines:
    xx.append(x.split('\t')[0])
    yy.append(x.split('\t')[1])
    zz.append(x.split('\t')[2])
    vx.append(x.split('\t')[3])
    vy.append(x.split('\t')[4])
    vz.append(x.split('\t')[5])
```

```
data.close()
deltaY=[]
TLE_prime=np.array(TLE_prime)
for i in range(len(xx)):
    aux=[float(xx[i]),float(yy[i]),float(zz[i]),float(vx[i]),
    vy(ola5[i]),float(vz[i])]
    aux=np.array(aux)
    aux = np.array(np.subtract(aux,TLE_prime))
    deltaY.append(aux)
m=np.divide((sum(deltaY)),len(xx))
pTCA=0
for i in range(len(xx)):
    aux2=deltaY[i]-m
    pTCA=+ np.multiply(aux2,aux2)
finalmatrix=(len(xx))
newList = [x / finalmatrix for x in pTCA]
#State error associated to the TLE prime
print("Final",np.divide(pTCA,finalmatrix))
```

Listing A.2 TLE Differencing

A.4 Initial Orbit Determination with Herrick-Gibbs Method

To compute the velocity of an object orbiting the Earth giving it three position vectors, it can be used the Herrick-Gibbs method.

This process can be called Initial Orbit Determination. Through the use of a formula represented in the book of ([74]) was created this script that by giving the three position vectors in meters and their corresponding epoch it returns the velocity vector at the epoch of the second position vector.

```
from datetime import datetime, timedelta
import numpy as np
import math as math


def datetime_to_jd(date):
    """
    Convert a `datetime.datetime` object to Julian Day.

    Parameters
    ----------
    date : `datetime.datetime` instance

    Returns
    -------
    jd : float
        Julian day.

    Examples
    --------
    >>> d = datetime.datetime(1985,2,17,6)
```

```
    >>> d
    datetime.datetime(1985, 2, 17, 6, 0)
    >>> jdutil.datetime_to_jd(d)
    2446113.75

    """
    days = date.day + hmsm_to_days(date.hour,date.minute,date.second,date.
    microsecond)

    return date_to_jd(date.year,date.month,days)


r1=[5502168.677709107, 3611959.6733120605,4031463.3886448448]
r2=[5350810.113175424, 3650100.551896426, 4194782.47860386]
r3=[5324336.5252980245, 3652893.2021886236, 4223699.776597945]


r1= np.array(r1)
r1=r1*10**-3

r2= np.array(r2)

r2=r2*10**-3
r3= np.array(r3)
r3=r3*10**-3
data3= datetime(2022,10,26,00,39,6,749777)
data2=datetime(2022,10,26,00,39,0,894421)
data1=datetime(2022,10,26,00,38,28,977031)

data3=datetime_to_jd(data3)
data2=datetime_to_jd(data2)
data1=datetime_to_jd(data1)

deltaT31=(data3-data1)*86400.00
deltaT32=(data3-data2)*86400.00
deltaT21=(data2-data1)*86400.00

miu=3.986004418E5 # Standard gravitational parameter of the Earth
z=np.cross(r2,r3)

zNorm=np.linalg.norm(z)
r1Norm=np.linalg.norm(r1)
r2Norm=np.linalg.norm(r2)
r3Norm=np.linalg.norm(r3)




Z23 = np.cross(r2, r3)
numerator = np.dot(Z23, r1)
denominator = np.linalg.norm(Z23) * np.linalg.norm(r1)
cos_alpha_cop = numerator / denominator
```

```
alpha_cop = 90 - np.degrees(np.arccos(cos_alpha_cop))
print("Com=",alpha_cop)


cos_alpha_12 = np.dot(r1, r2) / (np.linalg.norm(r1) * np.linalg.norm(r2))
alpha_12 = np.arccos(cos_alpha_12)
print("aplha_12=",np.degrees(alpha_12))
cos_alpha_23 = np.dot(r2, r3) / (np.linalg.norm(r2) * np.linalg.norm(r3))
alpha_23 = np.arccos(cos_alpha_23)
print("aplha_23=",np.degrees(alpha_23))

r1Cubo=np.linalg.norm(r1)**3
r2Cubo=np.linalg.norm(r2)**3
r3Cubo=np.linalg.norm(r3)**3
parte1DentrodosPar=(1/(deltaT21*deltaT31))+(miu/(12*r1Cubo))

parte1=-deltaT32*parte1DentrodosPar*r1


subtrai=deltaT32-deltaT21
parte2DentrodosPar=(1/(deltaT21*deltaT32))+(miu/(12*r2Cubo))

parte2=subtrai*parte2DentrodosPar*r2


parte3DentrodosPar=(1/(deltaT32*deltaT31))+(miu/(12*r3Cubo))

parte3=deltaT21*parte3DentrodosPar*r3




velocidade2=parte1+parte2+parte3


print("Velocity at the second point",velocidade2)
```

Listing A.3 Herrick-Gibbs Method